Real Virtual Humans

Gerard Pons-Moll
Max Planck Institute for Informatics

Max Planck for Informatics

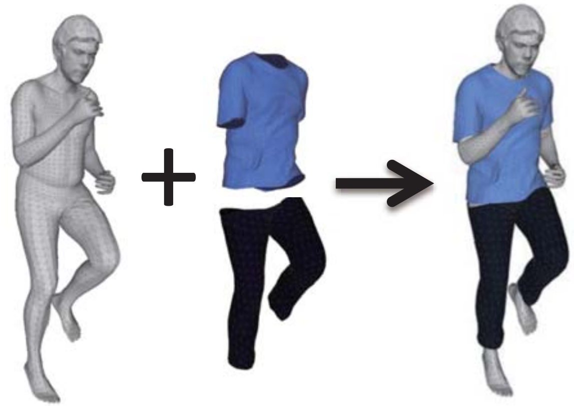
- Focus on basic research
- 5 departments
- 20 research groups

Saarbruecken (at the border of Germany and France)

Collaborators and students
(for the work I will present today)

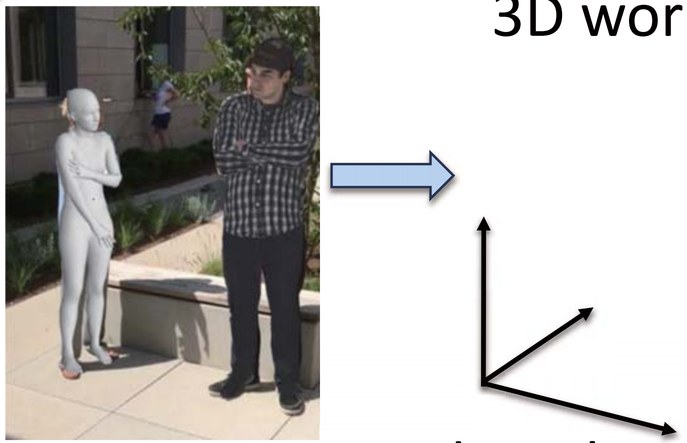
Thiemo Alldieck, Michael Black, Federica Bogo, Peter Gehler, Sonny Hu, Christoph Lassner, Yebin Liu, Mattwew Loper, Marcus Magnor, Timo von Marcard, Naureen Mahmood, Mohamed Omran, Javier Romero, Bodo Rosenhahn, Sergi Pujades, Bernt Schiele, Christian Theobalt, Yao Tu, Weipeng Xu.

Goal: Realistic virtual humans



Generation **(+ clothing)**

3D world

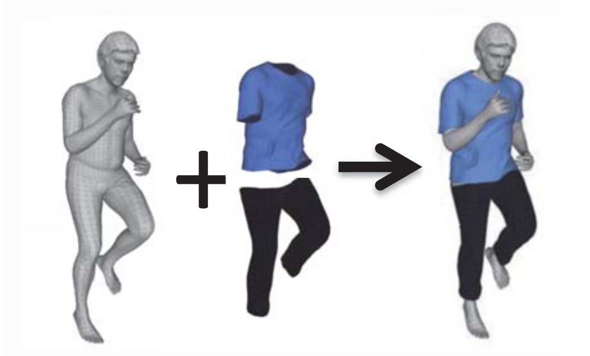


Capture and Analysis

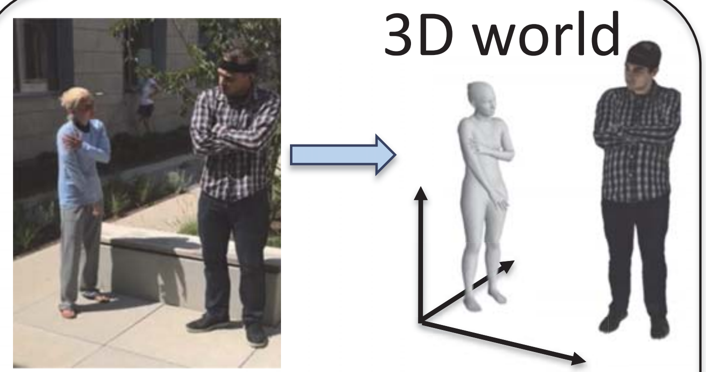
Realistic 3D people <u>models</u>:
- Move and look like real people
- Easy to control and animate
- Easy to fit to data

<u>Reconstruction</u> from images:
- Accurate
- Efficient
- Robust

Goal: Realistic virtual humans



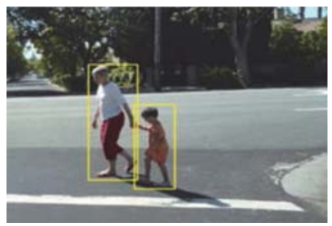
Generation **(+ clothing)**



Capture and Analysis

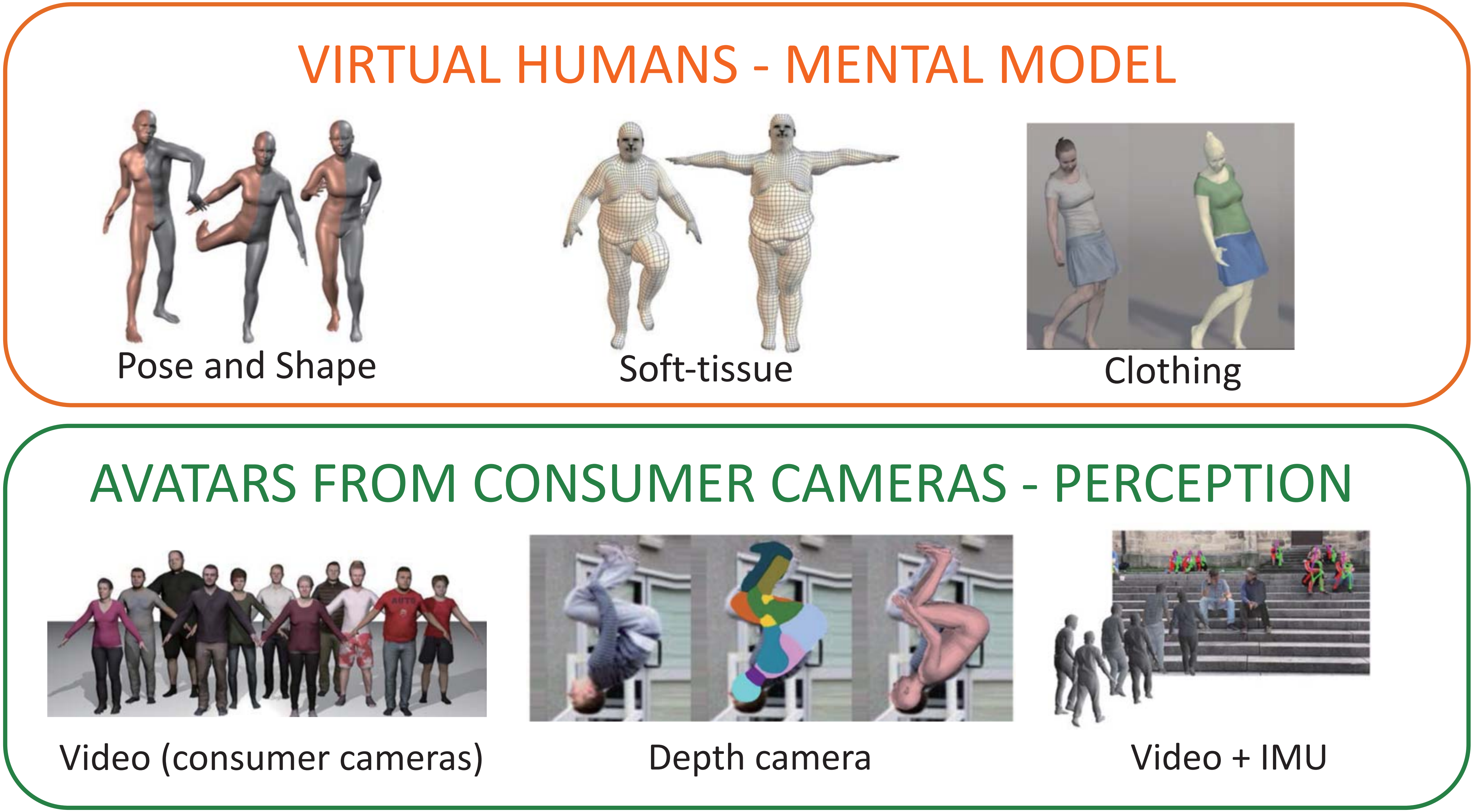
Virtual/Augmented Reality

Computer Vision



Pedestrian safety

Medicine and
self-perception

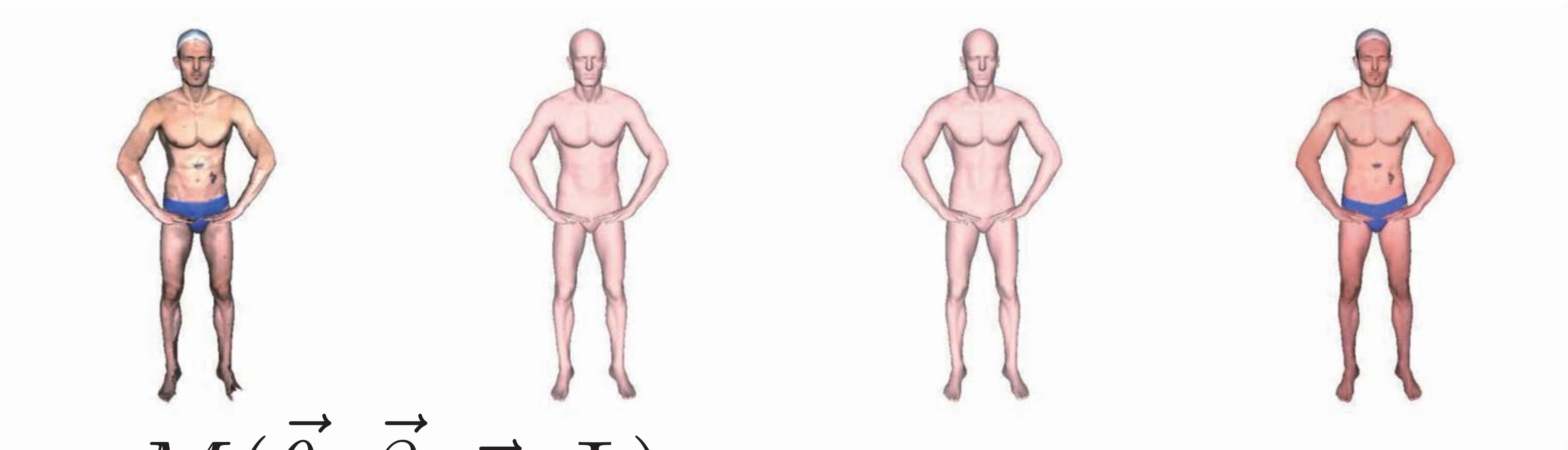

VIRTUAL HUMANS - MENTAL MODEL
Pose and Shape
Soft-tissue
Clothing
AVATARS FROM CONSUMER CAMERAS - PERCEPTION
Video (consumer cameras)
Depth camera
Video + IMU

Schedule

- Virtual human models
  - Kinematic Chains, Linear Blend Skinning, Blendshapes
  - SMPL & Dyna
  - ClothCap: Capturing people in clothing
- Capturing humans from consumer sensors
  - 3D human reconstruction from a video
  - 3D human pose and shape from images
  - 3D human pose from Inertial Measurement Units (IMU)

What is a virtual human model?

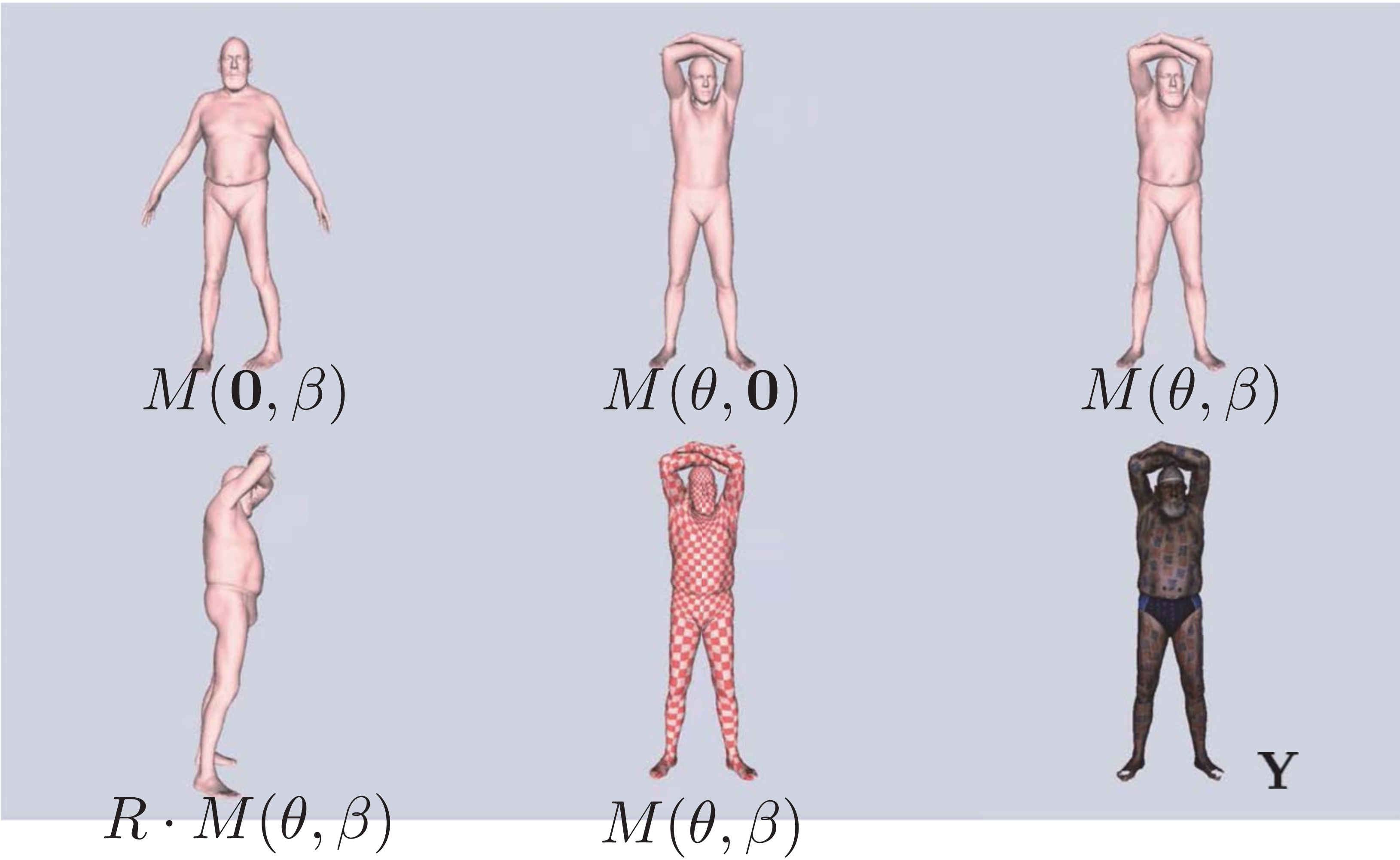
3D scan with texture | **Ground truth shape** | **Model** | **Model** with texture



$$M(\vec{\theta}, \vec{\beta}, \vec{u}; \Phi) \longrightarrow \text{3D mesh}$$

pose　shape　texture　Hyper-parameters to learn

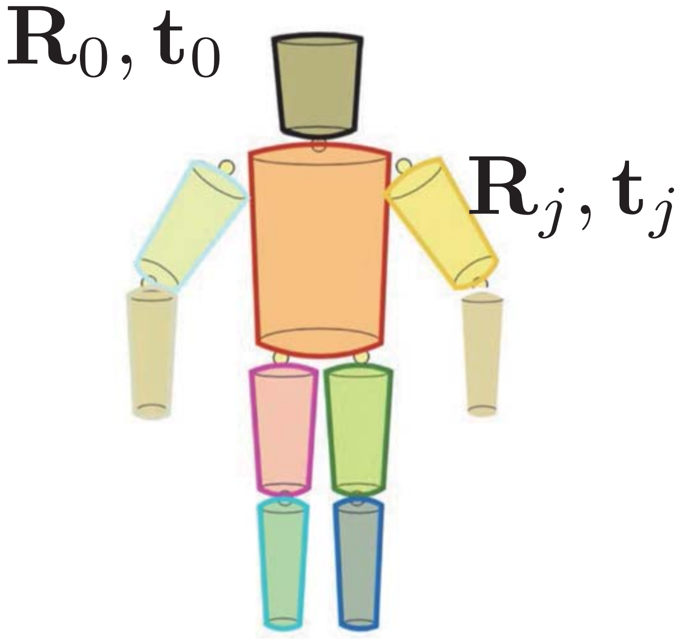
A virtual human is a function



Kinematic Chains

How do we parameterize pose ?

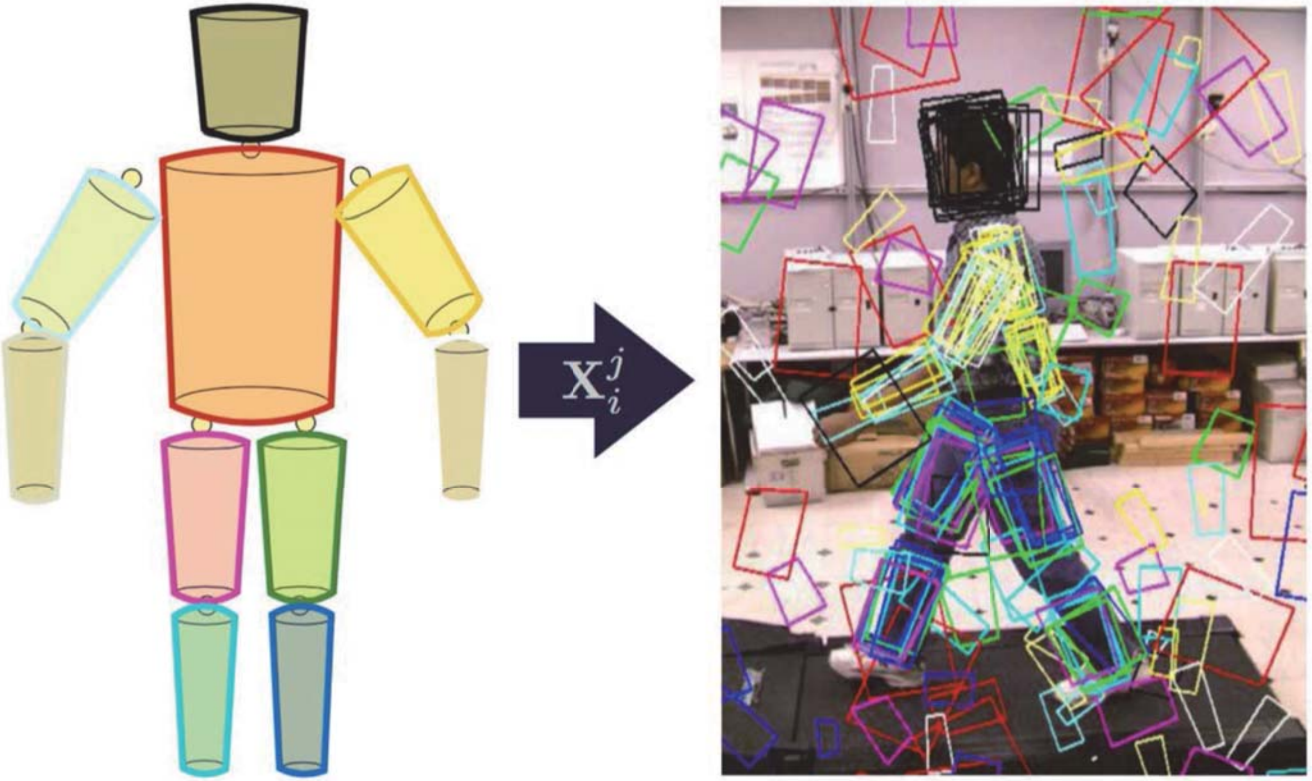
- Parameterize every body part separately ?



$$\mathbf{X}_{\text{pose}} = \{\mathbf{R}_0, \mathbf{t}_0, \ldots \mathbf{R}_N, \mathbf{t}_N\}$$

Problems ?

How do we parameterize pose?



Articulated constraints not satisfied!

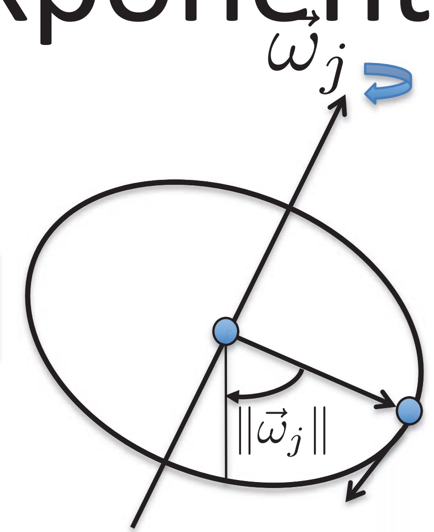
Rotation parameterization

- Rotations are composed of 9 numbers
- 6 **additional constraints** to ensure that the matrix is orthonormal with positive determinant
- **Suboptimal** for optimization

Rotation with Exponential Maps

$\|\vec{\omega}_j\|$: Angle of rotation

$\boxed{\vec{\omega}_j \text{ : scaled axis of rotation}}$

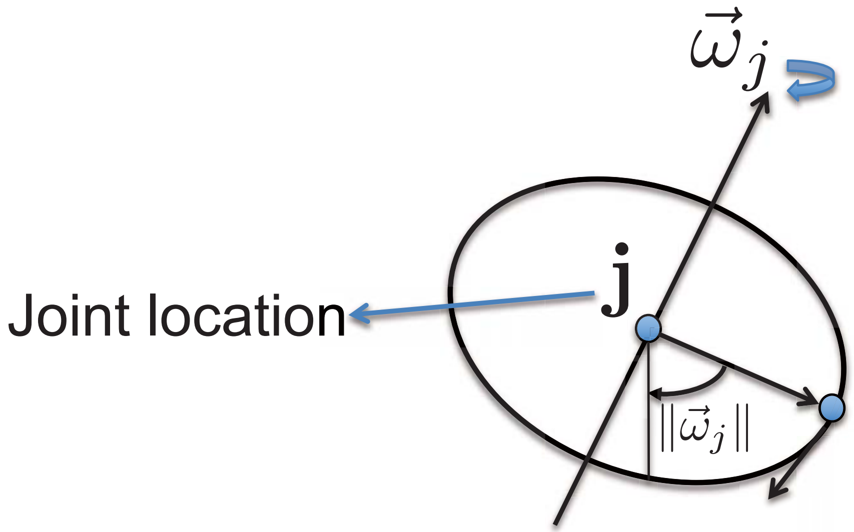


Rotation obtained with Rodrigues formula:

$$\mathbf{R} = e^{\widehat{\vec{\omega}}} = \mathcal{I} + \widehat{\omega}_j \sin(\|\vec{\omega}_j\|) + \widehat{\omega}^2 (1 - \cos(\|\vec{\omega}_j\|))$$

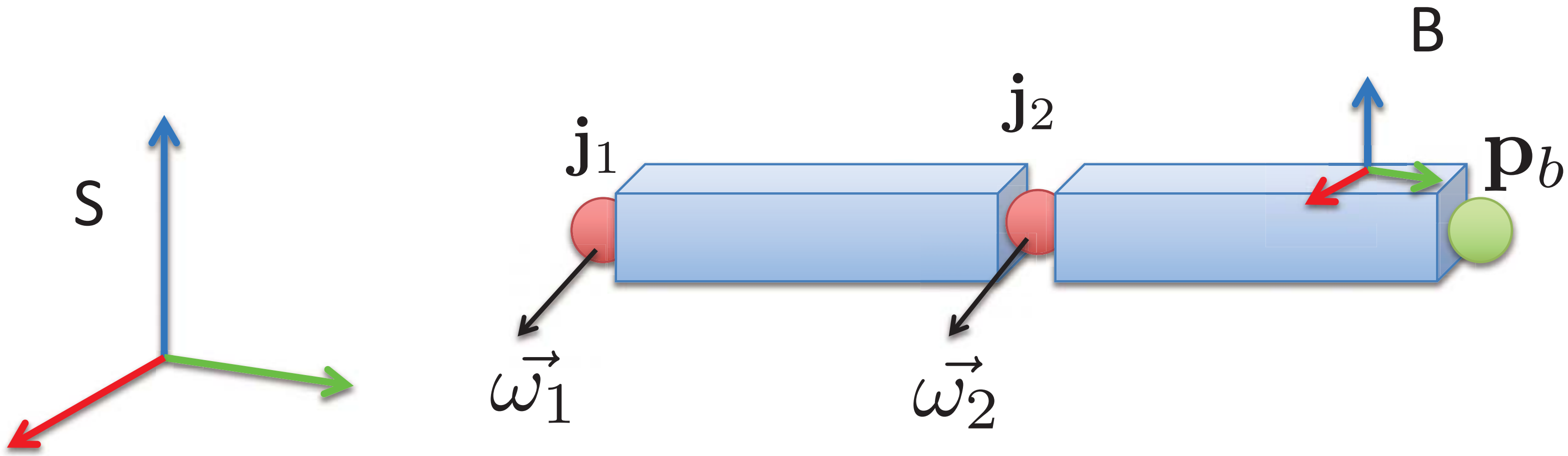
Joint Rigid Body Motion

The transformation associated with a rotational joint is

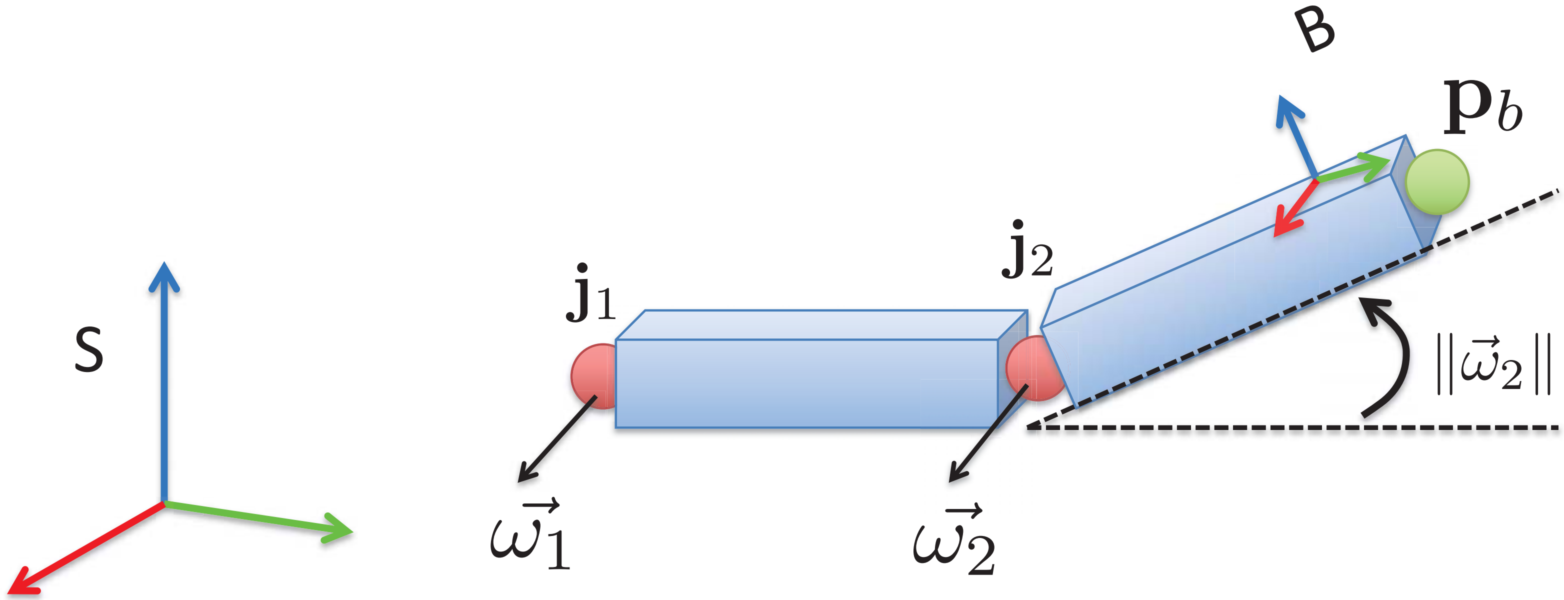


$$G(\vec{\omega}, \mathbf{j}) = \begin{bmatrix} [e^{\vec{\omega}}]_{3\times 3} & \mathbf{j}_{3\times 1} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \longrightarrow \text{Rigid Body Motion}$$

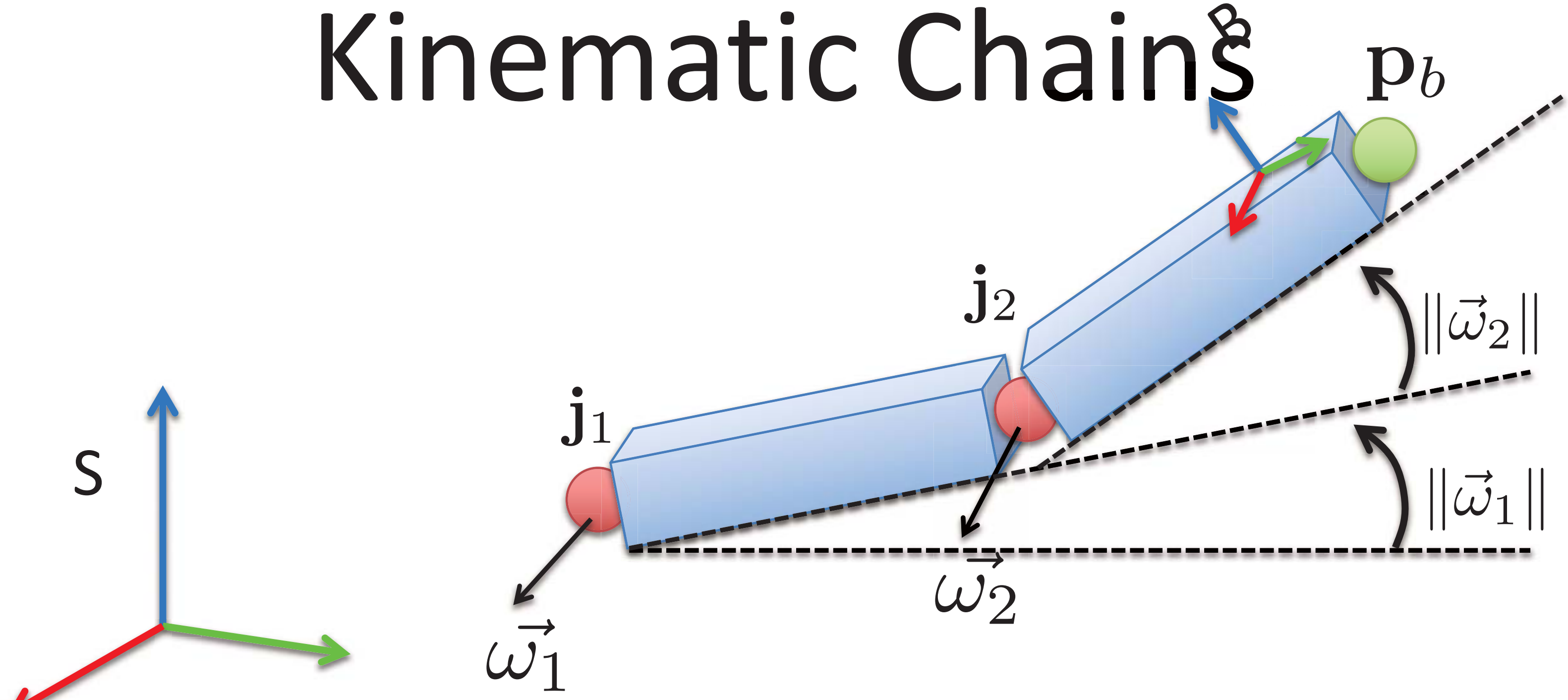
Kinematic Chains



Kinematic Chains



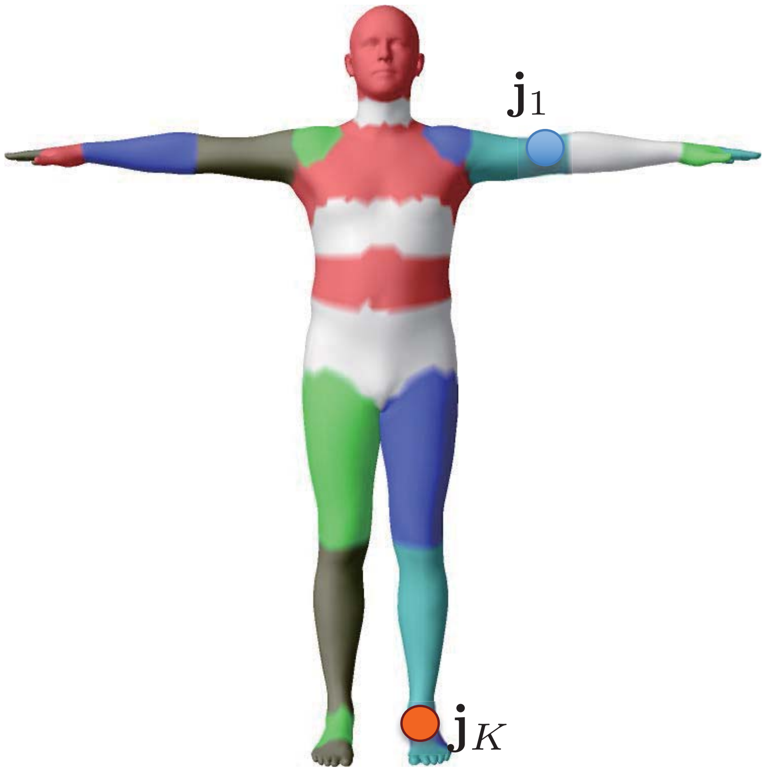
Kinematic Chains



The coordinates of the point in the spatial frame are:

$$\bar{\mathbf{p}}_s = G(\vec{\omega_1}, \vec{\omega_2}, \mathbf{j}_1, \mathbf{j}_2) = G(\vec{\omega}_1, \mathbf{j}_1)\boxed{G(\vec{\omega_2}, \mathbf{j}_2)}\bar{\mathbf{p}}_b$$

Pose Parameters



Given a set of joint locations

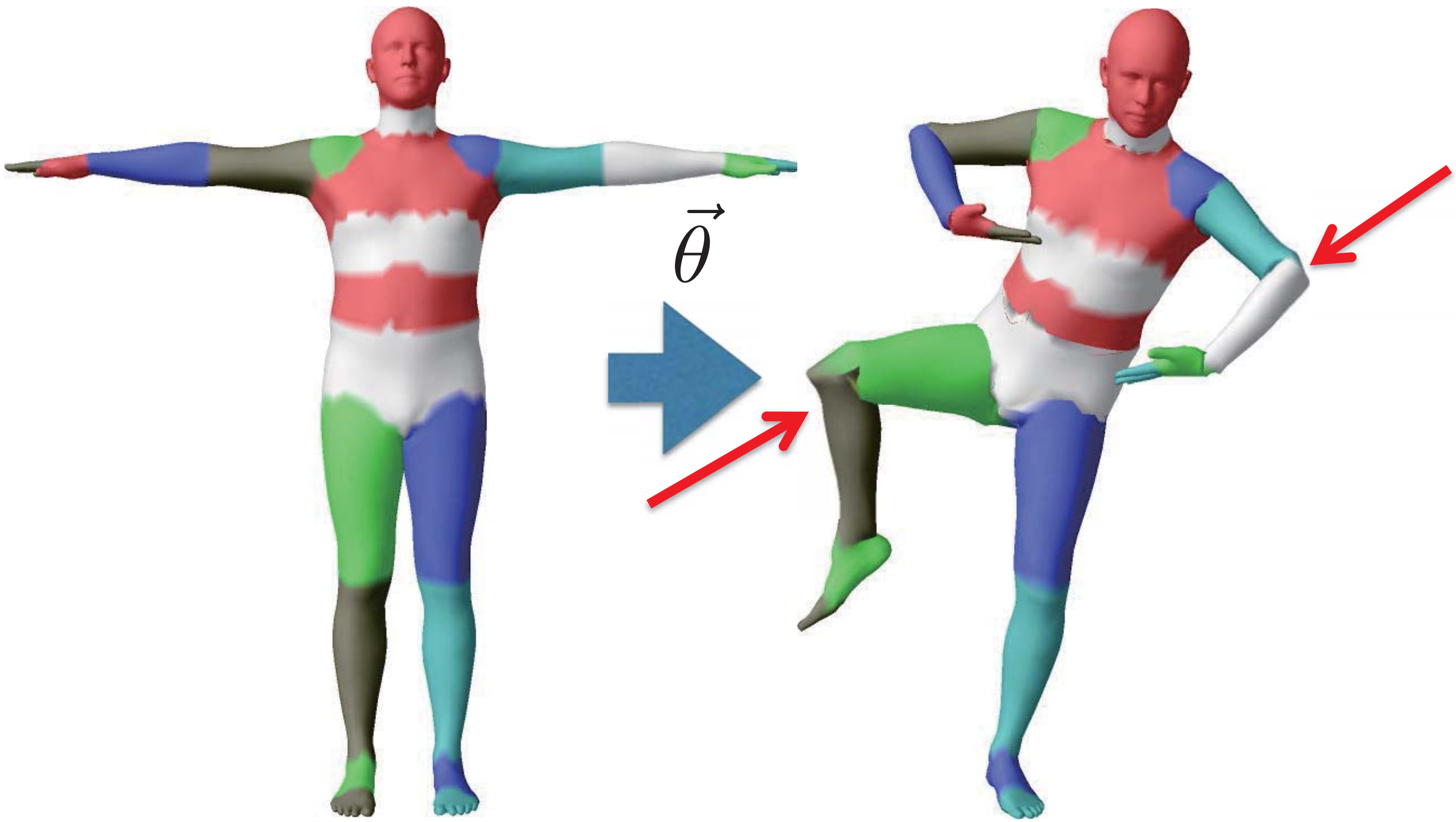
$$\mathbf{J} = (\underline{\mathbf{j}_1}, \ldots, \underline{\mathbf{j}_K})^T$$

The pose defined as the vector
of concatenated part axis-angles

$$\vec{\theta} = (\underline{\vec{\omega}_1}, \ldots, \underline{\vec{\omega}_k})^T$$

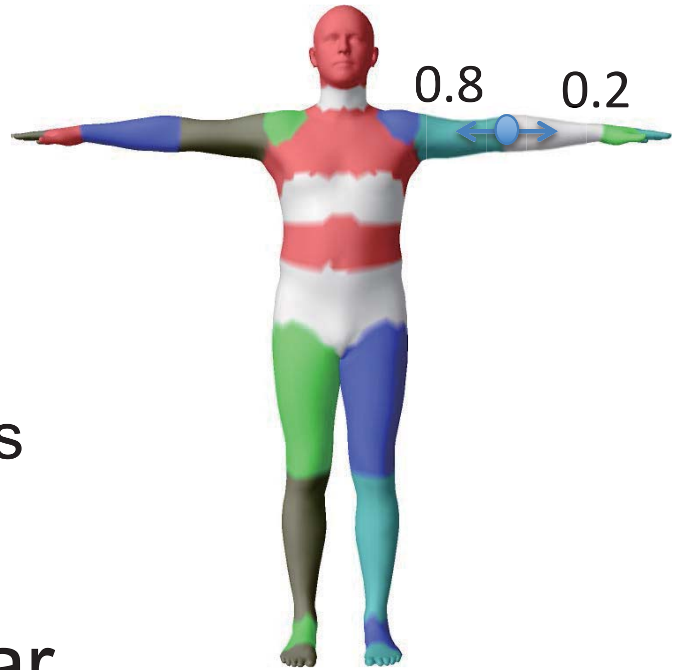
Pons-Moll & Rosenhahn 2011
Model-based Pose Estimation. Looking at People.

Kinematic Chain Problems



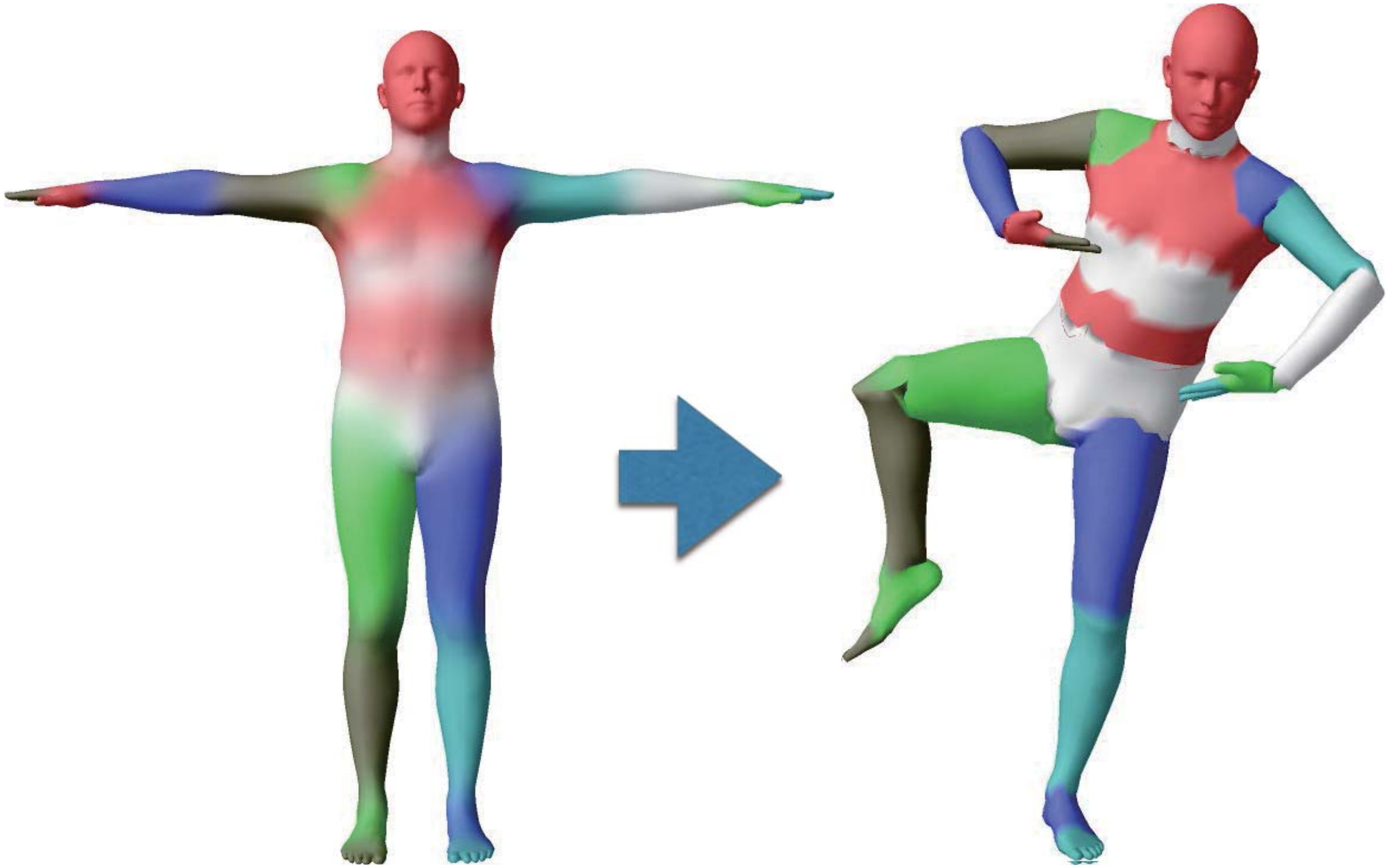
Linear Blend Skinning

$$\bar{\mathbf{t}}'_i = \sum_{k=1}^{K} \underbrace{w_{k,i}}_{\text{Blend weights}} \underbrace{G'_k(\vec{\theta}, \mathbf{J})}_{\text{Part transformations}} \bar{\mathbf{t}}_i$$



Points transformed as blended linear combination of joint transformation matrices

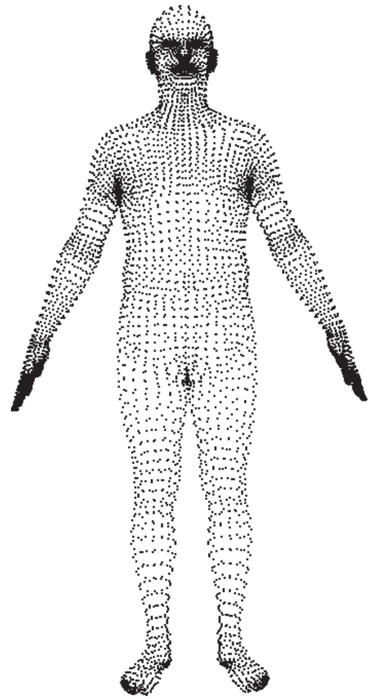
Linear Blend Skinning



Standard Skinning

Standard skinning produces vertices from…

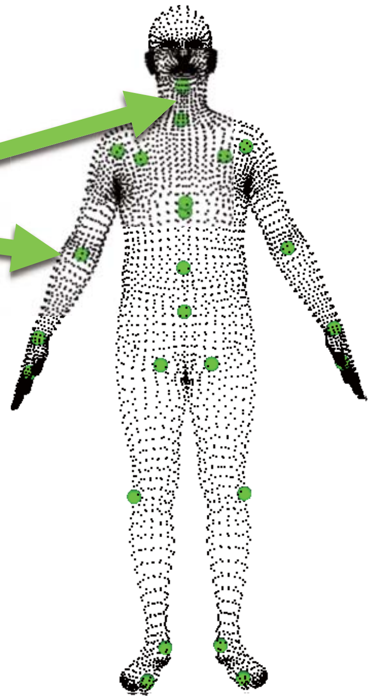
- Rest pose vertices: $\mathbf{T} \in \mathbb{R}^{3N}$
- Joint locations: $\mathbf{J} \in \mathbb{R}^{3K}$
- Weights: $\mathcal{W} \in \mathbb{R}^{N \times K}$
- Pose parameters: $\vec{\theta} \in \mathbb{R}^{3K}$



Standard Skinning

Standard skinning produces vertices from...

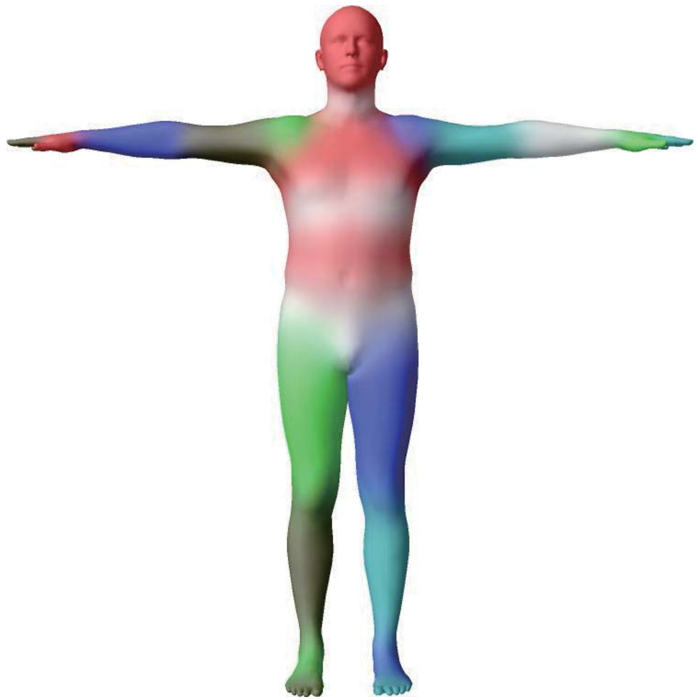
- Rest pose vertices: $\mathbf{T} \in \mathbb{R}^{3N}$
- Joint locations: $\mathbf{J} \in \mathbb{R}^{3K}$
- Weights: $\mathcal{W} \in \mathbb{R}^{N \times K}$
- Pose parameters: $\vec{\theta} \in \mathbb{R}^{3K}$



Standard Skinning

Standard skinning produces vertices from…

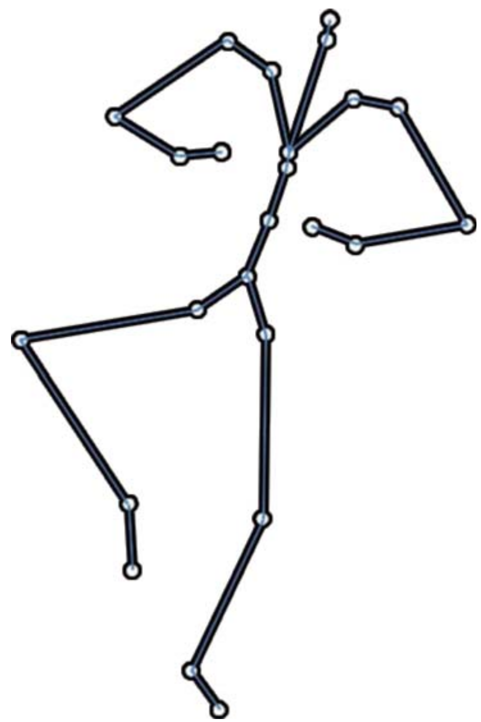
- Rest pose vertices: $\mathbf{T} \in \mathbb{R}^{3N}$
- Joint locations: $\mathbf{J} \in \mathbb{R}^{3K}$
- Weights: $\mathcal{W} \in \mathbb{R}^{N \times K}$
- Pose parameters: $\vec{\theta} \in \mathbb{R}^{3K}$



Standard Skinning

Standard skinning produces vertices from…

- Rest pose vertices: $\mathbf{T} \in \mathbb{R}^{3N}$
- Joint locations: $\mathbf{J} \in \mathbb{R}^{3K}$
- Weights: $\mathcal{W} \in \mathbb{R}^{N \times K}$
- $\boxed{\text{Pose parameters: } \vec{\theta} \in \mathbb{R}^{3K}}$

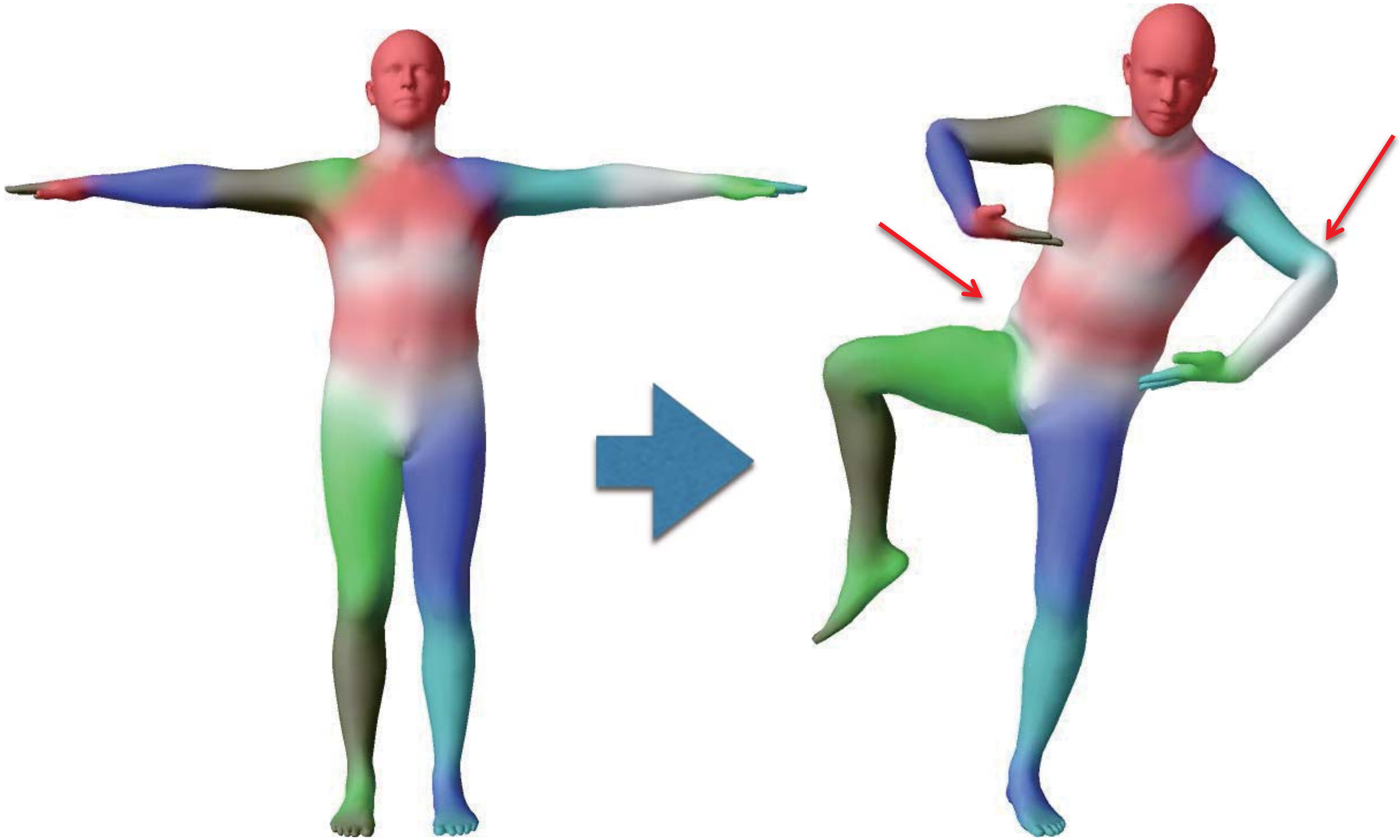


Skinning function

- Rest pose vertices: $\mathbf{T} \in \mathbb{R}^{3N}$
- Joint locations: $\mathbf{J} \in \mathbb{R}^{3K}$
- Weights: $\mathcal{W} \in \mathbb{R}^{N \times K}$
- Pose parameters: $\vec{\theta} \in \mathbb{R}^{3K}$

$$\boxed{W(\mathbf{T}, \mathbf{J}, \mathcal{W}, \vec{\theta}) \mapsto \text{vertices}}$$

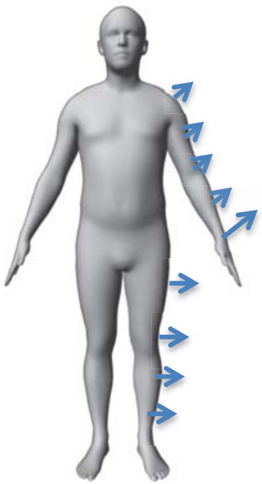
LBS problems



Blend Shapes

Solution: Blend Shapes

- A **blend shape** is a set of vertex displacements in a rest pose
- Pose blend shapes: correct for LBS problems

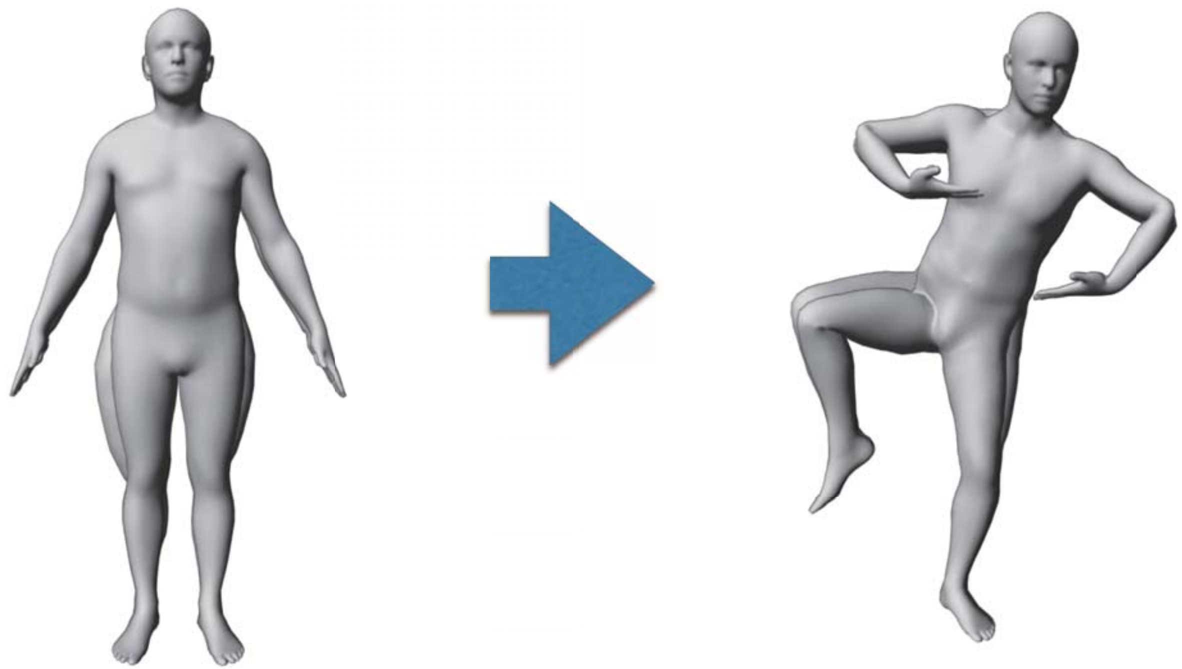


$$\mathbf{P} = \mathrm{vec}\left(\begin{bmatrix} \Delta x_1 & \Delta y_1 & \Delta z_1 \\ & \vdots & \\ & \vdots & \\ \Delta x_N & \Delta y_N & \Delta z_N \end{bmatrix}\right) \in \mathbb{R}^{3N}$$

Offset 1

Pose Blend Shapes

- **With** blend shape correction



How to predict Blend Shapes ?

- Animators sculpt it manually!
- Time consuming, does not scale

Can we learn them from captured real people ?

Problems

- How do we define pose blend shapes ?

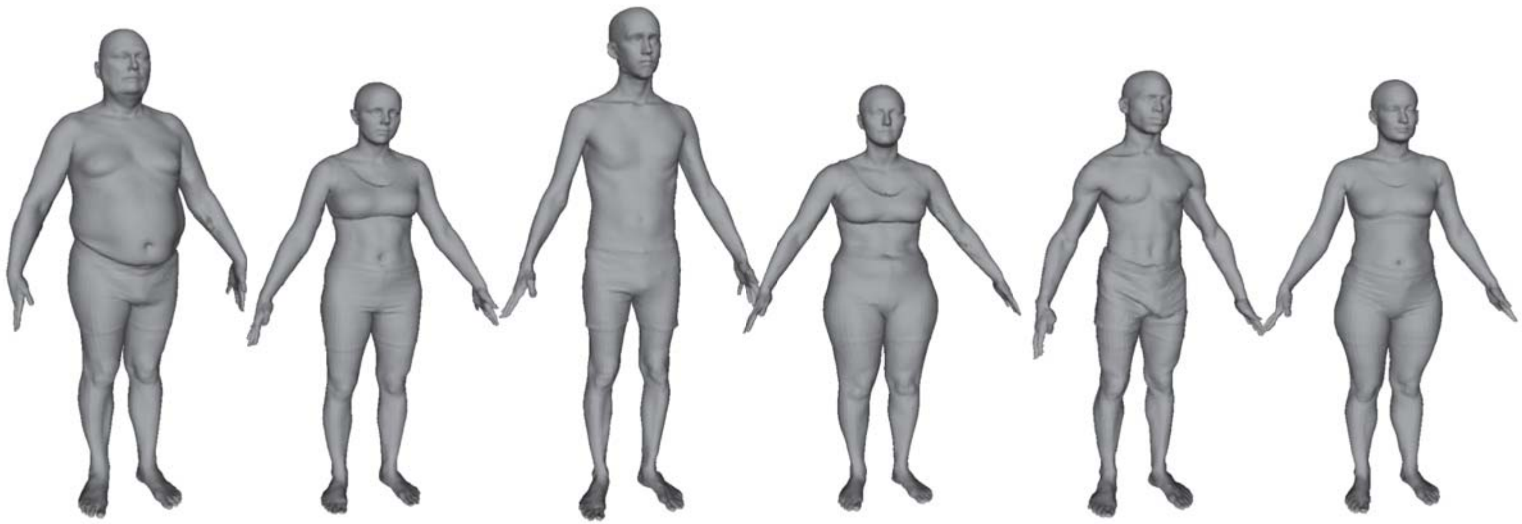
$$B_P(\vec{\theta}')$$

- How to set the skinning parameters ?

$$\mathbf{T} \in \mathbb{R}^{3N} \quad \mathbf{J} \in \mathbb{R}^{3K} \quad \mathcal{W} \in \mathbb{R}^{N \times K}$$

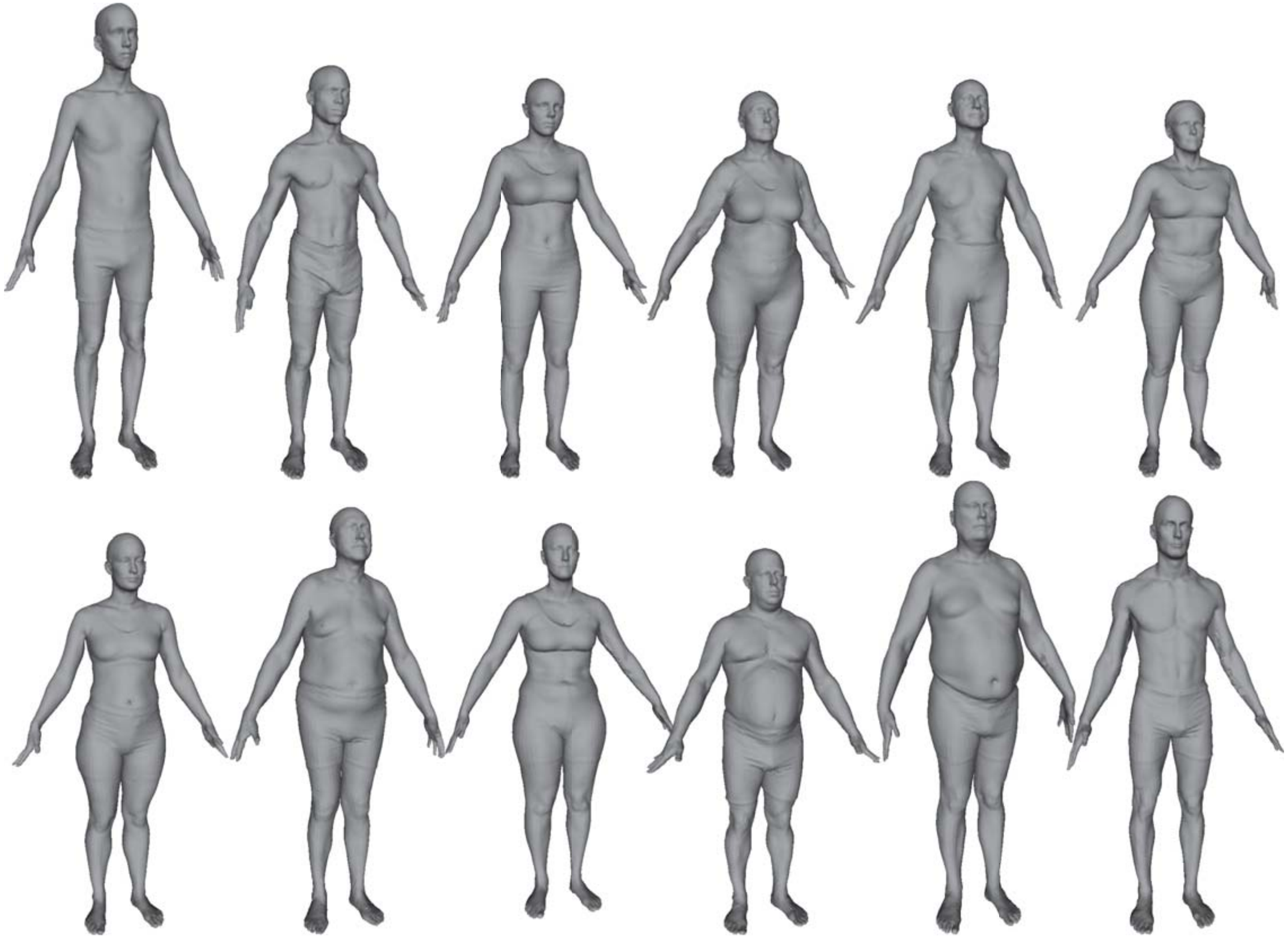
More Problems

How do we model shape identity variations ?



SMPL

Idea: Collect 3D scans from

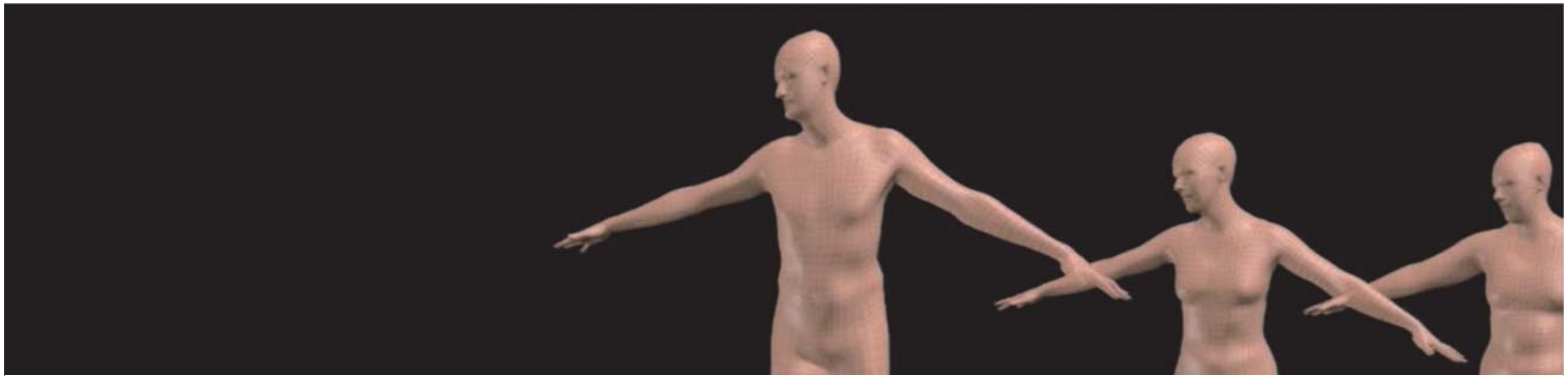


thousands of people…

and thousands of poses

1000’s of high-resolution scans of different shapes and poses

SMPL: A model of pose and shape



$$M(\theta, \beta; \mathbf{w}) : \mathbb{R}^{|\theta|+|\beta|} \mapsto \mathbb{R}^{3N}$$

$$\text{Latent parameters} \mapsto \text{vertices}$$

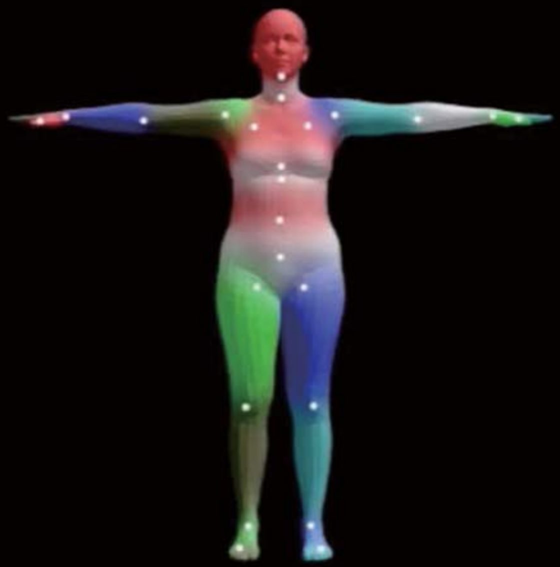
M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, M. Black **SIGGRAPH Asia'15**

SMPL Philosophy

We aim for the simplest possible model while having state-of-the-art performance

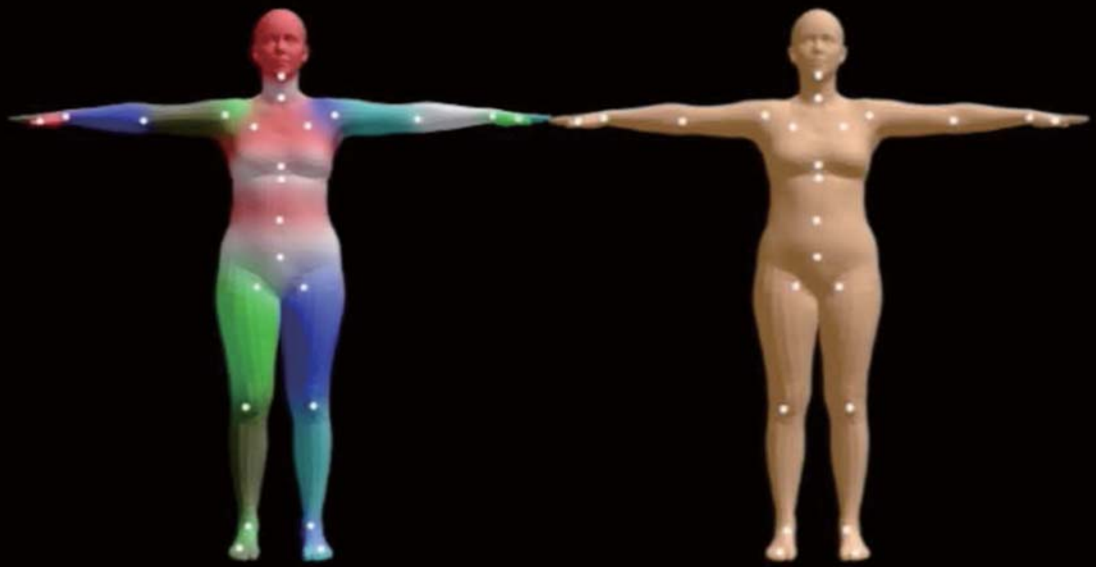
- Makes training easier
- Enables compatibility

SMPL Model Pipeline



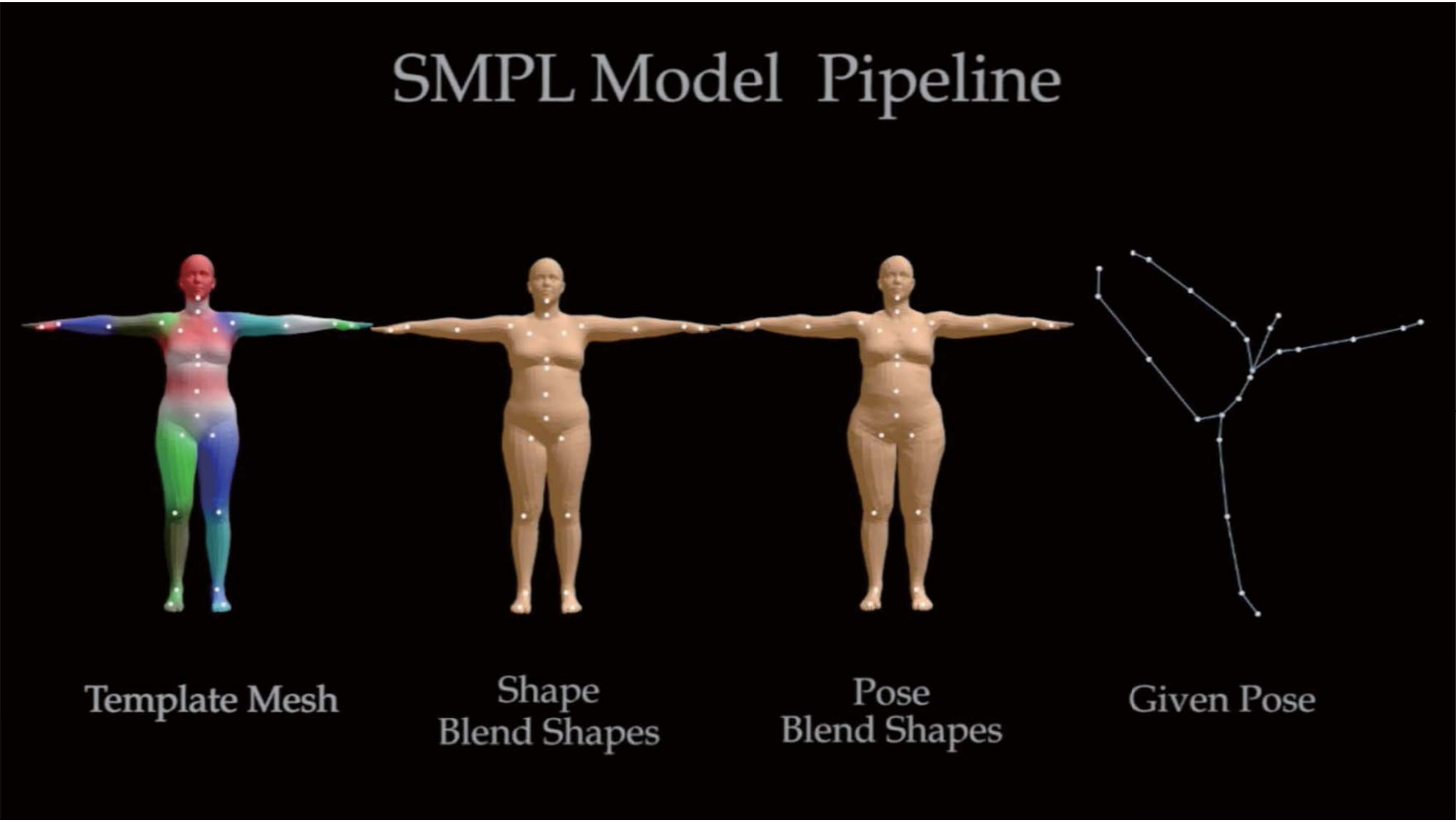
Template Mesh

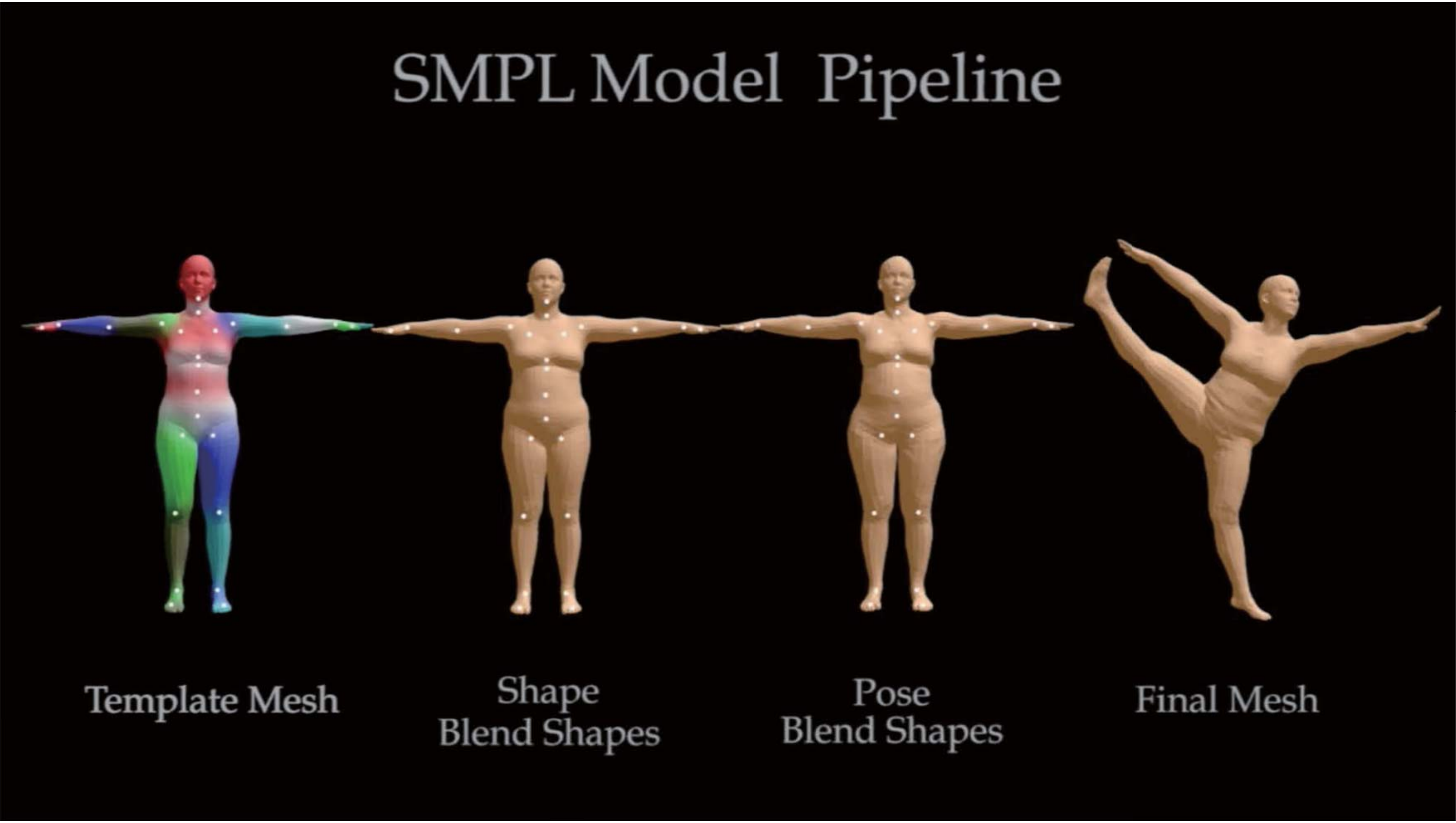
SMPL Model Pipeline



Template Mesh

Shape
Blend Shapes


SMPL Model Pipeline
Template Mesh
Shape
Blend Shapes
Pose
Blend Shapes
Given Pose


SMPL Model Pipeline
Template Mesh
Shape
Blend Shapes
Pose
Blend Shapes
Final Mesh

Parameterized Skinning

Standard skinning $W(\boxed{\mathbf{T}}, \boxed{\mathbf{J}}, \mathcal{W}, \vec{\theta}) \mapsto \text{vertices}$

SMPL model

$$M(\vec{\theta}, \vec{\beta}) = W(\boxed{\mathbf{T}_F(\vec{\beta}, \theta)}, \boxed{\mathbf{J}(\vec{\beta})}, \mathcal{W}, \vec{\theta}) \mapsto \text{vertices}$$

SMPL is skinning parameterized by pose $\vec{\theta}$ and shape $\vec{\beta}$

SMPL: BS are a parametric function of pose

- We parameterize the skinning equation by pose

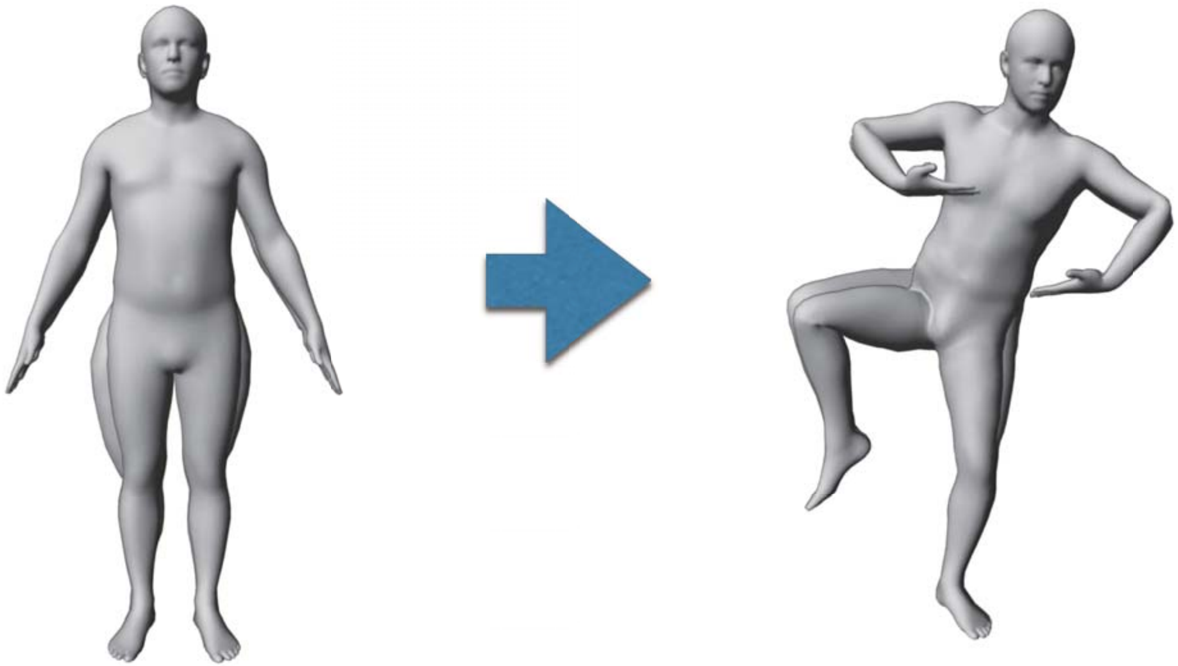
$$W(\mathbf{T}, \mathbf{J}, \mathcal{W}, \vec{\theta})$$

$$\Downarrow$$

$$W(T(\theta), \mathbf{J}, \mathcal{W}, \vec{\theta})$$

Remember: Pose Blend Shapes

- **With** blend shape correction



Parameterized Skinning

$$W(T(\theta), \mathbf{J}, \mathcal{W}, \vec{\theta}) \mapsto \text{vertices}$$

$$T(\vec{\theta}) = \mathbf{T} + B_P(\vec{\theta})$$

- Rest vertices are linear in $f(\theta)$

Each is a blend shape

$$B_P(\vec{\theta}) = \sum_{i}^{|f(\vec{\theta})|} f_i(\vec{\theta})\mathbf{P}_i$$

Parameterized Skinning

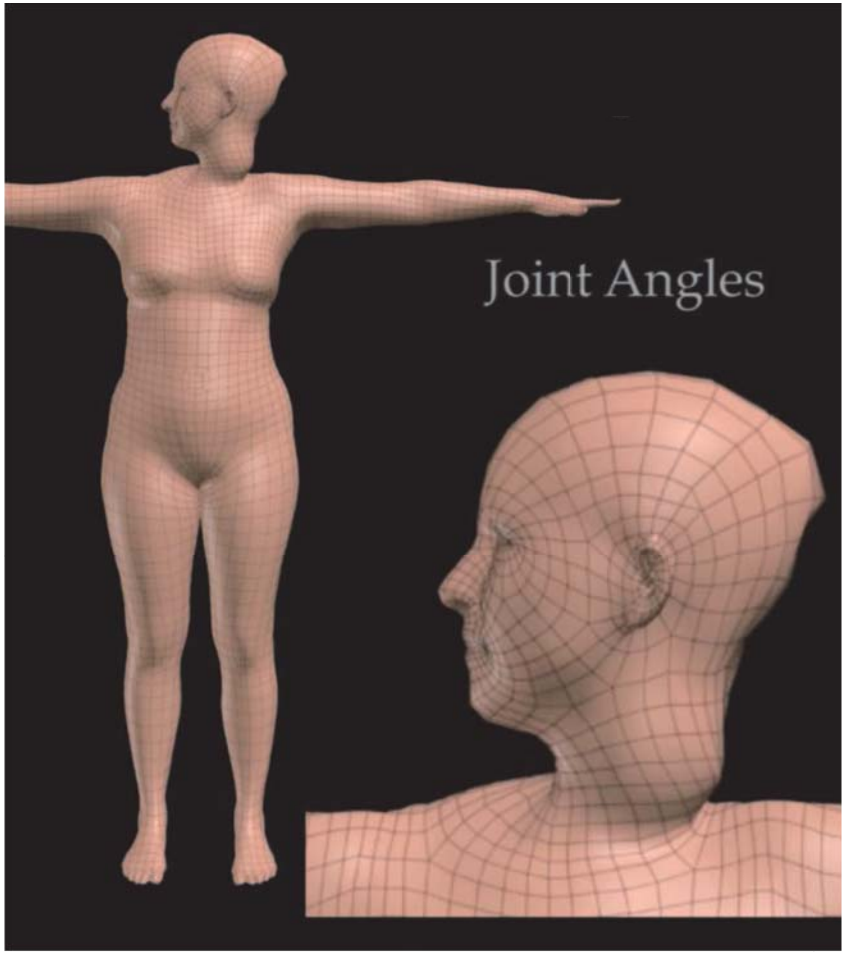
- What function $f(\vec{\theta})$?

$$B_P(\vec{\theta}) = \sum_i^{|f(\vec{\theta})|} f_i(\vec{\theta})\mathbf{P}_i$$

- Simplest possible:

$$f(\vec{\theta}) = \vec{\theta}$$

Neck Rotation



Parameterized Skinning

- What function $f(\vec{\theta})$?

$$B_P(\vec{\theta}) = \sum_{i}^{|f(\vec{\theta})|} f_i(\vec{\theta})\mathbf{P}_i$$

- Idea: we consider $f(\vec{\theta})$ as the vectorized joint rotation matrices

- Blend shapes are *linear in rotation matrix elements*

Pose Blend Shapes

$$B_P(\vec{\theta}) = \sum_{i}^{|f(\vec{\theta})|} f_i(\vec{\theta})\mathbf{P}_i$$

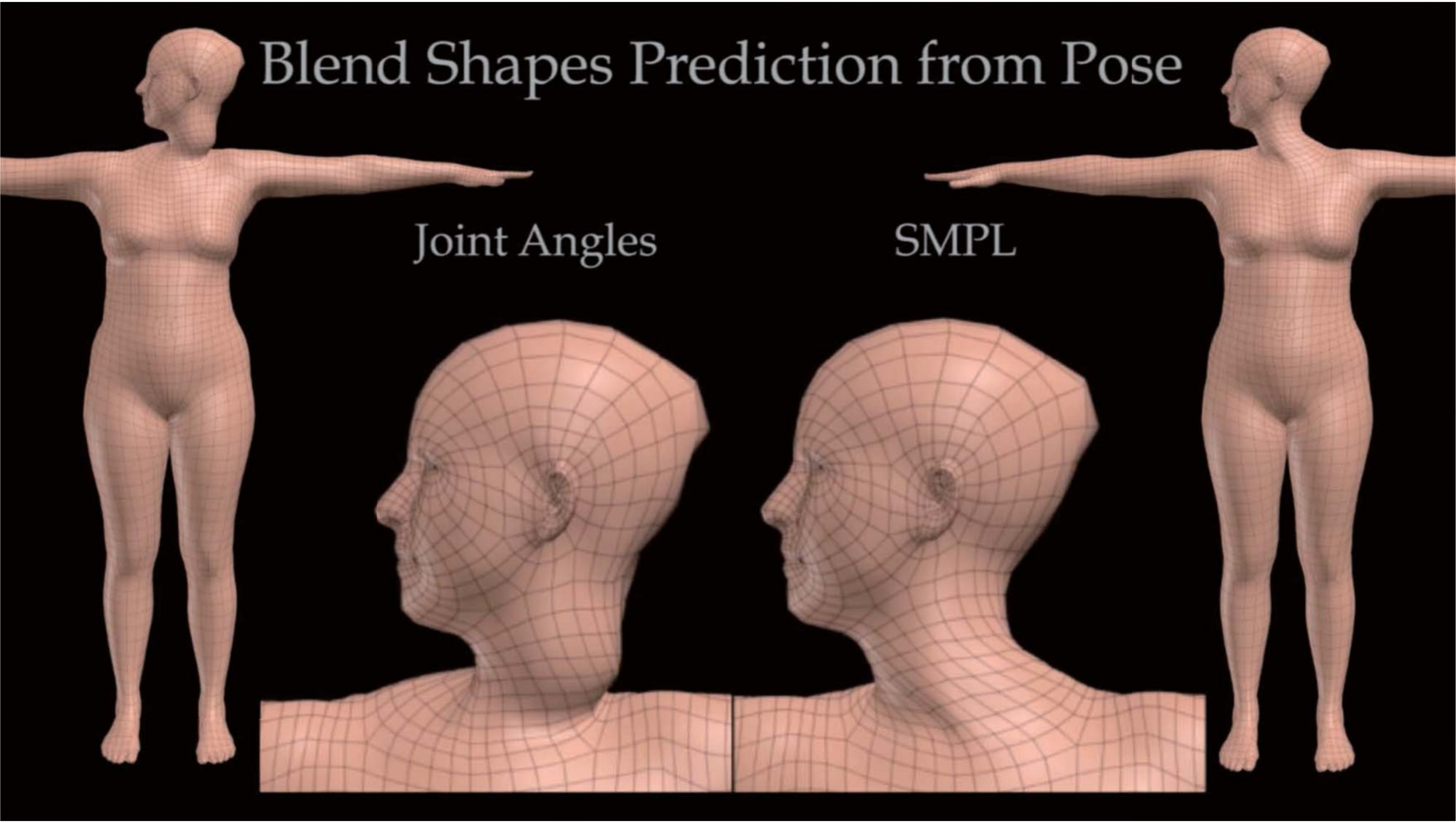
$$\vec{\theta} = (\vec{\omega}_1, \ldots, \vec{\omega}_k)^T$$

$$e^{\hat{\omega}_1} - \mathcal{I} \qquad e^{\hat{\omega}_K} - \mathcal{I}$$

Not a minus

$$f(\vec{\theta}) = [\underbrace{\bar{e}^{\hat{\omega}_1}_{1,1} \ldots \bar{e}^{\hat{\omega}_1}_{3,3}} \quad \ldots \quad \underbrace{\bar{e}^{\hat{\omega}_K}_{1,1} \ldots \bar{e}^{\hat{\omega}_K}_{3,3}}]$$

9 elements of the rotation matrix-> We learn 9xK=207 blendshapes

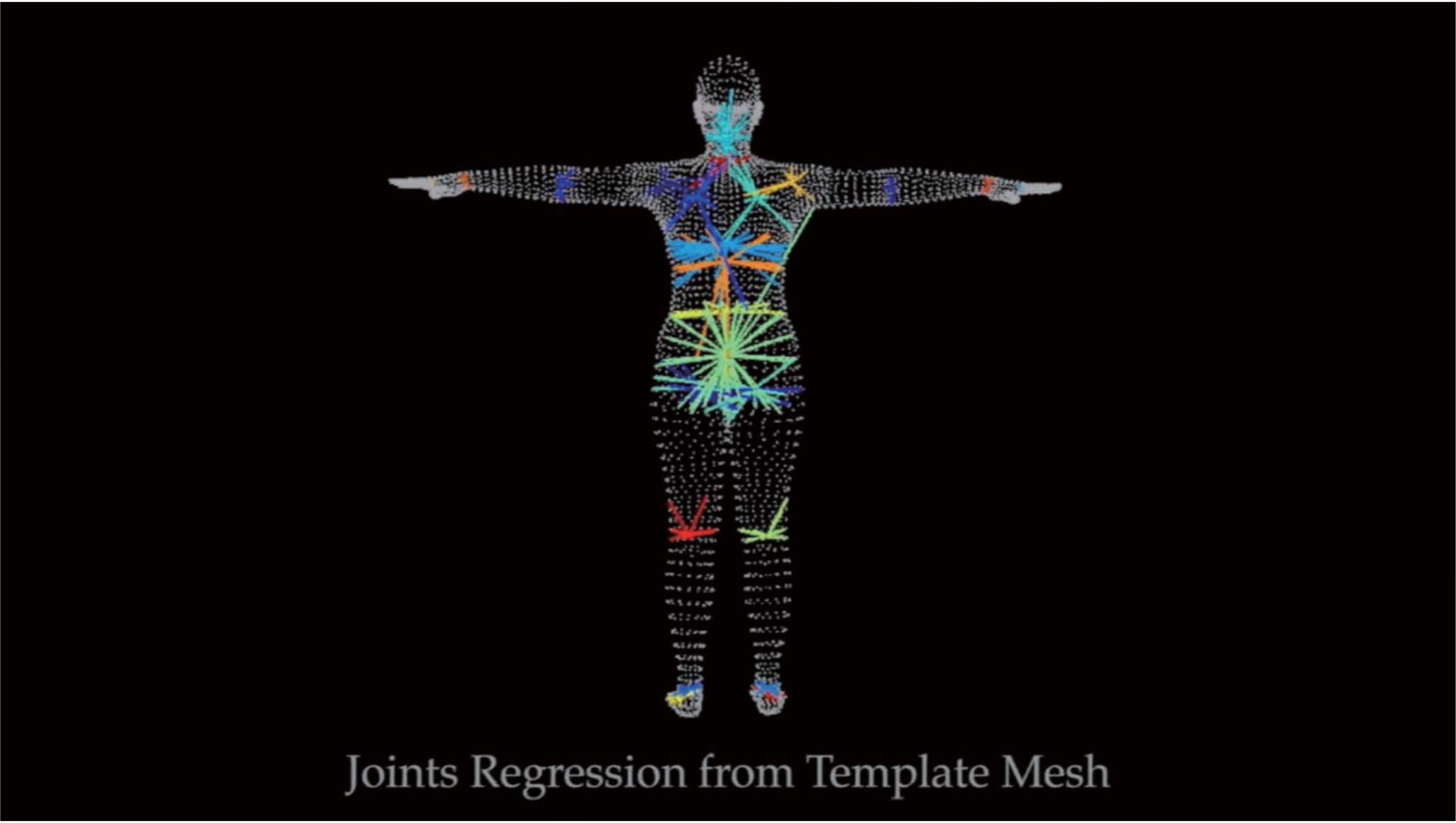

Blend Shapes Prediction from Pose
Joint Angles
SMPL

Joint Location Estimation

- How to get the joints $\mathbf{J}$ for a new shape?
- Joints are considered linear in rest vertices (much like in Allen et al. '06)

$$\mathbf{J} = J(\mathbf{T}; \mathcal{J}) = \mathcal{J}\mathbf{T}$$

Joint regressor matrix


Joints Regression from Template Mesh

SMPL Additive Model

$$\bar{\mathbf{t}}'_i = \underbrace{\sum_{k=1}^{K} w_{k,i} G'_k(\vec{\theta}, J(\vec{\beta}))}_{\text{Blendweights}} (\underbrace{\bar{\mathbf{t}}_i}_{\text{Vertices}} + \underbrace{\mathbf{b}_{S,i}(\vec{\beta})}_{\text{Shape-bs}} + \underbrace{\mathbf{b}_{P,i}(\vec{\theta})}_{\text{Pose-bs}})$$

Parameterized Skinning

Standard skinning $W(\boxed{\mathbf{T}}, \boxed{\mathbf{J}}, \mathcal{W}, \vec{\theta}) \mapsto \text{vertices}$

SMPL model

$$M(\vec{\theta}, \vec{\beta}) = W(\boxed{\mathbf{T}_F(\vec{\beta}, \theta)}, \boxed{\mathbf{J}(\vec{\beta})}, \mathcal{W}, \vec{\theta}) \mapsto \text{vertices}$$

SMPL is skinning parameterized by pose $\vec{\theta}$ and shape $\vec{\beta}$

SMPL

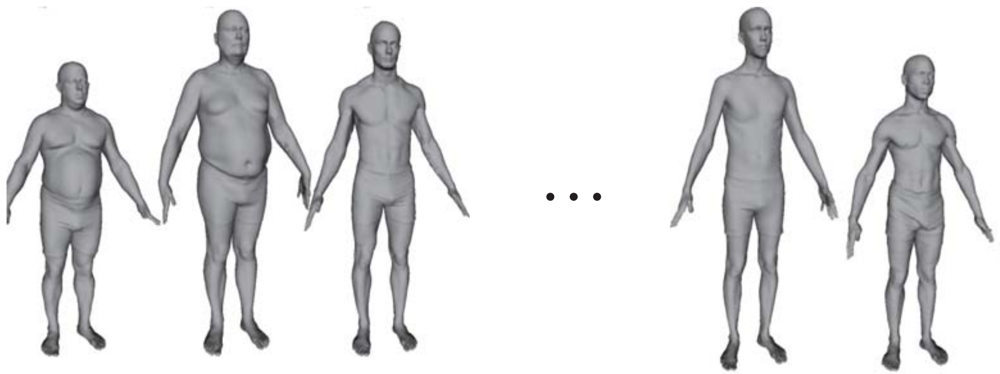
$$M(\underbrace{\overset{\text{pose}}{\vec{\theta}}, \overset{\text{shape}}{\vec{\beta}};}_{\text{Input}} \underbrace{\mathbf{T}, \mathcal{S}, \mathcal{P}, \mathcal{W}, \mathcal{J}}_{\substack{\text{Model parameters to} \\ \text{be learned from data}}})$$

$\mathbf{T}$	Template (average shape)
$\mathcal{S}$	Shape blend shape matrix
$\mathcal{P}$	Pose blend shape matrix
$\mathcal{W}$	Blendweights matrix
$\mathcal{J}$	Joint regressor matrix

DATA

Model Training

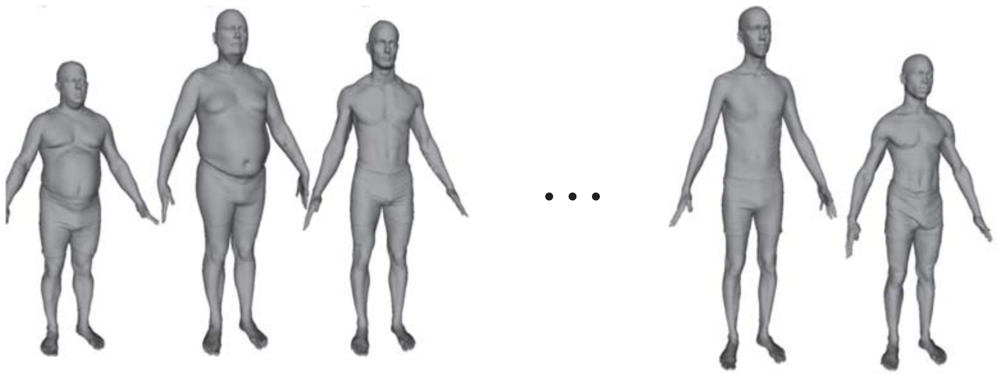
$$\mathbf{w} = \arg\min_{\mathbf{w}} \sum_j \| M(\vec{\theta}, \vec{\beta}; \mathbf{w}) - \quad \cdots \quad \|^2$$



$$\begin{bmatrix}\mathbf{V}_1 & \mathbf{V}_2 & \dots & \mathbf{V}_{N_{\text{subj}}}\end{bmatrix} = \mathbf{T} + \begin{bmatrix}\boxed{\mathbf{S}_1 \quad \mathbf{S}_2 \quad \dots} & \mathbf{S}_{N_{\text{subj}}}\end{bmatrix}\mathcal{B}$$

Average of shapes

Shape blend shapes are the first eigenvectors

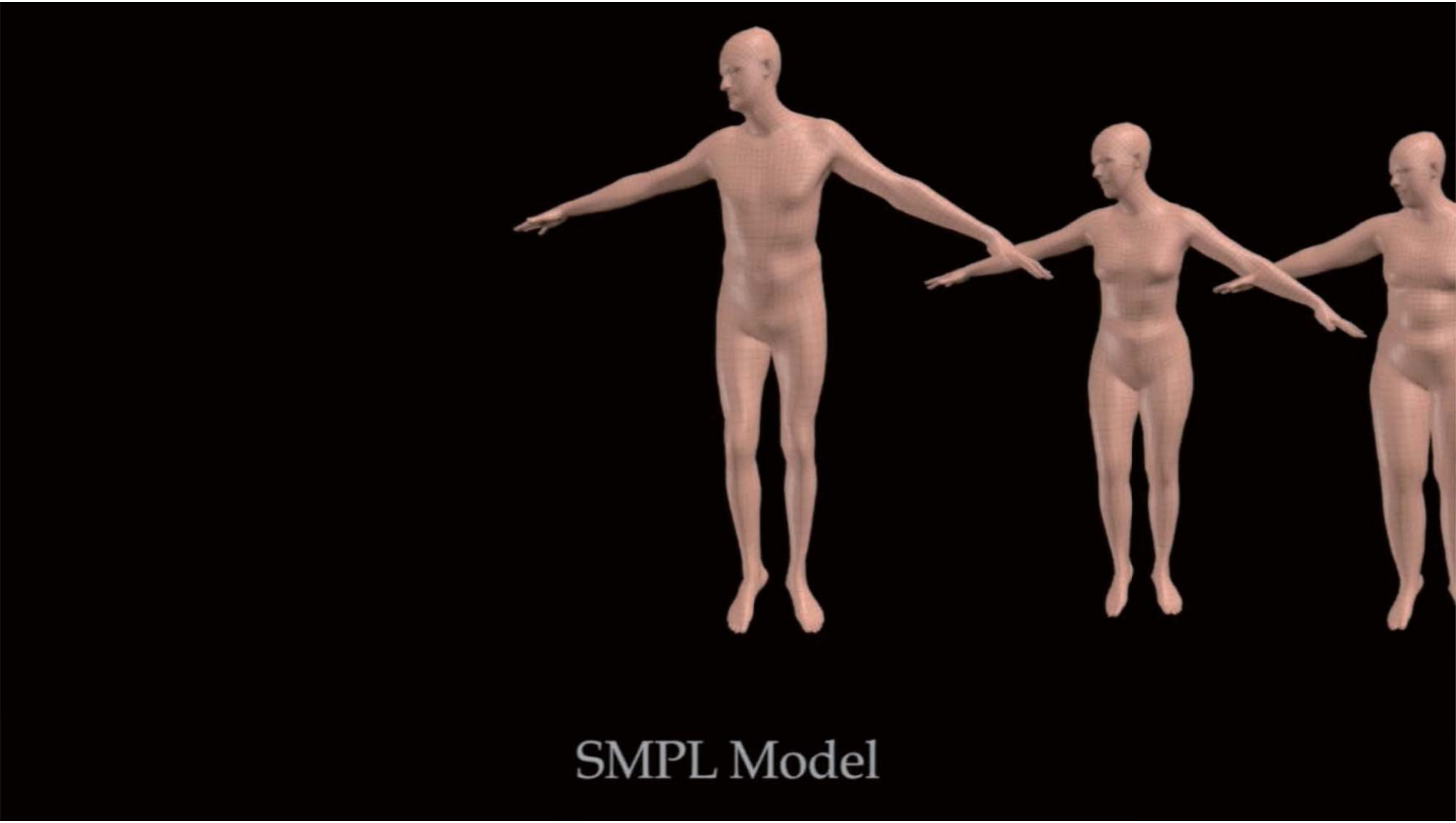


$$\left[\mathbf{V}_1 \quad \mathbf{V}_2 \quad \dots \quad \mathbf{V}_{N_{\mathrm{subj}}}\right] \approx \mathbf{T} + \mathcal{S}\mathcal{B}$$

Average of shapes

Shape blend shapes matrix

Before doing PCA all shapes have to be in the same pose (pose needs to be optimized)

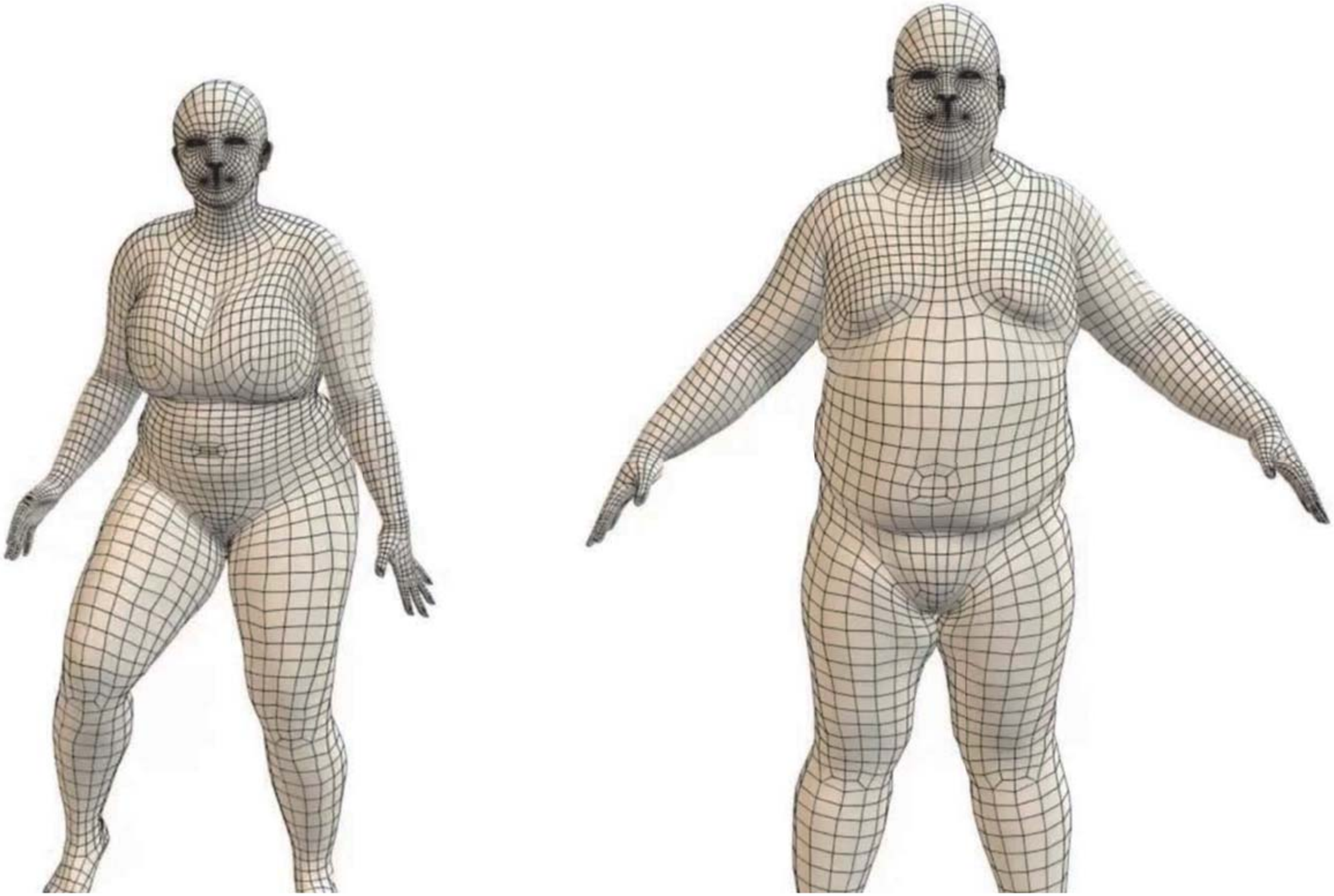

SMPL Model

SMPL conclusions

- **Speed**: fast run-time
- **Fidelity**: superior accuracy to Blend-SCAPE, trained on the same data
- **Compatibility**: works in Maya, Unity, ...
- Is publicly available for research purposes

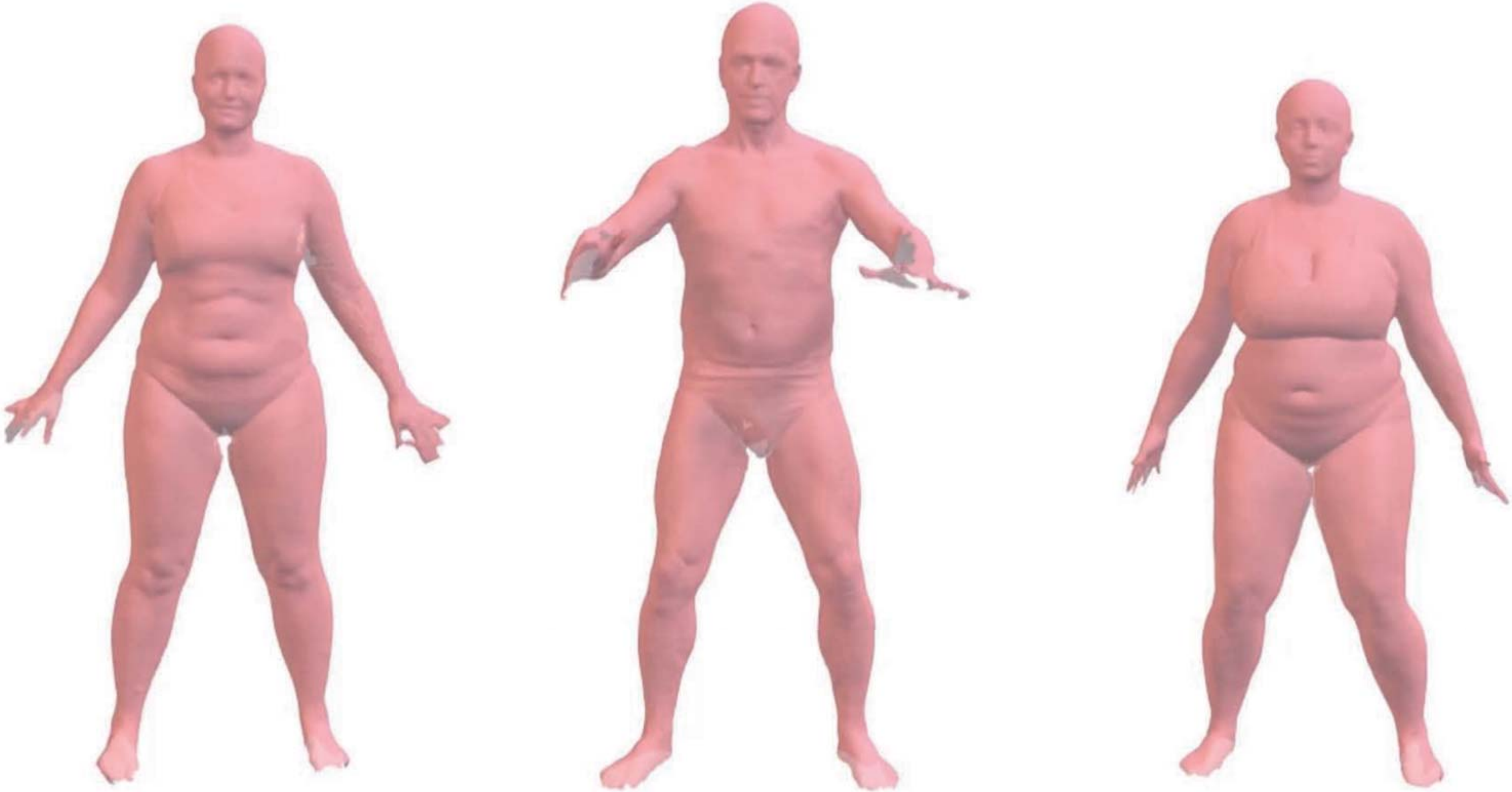
Download: http://smpl.is.tue.mpg.de

Dyna: A model of how we jiggle



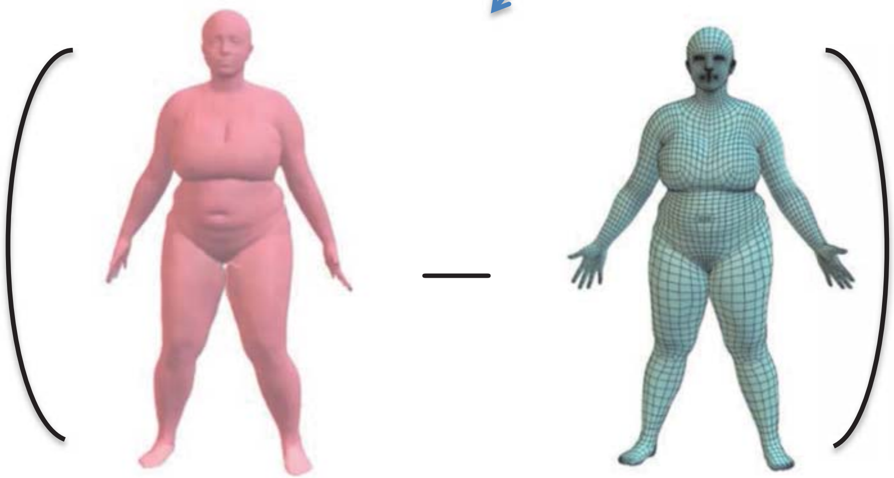
G. Pons-Moll,
J. Romero,
N. Mahmood,
M. Black
SIGGRAPH'15

Raw 4D scans



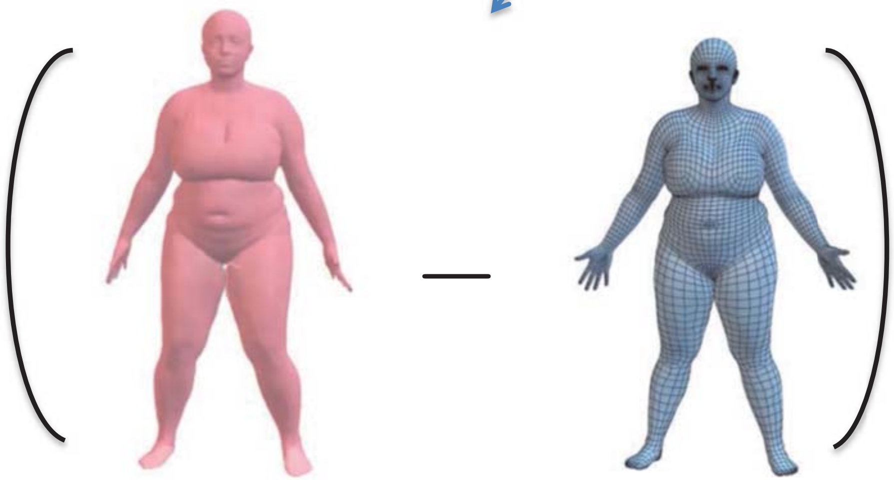
Registration

$$E(\mathbf{v}) = \sum_{\mathbf{s}_i \in \mathcal{S}} \mathrm{dist}(\mathbf{s}_i, \mathcal{A}(\mathbf{v})) + E_{\mathrm{prior}}(\mathbf{v})$$



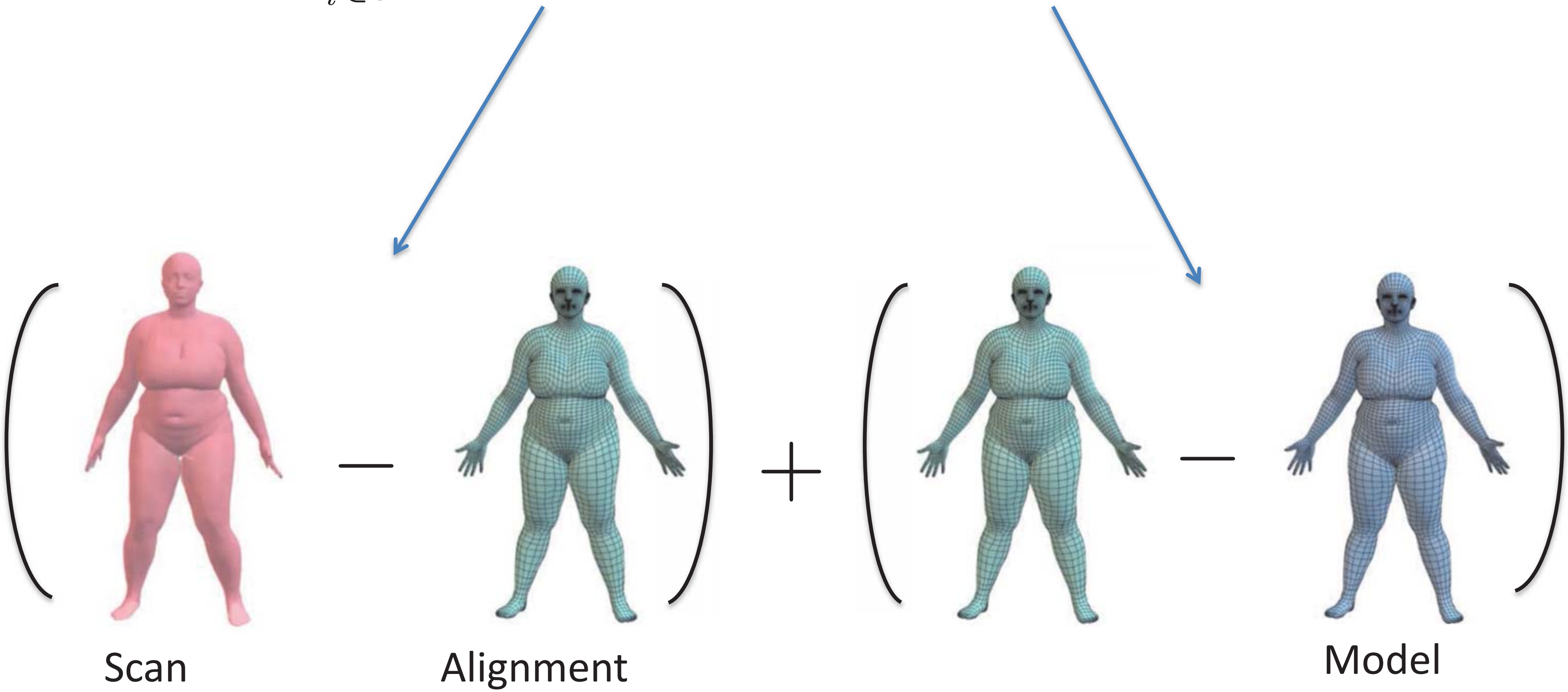
Scan Alignment

$$E(\theta, \beta) = \sum_{\mathbf{s}_i \in \mathcal{S}} \mathrm{dist}(\mathbf{s}_i, \mathcal{M}(\theta, \beta)) + E_{\mathrm{prior}}(\theta, \beta)$$

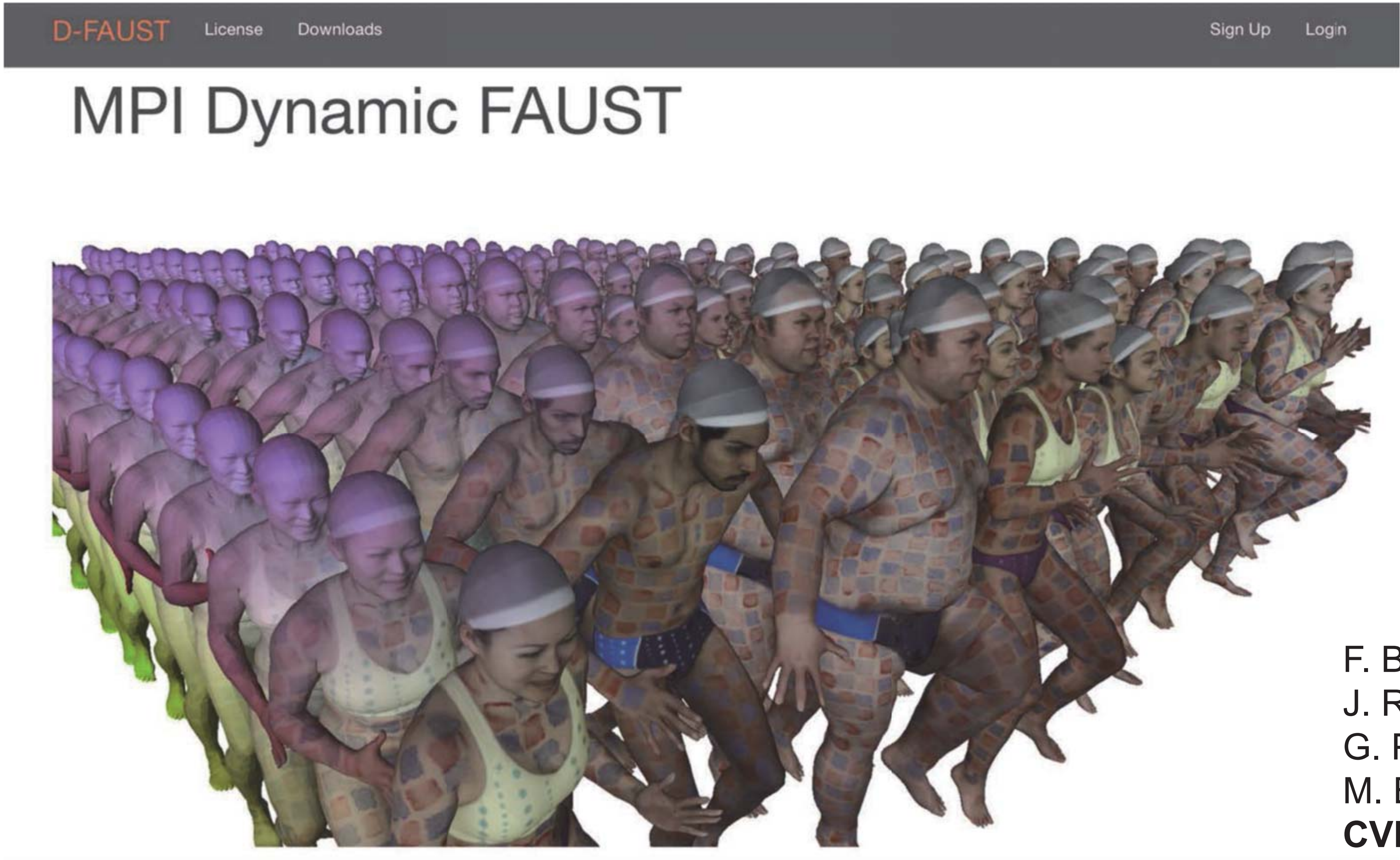


Scan Model

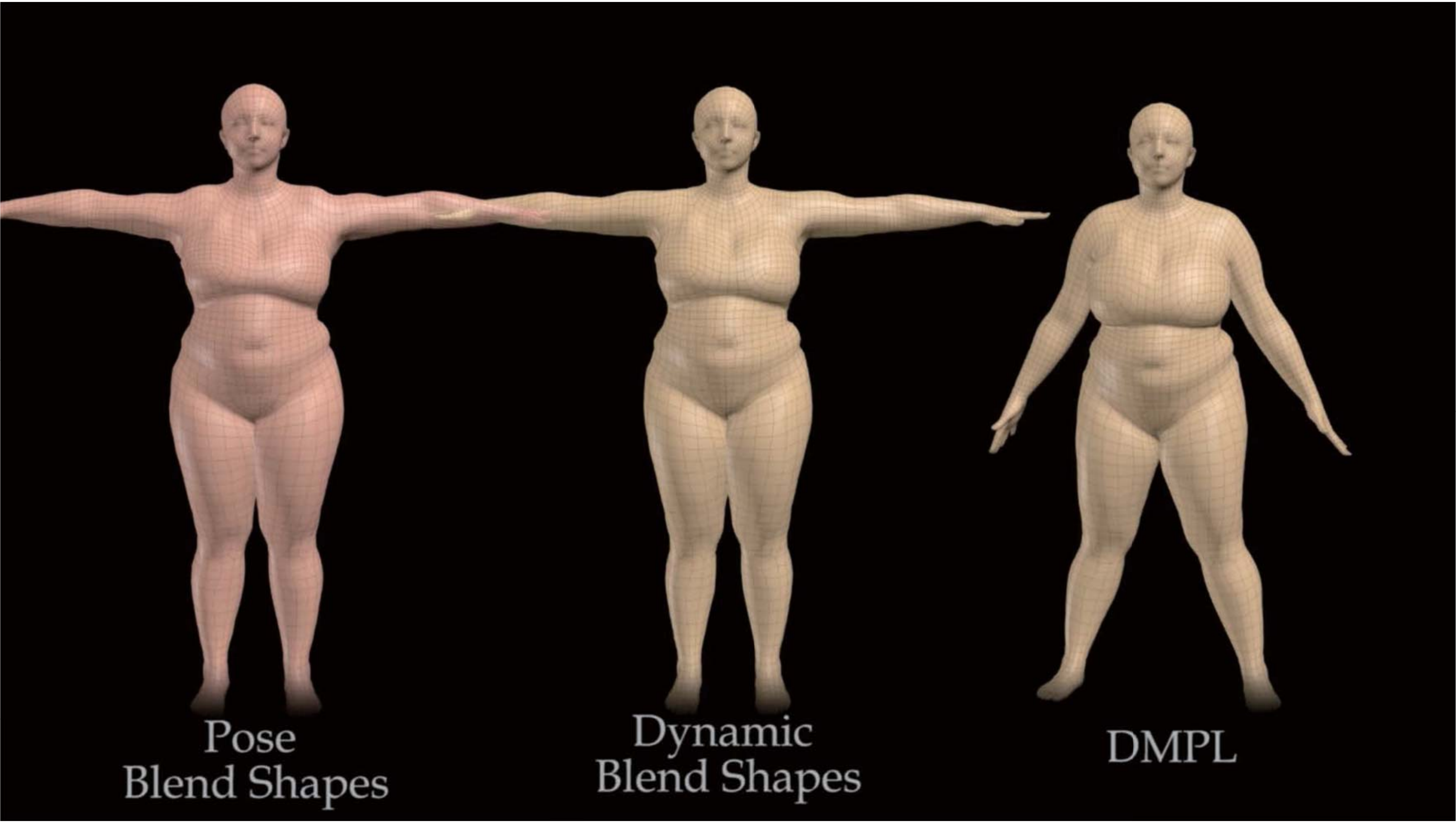
$$E(\theta, \beta, \mathbf{v}) = \sum_{\mathbf{s}_i \in \mathcal{S}} \mathrm{dist}(\mathbf{s}_i, \mathcal{A}(\mathbf{v})) + \mathrm{dist}(\mathcal{A}(\mathbf{v}), \mathcal{M}(\theta, \beta)) + E_{\mathrm{prior}}(\theta, \beta)$$



http://dfaust.is.tue.mpg.de



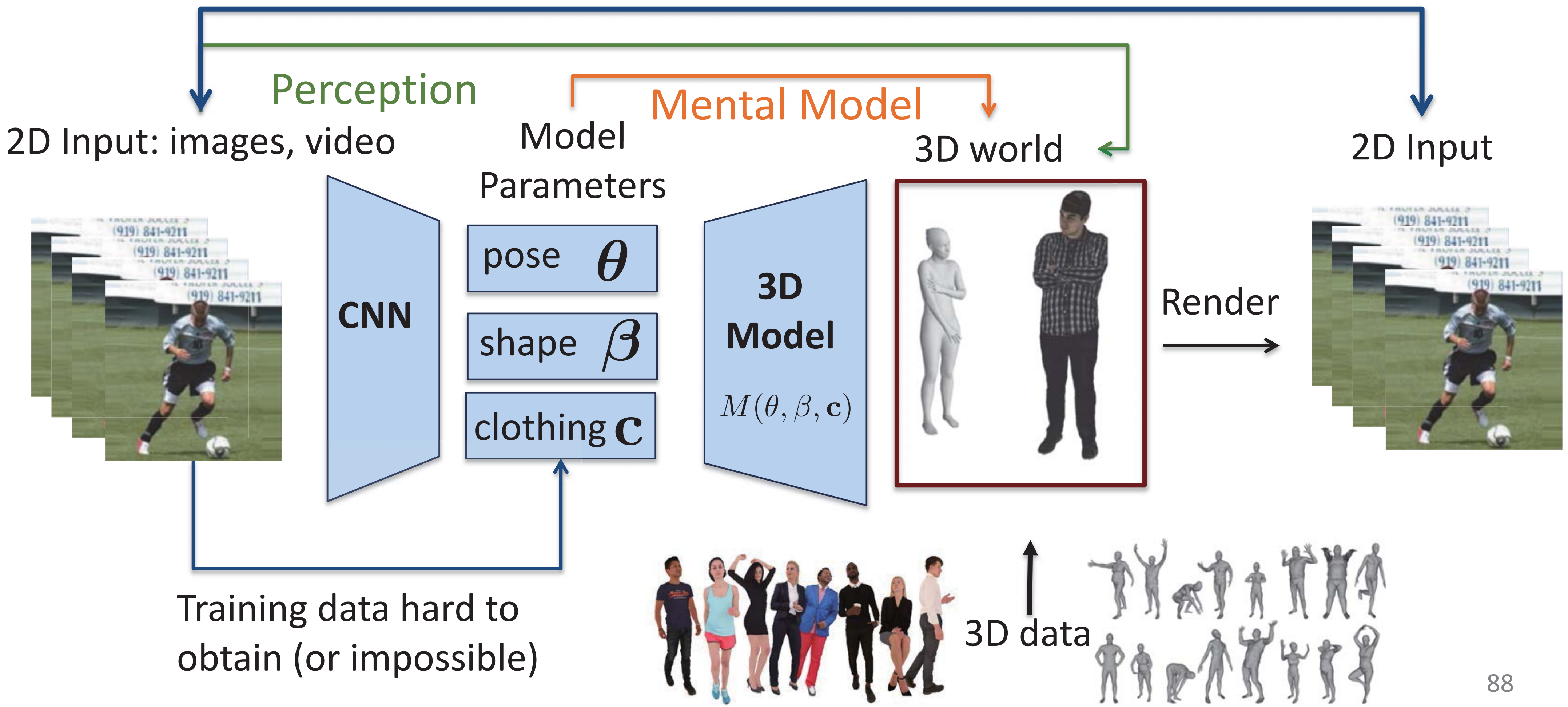
F. Bogo
J. Romero
G. Pons-Moll
M. Black
CVPR'17 [Oral]

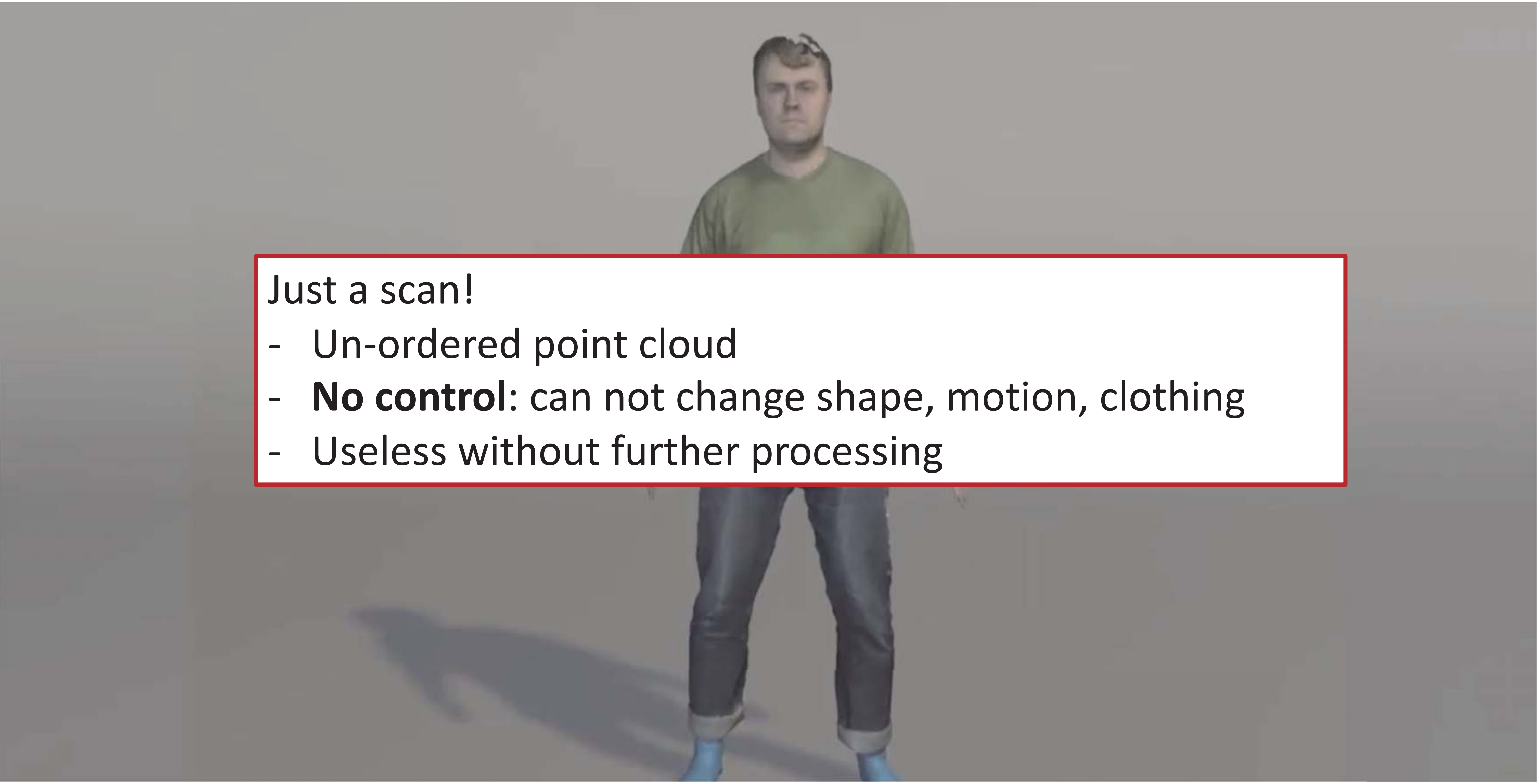

Pose
Blend Shapes
Dynamic
Blend Shapes
DMPL

DOES NOT SCALE TO THE REAL WORLD

Vision

Computer Vision + Computer Graphics + Learning





G. Pons-Moll, S. Pujades, S. Shu, M. Black – **SIGGRAPH ’17**

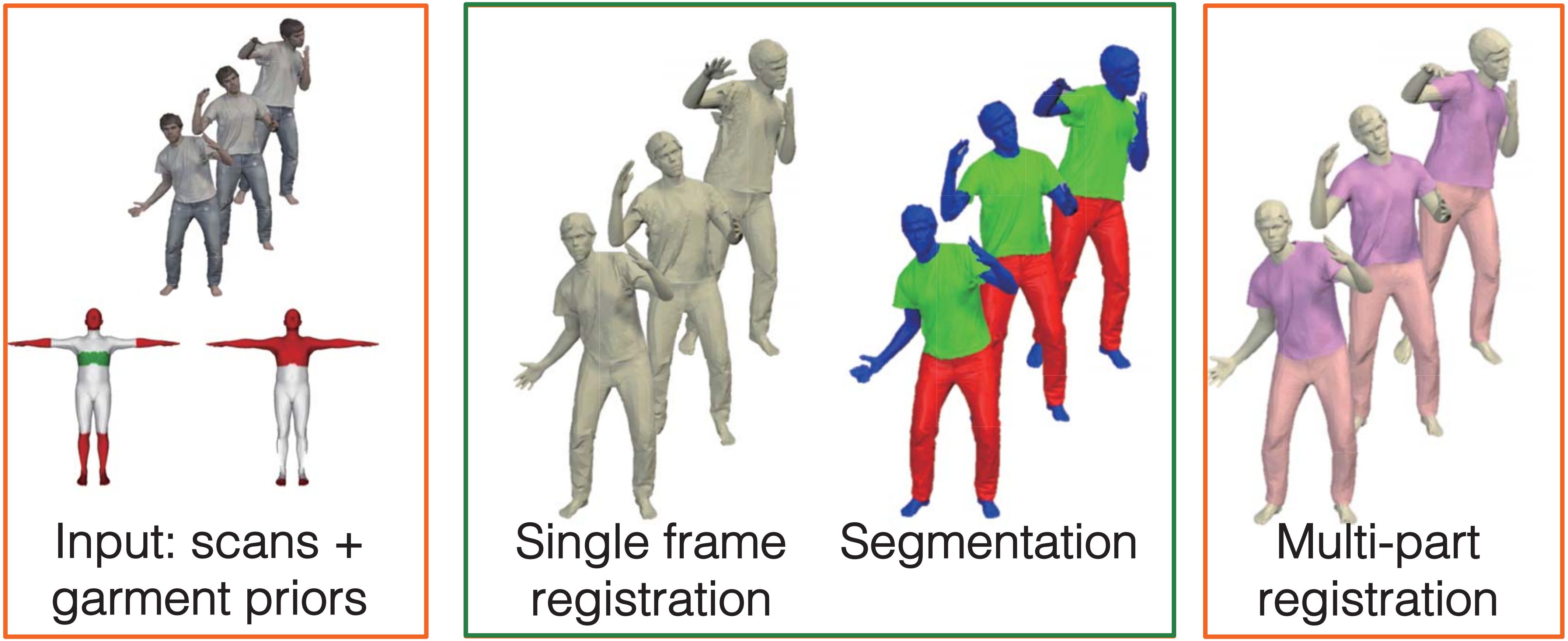

Cloth & Body
from ClothCap
Cloth from ClothCap
Body from ClothCap


4D Scan
ClothCap Result

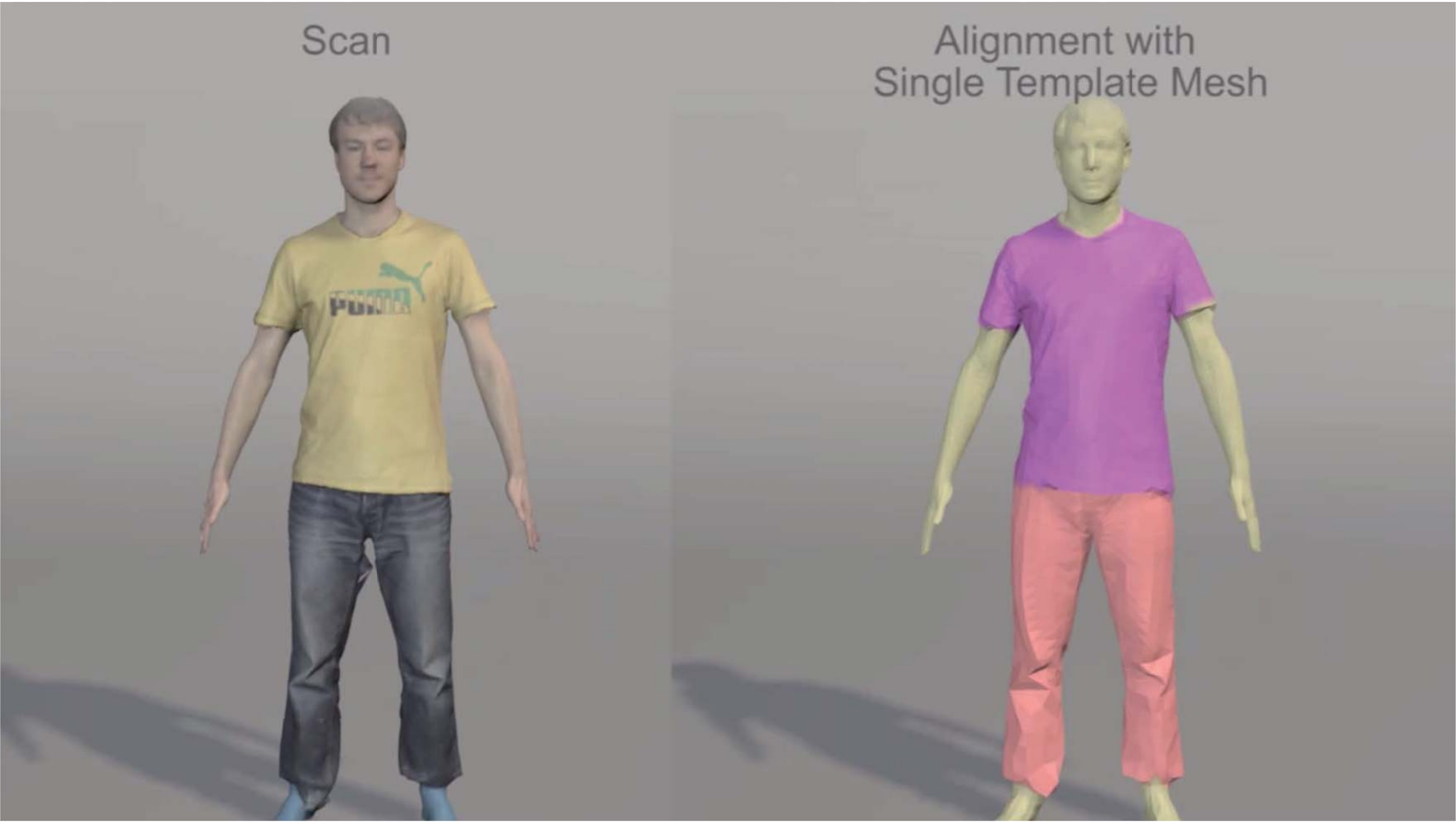

4D Scan
ClothCap Result
ClothCap Cloth on new Body


ClothCap Result
ClothCap Cloth on
new Body

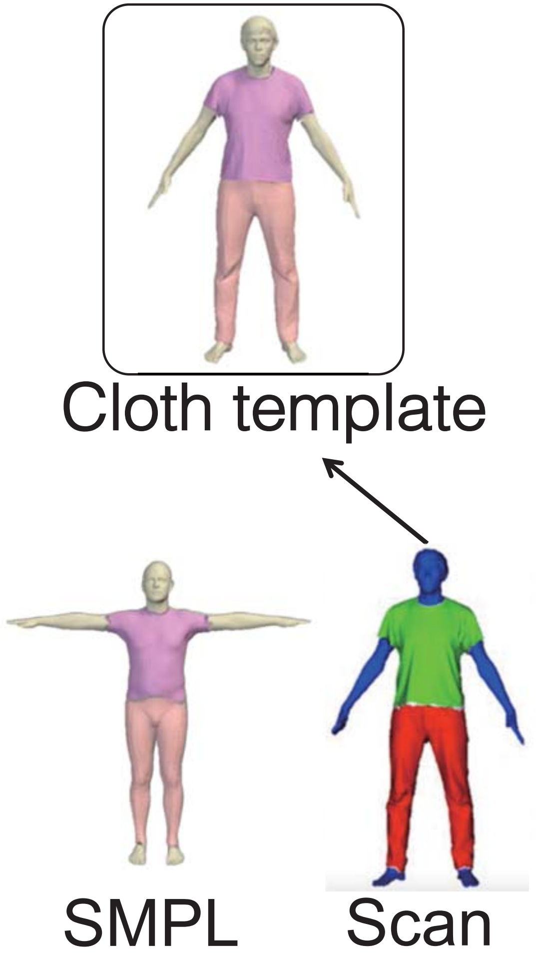
Overview


Input: scans +
garment priors
Single frame
registration
Segmentation
Multi-part
registration

Single Mesh Registration


Scan
Alignment with
Single Template Mesh
PUMA

Multi-part Mesh Registration

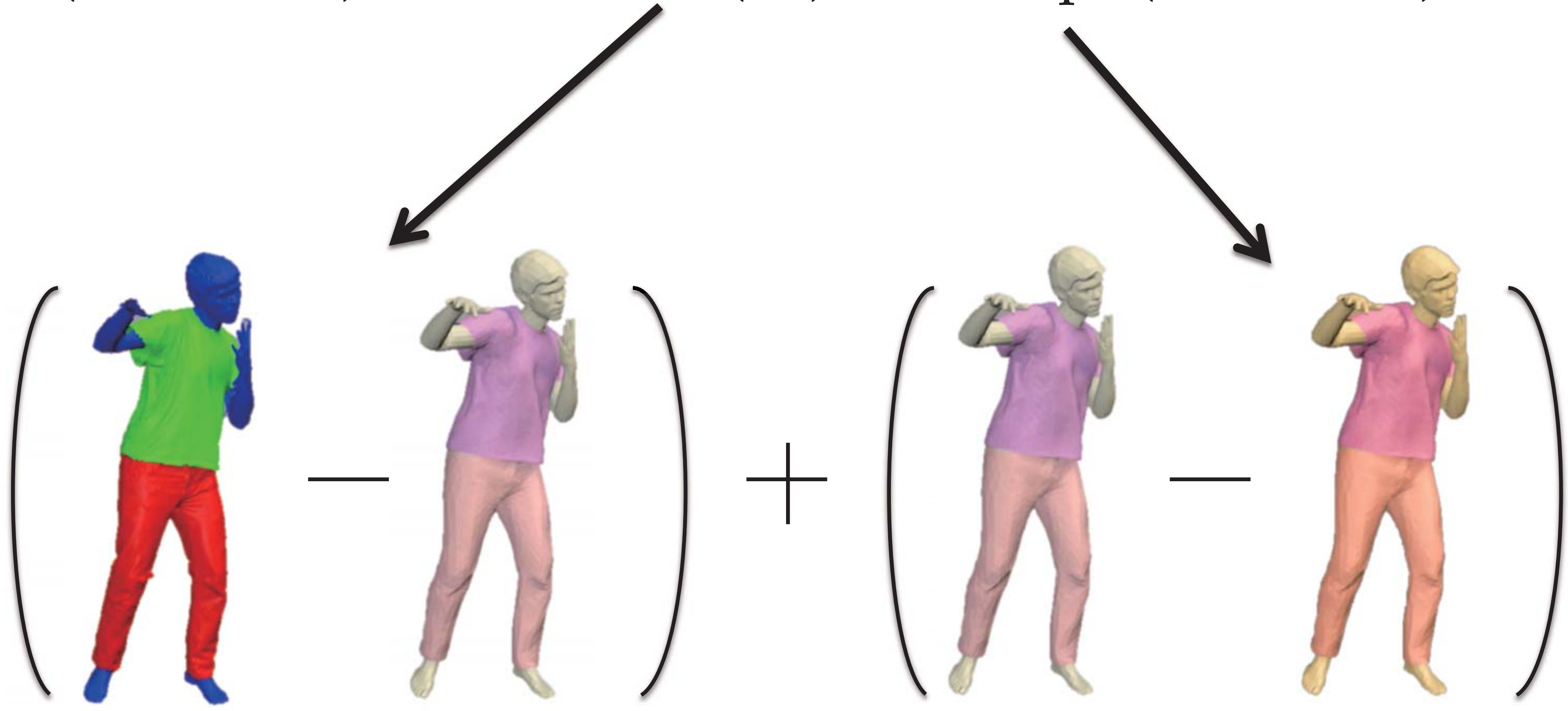


Multi-part Mesh Registration

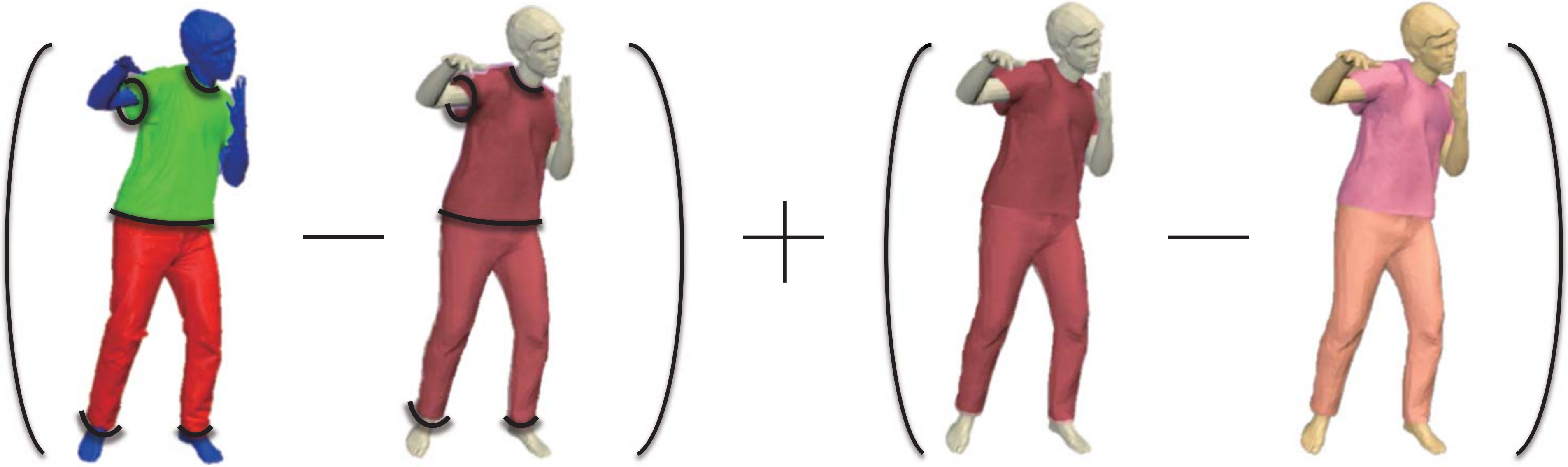
Multi-part Mesh Registration

Cloth
Template

$$E(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{v}) = E_{\mathrm{data}}(\mathbf{v}) + E_{\mathrm{cpl}}(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{v}) +$$



$$E(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{v}) = E_{\mathrm{data}}(\mathbf{v}) + E_{\mathrm{cpl}}(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{v}) + \\ + \underline{E_{\mathrm{boundary}}}(\mathbf{v}) + \underline{E_{\mathrm{lap}}(\mathbf{v})}$$

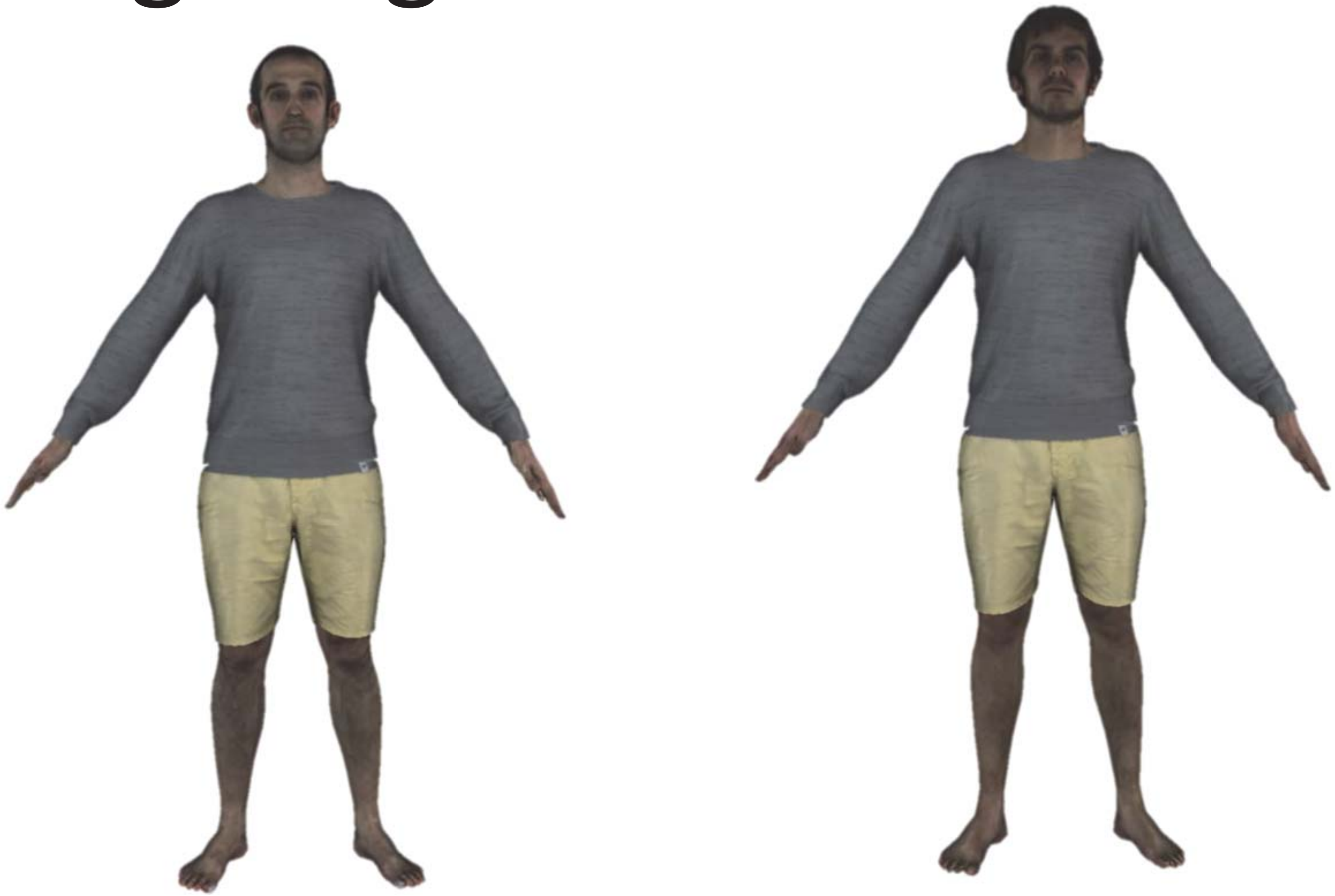


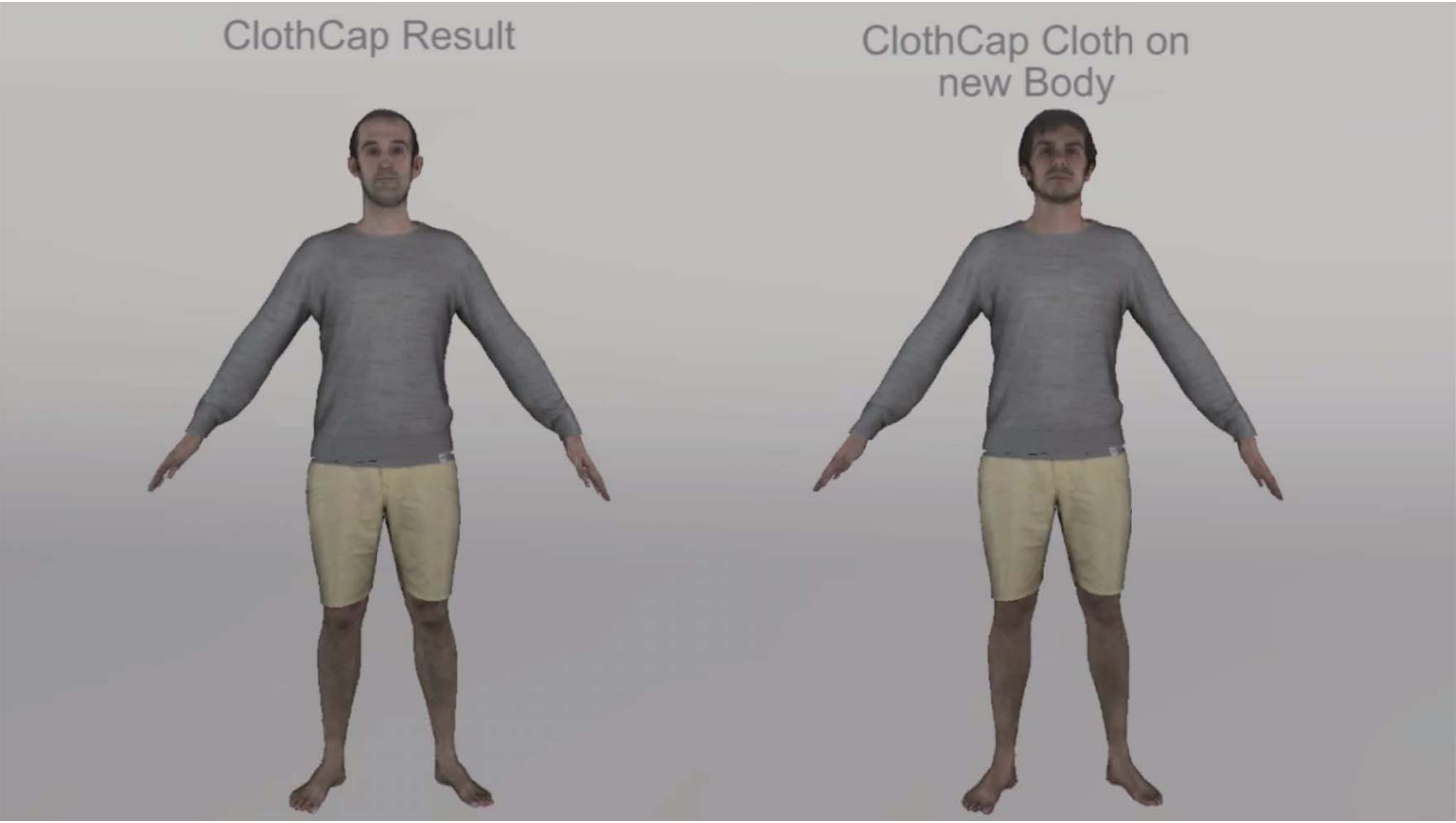
Retargeting Cloth to a New Body

Retargeting Cloth to a New Body

Retargeting Cloth to a New Body

Retargeting Cloth to a New Body

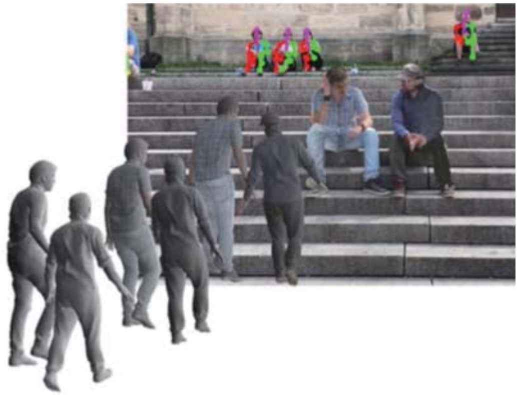



ClothCap Result
ClothCap Cloth on
new Body


CAESAR Dataset [Robinette, et al. 2002]
Male Subjects

Scans

Zhang et al. CVPR’17. BUFF http://buff.is.tue.mpg.de

Real Virtual Humans

http://virtualhumans.mpi-inf.mpg.de/

- Resources data and code available for research!
- **Open positions** in the areas of computer vision, machine learning and computer graphics with focus on analyzing and modelling people

Schedule

- Virtual human models
  - Kinematic Chains, Linear Blend Skinning, Blendshapes
  - SMPL & Dyna
  - ClothCap: Capturing people in clothing

- Capturing humans from consumer sensors
  - 3D human reconstruction from a video
  - 3D human pose and shape from images
  - 3D human pose from Inertial Measurement Units (IMU)

Challenges in Capturing Humans from Images and Video

- Depth ambiguities
- Articulation
- Clothing
- Illumination
- Background

Model Based Approaches

$$\arg\min_{\boldsymbol{\theta},\boldsymbol{\beta}} \mathrm{dist}(\hat{\mathbf{z}}\left(M(\boldsymbol{\theta},\boldsymbol{\beta})\right),\mathbf{z})$$

3D world

$$\xrightarrow{P(\cdot)}$$

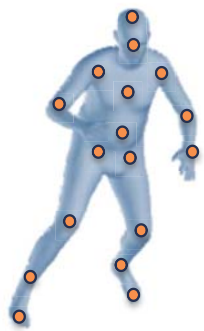
Image $\mathbf{z}$

Pons-Moll and Rosenhahn.
Model Based Pose Estimation 2011

Model Based Approaches

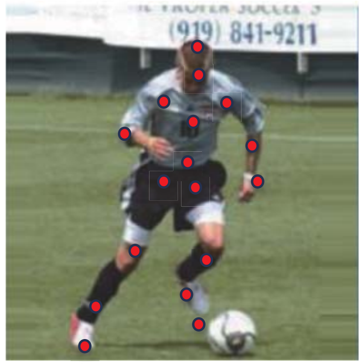
$$\arg\min_{\boldsymbol{\theta},\boldsymbol{\beta}} \mathrm{dist}(\hat{\mathbf{z}}\left(M(\boldsymbol{\theta},\boldsymbol{\beta})\right),\mathbf{z})$$

3D world



$\xrightarrow{P(\cdot)}$

2D Keypoints $\mathbf{z}$



Bogo et al. '16
Lassner et al. '17

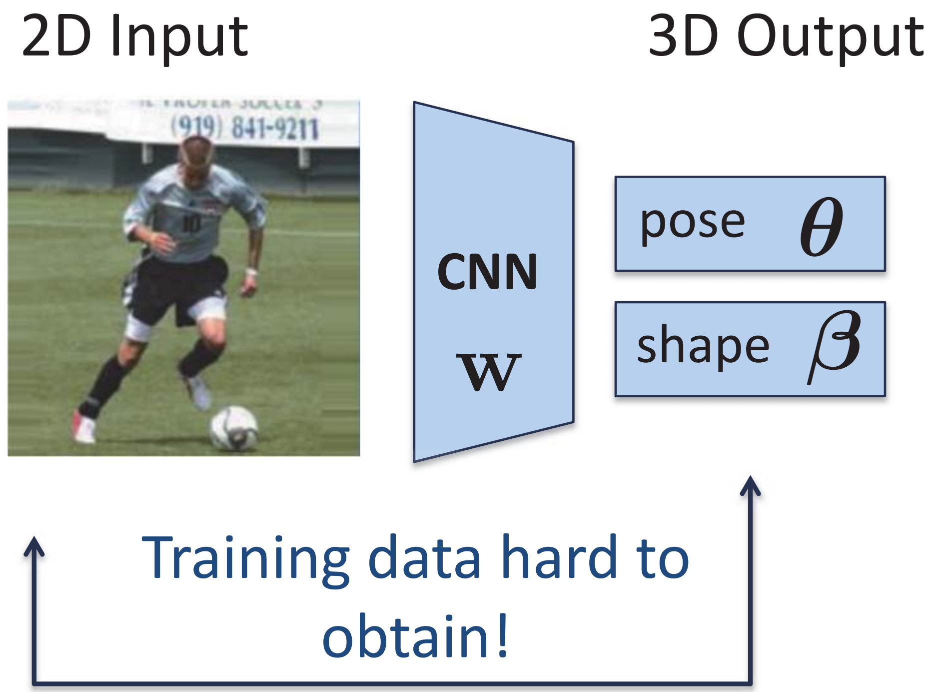
Model Based Approaches

$$\arg\min_{\boldsymbol{\theta},\boldsymbol{\beta}} \mathrm{dist}(\hat{\mathbf{z}}\,(M(\boldsymbol{\theta},\boldsymbol{\beta})),\mathbf{z})$$

Requires **careful initialization**
Optimization can be **slow**

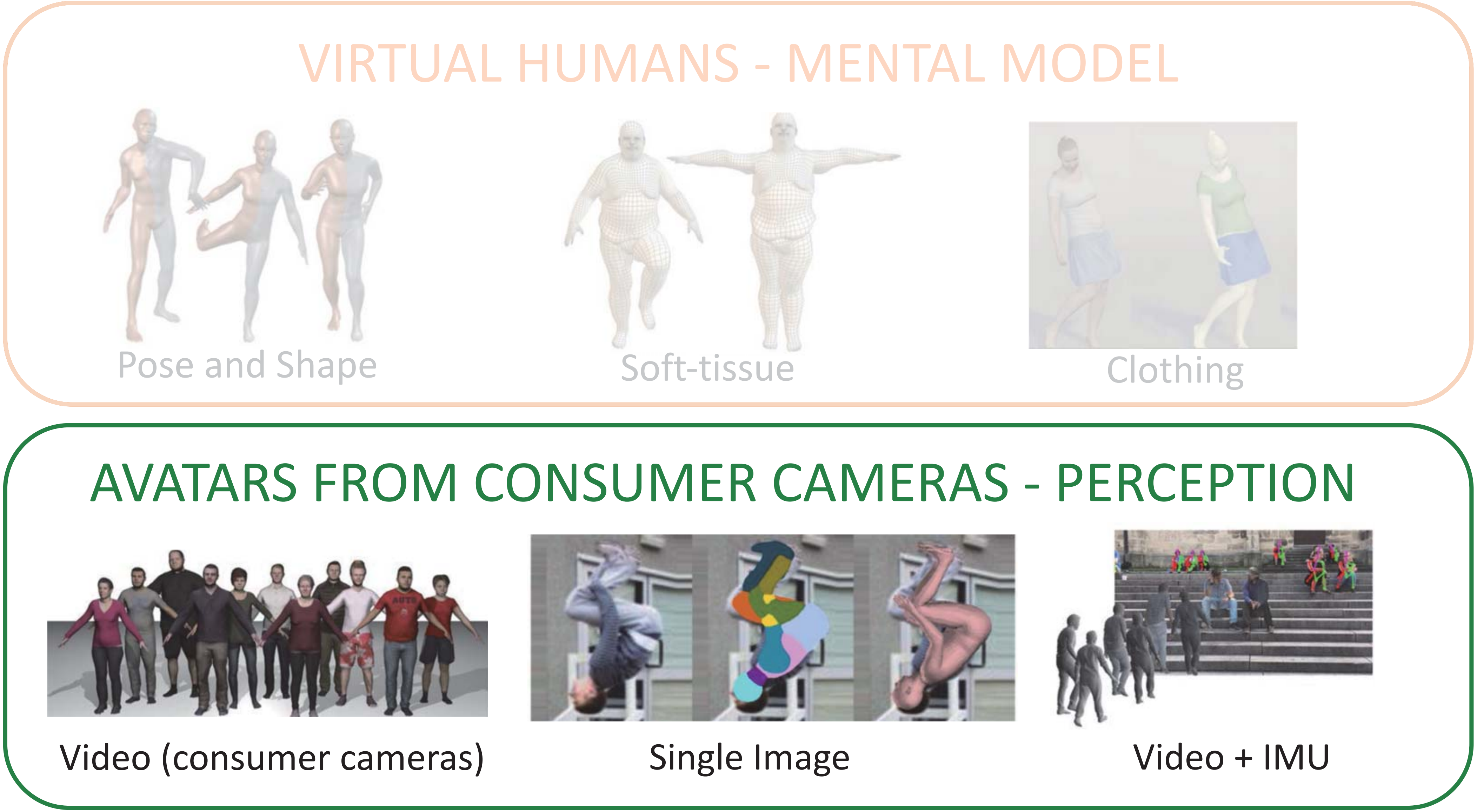
Bogo et al. '16
Lassner et al. '17

Learning Based Approaches



Remaining Problems

- Current methods can not recover personalized shapes: **no clothing, hair, appearance**
- Optimization can be **slow**
- Optimization requires **initialization**
- **Lack of 3D data** for learning methods

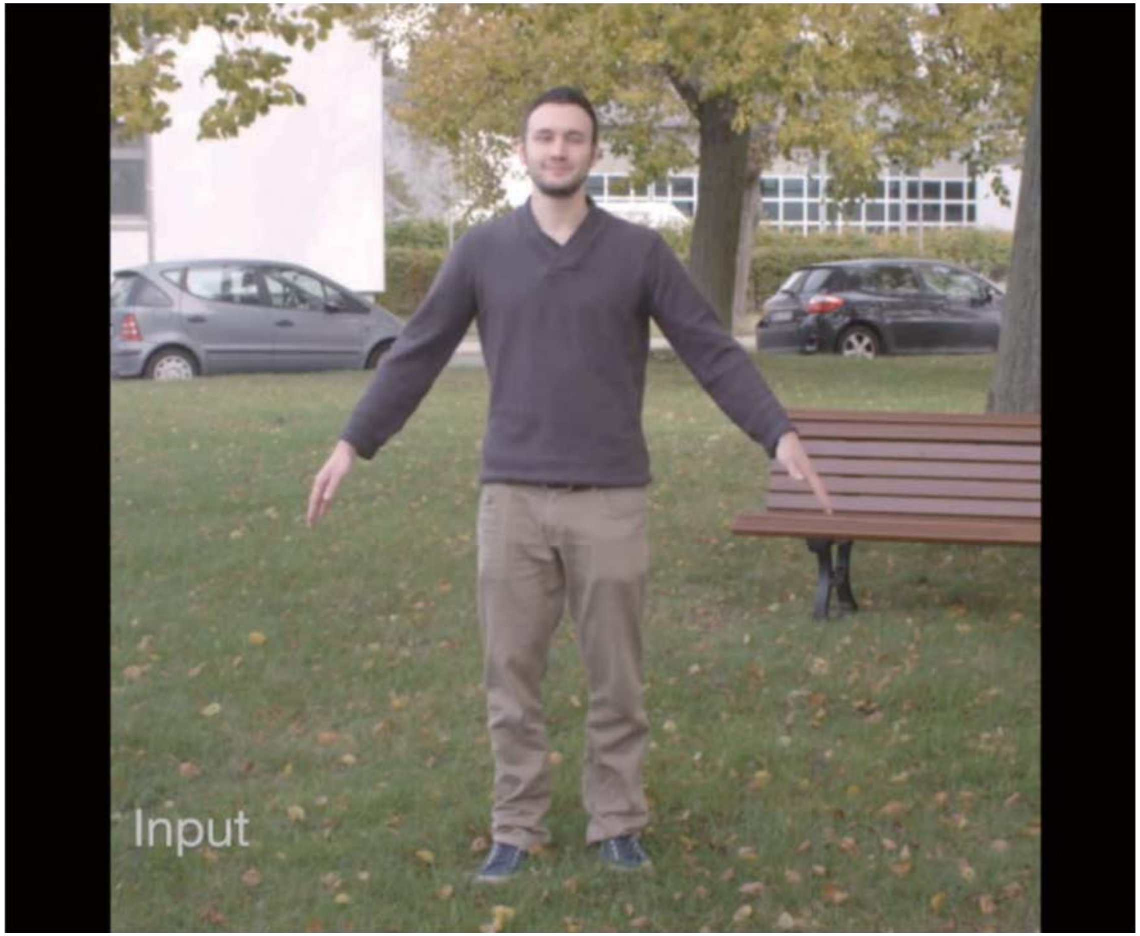

VIRTUAL HUMANS - MENTAL MODEL
Pose and Shape
Soft-tissue
Clothing
AVATARS FROM CONSUMER CAMERAS - PERCEPTION
Video (consumer cameras)
Single Image
Video + IMU

Video-Based Reconstruction of 3D People Models

T.Alldieck, M.Magnor, W.Xu, C. Theobalt, G. Pons-Moll

CVPR'18 [spotlight]

Goal: 3D Reconstruction of People from a Single Video



Previous Work

No clothing, no personalization!

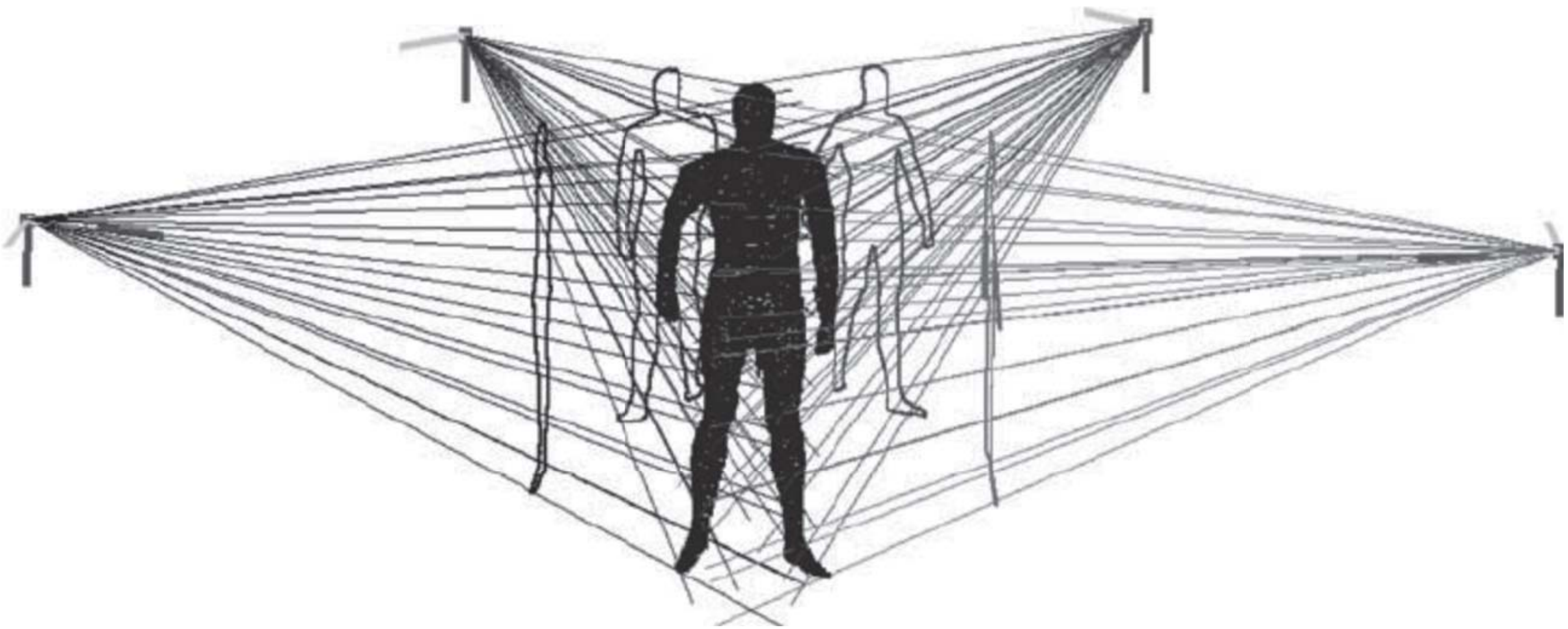
[Pavlakos et al. ’18]

[Kanazawa et al. ’18]

[Bogo et al. ’15]

Key Idea: Extend Visual Hulls to Dynamic Human Motion

Problem: standard visual hull requires a **static** object captured by multiple views

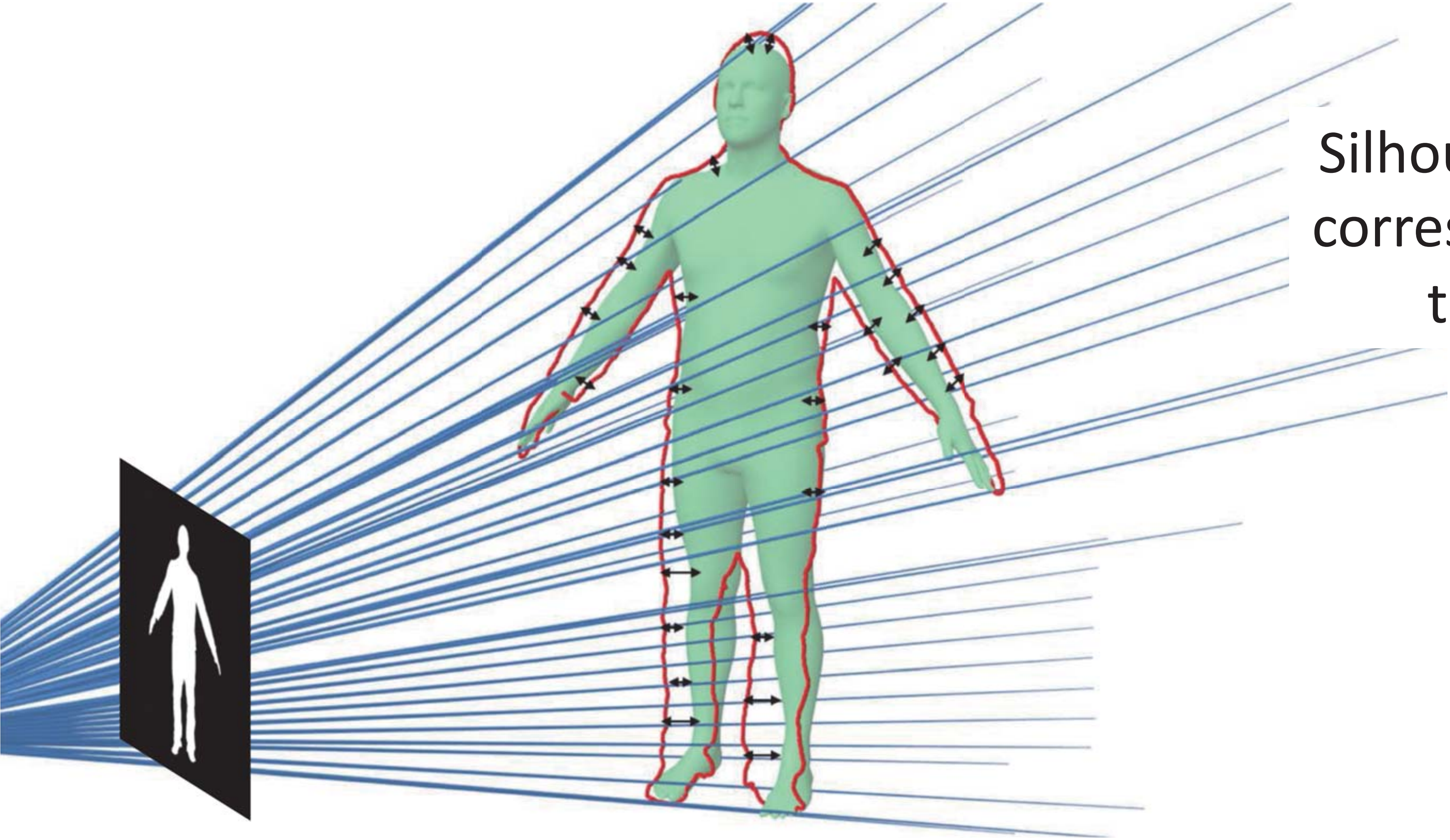


How Can We Generalize It to Dynamic Human Motion ?

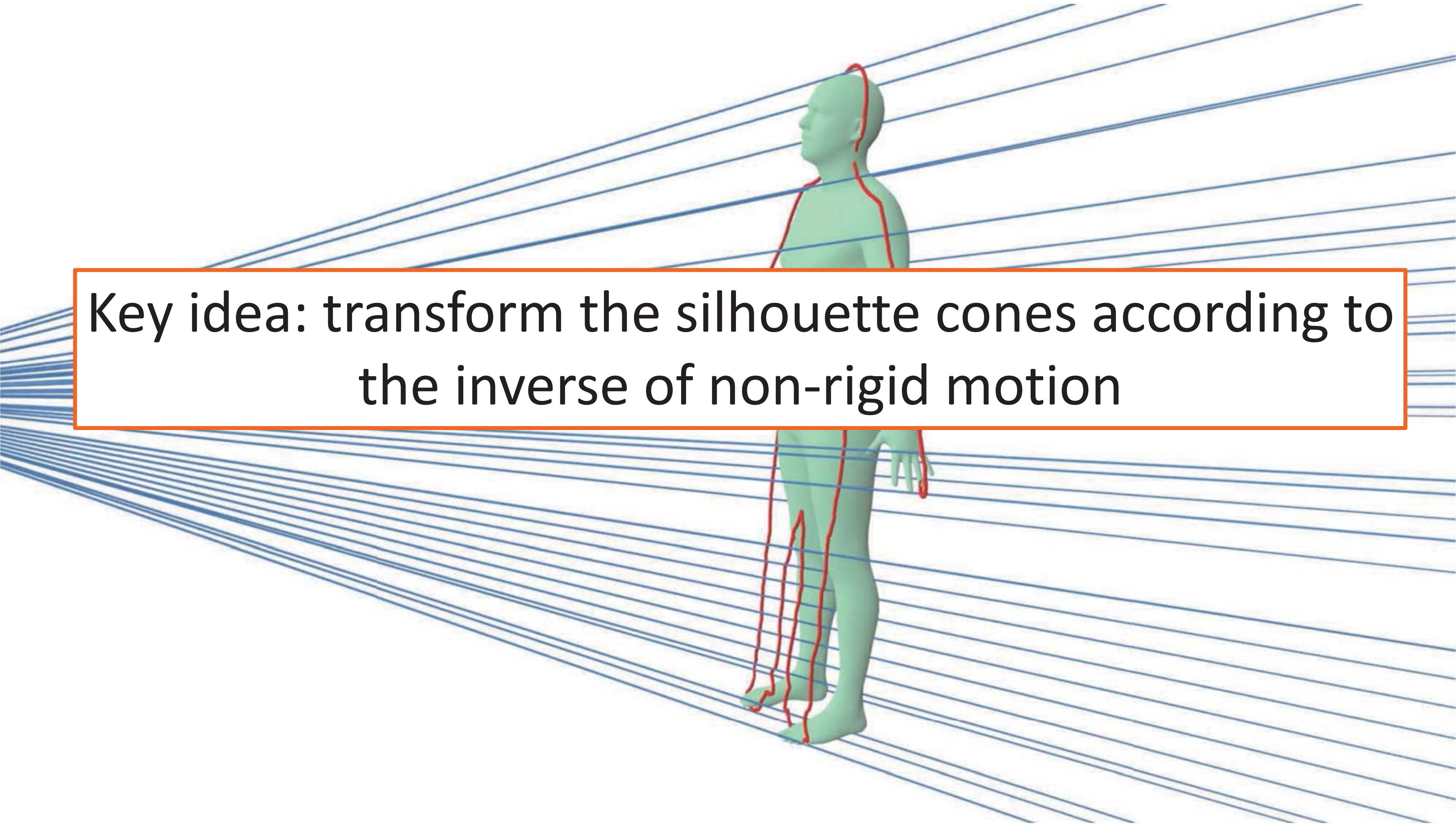
Person is moving!

How Can We Generalize It to Dynamic Human Motion ?

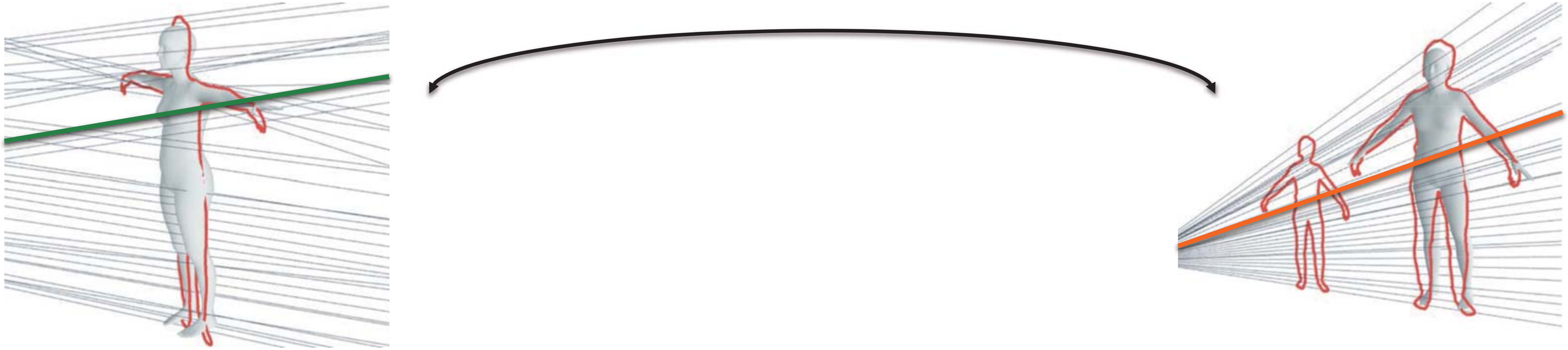
Estimate the 3D human pose and shape per frame



Silhouette rays with correspondences on the surface



Key idea: transform the silhouette cones according to the inverse of non-rigid motion



$$\underbrace{\boxed{\mathbf{r}}}_{\text{Ray in Canonical Frame}} = \underbrace{\left(\sum_{k=1}^{K} w_{k,i} G_k(\boldsymbol{\theta}, \boldsymbol{J}_{\boldsymbol{\beta}})\right)^{-1}}_{\text{Inverse of Articulated Motion}} \underbrace{\boxed{\mathbf{r}'}}_{\text{Ray}} - b_{P,i}(\boldsymbol{\theta}).$$

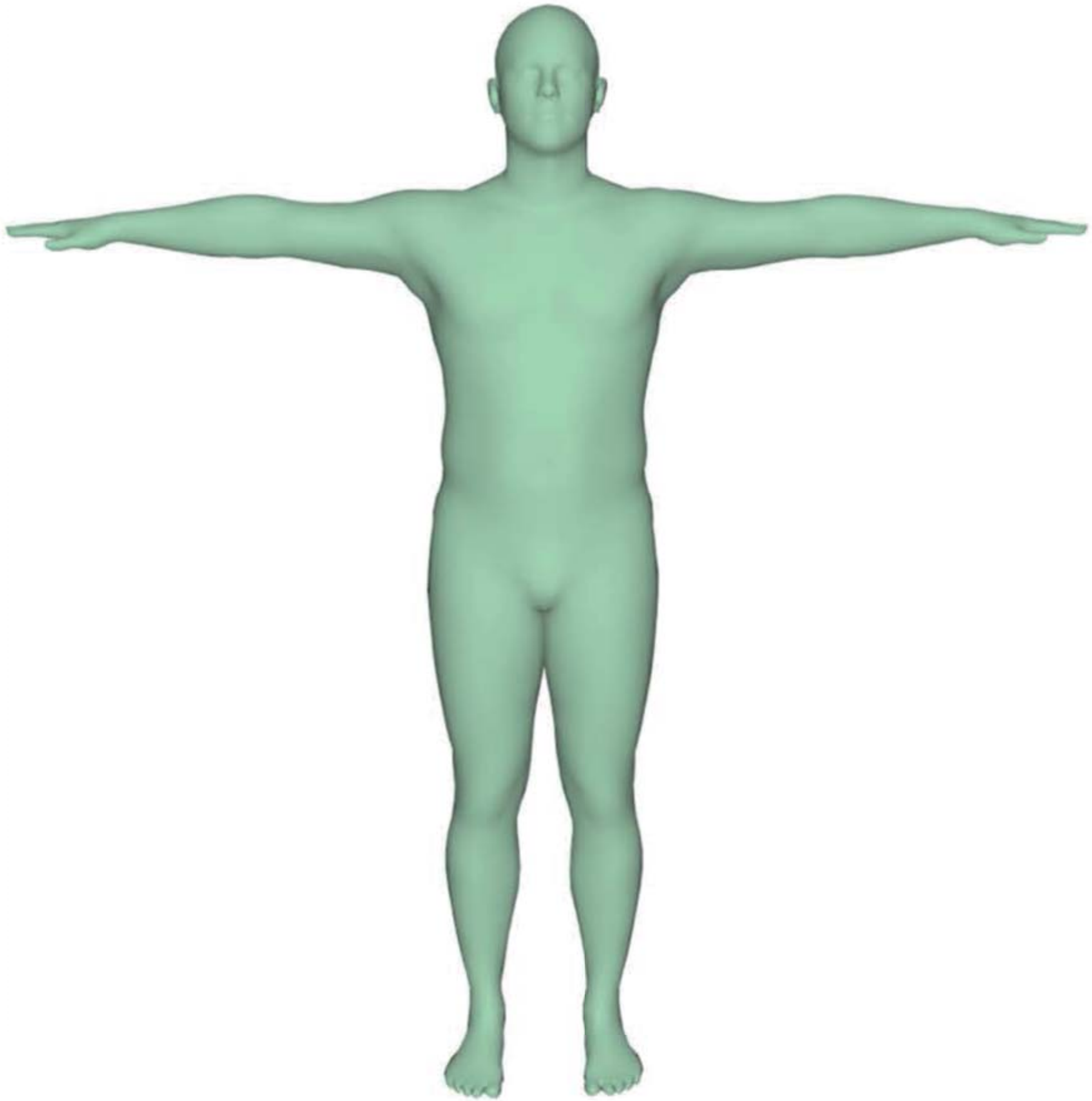
Optimize a Single Shape to Fit all *Unposed* Silhouette Cones

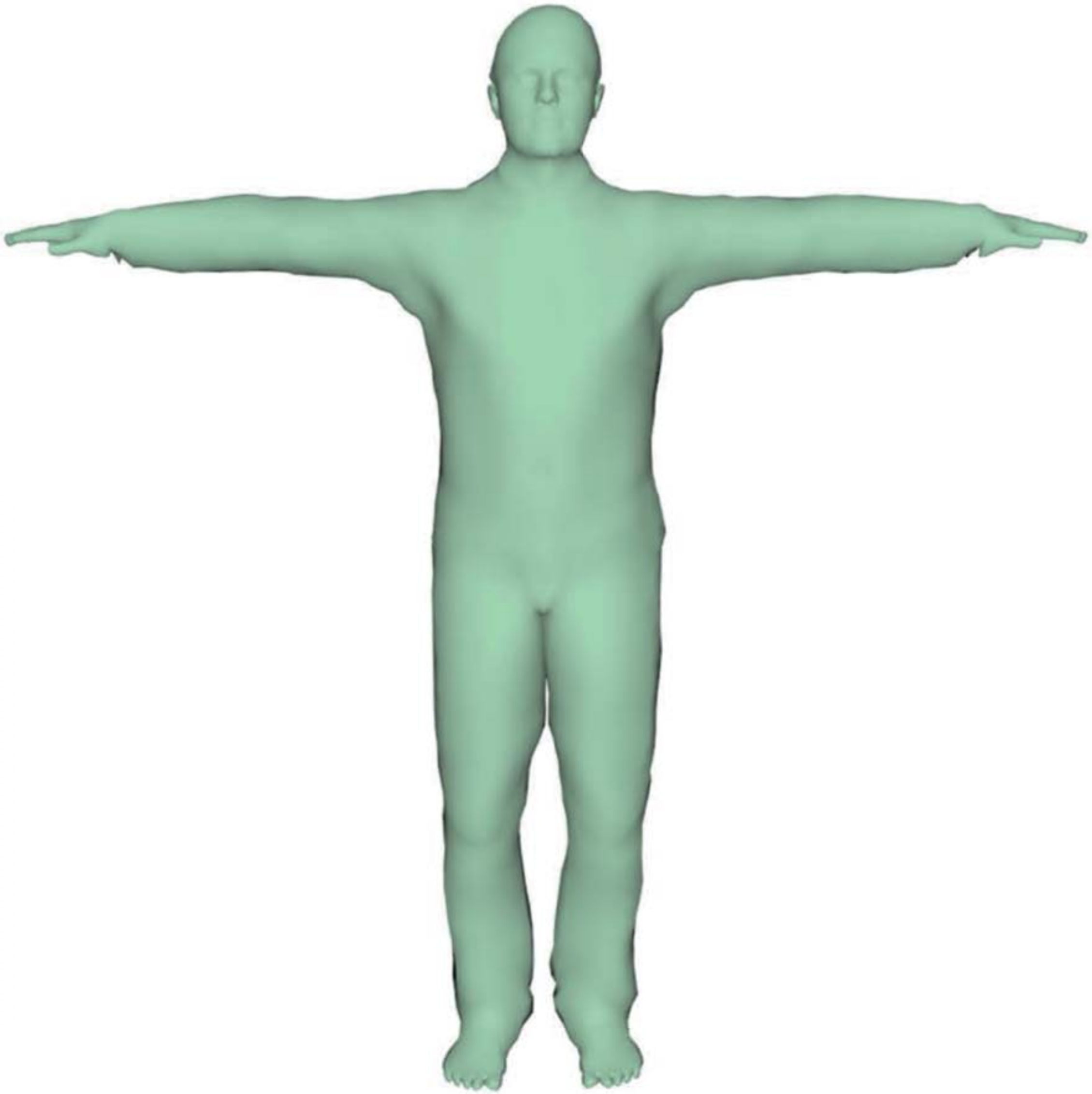
$$\boxed{\arg\min_{\boldsymbol{\beta},\mathbf{d}} E_{\text{cons}}(\boldsymbol{\beta},\mathbf{d})}$$

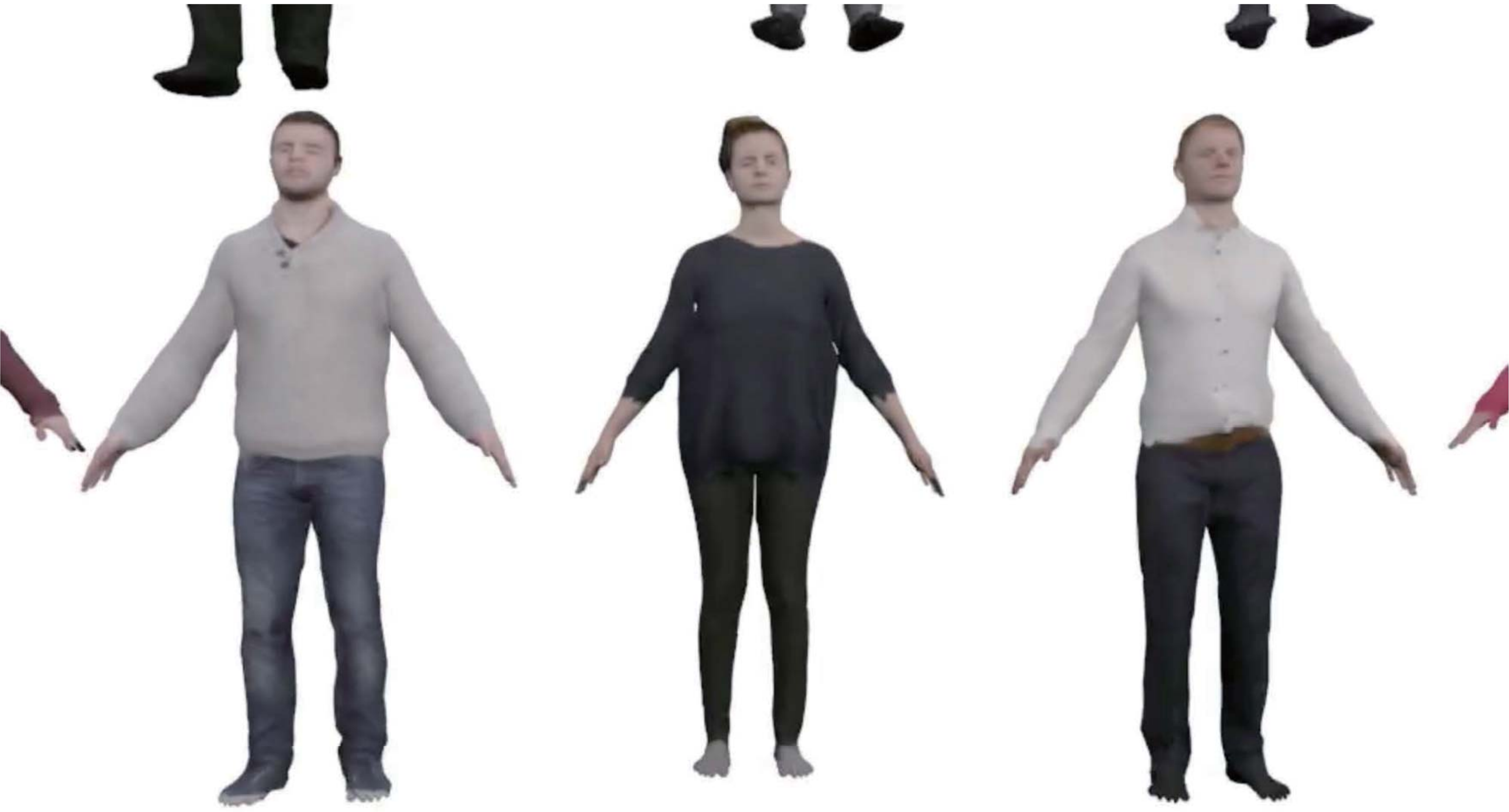
$$E_{\text{data}} = \sum_{(\mathbf{v},\mathbf{r})\in\mathcal{M}} \rho(\mathbf{v}\times\mathbf{r}_n - \mathbf{r}_m)$$

Sum of **point to line** distances

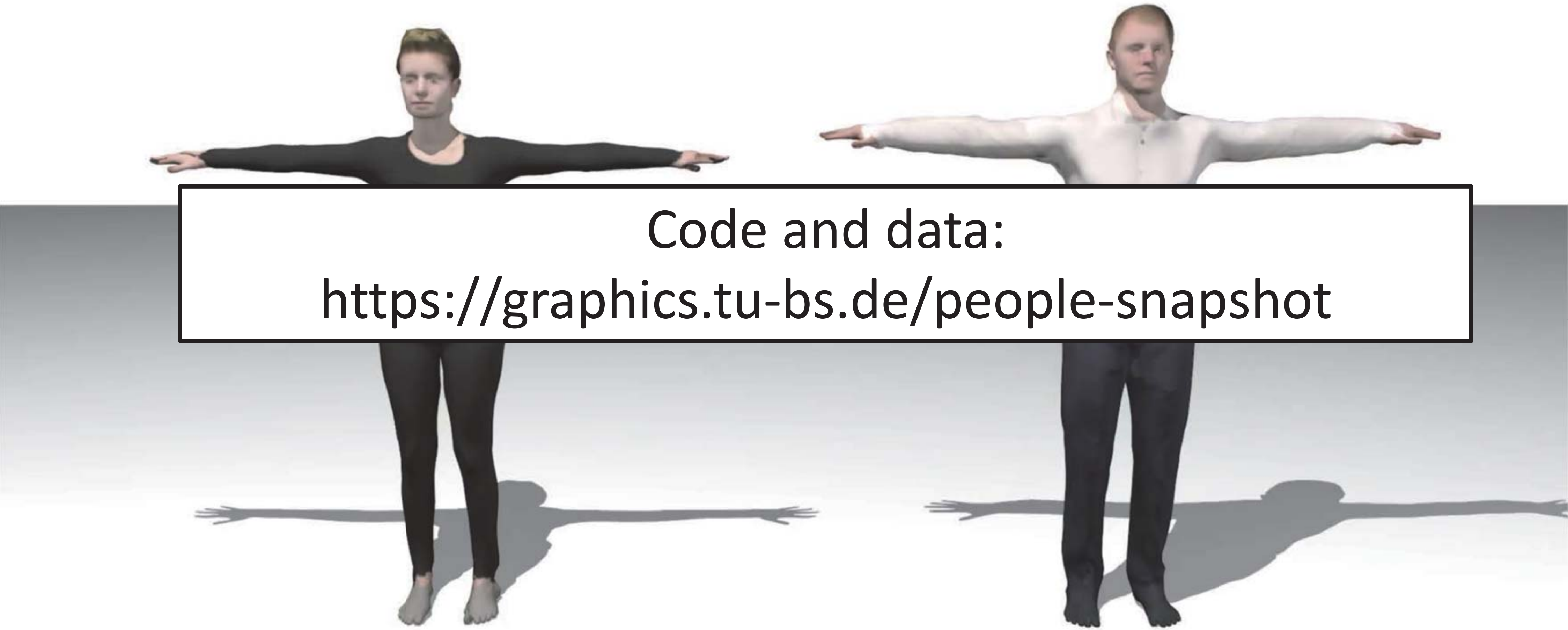
Prior Terms:
- Symmetry
- Prior on Shape
- Surface Smoothness





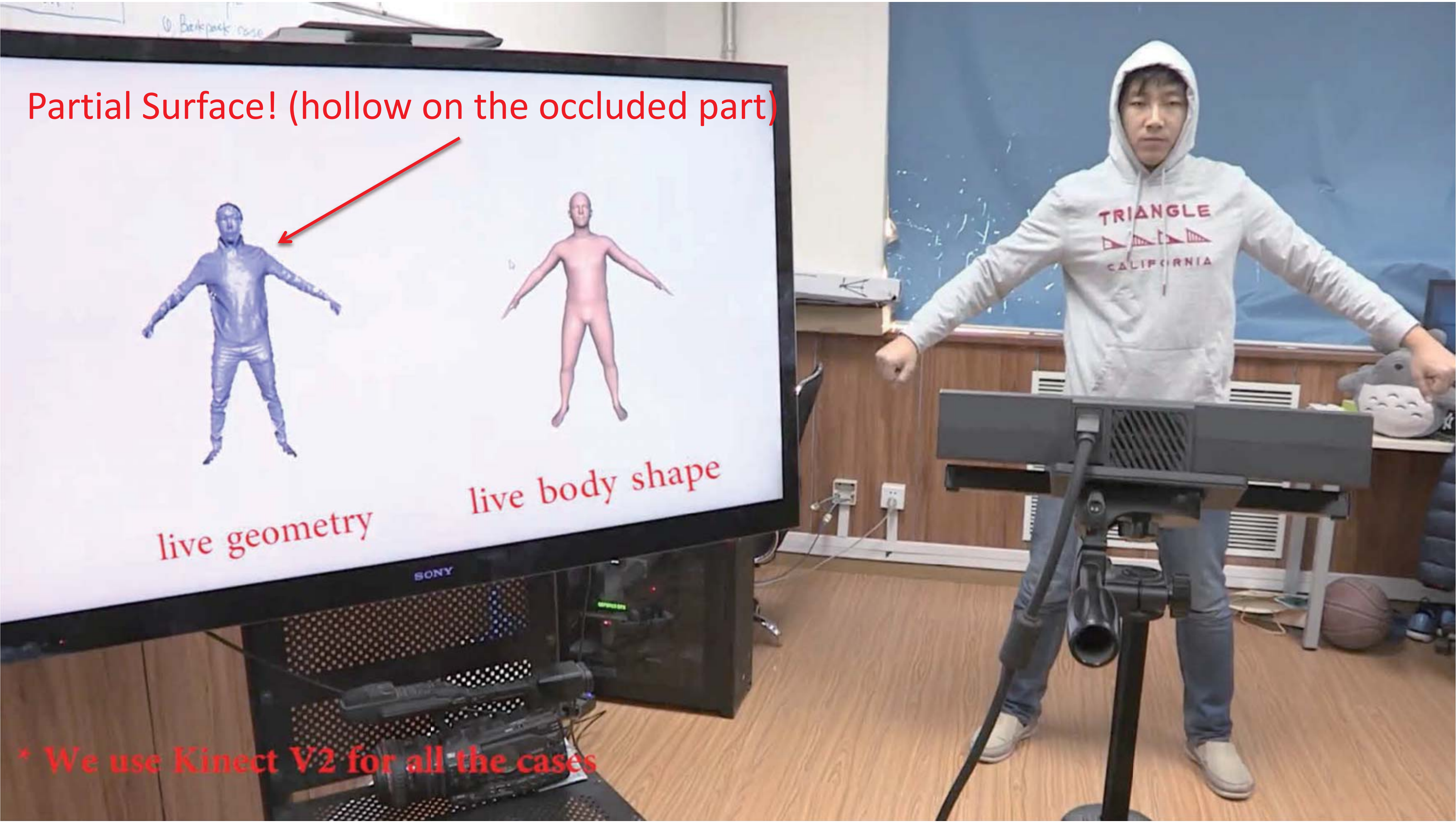



AUTO



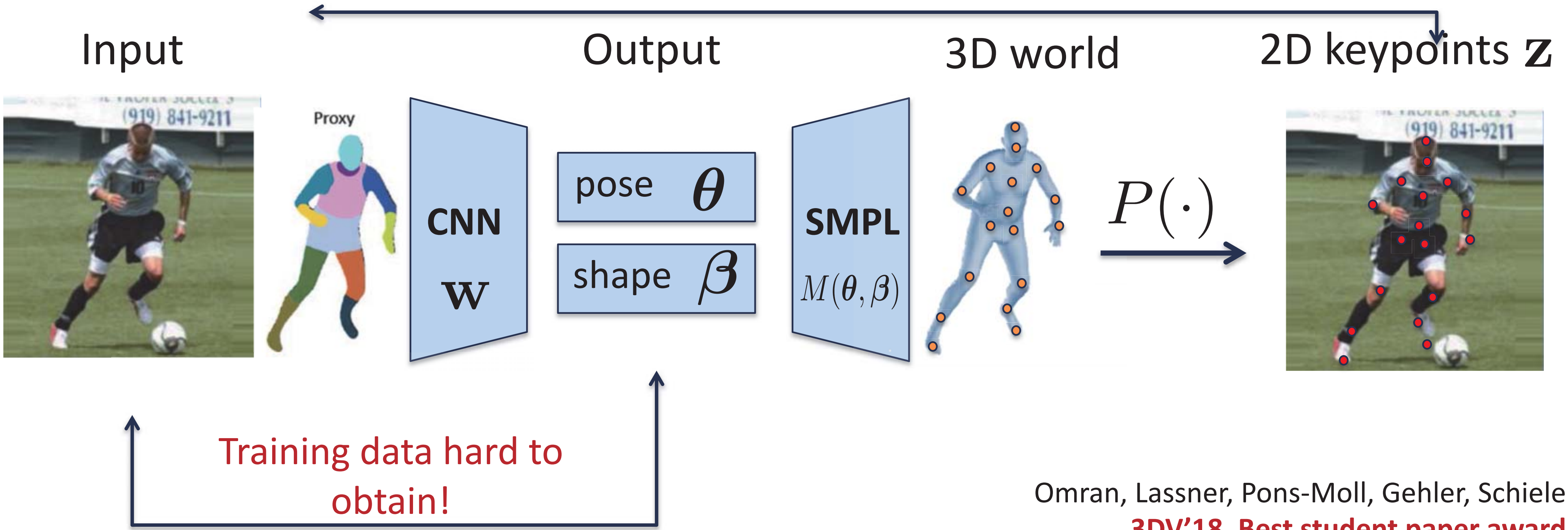
Code and data:
https://graphics.tu-bs.de/people-snapshot


Alldieck et al. 3DV ‘18

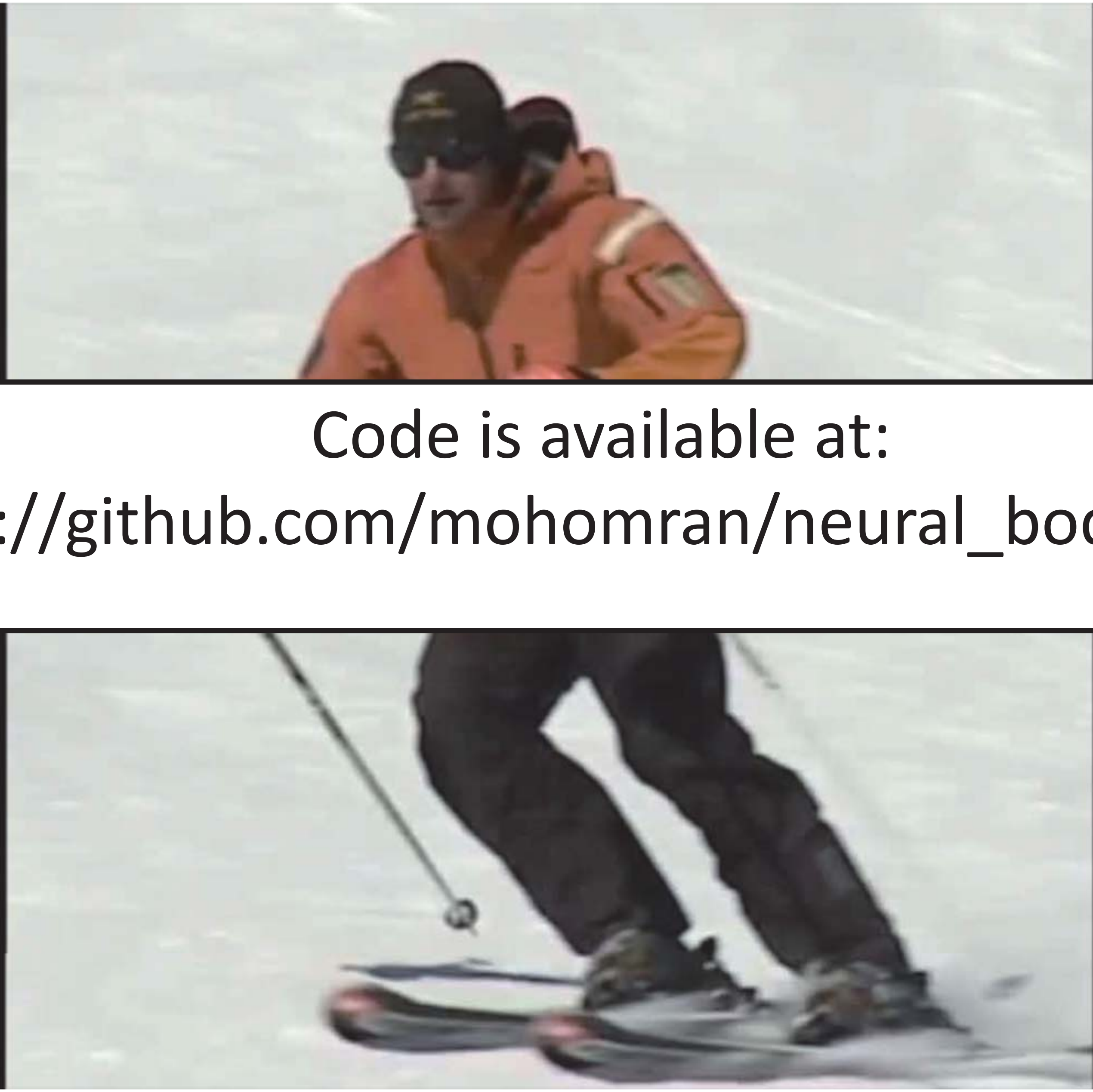

Partial Surface! (hollow on the occluded part)
live geometry
live body shape
SONY
TRIANGLE
CALIFORNIA
* We use Kinect V2 for all the cases

Neural Body Fitting
Body Pose and Shape from 1 Image

$$\arg\min_{\mathbf{w}} \|\hat{\mathbf{z}}(I, \mathbf{w}) - \mathbf{z}\|$$



Omran, Lassner, Pons-Moll, Gehler, Schiele
3DV'18, Best student paper award



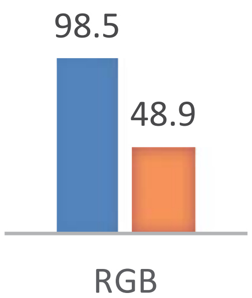
Code is available at:
https://github.com/mohomran/neural_body_fitting

Input Representation

Input (2D) Proxy representation Output (3D)

Would an intermediate representation help?
If yes, which?

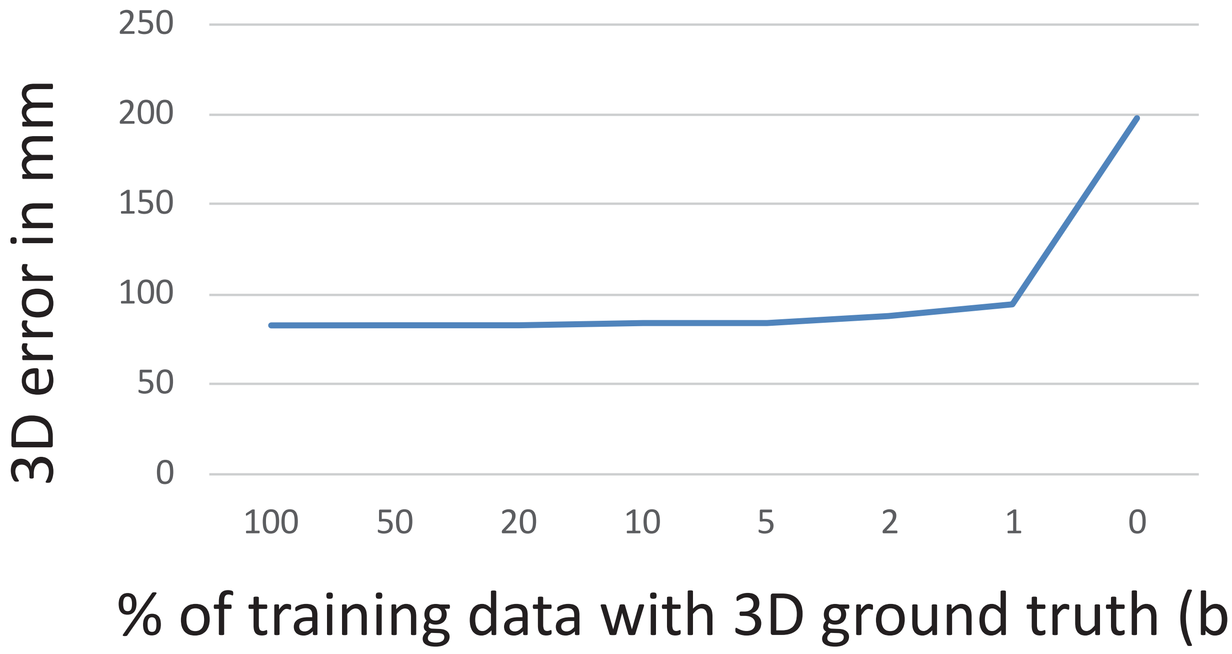
Input representation



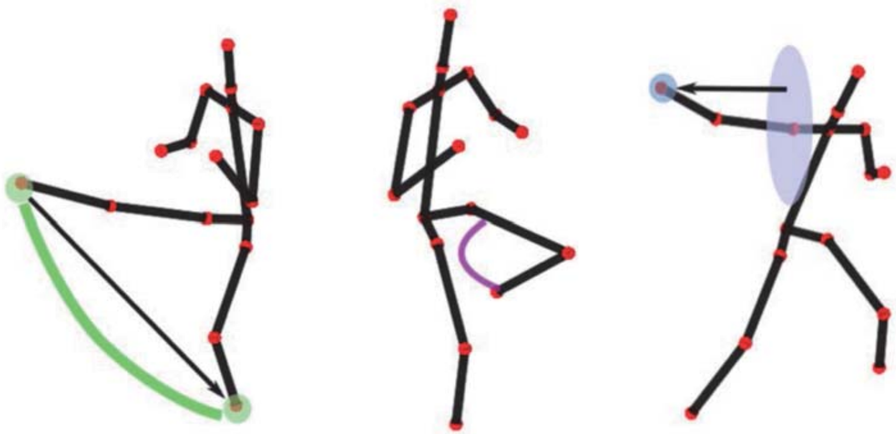
3D ERROR (IN MM) ■ UniteThePeople

How much 3D data is needed?

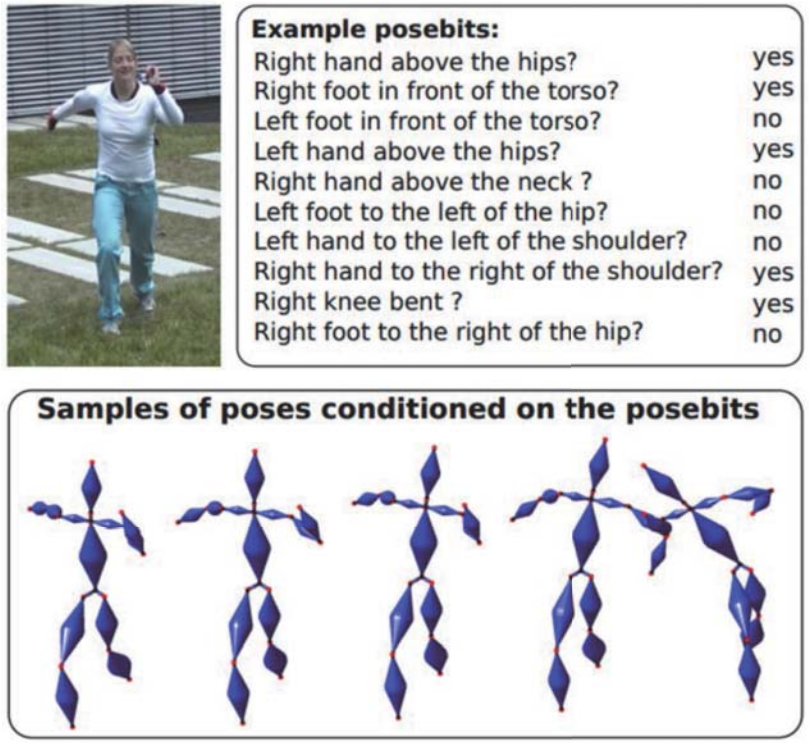
Experiment: given training data with 2D ground truth (keypoints)
vary size of subset that also has 3D ground truth (shape/pose)



Are 2D annotations enough?

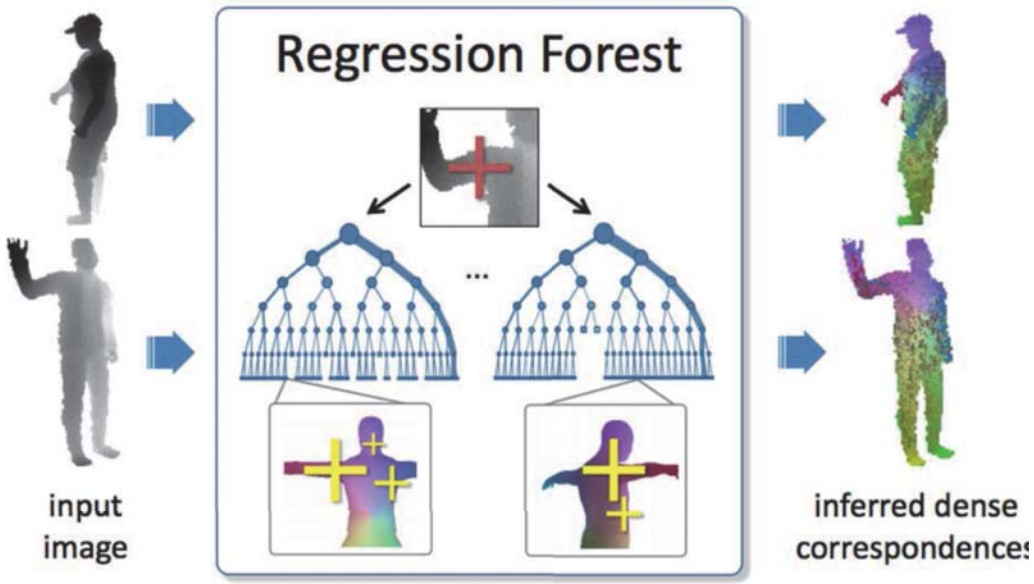


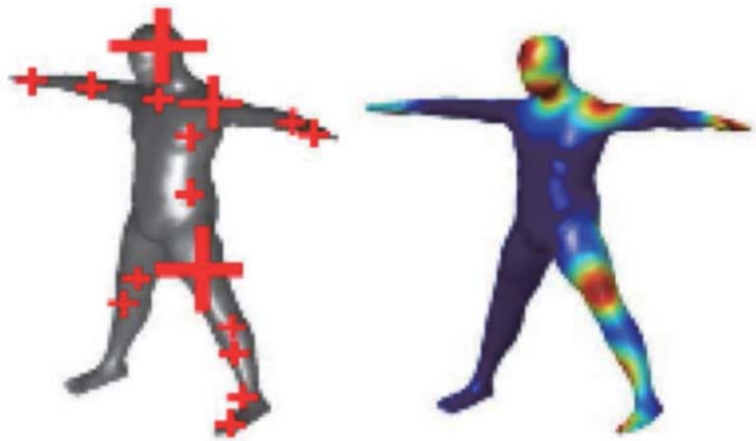
Posebits: pose descriptions



Pons-Moll, Fleet and Rosenhahn. CVPR'14

Dense Correspondences

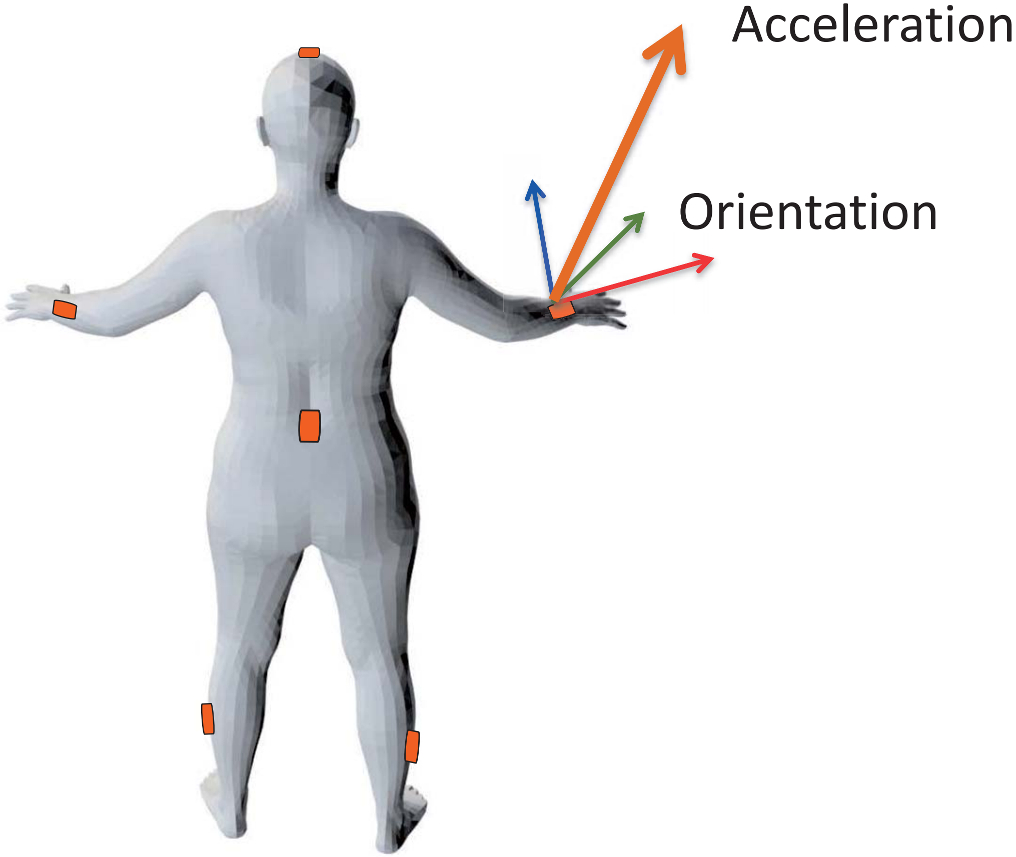




Taylor et al. CVPR’12
Pons-Moll et al.
- BMVC ’13 Best Science Paper Award
- IJCV15 – journal version

Motion Capture from Sparse IMUs

IMU = Inertial Measurement Unit
(Xsens)



Sparse Inertial Poser

Automatic 3D Human Pose Estimation from Sparse IMUs

Supplementary material

Eurographics'17
Paper ID 1112

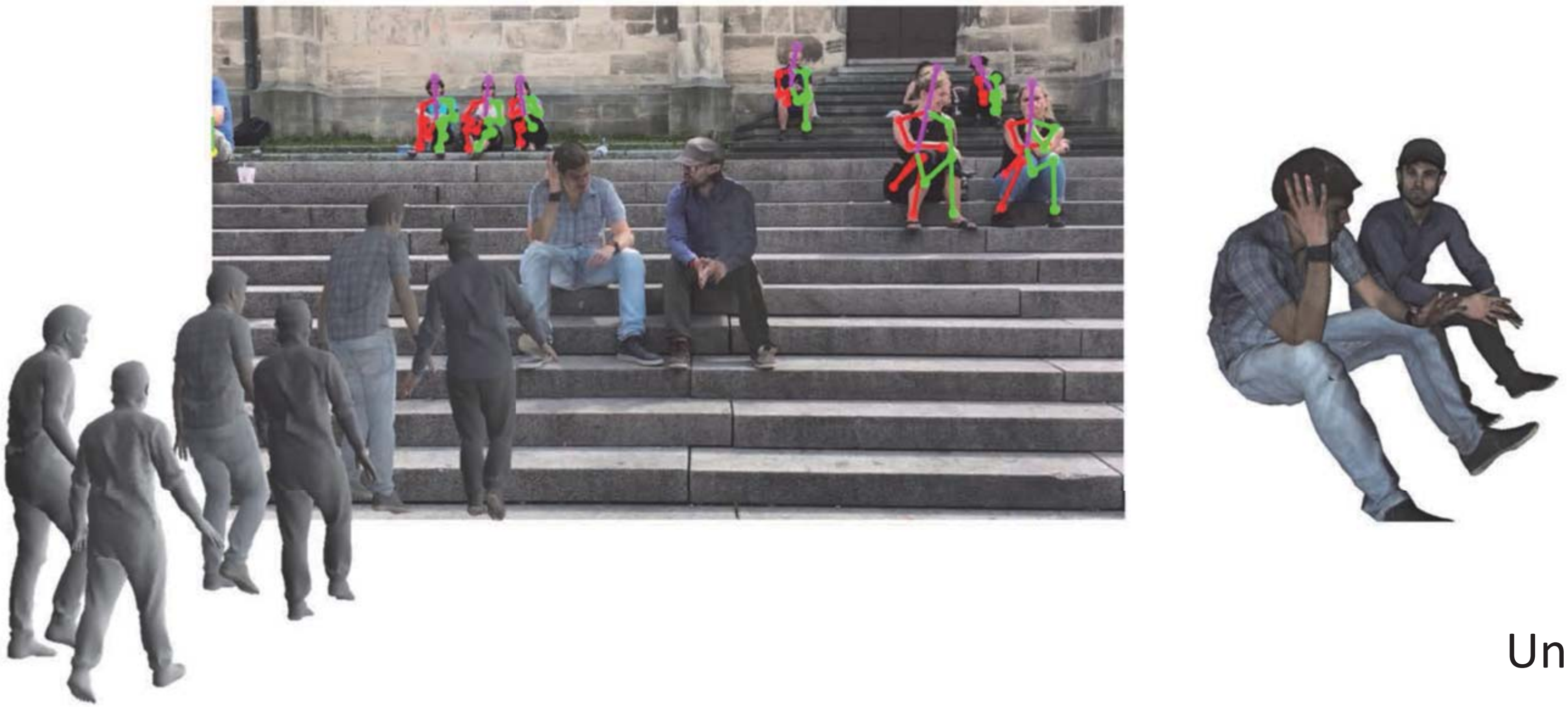
T. Marcard, B. Rosenhahn, M. Black, G. Pons-Moll. **Eurographics '17** **Best Paper Award**

Climbing

Single Phone Camera and IMUs?

Recovering Accurate 3D Human Pose from IMUs and a Moving Camera

von Marcard, T., Henschel R., Rosehnahn, Black, M., Pons-Moll, G.



Under review

Problem: limited datasets

MPII human pose

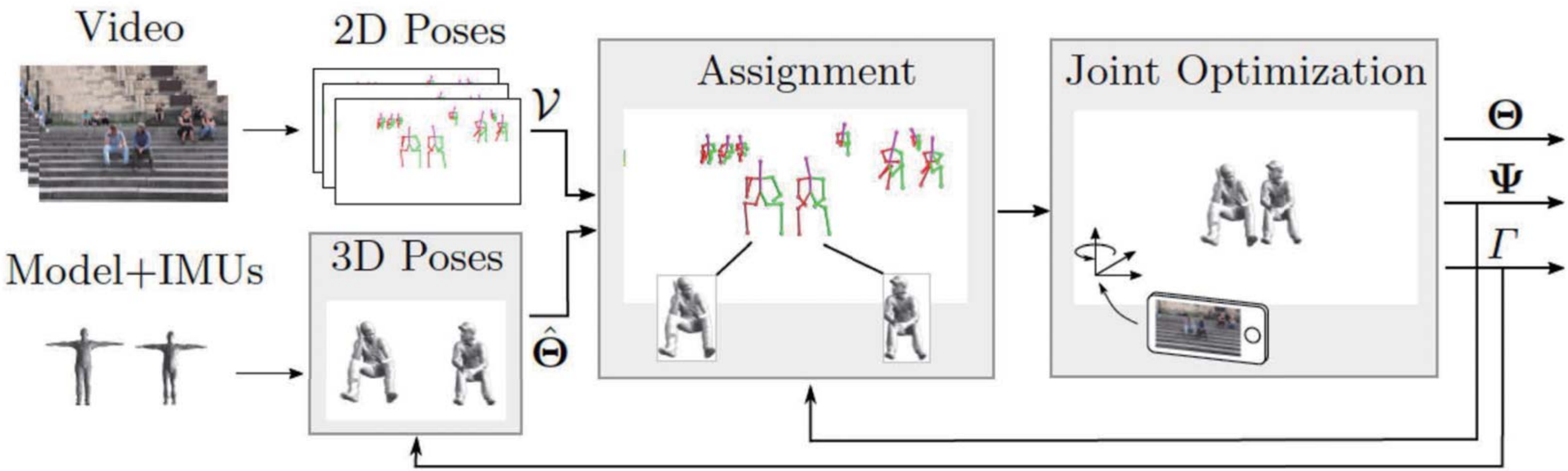
+ Variation
- Weak annotations

HumanEva

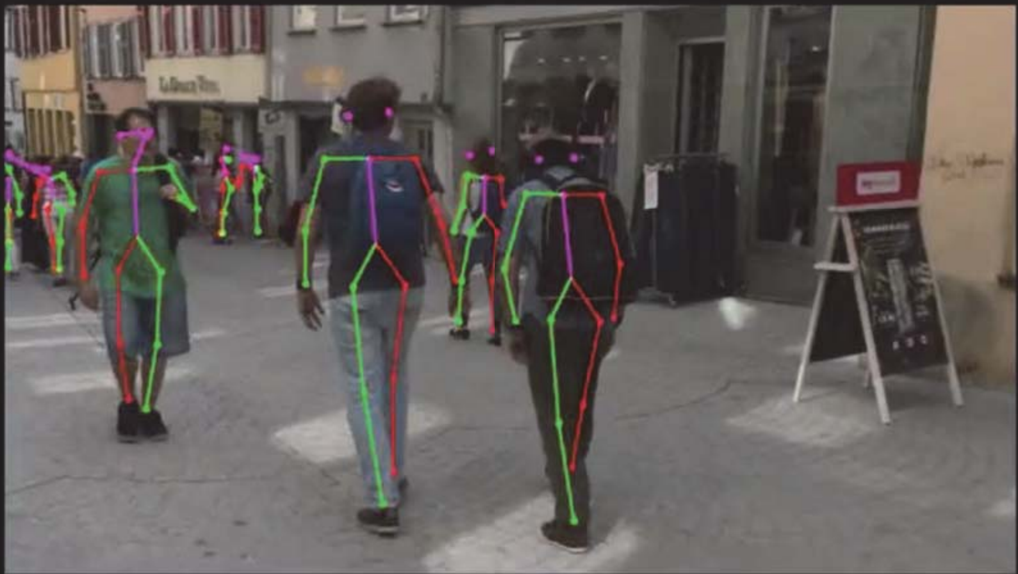
Human3.6M

+ 3D annotations
- Variation (very controlled indoor setups)

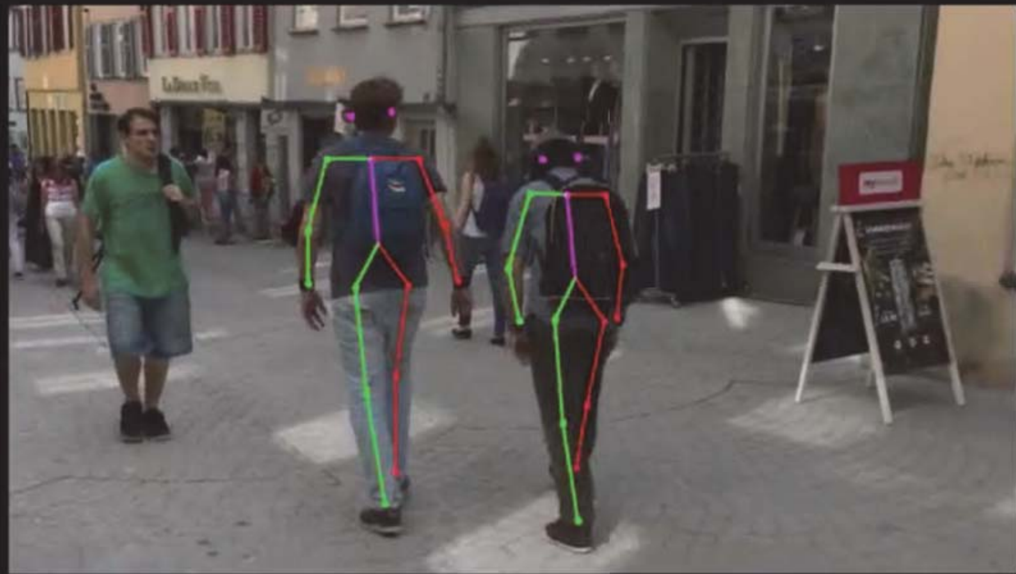
A single moving camera and IMUs on the person



Person Identification

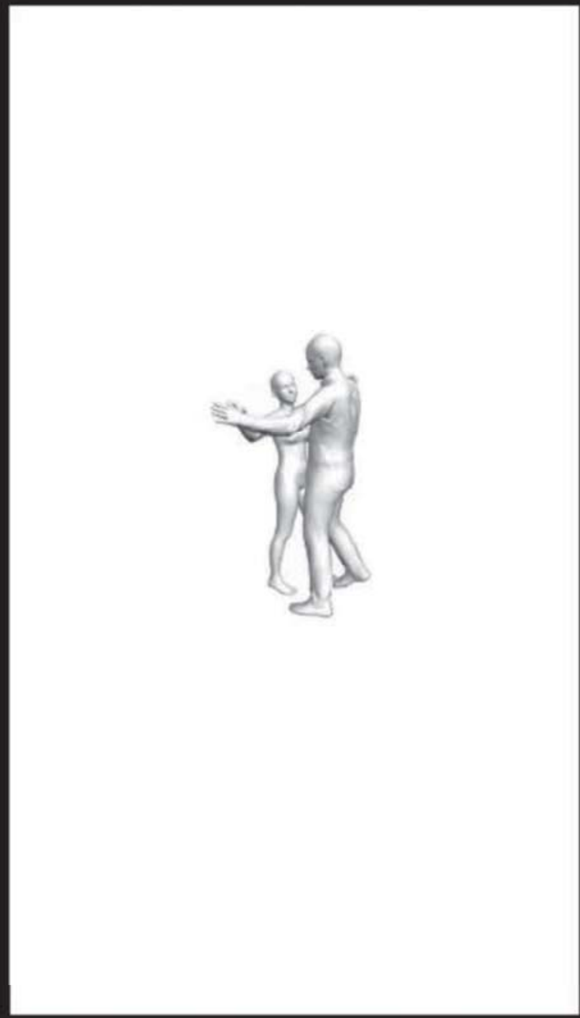


All 2D Poses



Assigned 2D Poses

3D Pose Estimation

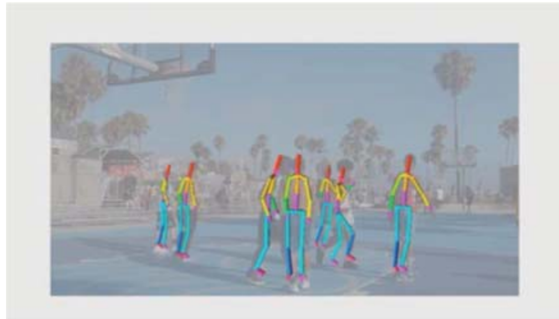


Full dataset available:
http://virtualhumans.mpi-inf.mpg.de/3DPW/

Not today…

Generating People with GANs

C.Lassner, G. Pons-Moll, P. Gehler ICCV’17



Multiple People (3DV’18)

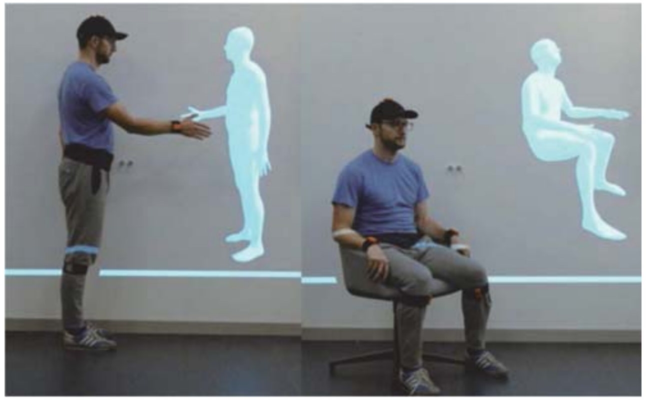
D. Mehta, O. Sotnychenko, F. Mueller, Weipeng Xu, S.Sridha, G.Pons-Moll, C. Theobalt

Shape and Motion from Markers

N. Mahmood, G. Pons-Moll, Ghorbani, N. Troje, M. Black

Real-Time Monocular Performance Capture

M. Habermann, W. Xu, M. Zollhoefer, G.Pons-Moll, C. Theobalt



Deep Inertial Poser Learning to Reconstruct Human Pose from SparseInertial Measurements in Real Time

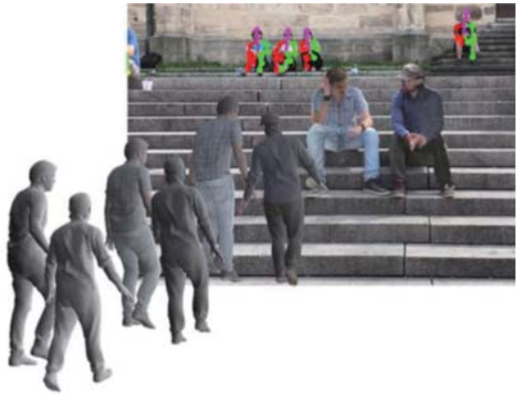
Y. Huang, M. Kaufmann, E. Aksan, M. J. Black, O. Hilliges, G. Pons-Moll Sigg. Asia ‘16

Fashion is taking shape

H. Sattar, G. Pons-Moll and M. Fritz (WACV’18)

CONCLUSIONS

- 3D virtual humans are powerful for a number of applications
- To achieve **realism** we need to **learn** digital humans by capturing **real** ones
- **Clothing** is one of the main **missing** components in current statistical body models → **capture** from **consumer cameras**!
- We need **perception** algorithms that **reason** about the **3D world**, not about pixels



Real Virtual Humans

http://virtualhumans.mpi-inf.mpg.de/

- Resources data and code available for research!
- **Open positions** in the areas of computer vision, machine learning and computer graphics with focus on analyzing and modelling people

Resources, data, and code

- Shape/cloth 3D avatar from RGB-video: https://graphics.tu-bs.de/people-snapshot
- https://graphics.tu-bs.de/upload/publications/alldieck2018videopeople.pdf
- Single image human pose and shape (code): https://github.com/mohomran/neural_body_fitting
- 3DPW (3D Poses in the wild): https://virtualhumans.mpi-inf.mpg.de/3DPW/
- SMPL: http://smpl.is.tue.mpg.de
- DYNA: http://dyna.is.tue.mpg.de
- CLOTHCAP (Tracking people in clothing with layers/parts): http://clothcap.is.tue.mpg.de
- Shape under clothing (>11.000 cloth-people scans): http://buff.is.tue.mpg.de
- DFAUST (40.000 scans and registrations): http://dfaust.is.tue.mpg.de
- SIP – 3D pose from 6 IMUs: https://ps.is.tuebingen.mpg.de/uploads_file/attachment/attachment/345/sparseInertialPoser.pdf
- Data driven physics: https://ps.is.tuebingen.mpg.de/publications/meekyoung-siggraph
- DoubleFusion: Online 3D pose, shape, detailed geometry from depth
- http://www.liuyebin.com/doublefusion/doublefusion.htm
- Generative Model of People (Variational Autoencoder): http://files.is.tuebingen.mpg.de/classner/gp/
- Multiple People 3d Pose (3DV'18): https://arxiv.org/abs/1712.0345
- Detailed Human Avatars from Monocular Video:
- https://virtualhumans.mpi-inf.mpg.de/papers/alldieck2018detailed/alldieck2018detailed.pdf