Computer Animation
Humans SS 18

Prof. Dr. Charles A. Wüthrich,
Fakultät Medien, Medieninformatik
Bauhaus-Universität Weimar
caw AT medien.uni-weimar.de

Introduction

- Humans are especially difficult to simulate
  - We are very familiar with them
  - The human body is very complex, with >200 bones and tree times as much muscles
  - Humans are deformable, not rigid
  - Motion is diversem, depending on the person walking
  - Except in sports, little analysis of walk/movement has been done

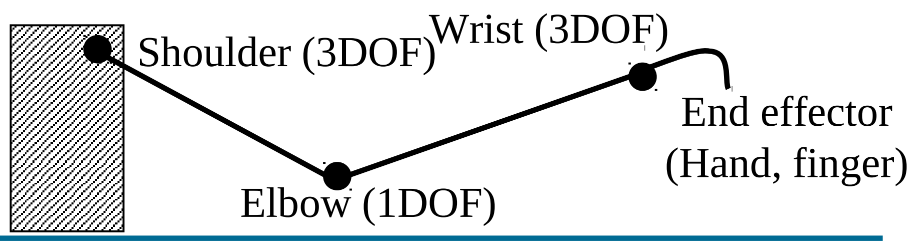
Reaching and grasping

- The first animation task one can do is move its upper limb to reach and grab something
- For the first moment, one can consider the arm to be independent of the movement of the rest of the body
- This is a rough approximation, but we start from this easy simplification
- We therefore start considering the arm as isolated

Modeling the arm

- An arm can in first instance be described as a 7 DOF system
- These joints are limited: for example the elbow can extend only from 20 to 160 deg.
- A configuration (or pose) of the arm is the set of the seven joint angles
- In humans, a rotation is distributed along all the arm, ie.e if I rotate the wrist, the rest will be rotated a bit, so as to distribute strain
- Also, the joints will avoid to reach maximum extension and twisting if possible

- To solve the problem, one can use forward or inverse kinematics (in case I have a goal to reach)
- Note that since the elbow has 1 DOF, shoulder, elbow and wrist are on a plane
- One can use this plane to control the movement
- Recent studies say that first we twist the wrist to get best grabbing, then we adjust the arm plane

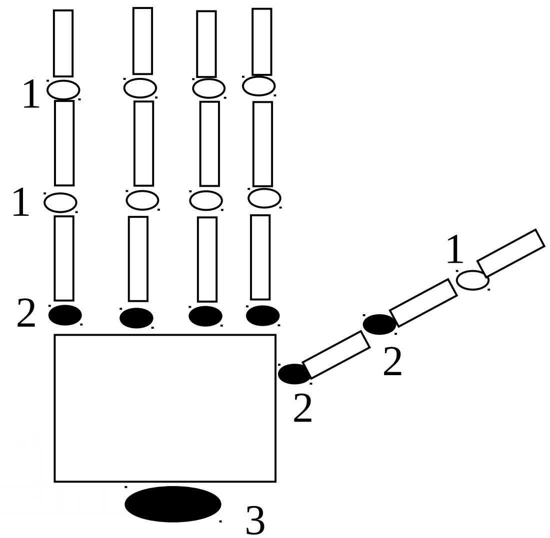


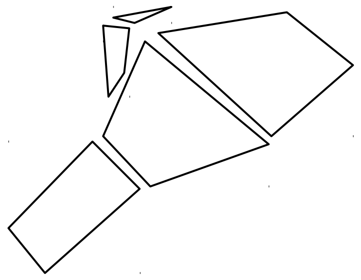
Modeling the shoulder

- There are two basic ways of modeling the shoulder:
  - One models it as a ball joint with 3 coinciding DOF
  - A more precise ones, binds the clavicle and the scapula to the system of 3 coincident joints to add realism to the motion

Modeling the hand

- A fully articulated hand requires many more joints
- The first picture depicts the basic DOF of a hand model
- An opposable thumb provides great and complicated dexterity
- Simplified models can be used, gruoping finger like in a 4 finger glove
- Complete models remodel the anatomical bone structure





Coordinating movement

- Once one has understood how to model the different parts, it is necessary to coordinate the whole movement among the different parts
- When we reach to grab, we also turn the torso
- If we are standing, this implies even a small rotation of the legs
- Even when walking, we use our arms to counterbalance the weight

Obstacles when grabbing

- If obstacles are present and clutter the space, the problem of reaching becomes complicated
- Not only the end effector (hand) must reach the right spot, but also the whole parts of the arm must follow a collision free path
- To compute such a path, often a potential field is added to the enviroment, so that the path chosen can follow the gradient to the field
- However, one can get trapped in local minima
- To avoid this, genetic algorithms are used.
- Unfortunately, these strategies tend to look unnatural, since they tend to pass too close to the obstacles, and humans stay as clear from them as possible
- Moreover, human reaching behaviour varies depending if the goal can be seen or not: if we do not see, we are much more cautios
- Finally, human reaching is usually done to apply strength, and grabbing is optimized to allow enough force to be applied for the task due

Walking

Copyright (c) 1885
Eadweard Muybridge

Source: http://web.inter.nl.net/users/anima/↵
chronoph/first-15/index.htm

Walking

- Walking is done by each human in his own way
- We will first take a look at walking, subsequently at running
- Walking can be seen as „controlled falling“
- Walking is an almost „cyclical“ movement
- Let us try to analyze in detail how walking is done:
  - We will break up the walking by distinguishing the relationships between feet and contact points on the ground
  - We call a *stride* the sequence of motions between two repetitions of a body configuration (i.e. the movements done between same configurations)

Source: http://web.inter.nl.net/users/anima/↵
chronoph/first-15/index.htm

Walking

- Walking is done by each human in his own way
- We will first take a look at walking, subsequently at running
- Walking can be seen as „controlled falling“
- Walking is an almost „cyclical“ movement
- Let us try to analyze in detail how walking is done:
  - We will break up the walking by distinguishing the relationships between feet and contact points on the ground
  - We call a *stride* the sequence of motions between two repetitions of a body configuration (i.e. the movements done between same configurations)

Source: http://web.inter.nl.net/users/anima/↵
chronoph/first-15/index.htm

Walking/Running

- The left stance starts when the left heel starts to hit the ground
  - Then the right foot is lifted, and the right stance starts
  - Once the right foot lifts, it swings forward and finally strikes the ground again. This is called right swing phase (of the left stance)
  - After the right foot hits the ground, the right foot stance begins
- Running is done basically in the same way
  - Except that in the swing phase both feet are lifted from the ground

Pelvic movement/Knees/Feet

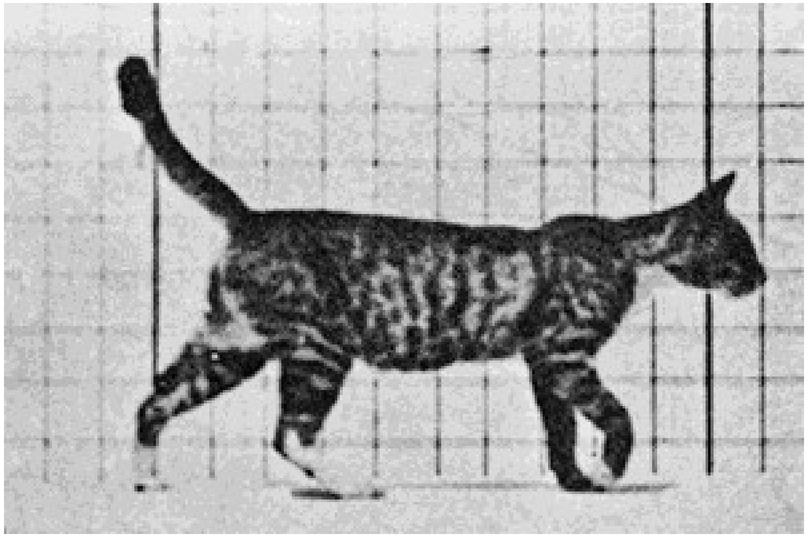
- The pelvis lifts to aid the walking process
- To avoid the leg hitting the ground in the swing, the knees have to be flected
- Finally, the ankle and the toes joints are flected to complete the movement and flatten the feet so it lands flat and to absorbe the shock
- The toe joint is used to roll over it before lifting the foot

Physical simulation

- Dynamic simulation can be used to compute a walking figure
- However, often animators want more control
- Here, dynamics must help the animator, not take over the whole animation.
- For dynamic simulation, all body parts have mass, and the feet muscles impart forces to the upper part of the body
- For simplification sake, sometimes the leg dynamics can be modeled with a telescopic joint during the stance phase

Humans are not animals (walkwise)

- Of course, every species has its own type of walking, which makes it more and more difficult



Source: http://web.inter.nl.net/users/anima/chronoph/first-15/index.htm

Facial animation

- If you thought walking is scary, facial animation is even more scary
- First, faces are very different in shape, and even expressions are different among individuals
- A face is a complex deformable shape with distinct articulated parts
- It is very important in animations, since it defines the character (we look at faces more than elsewhere)
- Lip syncing has to be done on faces
- Good facial models allow to geometrically represent a specific person (conformation)

Facial animation

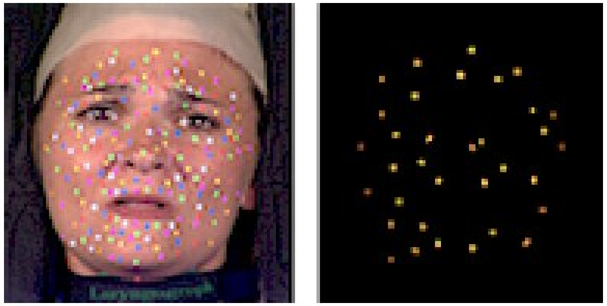
- Wide range of applications:
  - Cartooning:
    - expression and personality have to be conveyed,
    - but shape can be simple (a sphere).
    - Expressions can be mapped as a texture map
  - Realistic character animation:
    - anatomy has to be modeled
    - Complex geometry has to be done, as well as movement
  - Telecomm. To reduce bandwidth
  - For HCI purposes:
    both these last ones require real time behaviour

Facial models

- When modeling a face, two factors are important:
  - Static properties
  - Dynamic properties
- Usually, polygonal models are used
  - Easy to create, but show often too coarse detail, or too complex
  - Recently, data acquisition and curve fitting used
- For smooth surfaces, splines are used
  - Reduced data
  - Splines scale not well: tradeoff between detail and data complexity has to be found
  - One can use subdivision surfaces to achieve detail where needed
  - Recently, hierarchical B-splines have been used with success

Creating facial models

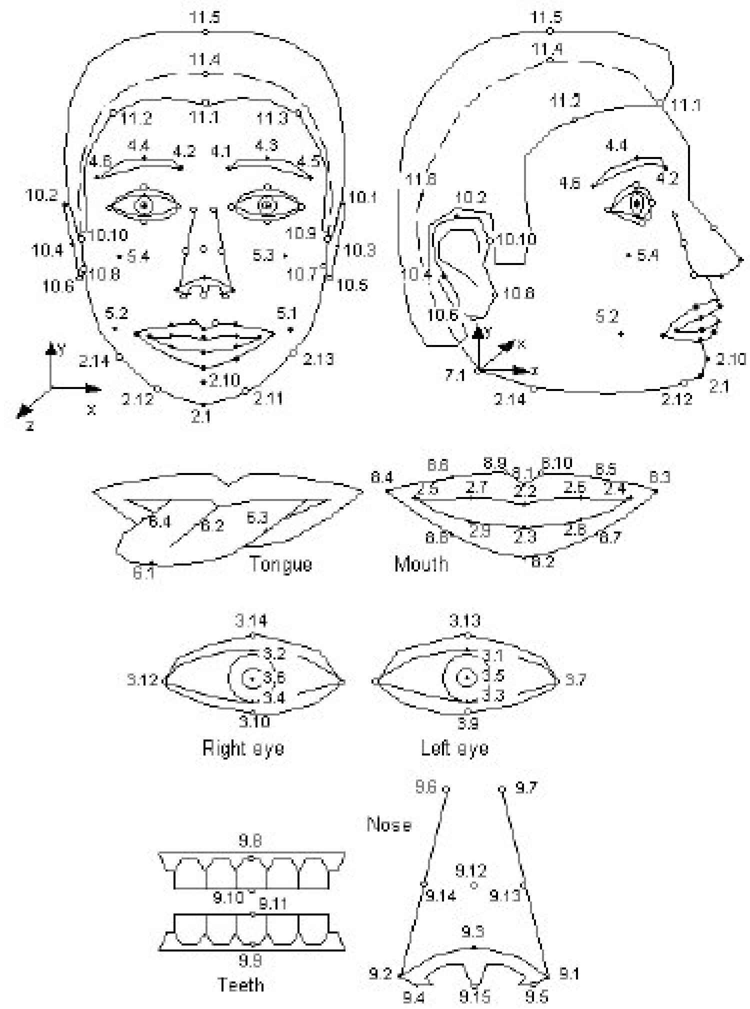
- Two main ways:
  - Through a 3D modeler
  - Through data acquisition
    - From a clay model, through 3D scanning
    - From photographs or cameras, through 3D recognition
- Parke developed a parametrized model in which he subdivides in
  - Conformational parameters: distinguishing features of each face. He outlined 22 parameters, of which
    - 5 control shape of forehead, cheekbone, cheek hollow, chin and neck



    - 13 are scale distances between facial features
    - 5 parameters control feature translation in space
  - Expressive parameters: include things like
    - Upper lip position
    - Eye gaze
    - Jaw rotation
    - Eyebrow separation
  - Most of them are concerned with the eyes and mouth, since this is where most expressions are done

Creating facial models: MPEG-4

- MPEG-4 proposes a set of *facial definition parameters* (FDP) devoted to facial animation for videoconferences
- Take a look at the feature points in the picture
- Once the model is defined, it can be animation through a set of *Facial animation parameters* (FAP)
- MPEG-4 defines 68 FAPs, controlling the rigid roatation of the head, eyeballs, eyelids and mandible



Face textures

- Often a video of a real human is mapped in real time onto a 3D face model
- This allows the full expressivity of expressions
- However, finding mapping points is not a trivial task
- Registering the texture with the 3D model is a complex work
- Also, lighting conditions of video have to be reproduced in the virtual model
- A simpler form is done by mapping a scanned image (in 2D) to the 3D model

Animating faces

- Simplest approach: selecting key poses and interpolating between them
- This of course is very limited, but if one uses the weighted sum of more poses instead (whereby their sum must be 1), one can interpolate more completely in key position space
- Moreover, when the animator allows a wide variety of key poses (= lots of expressions), their number grows too much
- Which leads to the question: what are the primitive motions of the face?

The Facial Action Coding System

- Developed by psychologists (Ekman & Friesen)
- Deconstructs all facial expression into a set of basic facial movements, called action units (AU)
- There are 46 AUs, like „brow lowerer“, „inner brow raiser“, „wink“,....
- Given the AU, the animator can build the corresponding movement
- A parameter controls the amount of facial animation that results from the associated AU
- While nice work, FACS is difficult to apply in practice, because this work was done for analysis, not for generation

Parametrized models

- Parametrizing the facial model according to primitive actions and controlling the parameter values in time is the most common way of implementing facial animation
- Here, the basic work used is the already mentioned work by Parke with conformational and expressive parameters

Muscle models

- Some authors did a complete modeling of the muscles of the face, and defined the skin as an elastic medium moved by such muscles
- While results are impressive, the computational load by such a model is enormous (given the high number of muscles in a face)
- Moreover, there are different kinds of muscles: circular (sphinteres) and linear, and this adds to complexity
- Of course, the more complex the model, the more difficult for an animator is to control movement

Muscle models

Human models

- Basically, for representing whole humans the model geometry can be represented as
  - Polygonal model
  - Higher order curves models (splines)
  - High level model (arm, hand... etc.)
- Recently data acquisition methods have been used to get more individual models
- Moreover, multilayered models have been created (bones + tissue)
- However, hair models have been always a problem, because hair complexity and the complexity of their movement is very high
- Finally, also clothing represents a big problem, because clothes behave elastically and adhere to the body, and to produce realistic clothing one needs to implement the physics of clothes
  - Draping
  - Dynamical models
  - Collision detection

Motion Tracking

- As we could see from the last part, making synthetic animations of humans is complicated
- An alternative to the synthetic generation of movement is the recording of the movement of real humans.
- This is called motion capture:
  - Real humans are equipped with sensors (or markers) applied to the different body parts
  - The xyz positions of these markers in time are recorded while the „actor“ is performing movement

Motion Tracking: Magnetic

- There are basically two ways of doing motion tracking:
- Electromagnetic sensors:
  - uses sensors positioned at the joints that transmit their position and orientation
  - Transmission is done either by cable (= limit freedom of movement) or by wireless
  - De facto real time
  - Main problem: room must be free of field distortions
  - Limited range, accuracy problems
  - High purchase cost

Motion tracking: optical

- Optical tracking:
  - Uses video cams to record motion of the subject
  - Easier to wear (reflective markers are applied to subject)
  - Wider range
  - No cables
  - Real time difficult
  - Data is noisy and error prone
  - Because orientation is not directly generated, more markers are required than with magnetic trackers
  - Cameras may vary in quality and principle:
    - Infrared
    - Very high resolution
    - But also available for consumer videocams => cheap!
  - In the next, we will take a look at how optical tracking works

Motion tracking: optical

- Objective is to reconstruct the three-dimensional model of a motion and apply it to a synthetic model
- Work can be subdivided in 3 tasks:
  - Image processing: Images need to be processed so as to be able to locate, identify and correlate the markers
  - Camera calibration: 3D locations of markers have to be extracted from the 2D images
  - Constraint satisfaction: The 3D marker locations have to be constrained to the physical model whose motion is being captured

Optical tracking: Image Processing

- Optical markers can be of different shapes: pingpong balls, other markers...
- Stuck to the joints with velcro/tape
- One of the problem is that they stick out of the body, so there is a difference between where they are and where the real joints are
- Moreover, they can moveWRT the real joint too
- Once video digitized, it can be analyzed
- If background static, it can be subtracted
- Once this is done, the marker gets searched for
  - Of course, with more markers it is more complicated, because they may get occluded
  - Therefore one has to track the markers across the frames

Optical tracking: Image Processing

- Tracking trackers across the frames is also difficult
- One can use frame coherence, which works as long as the subject moves slowly enough
- One can also use logical coherence, i.e. when walking feet are always at the floor
- One can use also prediction methods: if I know how fast the subject is, I can try to „guess“ the whereabouts of the marker in the next frame
- Occlusion is a further problem: if more markers disappear, it is difficult to know which is which when they reappear
- Also, when markers pass near each other, they might be swapped next frame
- This might generate markers swapping positions
- Sometimes, this can be solved by taking a 3D image (with 2 cameras).
- Other times, human intervention is necessary

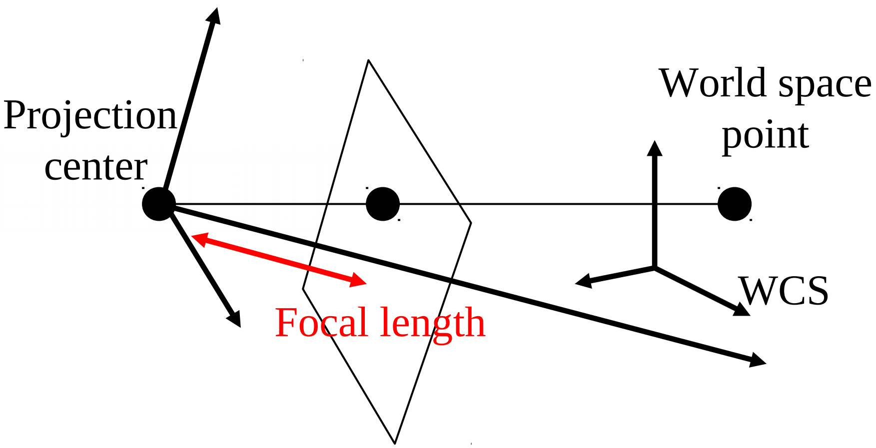
Optical tracking: Image Processing

- There are new markers that flash a unique sequence to avoid tracking errors
- Another problem is given by noise, due to the fact that markers are sticked, not sewed to the actor‘s body, so they swing
- Sampling errors also occur
- Sometimes marker positioning errors are also done

- To deal with noise, the user can condition the data before analyzing it
  - Throw out inconsitent data
  - Filter the rest of data
  - Ideal result: smooth data without throwing away features
  - Weighted average of a certain number of adiacent points can be used to smooth

Optical tracking: Camera calibration

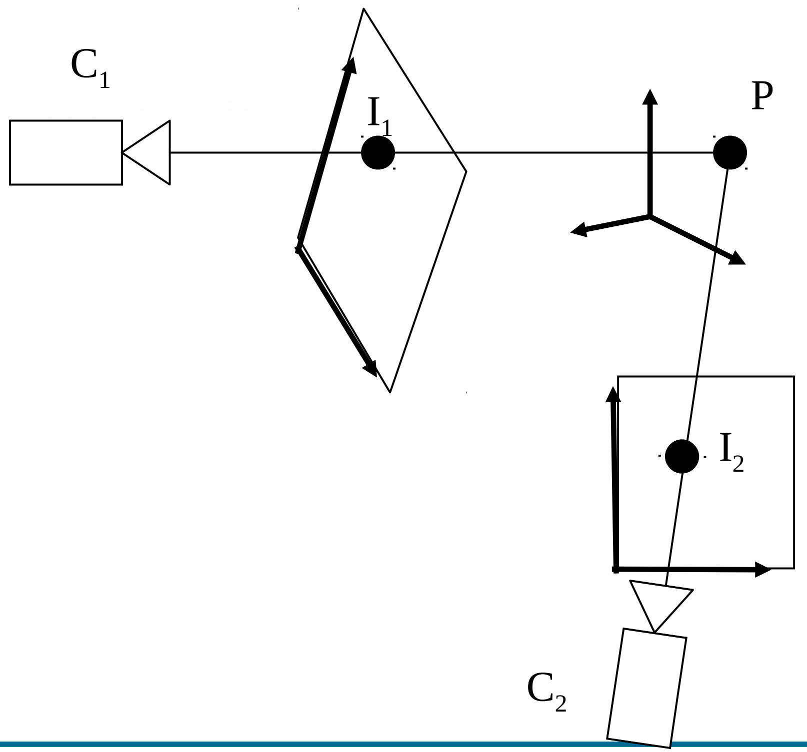
- Before the 3D position of a marker can be reconstructed, one needs to know
  - location and orientation of the cameras in world coords
  - Focal length, image center and aspect ratio have to be known
- The camera system is modelled like in Computer Graphics
- The image of a point is done by projecting a ray from the point to the center of projection
- Calibration is done by recording a number of known points in space



Opt. tracking: position reconstruction

- At least two views are needed to reconstruct 3D
- Since we know I1 and I2, we deduce
  P=C1+k1(I1-C1)
  P=C2+k2(I2-C2),
  thus
  C1+k1(I1-C1)= C2+k2(I2-C2)
  which are 3 equations in 2 variables, and this solvable
- Unfortunately, noise complicates it, because the two straight lines do not necessarily touch

- This can be solved by finding P_1 and P_2 ⊥ to the lines through the other cameras, and computing the midpoint of the segment P_1P_2



Opt. tracking: position reconstruction

- As few as 14 markers can provide some simple tracking of a human figure
- Complete marking sets include 31 markers, including elbow, kneews, chest, hands, toes, ankles, and spine, as well as scapulae and more...

- The more markers one has, the more it is necessary to have more than 2 cameras, so as not to have marker occlusion
- Each marker at each frame needs to be seen by at least two cameras
- A typical system would have 8 cams
- Multiple cams requre some more effort in synchronizing them

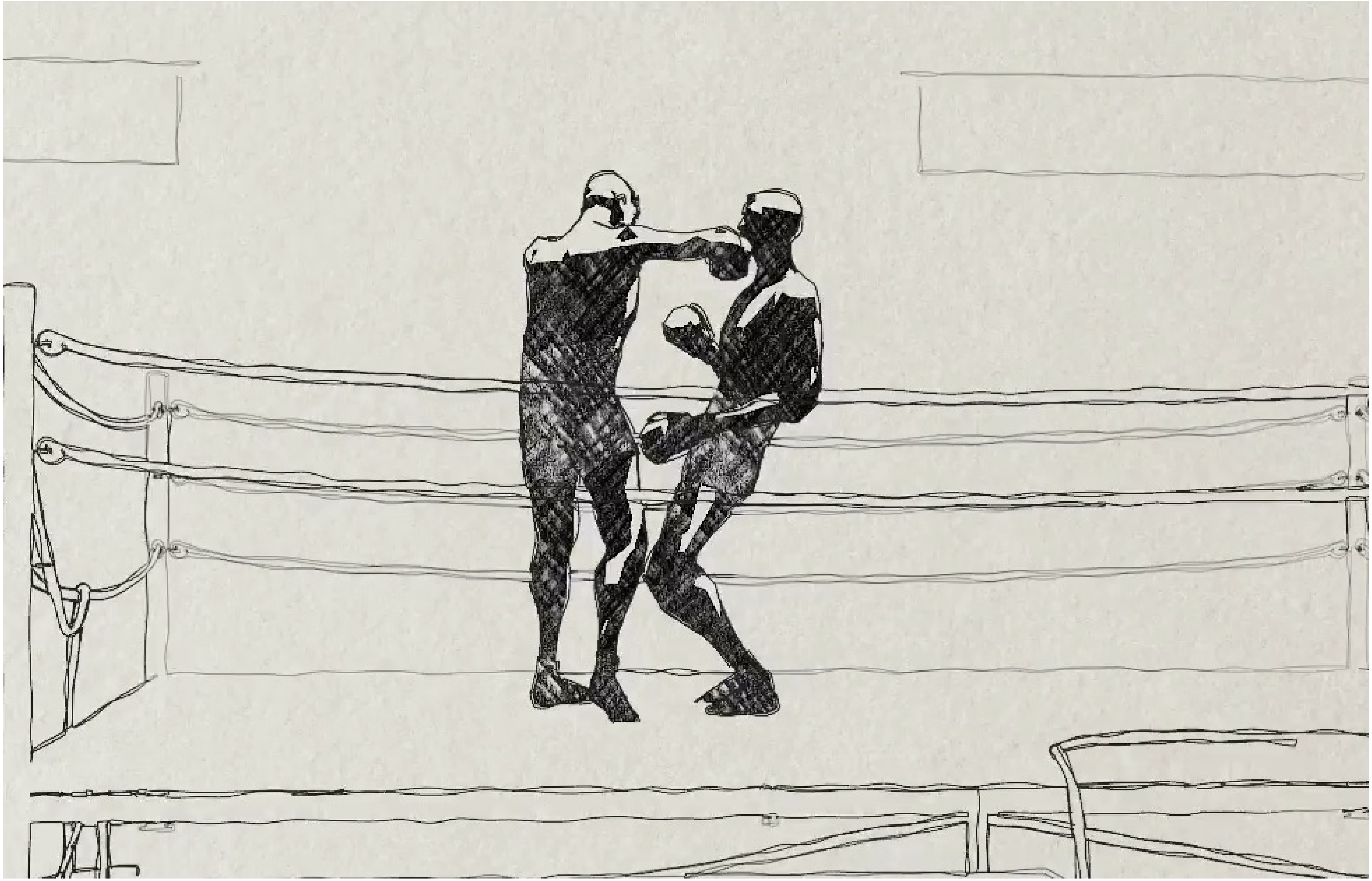
Optical tracking: fitting to skeleton

- The next step is to attach the markers to the skeleton
- One could do it directly, but unfortunately it does not work well, because in general, marker distances are not preserved
- Markers are not exactly on the joints, but on the skin
- One can compensate for that by setting markers at their right positions, but it is still imprecise because the body is elastic
- Another solution is to put two markers on the sides of the joint
- This works well (but doubles complexity), but not for joints which are inaccessible
- Simple geometric calculations lead to deduce the correct joint-marker mutual positions
- Once this is known, the movement can be applied to the skeleton
- Watch out for imprecisions of the data obtained, that can lead to visible artifacts (avoid floor penetration)

IR tracking with fitting to skeleton

- Recently, cheap devices have been introduced that are capable of tracking (Microsoft Kinect)
- Based on IR depth maps and laser
- They deliver quite precise data but at relatively small distance (1.5-3m), area covered $6m^2$.
- Software allows person recognition (even more than one person)
  - Deliver face , gestures, voices

A Kinect Example

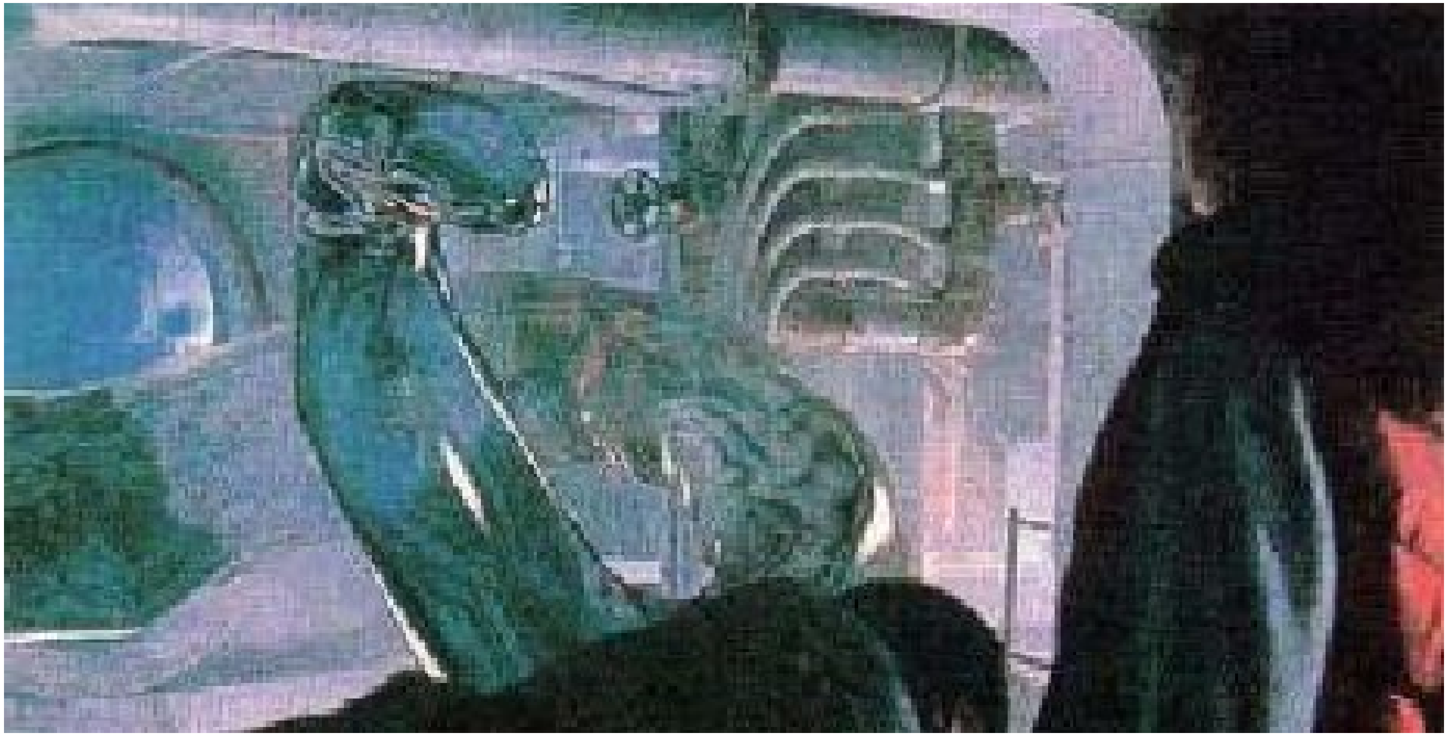


Future work and review

- Recently, work has been done on fitting motion data to different skeletons (taller, smaller, etc)
- This brings the hope of getting parametrized libraries

- For sure, motion capture is an interesting approach
- However, it will never replace a creative animator, maybe only support him
- Current research tries to recognize parts of the body from videos without markers
- However, we are not yet to the point that this is usable

End



+++ Ende - The end - Finis - Fin - Fine +++ Ende - The end - Finis - Fin - Fine +++