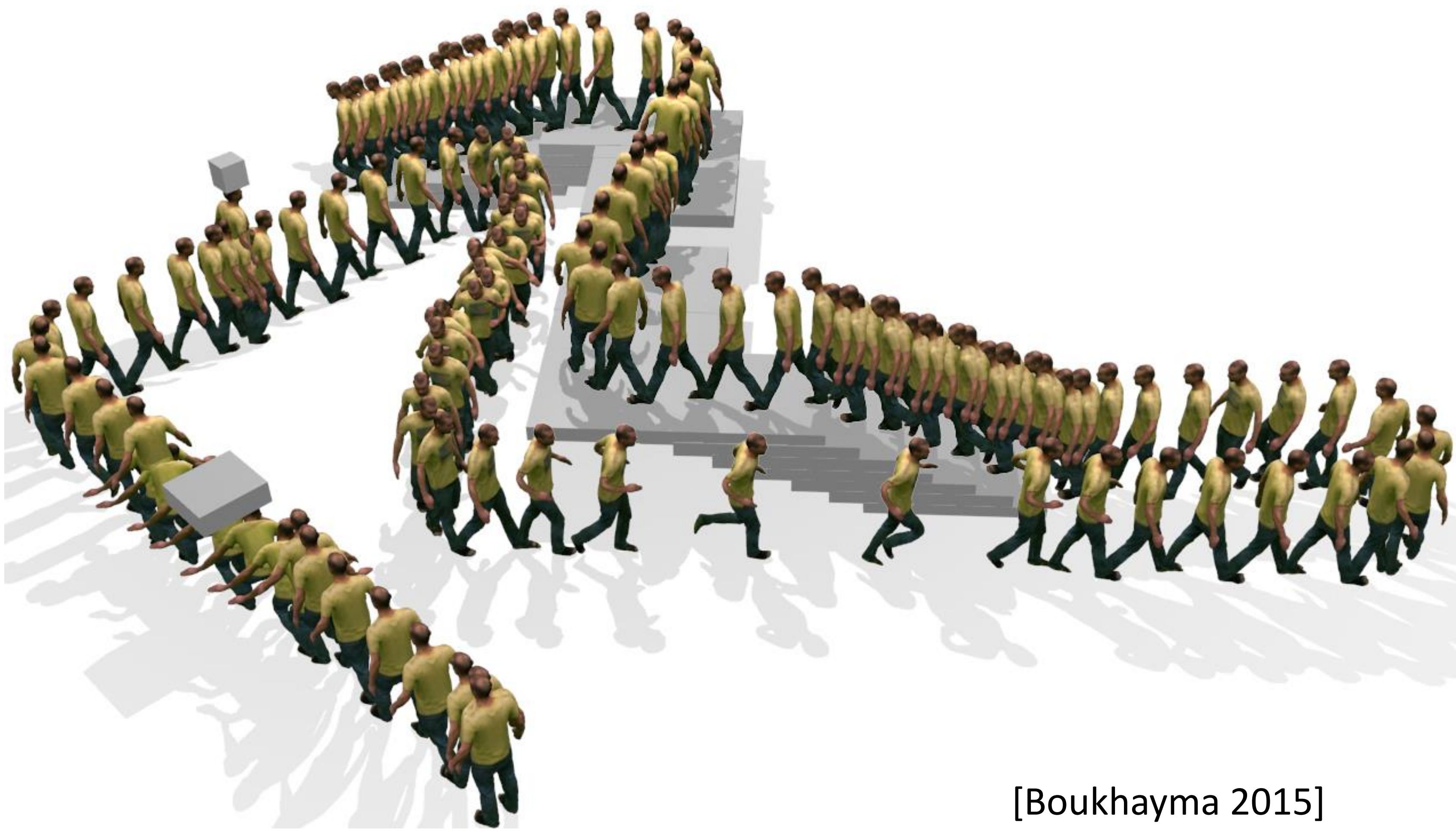
Advanced Animation



[Boukhayma 2015]

Topics

1. Advanced & non-rigid capture techniques
2. Data driven content reanimation
3. Layered animation models for complex scenes

Advanced & non-rigid capture techniques

Remember: Motion capture

- Capture animation based on actor movements
- Traditionally based on markers
- Traditionally used to infer kinematic bone movement

Limitations

- Density: going beyond bones
- Combining different motions
- Adapt to different morphologies

Difficulties with traditional pipelines

- Manually define animation trajectories
- Traditional capture helps but still requires manual intervention
- Animating non rigid objects is still tedious (faces, clothes…)
- Requires expertise and time, expensive

Source: Felix Ferrand

Automatic dense capture

Main ideas

- Recover 3D motions with little or no manual input
- Densely observe real shapes for non-rigid effects
- Solve an alignment problem, between
  1. A digital 3D deformable model
  2. Real shape surface trajectories observed

Challenges

- How to define a proper deformable model?

- How to match trajectories between real and digital model?
  - Huge search space : model vertices vs sensor data

- How to properly constrain the motion?
  - Real shapes don't move randomly, can we use this?

Shape alignment principles

- Design or acquire a 3D shape model
  - See previous courses
- Use a low dimensional motion parameterization
  - Element subdivision whose positions parameterizes the motion, or shape subspace model
- Create/identify matching handles
  - Other subdivision, not necessarily same as above
  - Can be landmarks, vertices…

Solving the shape alignment

Algorithm template

0. Initialize deformation elements close to observed
1. Match model handles to their real counterparts
2. Update parameters of deformation to minimize handle distances
3. Iterate 1 & 2 until convergence, for each new frame

Note: can be seen as alternating minimization problem

argmin E with E = Em + Ed

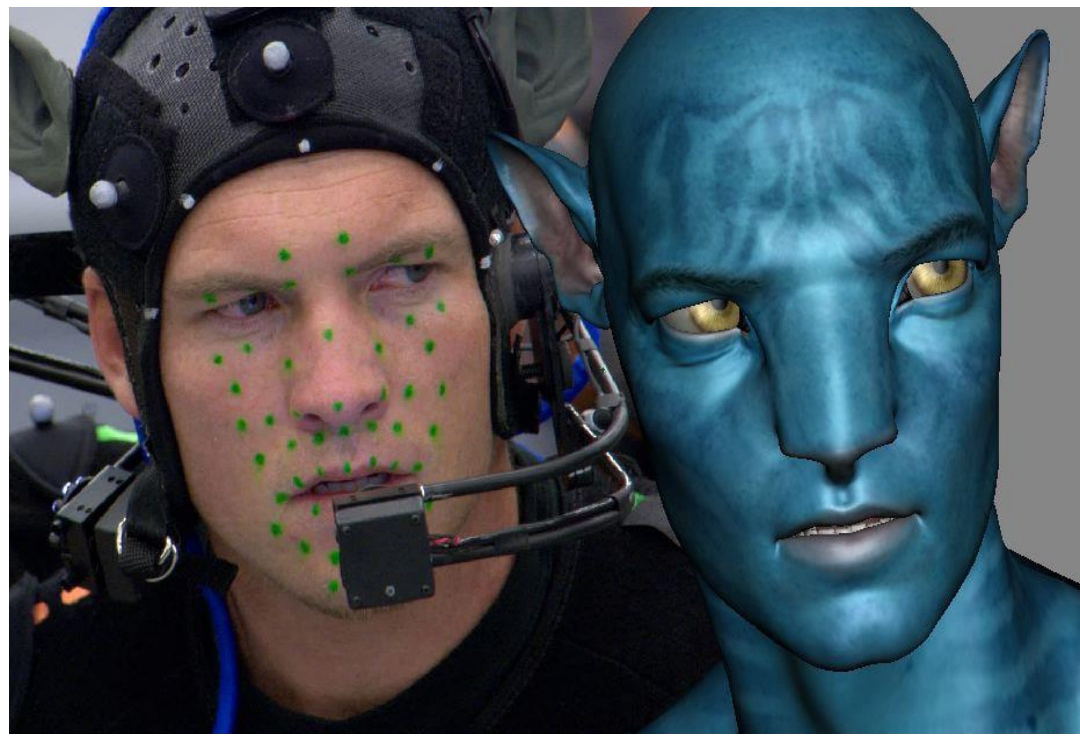
Matching energy. Deformation energy.

Example: face capture

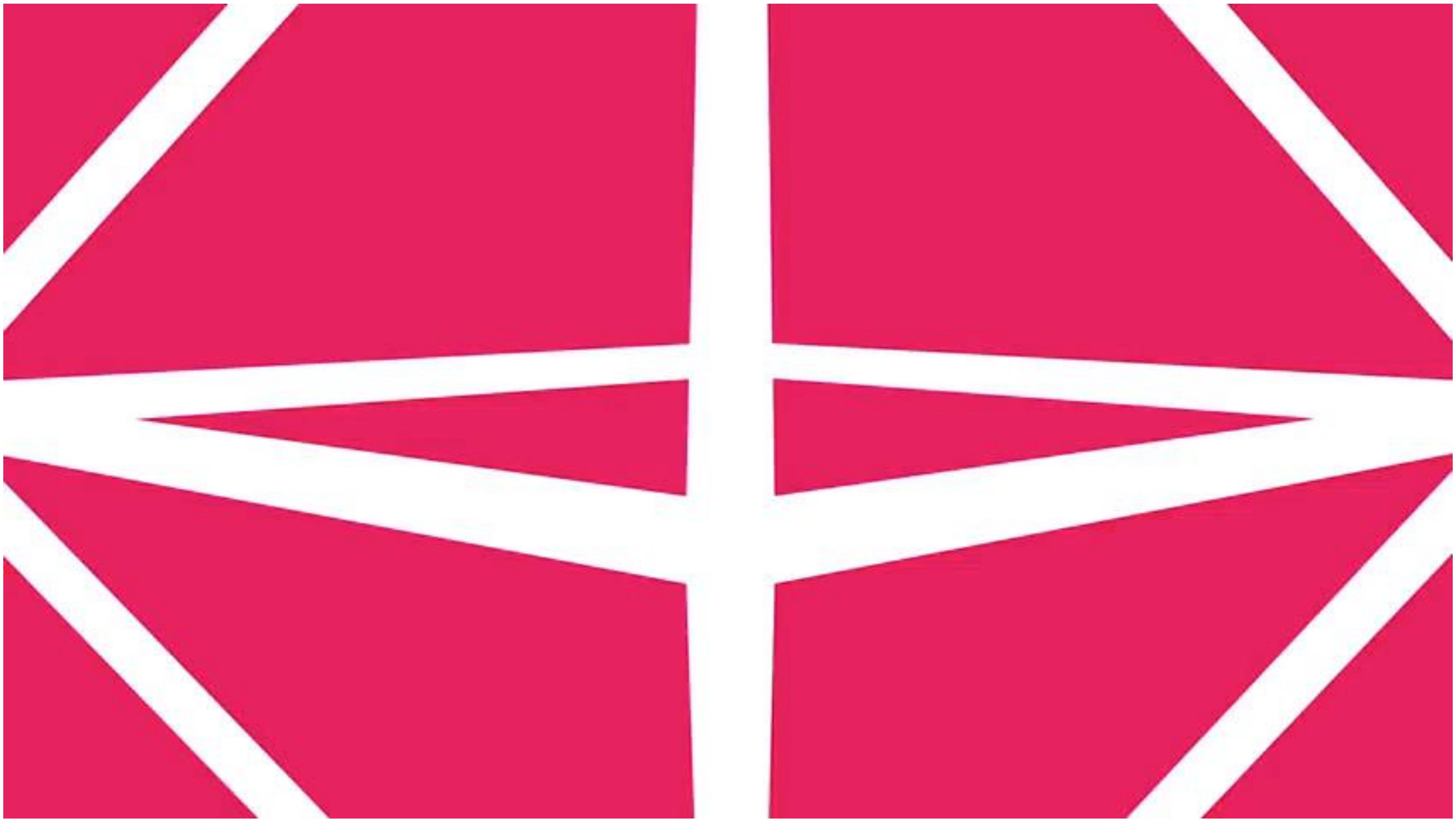
- Observations: manually placed face markers
- Shape model: head and face mesh model
- Deformation model: vertex keys, as rigid as possible energy …
- Handles: pre-identified face landmarks

Limitations

- Marker occlusion, camera placement
- Manual post-processing usually needed



Face capture illustration

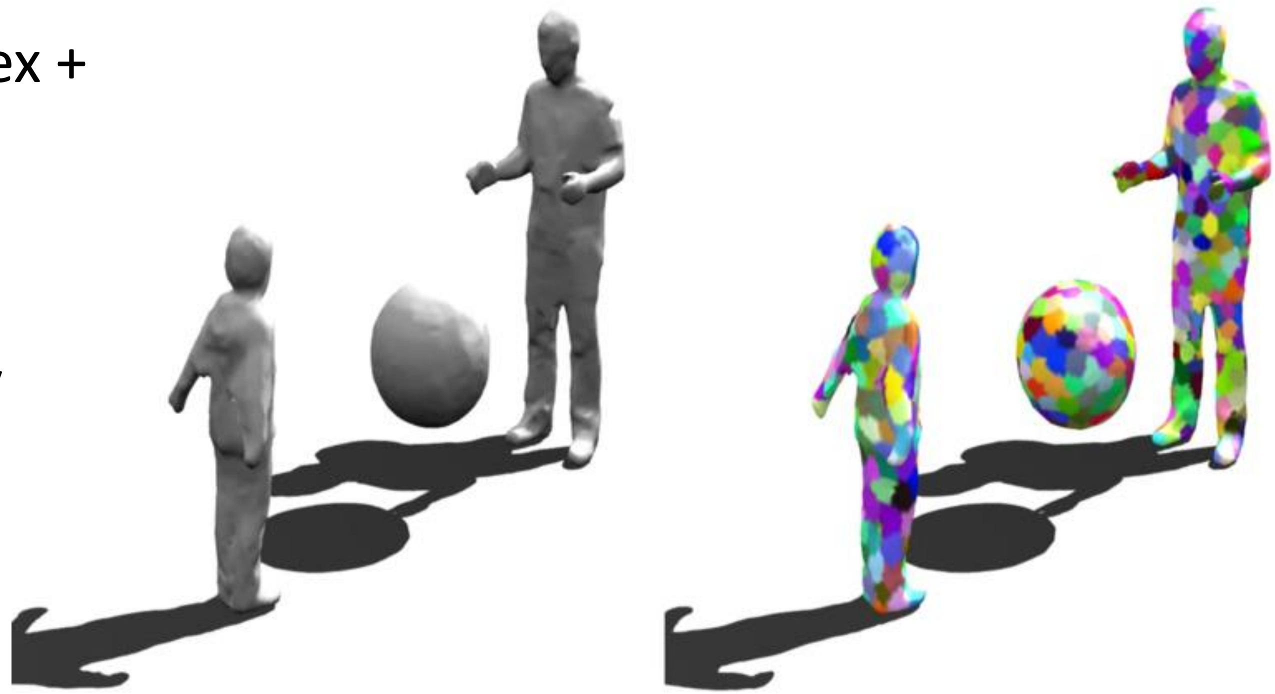


Example: patch-based body deformation capture

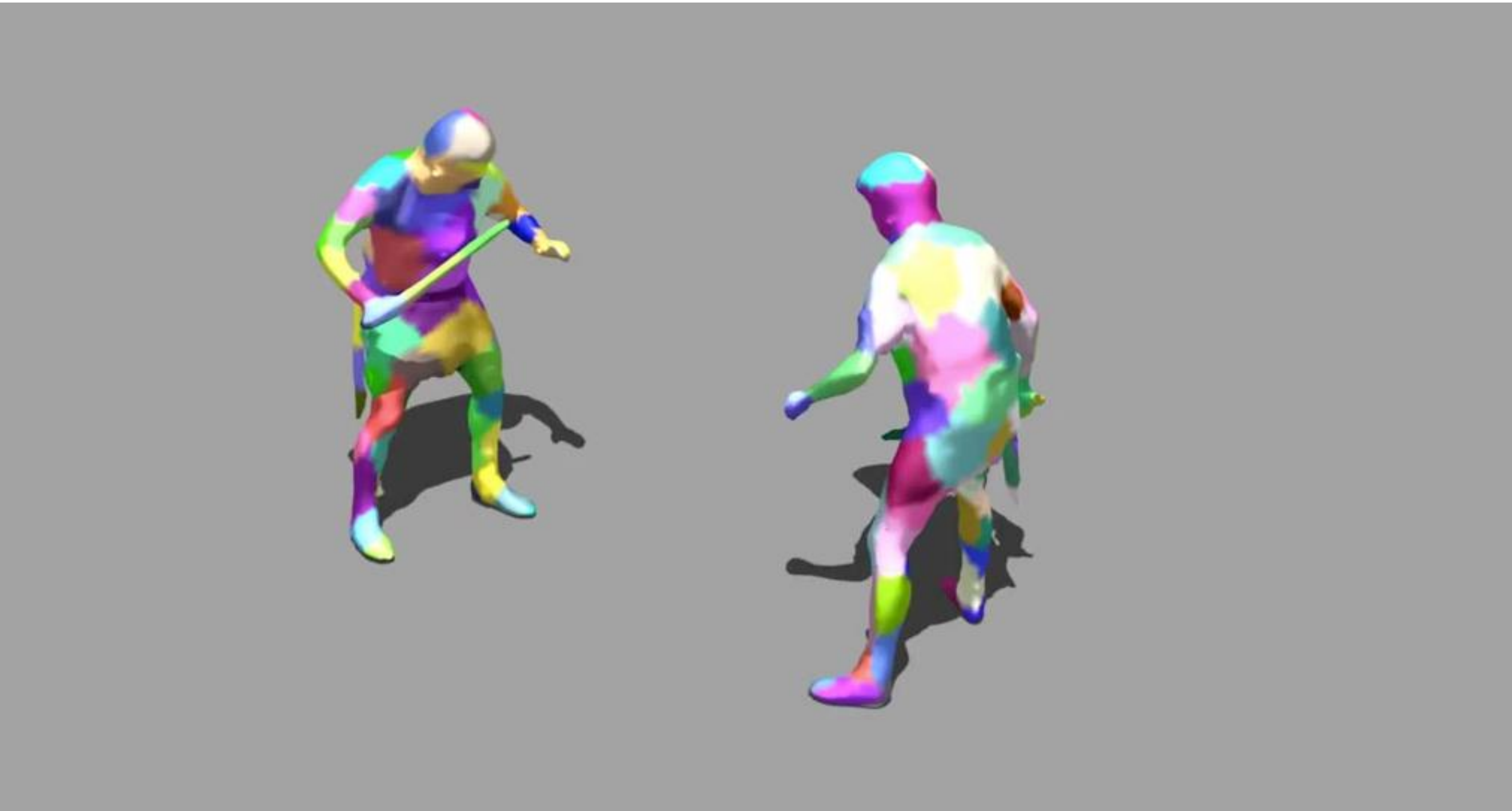
- Observations: silhouettes from multiple cameras
- Shape model: body mesh model
- Deformation model: surface patches with elastic tension
- Handles: surface vertex + silhouette proximity

Limitations

- Geometric fitting only
- Exercise: other limitations?



Patch-based capture example

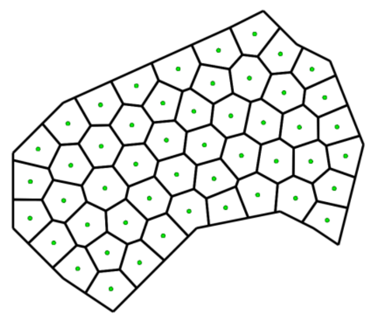


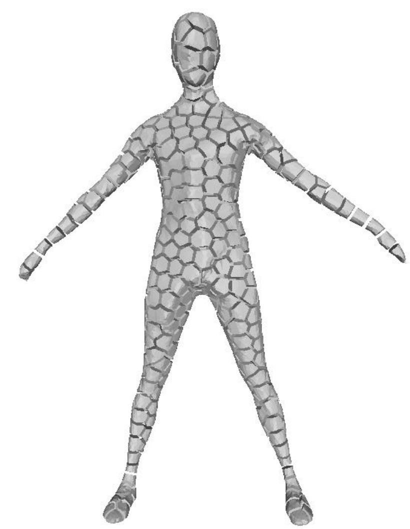
Examples: volume-based deformable shape capture

- Observations: silhouettes from multiple cameras
- Shape model: **volumetric** body mesh model based on CVTs
- Deformation model: **volumetric** patches with elastic tension
- Handles: surface vertex + silhouette proximity

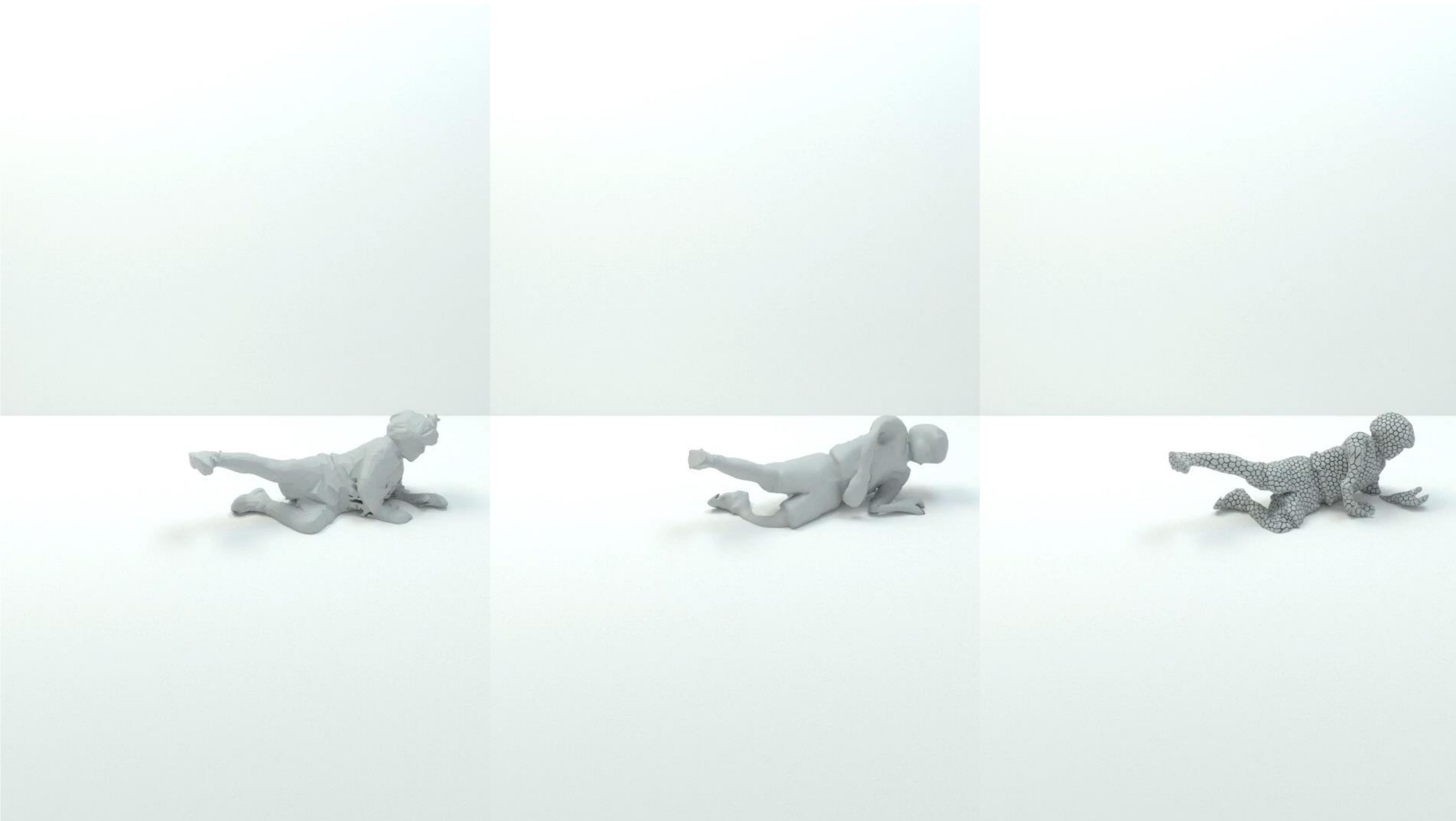
Limitations

- Geometric fitting only
- Exercise: other limitations?





Volume-based deformable capture example



Data driven content reanimation

Problems

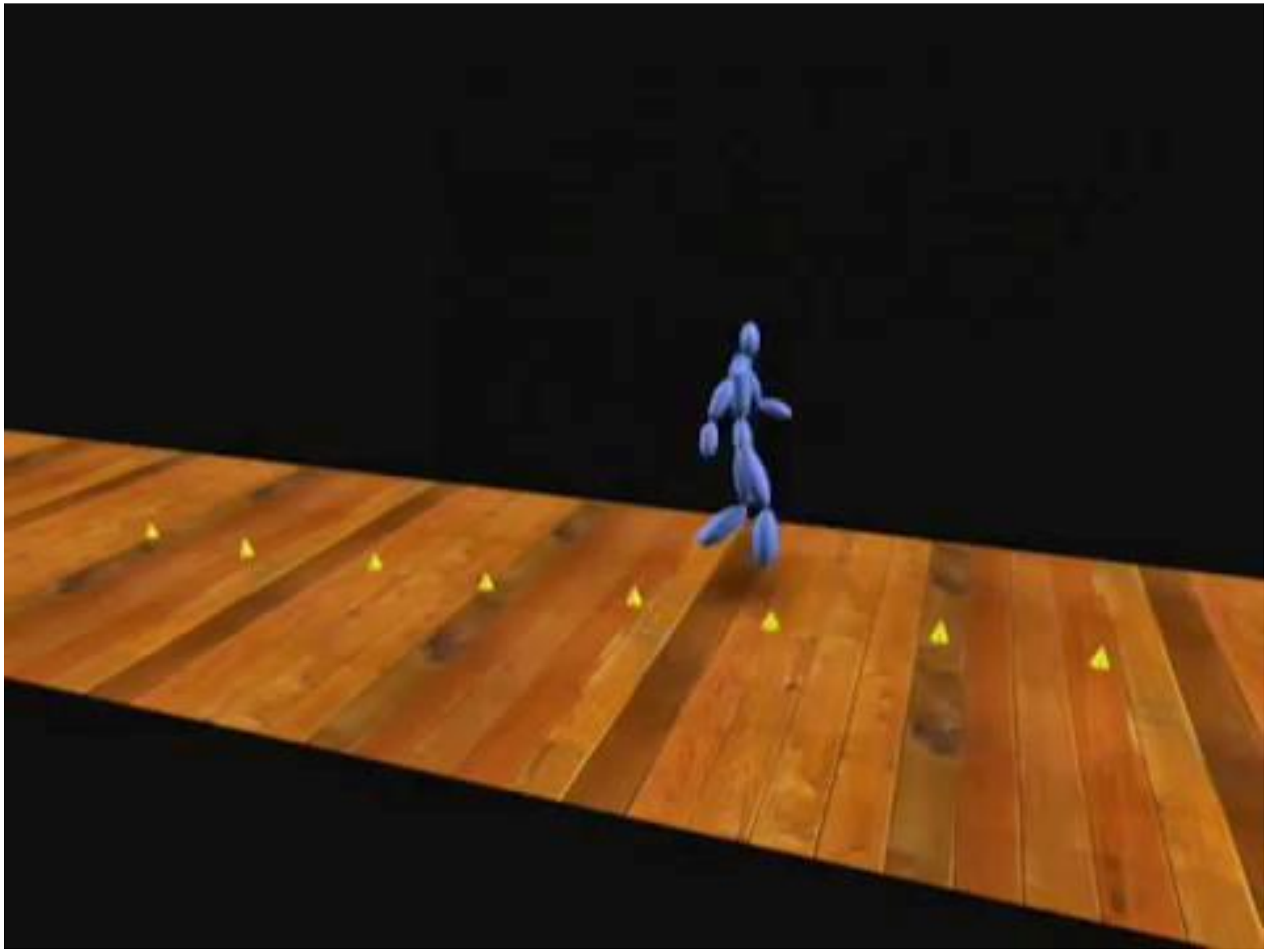
Capture and character animation don’t scale well

- Adapt capture to different morphology of virtual character
- Abstract control of animation with many degrees of freedom
- Generate large corpus of data

Can we automate these instead of all manual adaptations?

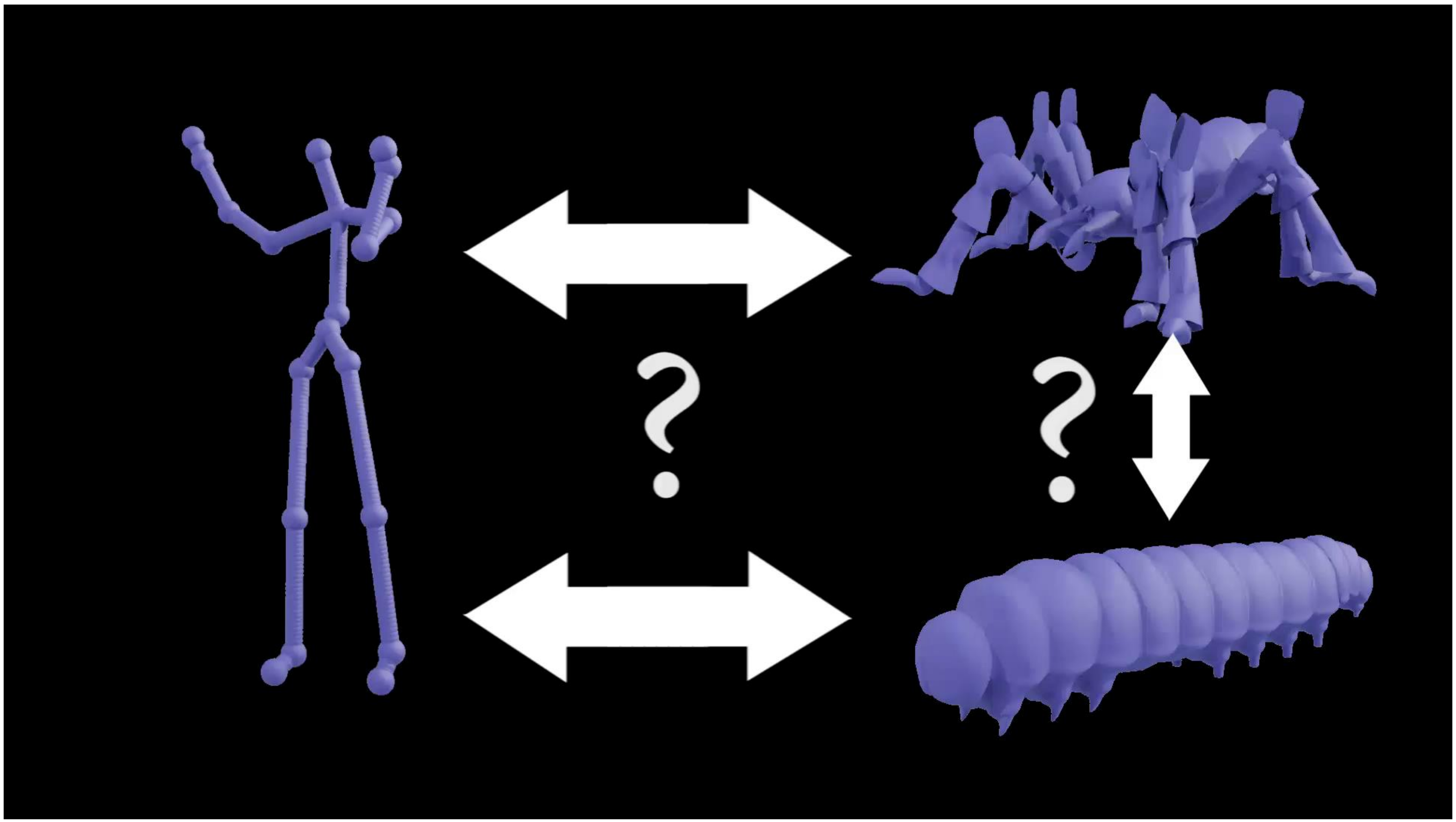
Different morphologies: retargeting

- Principle : preserve angular information of capture and bone lengths of target model [Gleicher 1998]



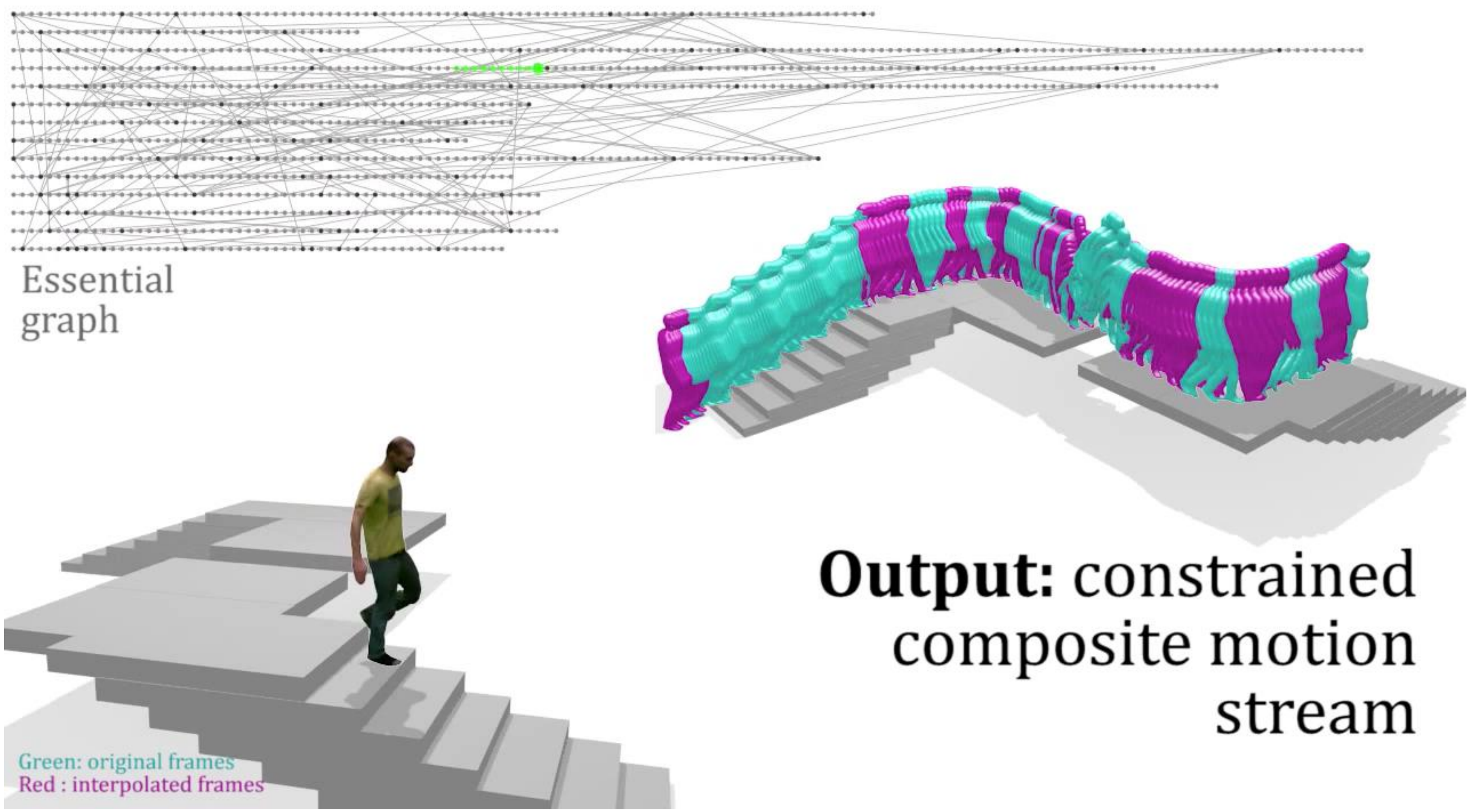
Animation control: Motion mapping

- Principle : track and detect user movement, remap it to character degrees of freedom [Rhodin 2014]



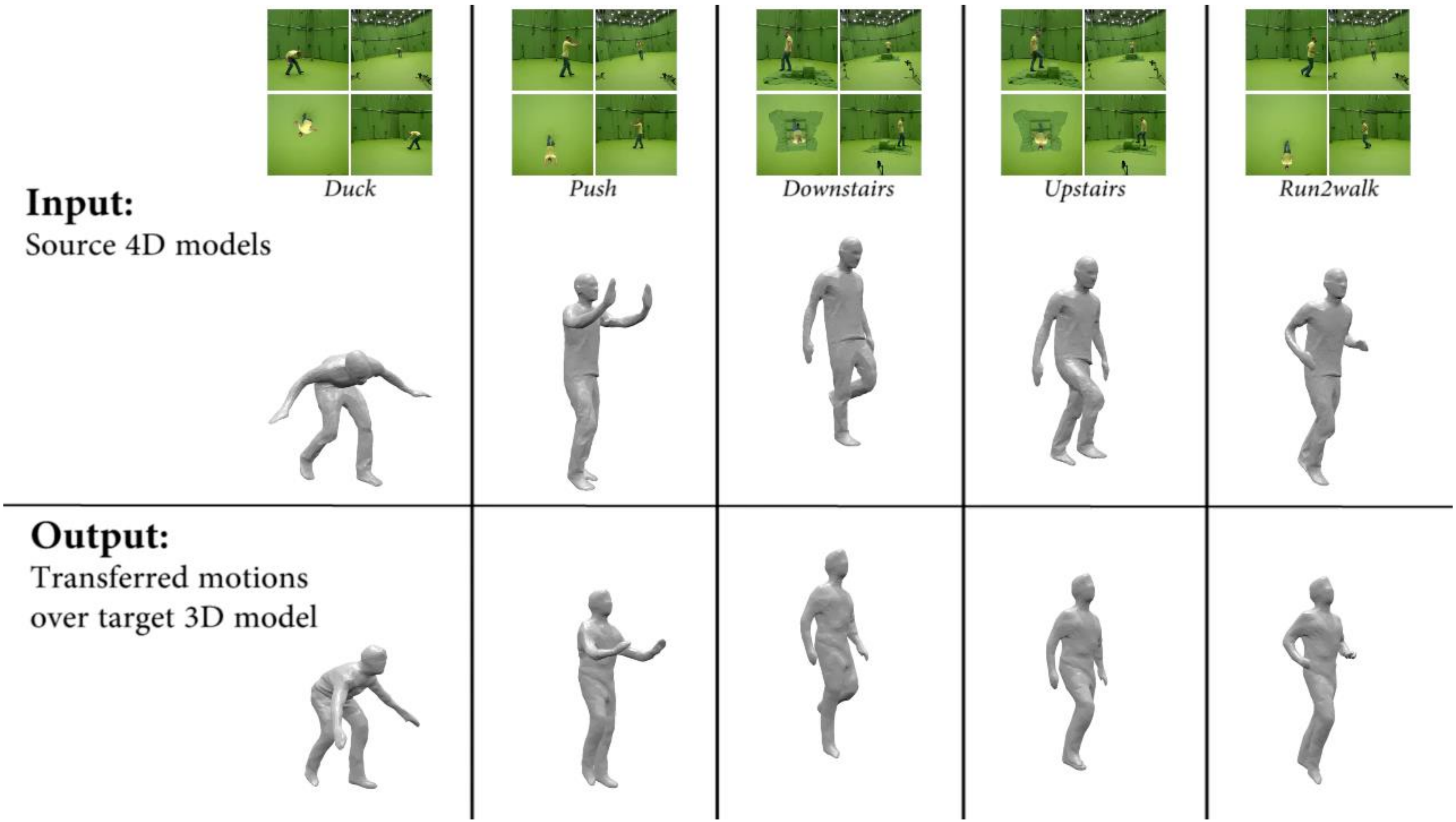
Generate large corpus: motion graphs

- Principle : build smooth composite sequence from several input sequences of a real captrued character [Boukhayma15]



Generate large corpus: Capture transfer

- Principle : transfer corpus of captures to a different capture with some matching sequences, based on direct sequence regression [Boukhayma16]



Animating Complex Scenes

Animating Complex Scenes

- Grass blowing in the wind, interacting with the feet
- Trees, clouds…
- Characters

Procedural model?

Descriptive animation?

Geometry / physics?

Animating Complex Scenes

Solution : « layered model»

Successive animation layers

each one models a specific feature

- Eases conception & control
- Best model for each layer
- Possible retro-action

Layered models

General methodology

1. Observe & identify the sub-phenomena to reproduce
2. Represent them independently
   - Choose the best model for each feature
     Physics, kinematics, geometry, textures…
   - Use different time & space sampling if necessary
3. Couple them together

Animation loop

Successive update of each layer + possible retroaction

layered models: case studies
1. Natural phenomena

Examples

- Grass blowing in the wind
- Ocean Waves

Layered models for Natural Phenomona
Prairies blowing in the wind

View of a walker in real-time?

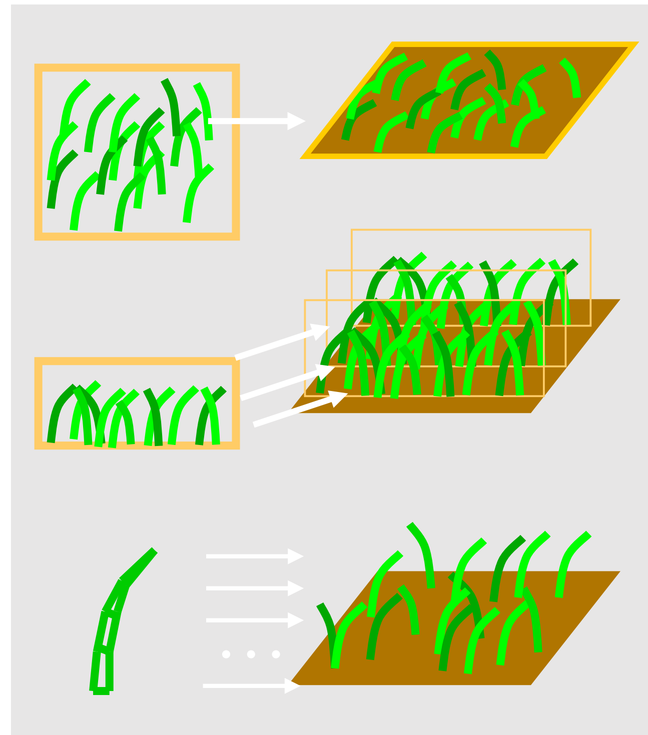
Difficulties

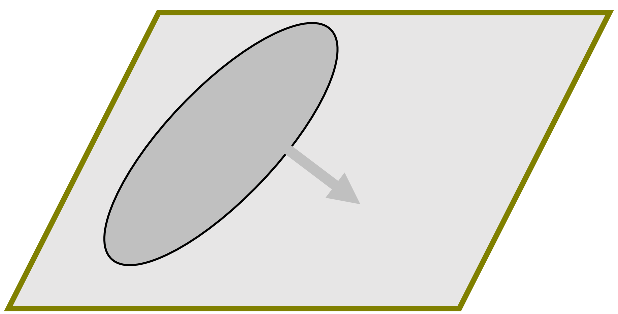
- Number of blades of grass
  - Rendering: aliasing problems
- Control of the wind
  - Breeze, gusp of wind, wirld wind
- Plausible action

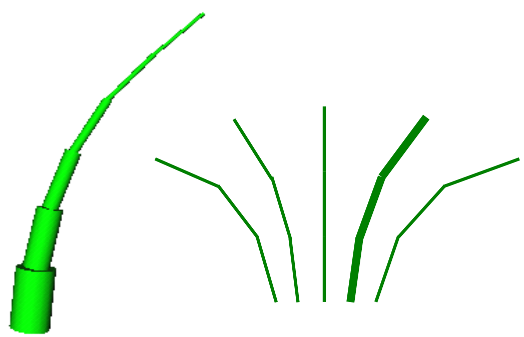
Layered models for Natural Phenomona
Prairies blowing in the wind

Sub-models

- Grass: 3 levels of detail
- Wind model : mask + action
  - Breeze, gusp of wind, wirld wind
- Receever : blade of grass
  - deformations : pre-simulation



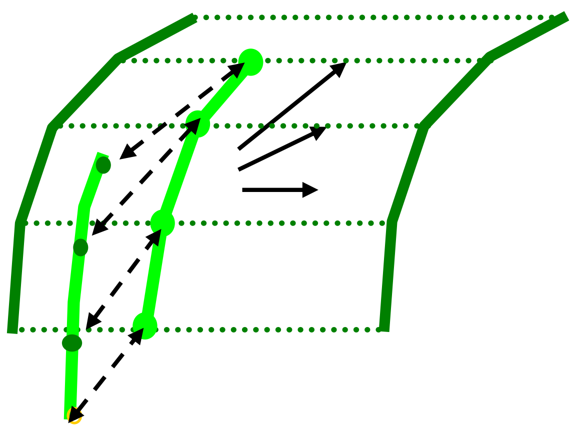


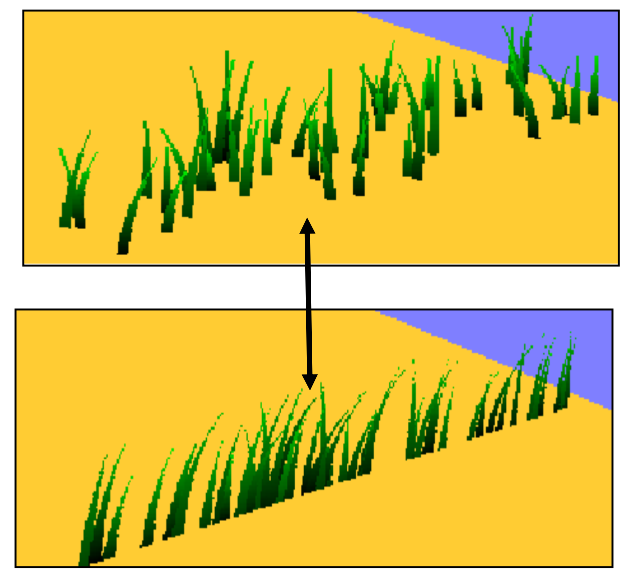


Layered models for Natural Phenomona
Prairies blowing in the wind

Transitions between levels of details

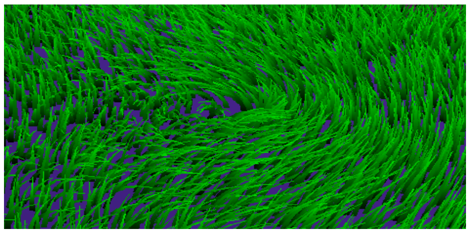
- 3D blades of grass / texture 2D1/2
- texture 2D1/2 / texture

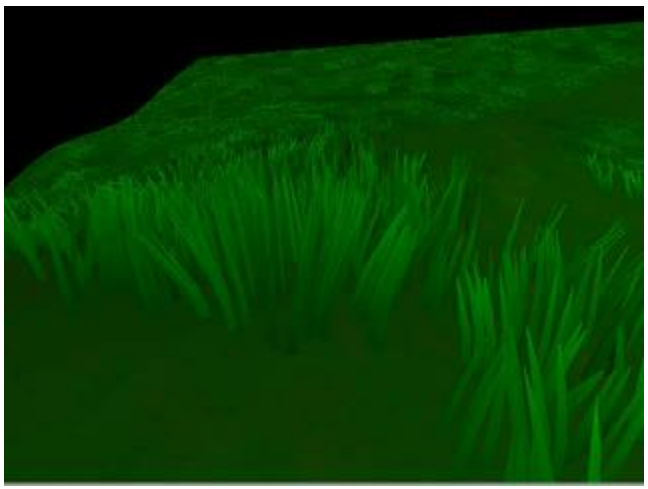




Layered models for Natural Phenomona
Prairies blowing in the wind

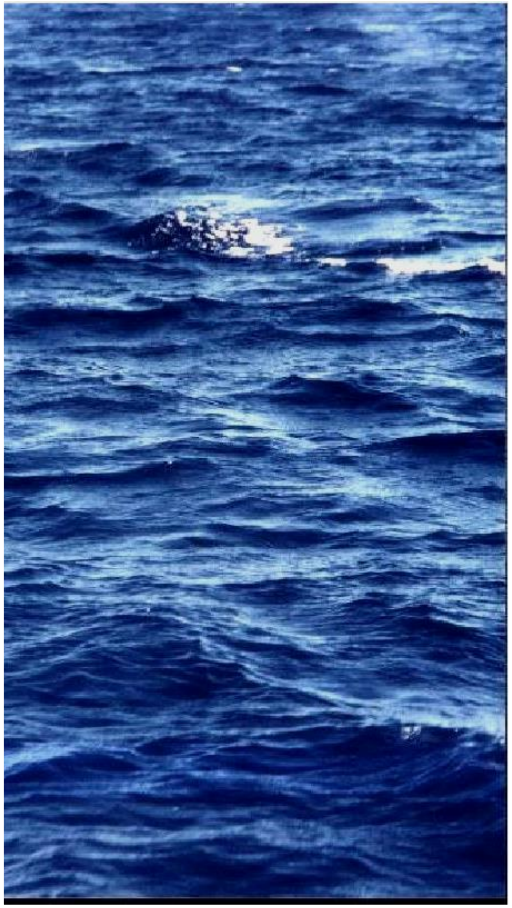
[Perbet Faure Cani 02]





Animating Ocean Waves

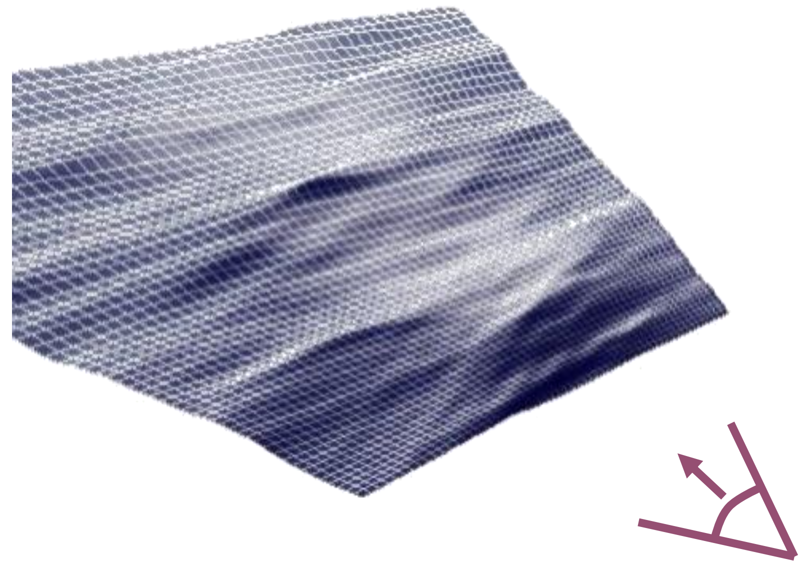
- Aims
  - Tunable compromise realism/efficiency
  - Camera motion
  - Unbounded ocean
- Difficulties
  - Complex deformations
  - Close to far view
  - Aliasing

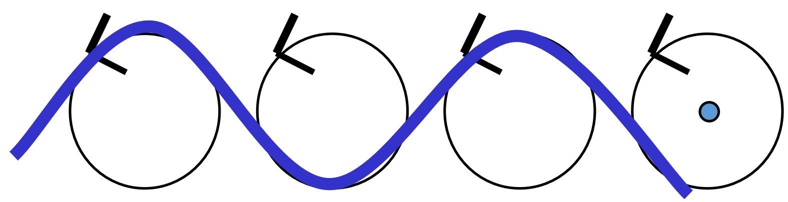


Animating Ocean Waves

Sub-models

- Receivers
  - Sampled surface
  - Projection of screen pixels
- Wave trains
  - mask + action

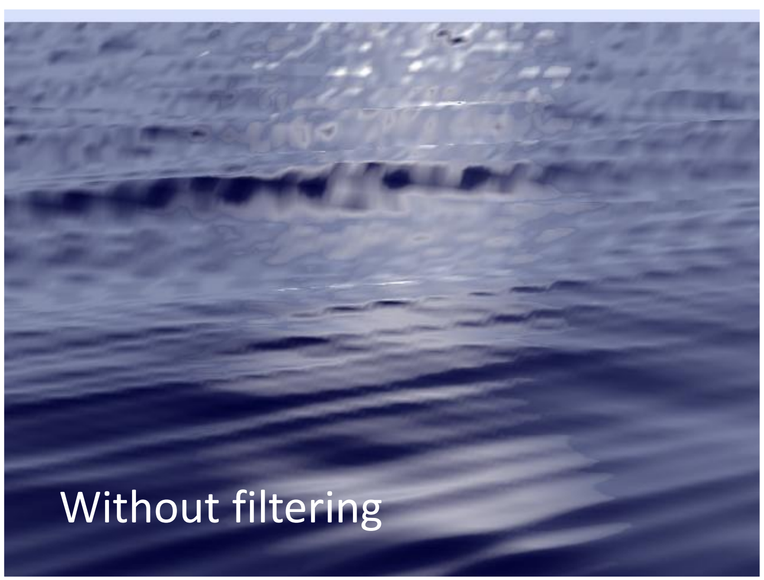


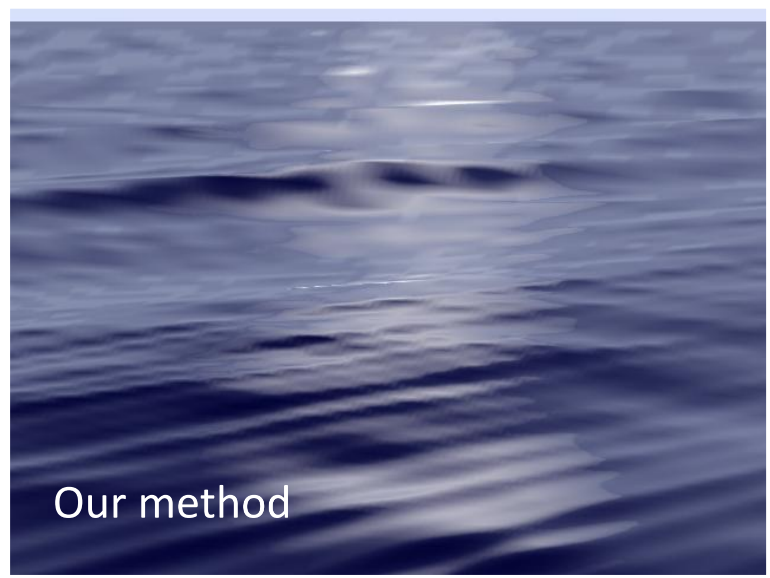


Animating Ocean Waves

Animation : Levels of detail

- Filtering wave trains with the distance
  - Increases efficiency and reduces aliasing





Animating Ocean Waves

[Hinsinger Neyret Cani, SCA'02]

Case study 2: animated characters

Need of different layers for

1. Brain, decision taking
2. Moving the skeleton (walking, gestu
3. Deforming flesh & skin
4. Hair
5. Clothing

Exo: *Which models would you use?*

Is retro-action necessary?

Layer 1: Behavioral model
(brain, decision taking)

Example: crowd animation

Particle systems

- Attraction towards an objective
- Repulsive obstacles
- Avoid inter-collisions (fluids)

Techniques from artificial intelligence (AI)

- Individual behavior : rules, emotions, personality
- Social behavior for crowds

Layer 2 : animating the skeleton

From the behavioral model

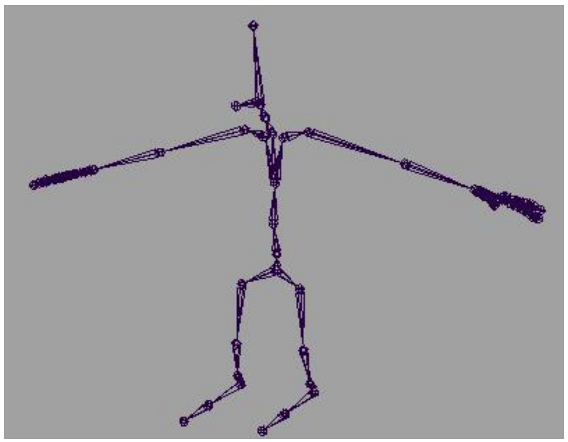
1. Coordinate the different actions (finite automata)
2. Call elementary motions

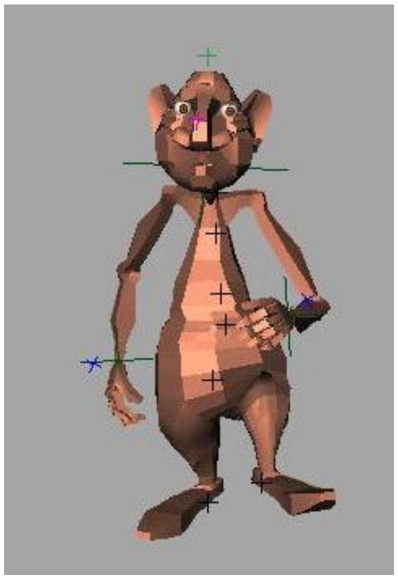
Choose a model for elementary motions

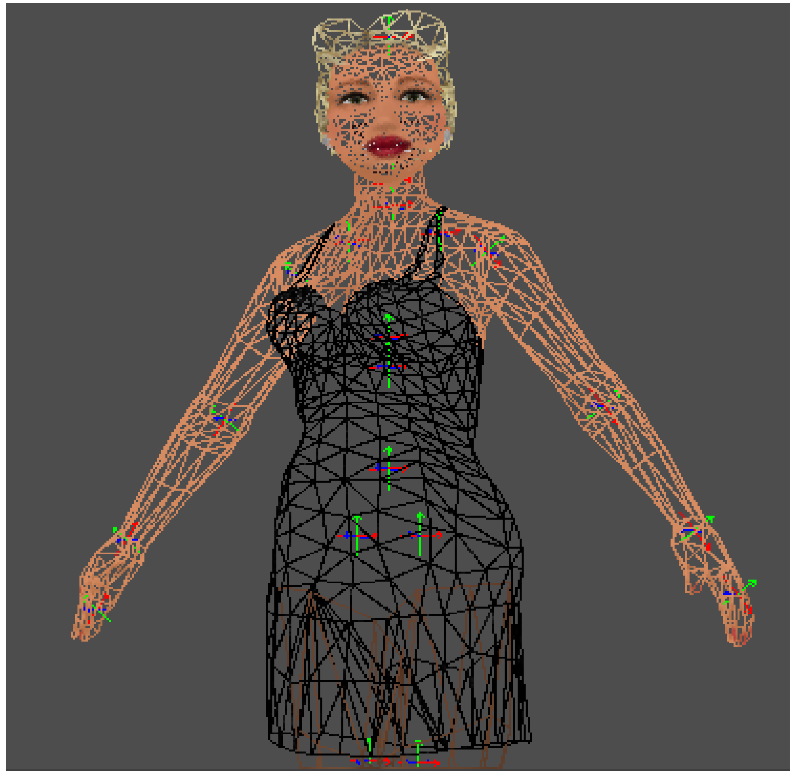
- Descriptive methods
  - Direct and inverse kinematics
  - **Motion capture**
- Procedural models
  - Physically-based animation ***+ control***

3. Flesh & skin deformation
Smooth skinning

- Single mesh
- Deformed by the skeleton
  (hierarchy of joints)







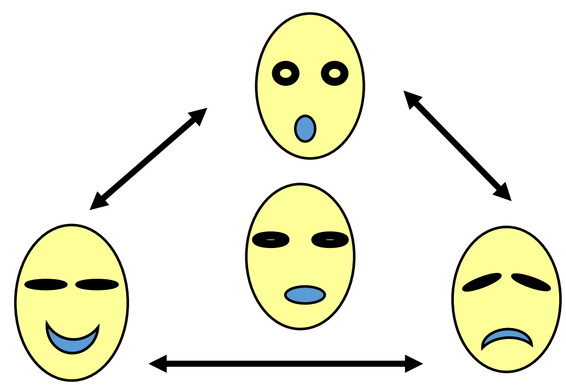
3. Flesh & skin deformation
Key frames vs Blend Shapes

Example of an animated face

- Key frames = Temporal interpolation
  - Model and store all successive key- faces
- Blend shapes = Multi-target interpolation
  - Model a few « extreme faces » from a « neutral face »
  - Animate a trajectory in this space

  For each mesh point,

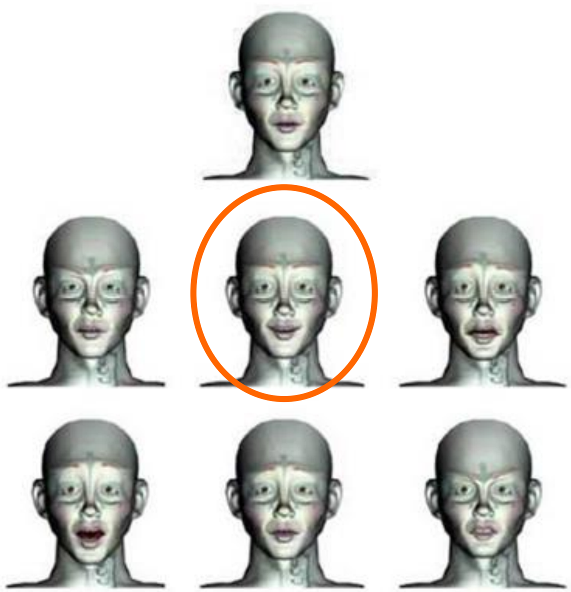
  compute successive barycenters on the fly

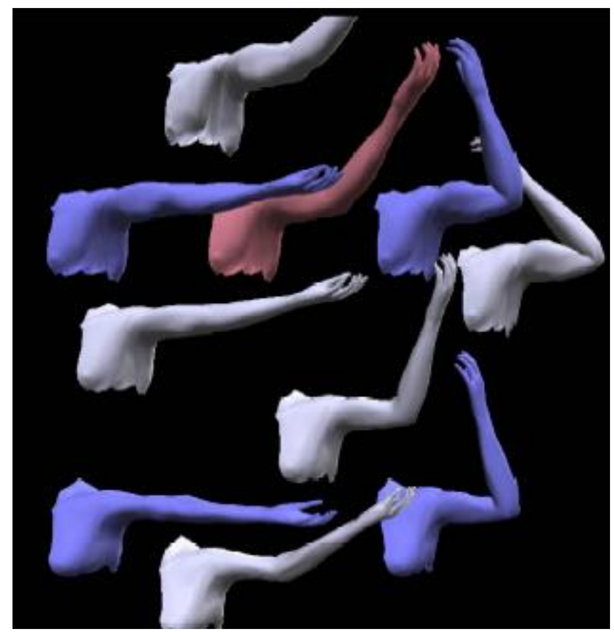


3. Flesh & skin deformation
Multi-target interpolation

Advantages

- Fast interpolation
- No need to model something repetitive



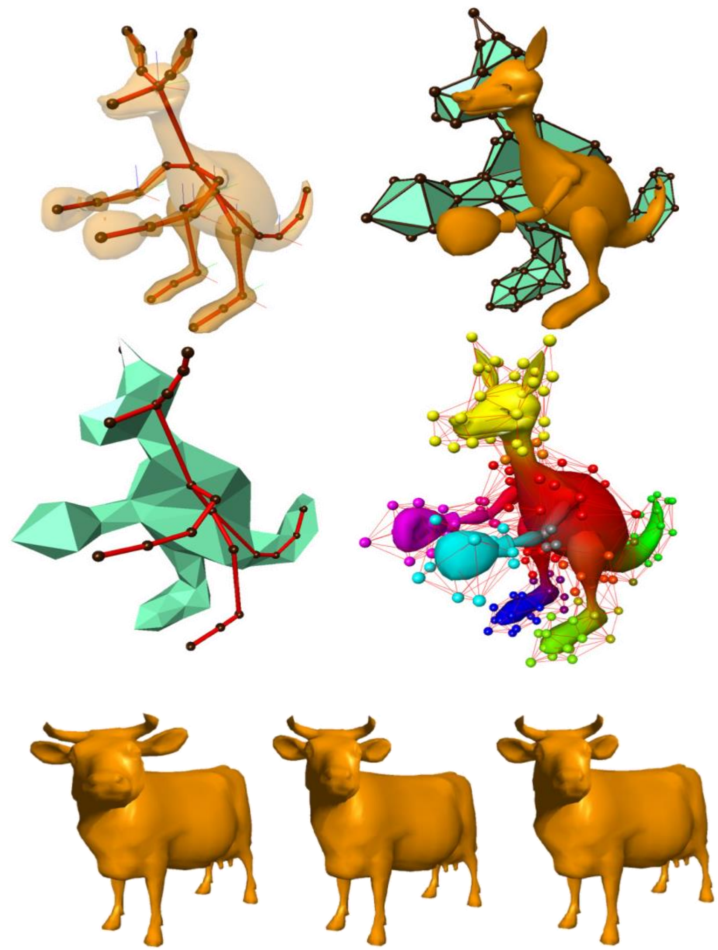


3. Flesh & skin deformation
Adding dynamics to the flesh

Using finite elements

[Capell et al. SIGGRAPH 03]

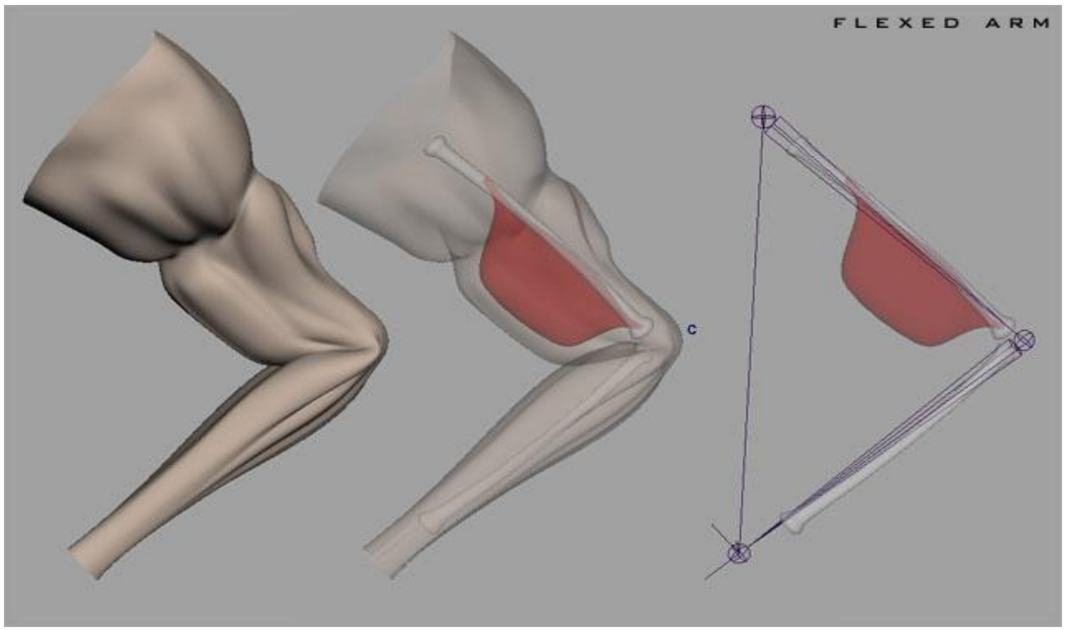
- Associate each cell with a bone
- Linear elasticity for local models
- Constraints to paste cells together



3. Flesh & skin deformation
Anatomical simulation

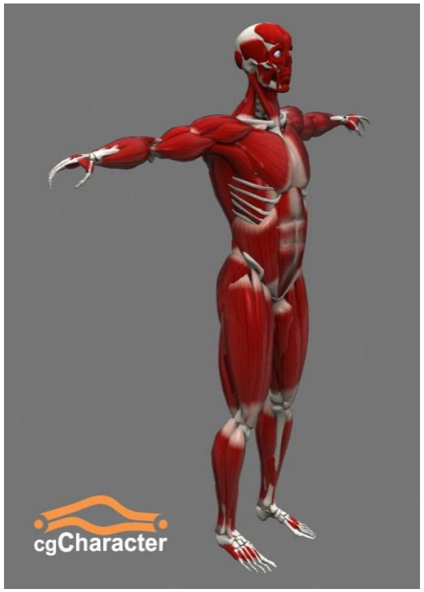
- Realistic model for each layer

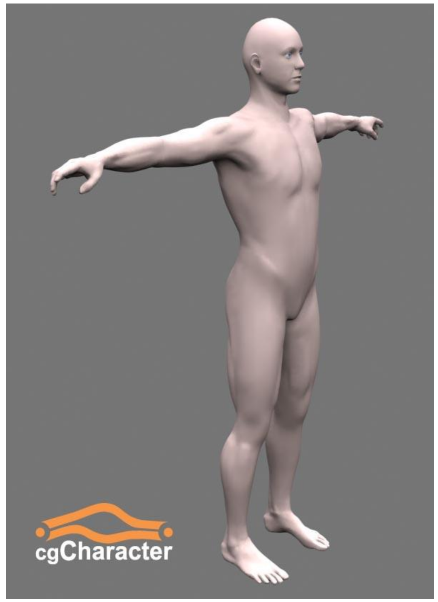
  skeleton, flesh, skin



3. Flesh & skin deformation
Anatomical simulation

- Advantage : realism, possibility to simulate muscles
- Drawback : computational time!





4. Clothes and hair

Physically-based models

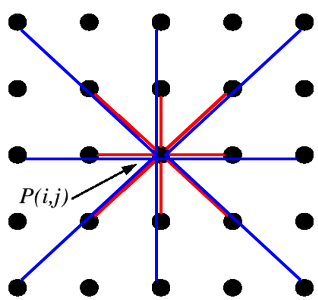
1. Difficulties for clothes
   - Collisions between thin objects
   - Non-extensible: should fold!
   - Numerical integration with stiff springs?
2. Difficulties for hair
   - 100 000 strands

   Exploit spatial coherency!

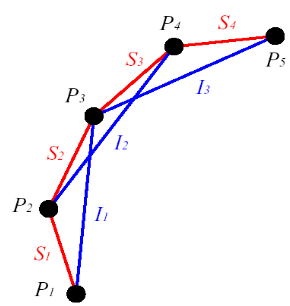
4. Clothes
Ease formation of folds

[Choi and Ko 02] Stable but responsive cloth

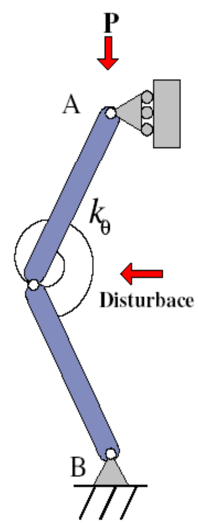
- Rotation when compression force in the plane of the cloth



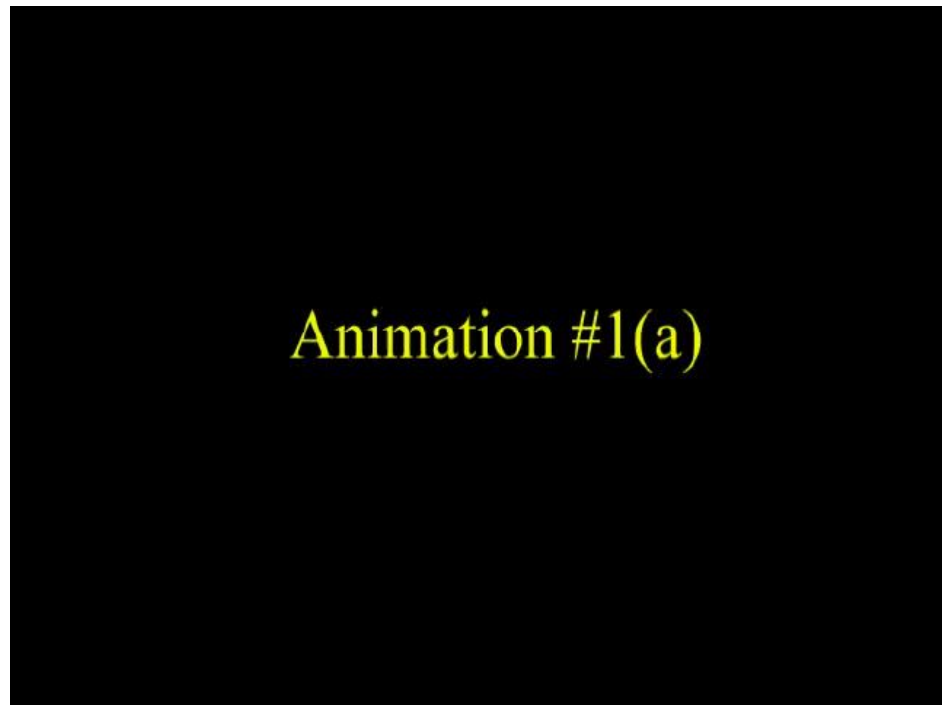
(a) all the connections for $P(i, j)$



(b) connections among neighboring particles in a particular direction



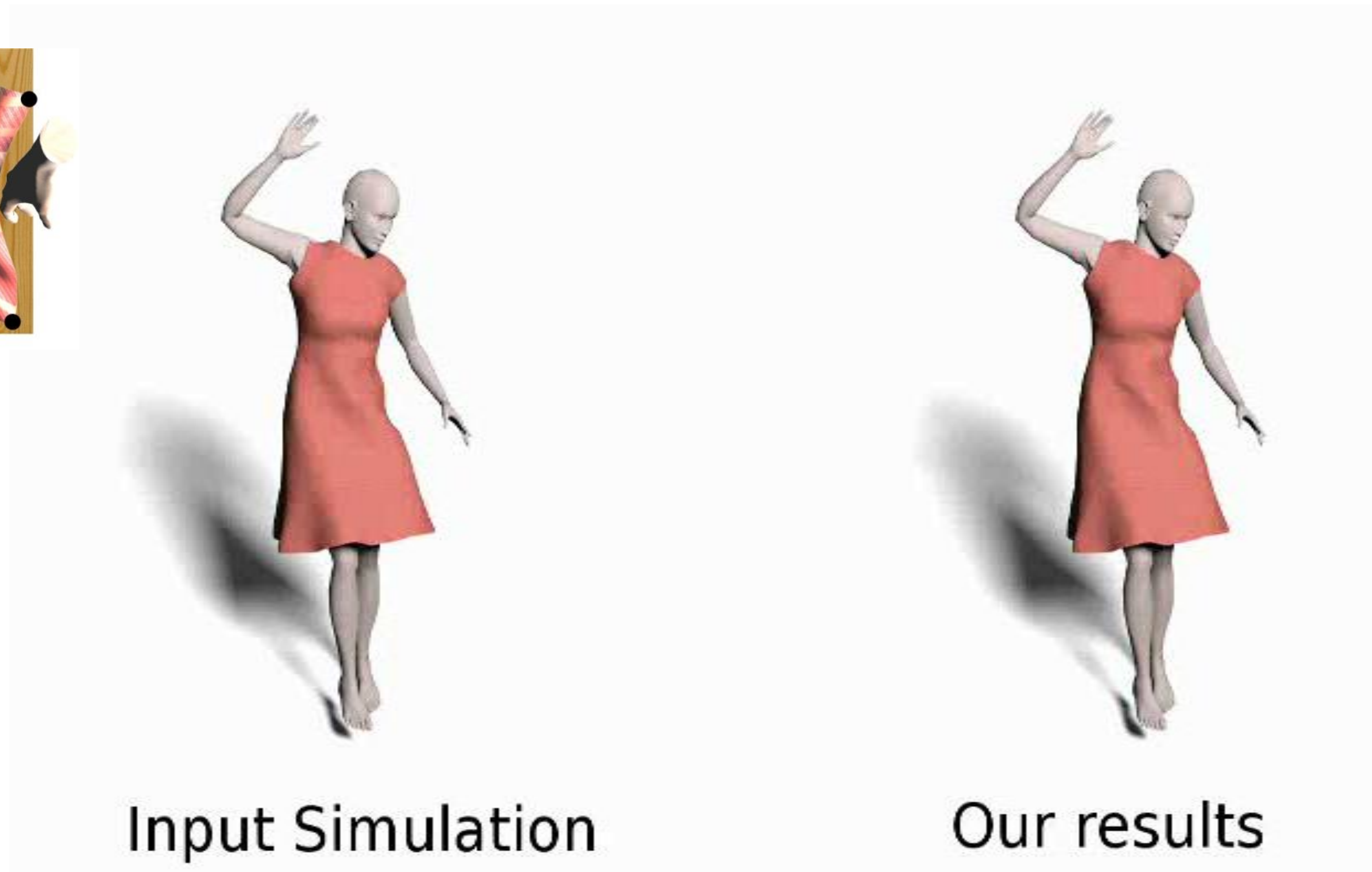
(c)



Layered model for clothes

[Rohmer, Popa, Cani, Hahmann, Sheffer, SIGGRAPH Asia 2010]

Coarse mesh deformed by convolution skeletons to add folds

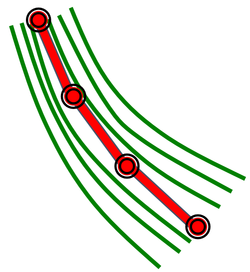


5. Hair

Hair animation

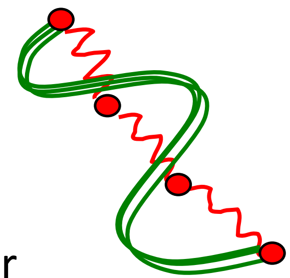
Physically-based models

- Rigid sticks
- Mass-springs



Geometry

- Hair wisps
- Interpolate between guide hair



- Not realistic without collisions