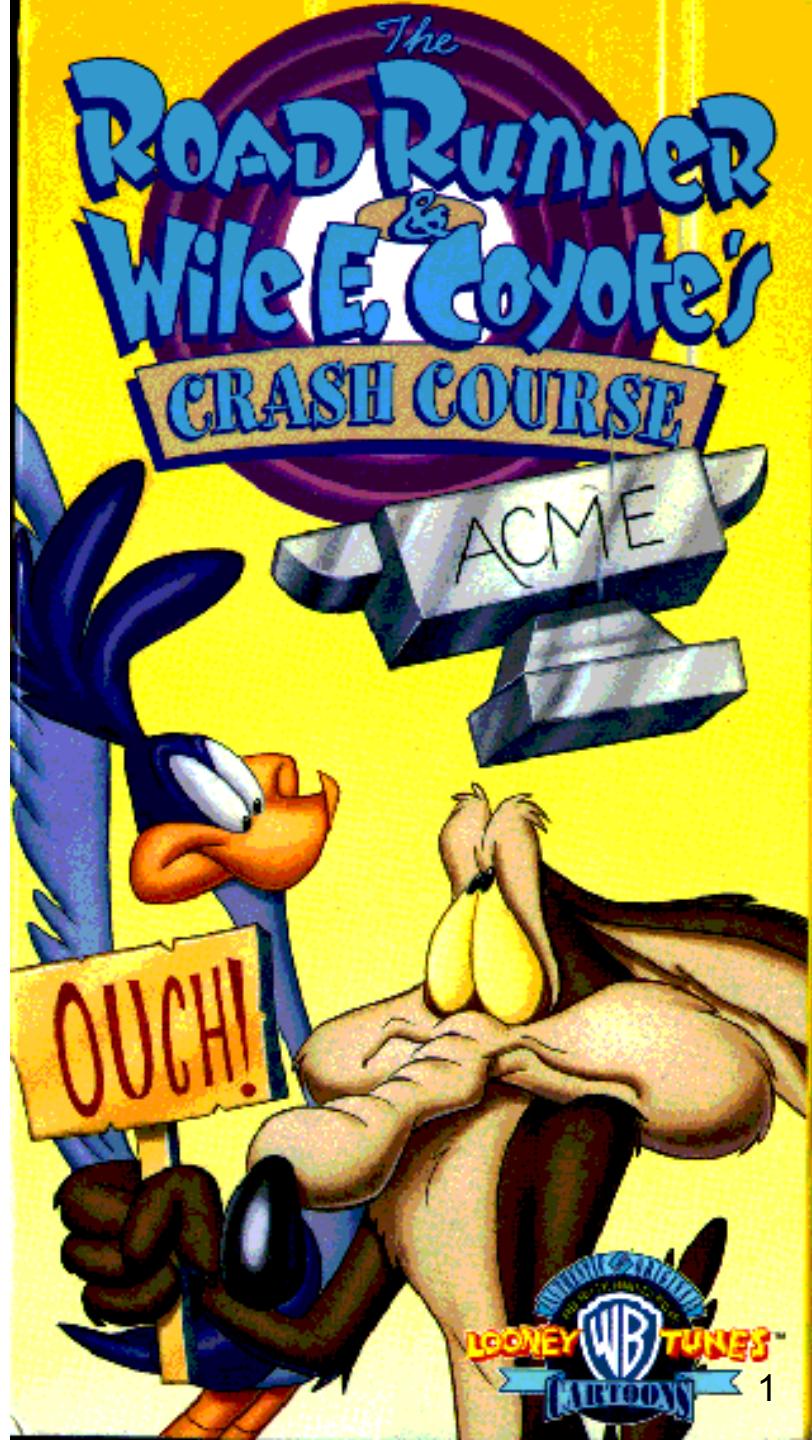


Basics of Computer Animation

7.2 Animation Controls

Jaakko Lehtinen

Many slides courtesy of Jovan Popovic, Ronen Barzel, and Frédo Durand



In This Video

- User-friendly Controls for Animation
- “Rigging” = Building Animation Controls
- Keyframing as a General Approach

How do you Control Animation?

- Animation is (usually) specified using some form of low-dimensional **controls**...

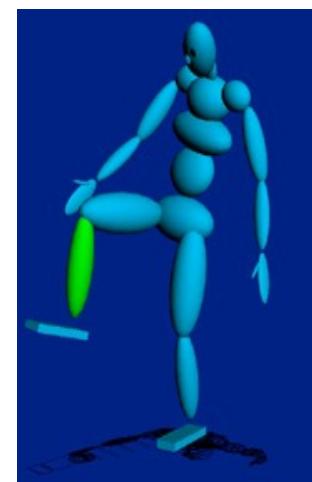
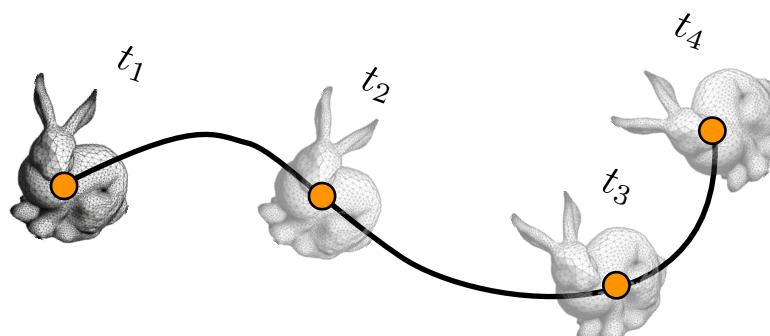
How do you Control Animation?

- Animation is (usually) specified using some form of low-dimensional **controls**... as opposed to remodeling the actual geometry for each frame.

Can you think of examples?

Animation Controls

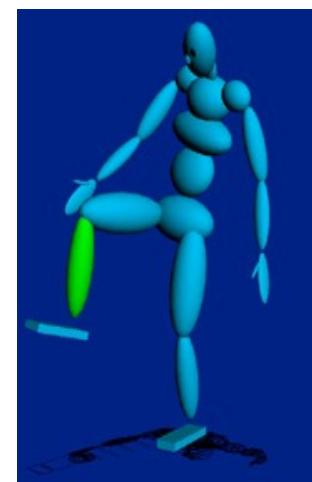
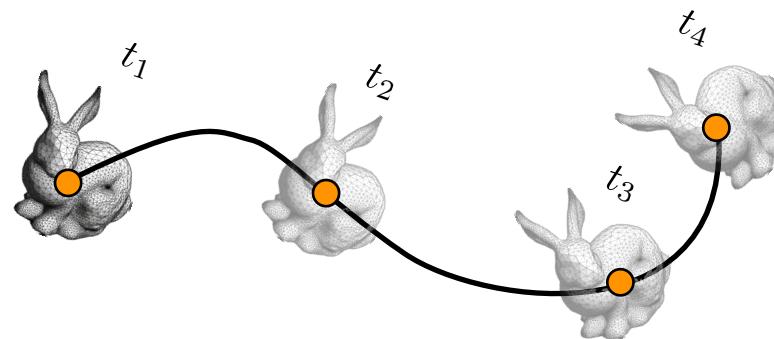
- Animation is (usually) specified using some form of low-dimensional **controls**... as opposed to remodeling the actual geometry for each frame.
 - Example: The joint angles (bone transformations) in a hierarchical character determine the pose
 - Example: A rigid motion is represented by changing the object-to-world transformation (rotation and translation).



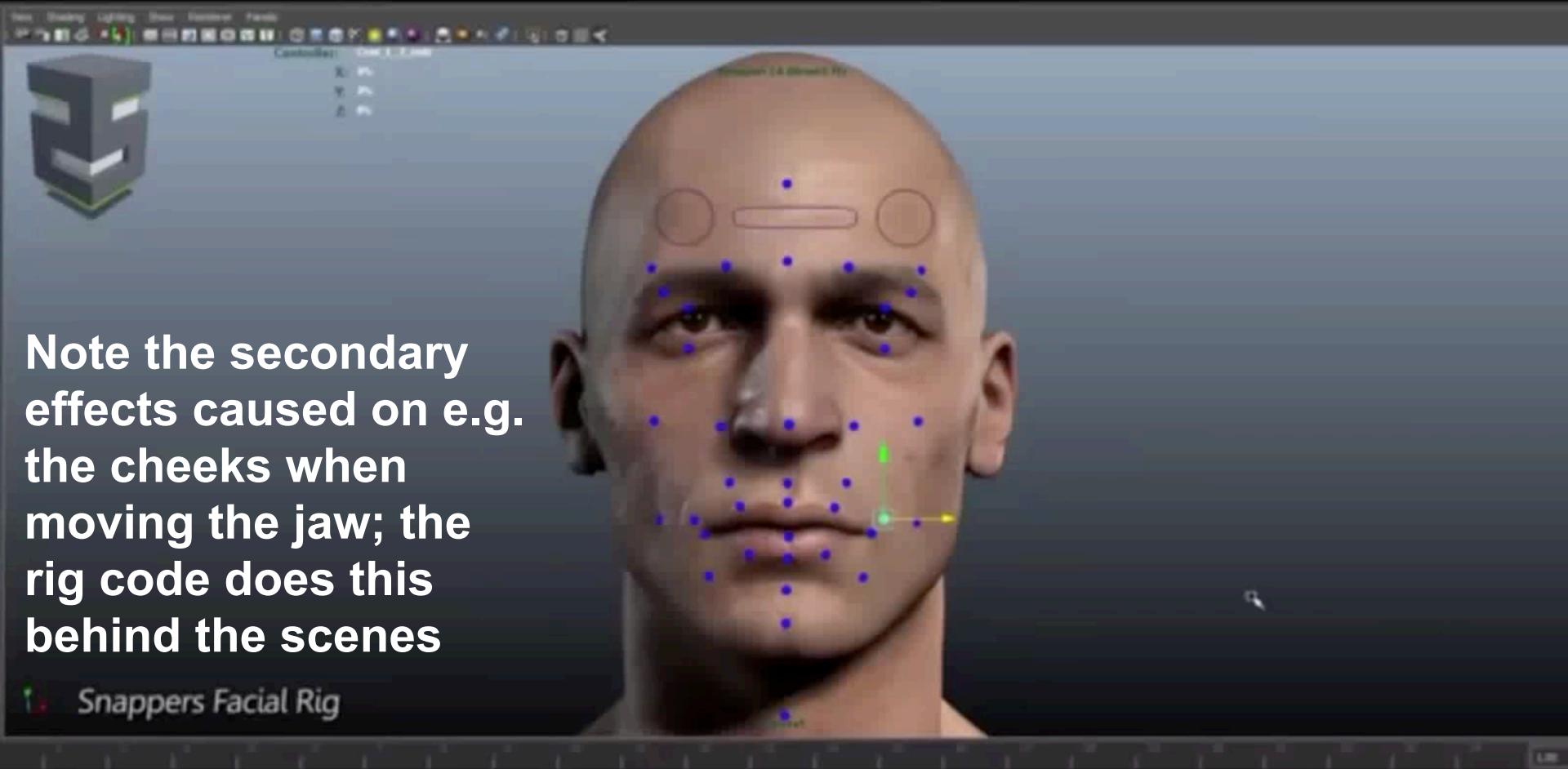
Animation Controls

- Animation is (usually) specified using some form of low-dimensional **controls** as opposed to remodeling the actual geometry for each frame.
 - Example: The joint angles (bone transformations) in a hierarchical character determine the pose
 - Example: A rigid motion is represented by changing the object-to-world transformation (rotation and translation).

“**Blendshapes**” are keyframes that are just snapshots of the entire geometry.

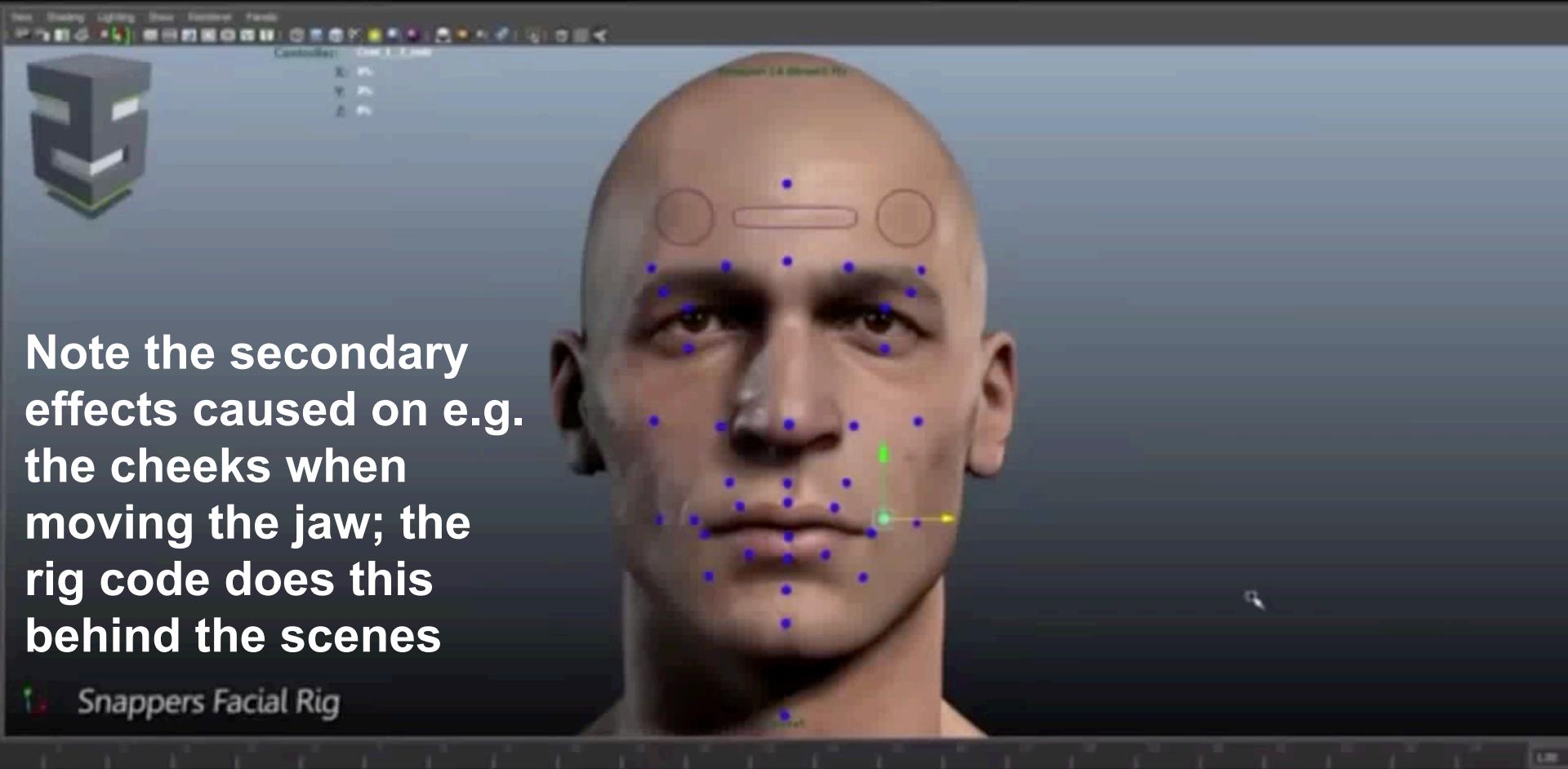


Modern Face Rig (link to video)



Note the secondary effects caused on e.g. the cheeks when moving the jaw; the rig code does this behind the scenes

Modern Face Rig (link to video)



Note the secondary effects caused on e.g. the cheeks when moving the jaw; the rig code does this behind the scenes

(Intuitive Control, non-animation)

- Brits vs. the rest of us

2 DOF



2 DOF



Which one is easier to use?

One does not simply
QWOP
into Mordor



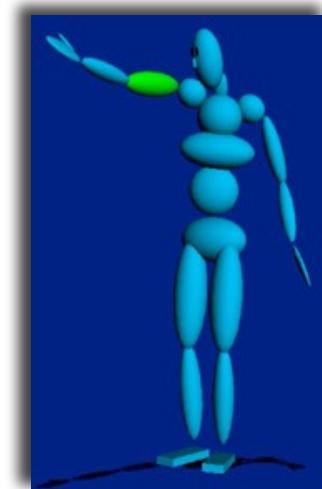
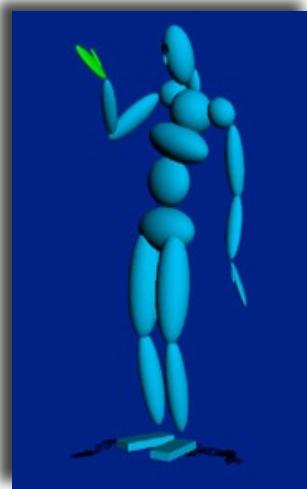
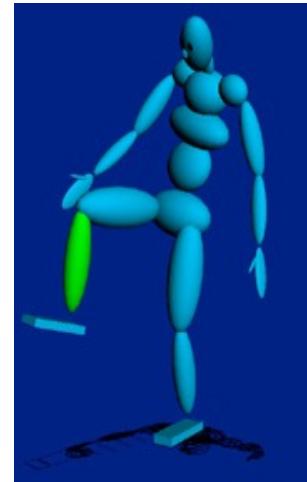
- (Google QWOP if you didn't get that)

Building 3D models and their animation controls is a major component of every animation pipeline.

Building the controls is called “rigging”.

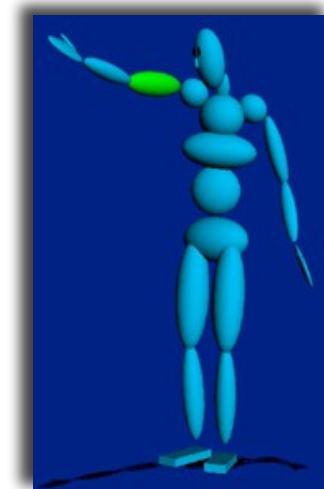
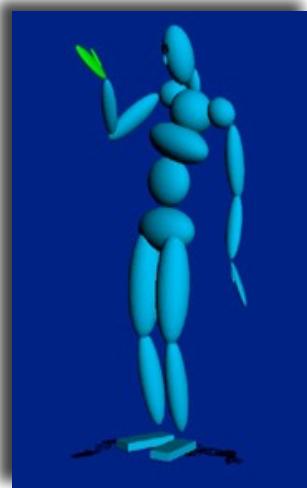
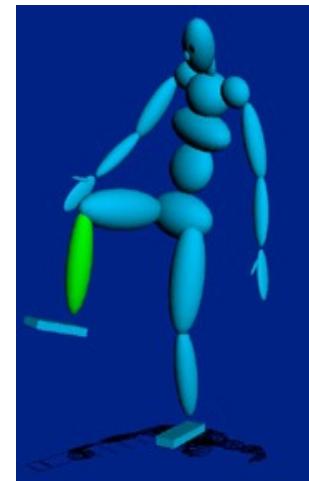
Articulated Character Models

- Forward kinematics describes the positions of bones as a function of joint angles
 - **Angles are the low-dimensional control.**



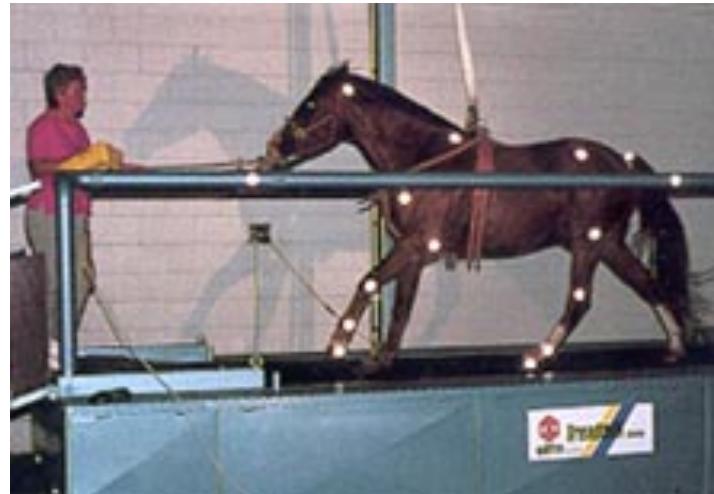
Articulated Character Models

- Inverse kinematics specifies target locations for bones and solves for joint angles
 - IK target positions are also routinely used as animation controls
 - That's the point in IK: to make control easier! 11



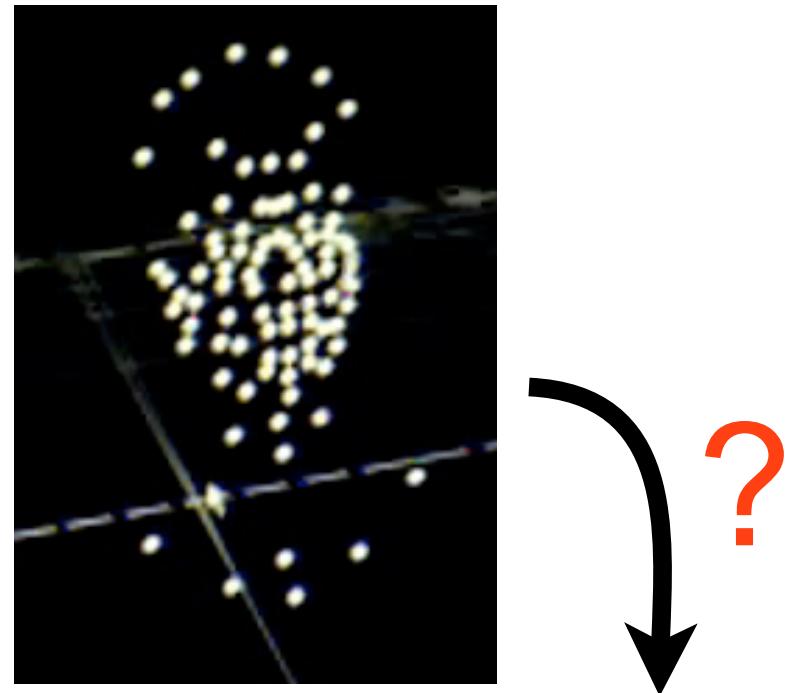
Motion Capture is a form of IK

- Usually uses optical markers and multiple high-speed cameras
- Triangulate to get marker 3D position
 - (Again, structure from motion and projective geometry, i.e., homogeneous coordinates)
- Captures style, subtle nuances and realism
- But need ability to record someone



Motion Capture

- Motion capture records 3D marker positions
 - But character is controlled using animation controls that affect bone transformations!
- Marker positions must be translated into character controls (“retargeting”)
 - A kind of inverse kinematics!



Video: Facial Mocap

created by: Motek Entertainment
in collaboration with: Dynamixyz
modeling/shading: Ivo Diependaal



Video: Facial Mocap

created by: Motek Entertainment
in collaboration with: Dynamixyz
modeling/shading: Ivo Diependaal



Skinning Characters

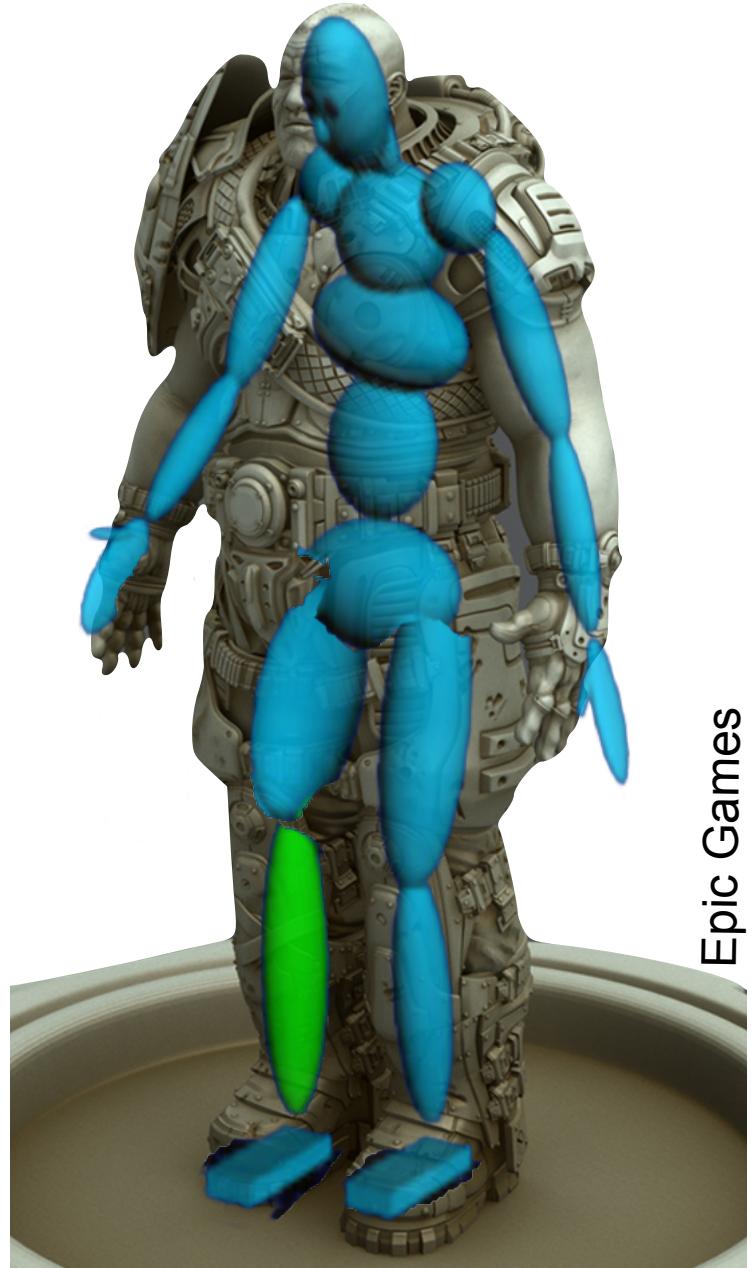
- Embed a skeleton into a detailed character mesh
- Animate “bones”
 - Using IK or other controls
- Bind skin vertices to bones
 - Animate skeleton, skin will move with it
- **Skeleton is the low-dim. control for the skin**



Epic Games

Skinning Characters

- Embed a skeleton into a detailed character mesh
- Animate “bones”
 - Using IK or other controls
- Bind skin vertices to bones
 - Animate skeleton, skin will move with it
- **Skeleton is the low-dim. control for the skin**



Epic Games

Production-Level Facial Performance Capture Using Deep Convolutional Neural Networks

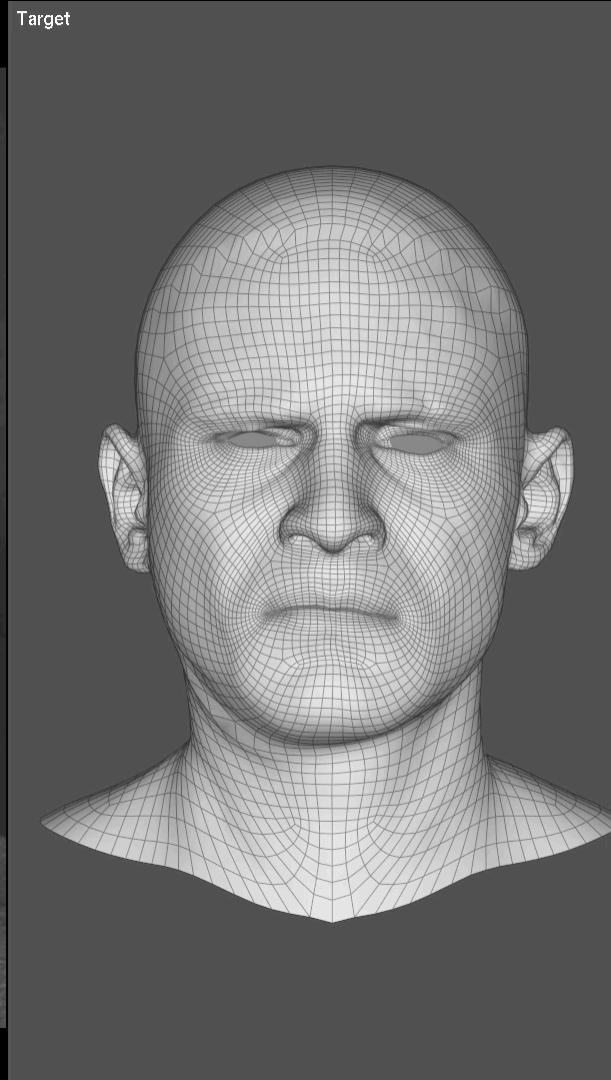
Samuli Laine, Tero Karras, Timo Aila, Antti Herva, Shunsuke Saito, Ronald Yu, Hao Li, and Jaakko Lehtinen

Proc. SIGGRAPH/EUROGRAPHICS Symposium on Computer Animation (SCA) 2017, 2017.

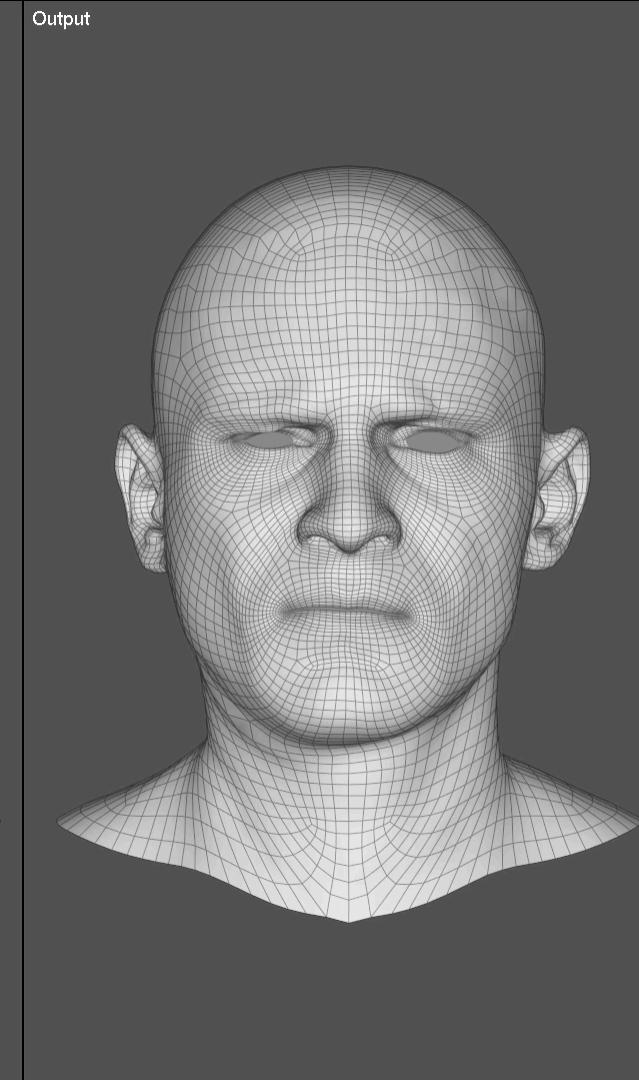
Input frame



Target



Output



Input

Classical heavy CV
+ manual cleanup

Our deep neural net
(90% total time save)

Production-Level Facial Performance Capture Using Deep Convolutional Neural Networks

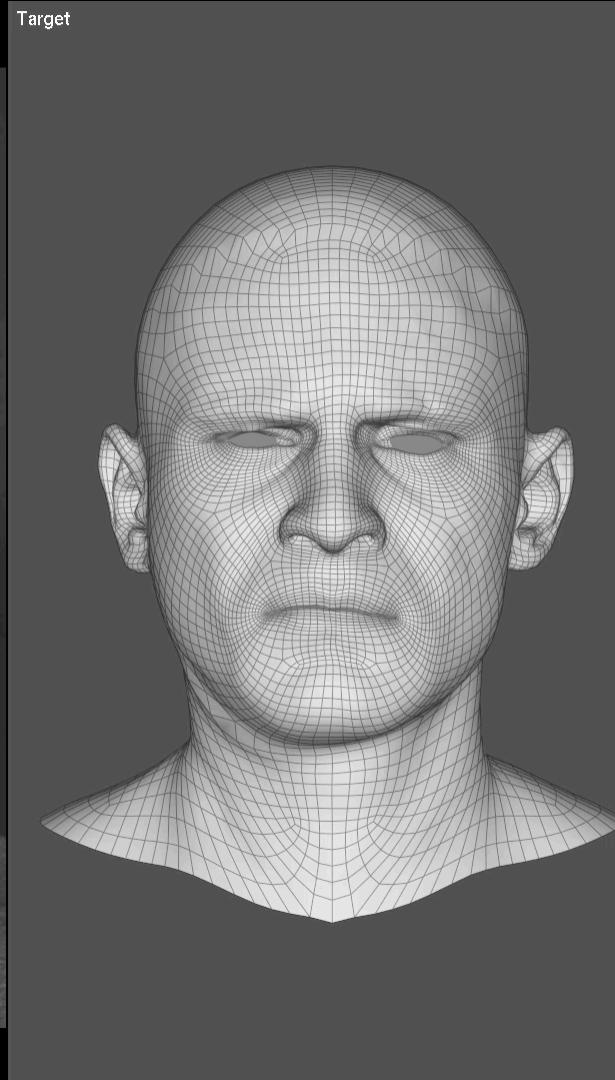
Samuli Laine, Tero Karras, Timo Aila, Antti Herva, Shunsuke Saito, Ronald Yu, Hao Li, and Jaakko Lehtinen

Proc. SIGGRAPH/EUROGRAPHICS Symposium on Computer Animation (SCA) 2017, 2017.

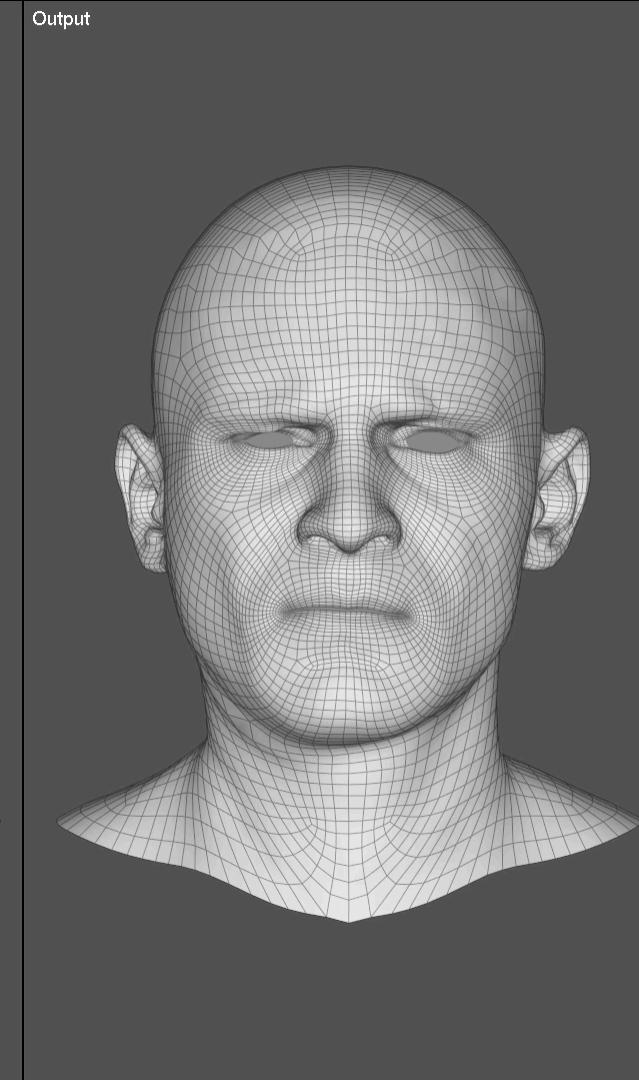
Input frame



Target



Output



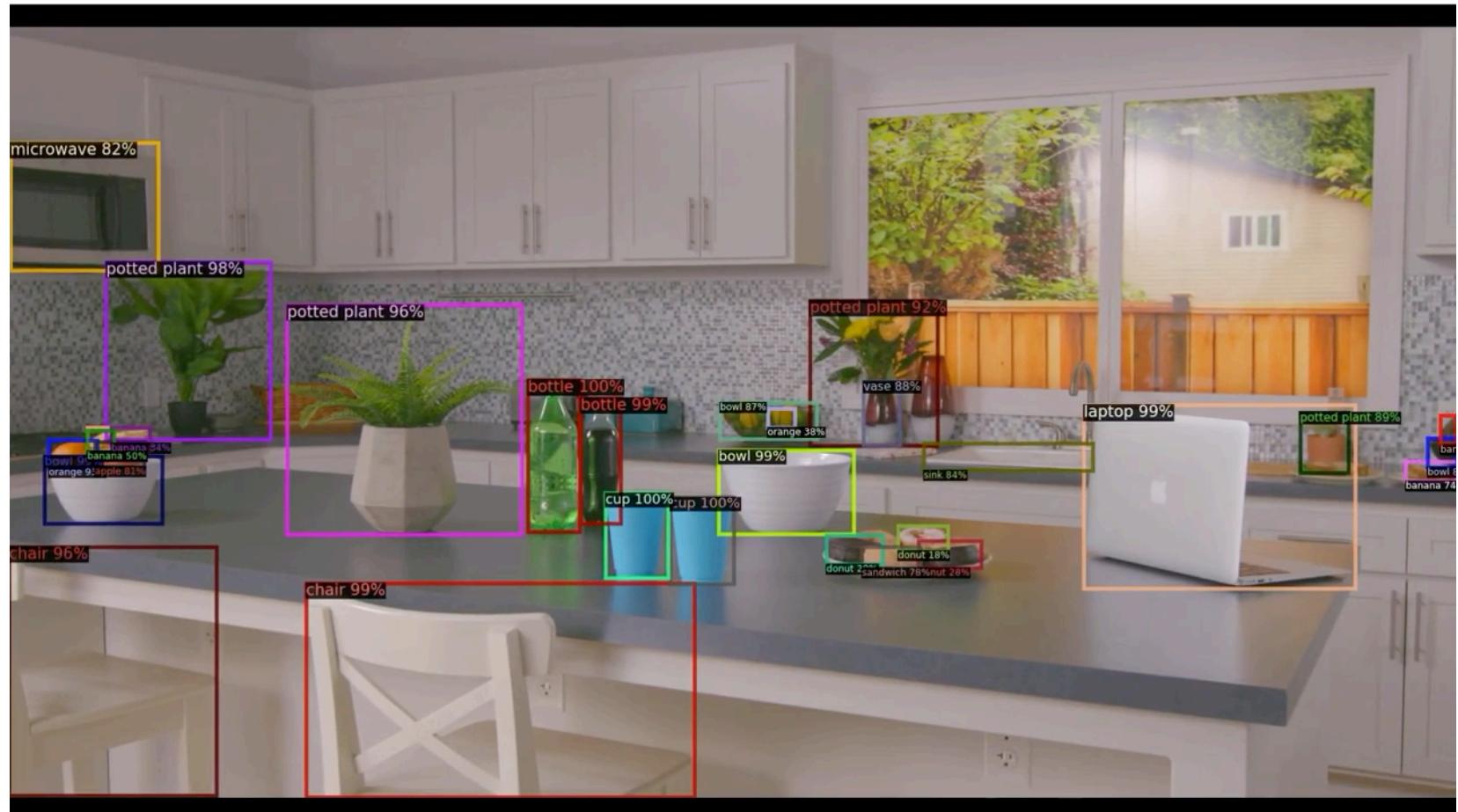
Input

Classical heavy CV
+ manual cleanup

Our deep neural net
(90% total time save)

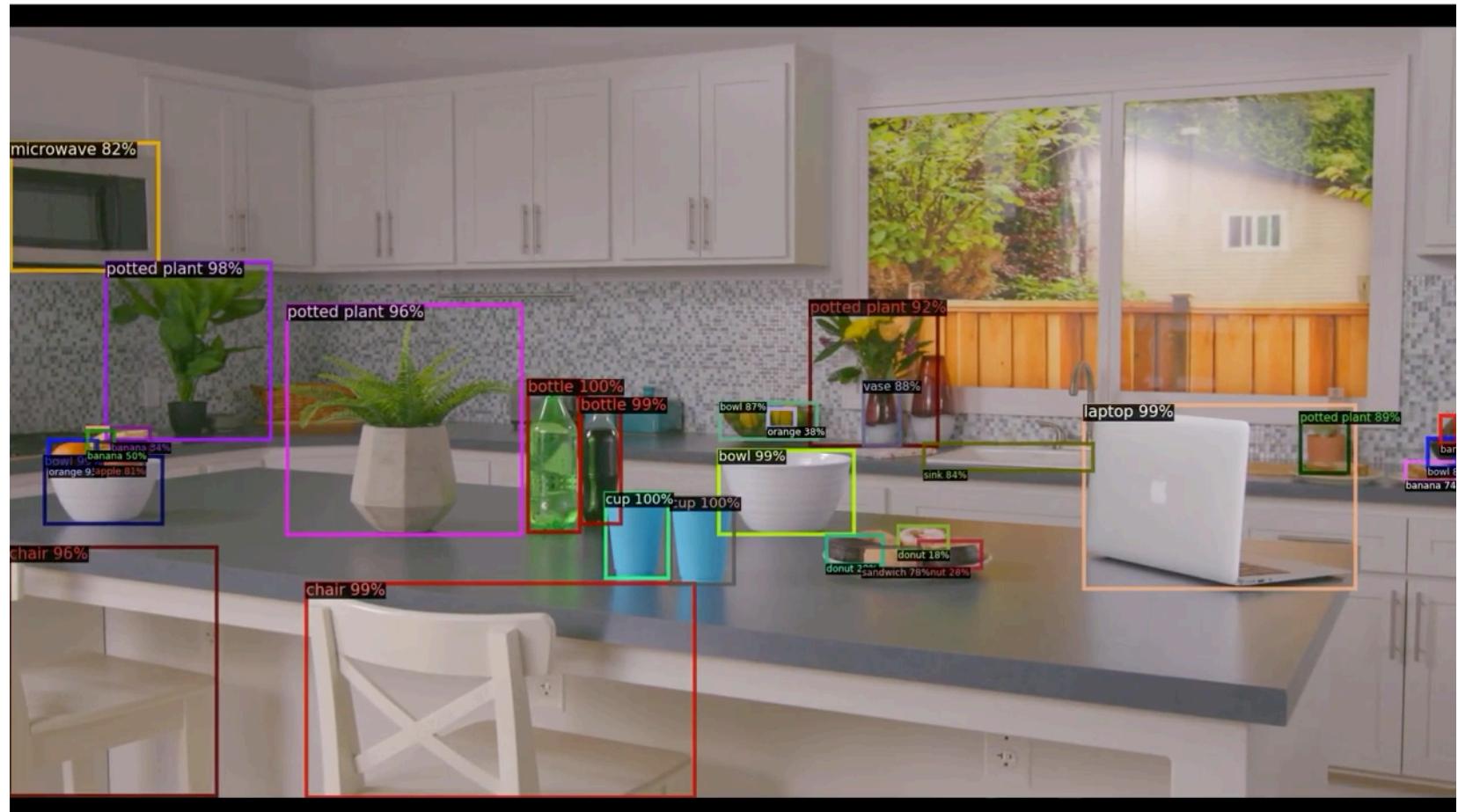
Modern & Lightweight: Monocular Tracking by Deep Learning

- Detectron2 video (use this for extras in Assn'3!)

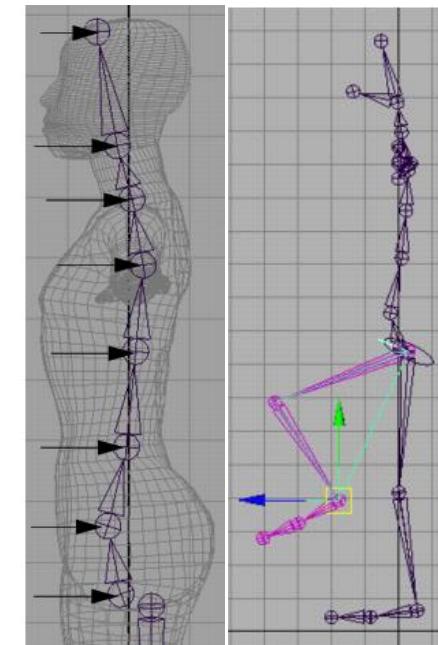


Modern & Lightweight: Monocular Tracking by Deep Learning

- Detectron2 video (use this for extras in Assn'3!)



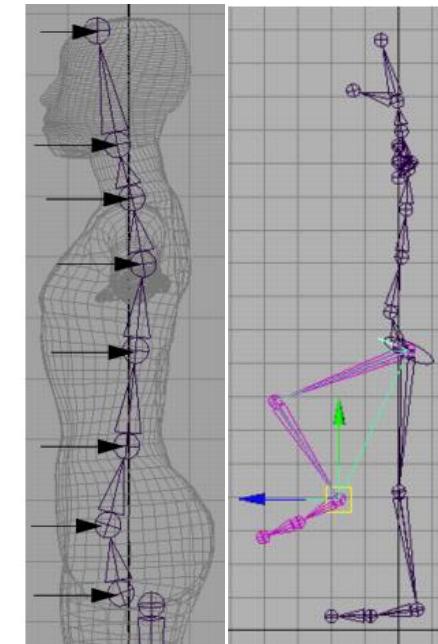
What Values are Keyframed?



Images from the Maya tutorial

What Values are Keyframed?

- Camera and object positions and orientations
 - Use quaternions for orientations
- Skeletal joint angles for skinning
 - Bone transformations
- Higher-level controls
 - Smile control, eye blinking, etc.
- Inverse kinematics
 - Endpoint positions
- Sometimes even geometry (vertex positions, normals)
 - “Blendshapes”



Images from the Maya tutorial

That's All!



Remedy / Microsoft / ign.com