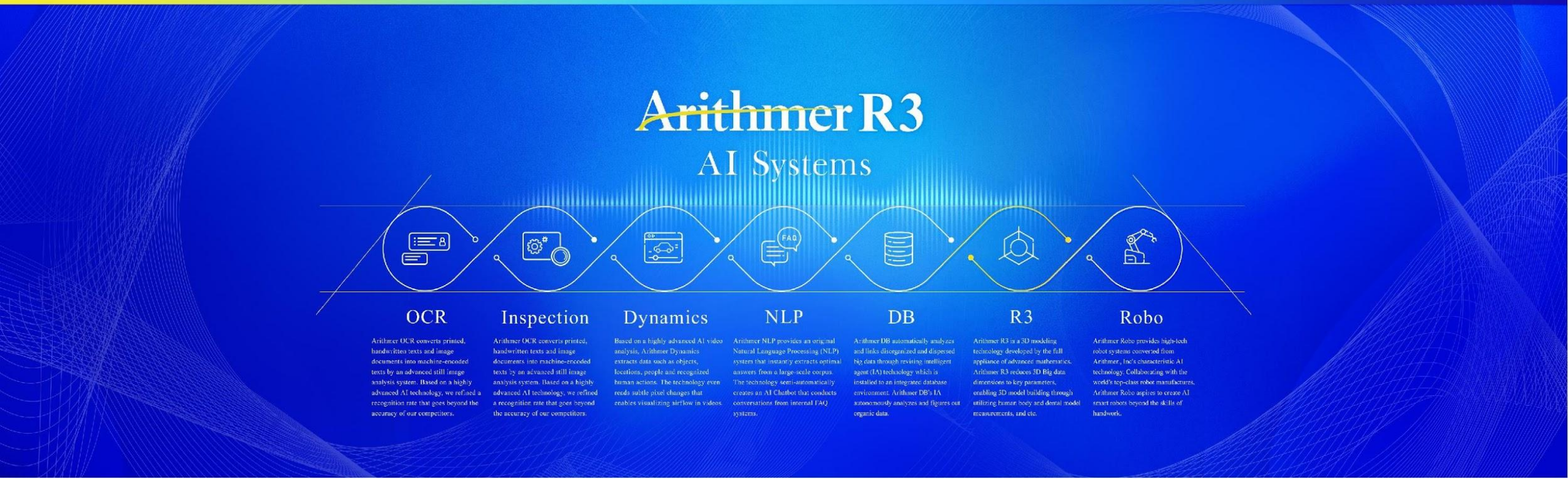


3D human body modeling from RGB images

Arithmer R3 Div. - Enrico Rinaldi

July 9 and 16, 2020

Human Pose Estimation WG

Enrico Rinaldi (Ph.D.)

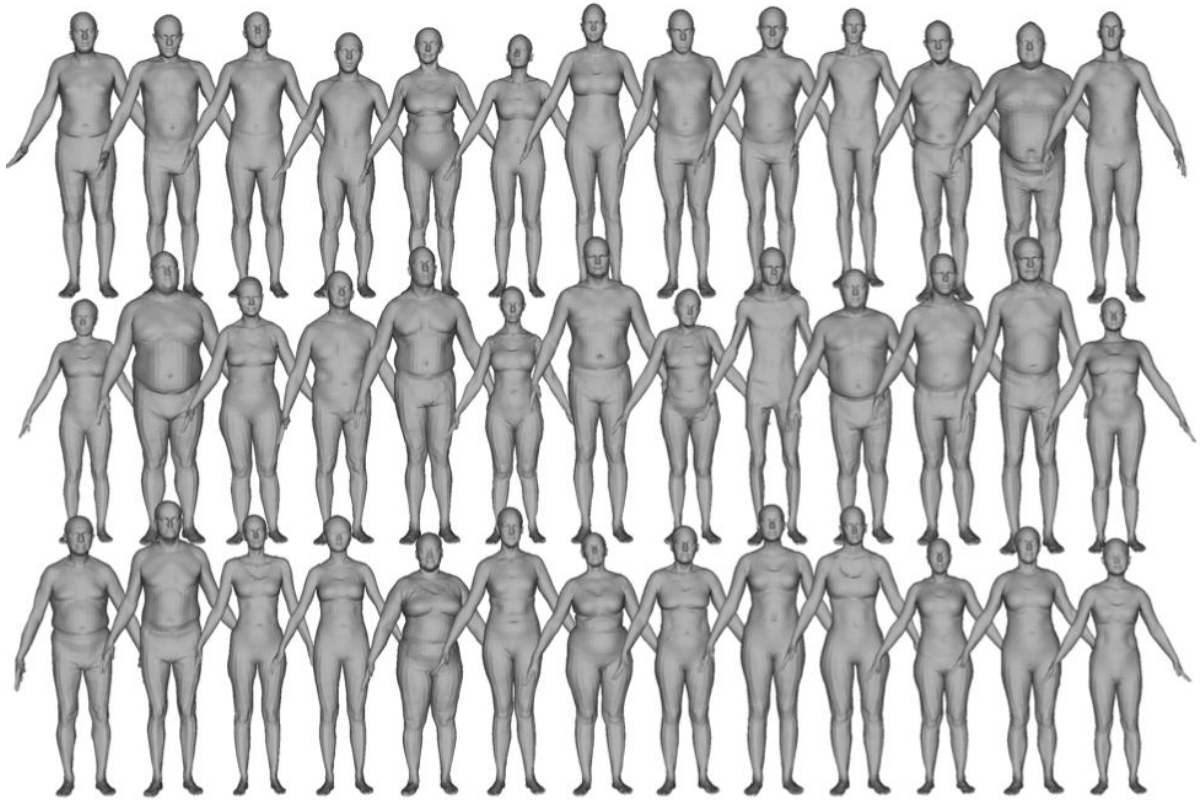
- **Education**
  - Bachelor of Science in Physics from the University of Milan
  - Master of Science in Theoretical Physics from the University of Milan
  - Ph.D. in Theoretical Particle Physics from the University of Edinburgh
    - Computational Physics and (Big) Data Analysis
    - Particle Physics with Composite Higgs, Dark Matter, and Extra Dimensions
    - Supported by Scottish Universities Physics Alliance (SUPA) fellowship and Japanese Society for the Promotion of Science (JSPS) fellowship at the Kobayashi-Maskawa Institute of Nagoya University
- **Associate Researcher** at the Lawrence Livermore National Laboratory
  - High-Performance Computing simulations of particle physics, nuclear physics and string theory
  - Markov Chain Monte Carlo sampling and Optimization problems
- **Special Postdoctoral Fellow** at the RIKEN BNL Research Center
  - Junior PI of project on numerical simulations for composite dark matter theories
  - GPU computing for HPC simulations of nuclear physics properties
  - Leveraging machine learning techniques to accelerate physics discoveries
- **R&D Researcher and Engineer** at Arithmer Inc.
  - Research on Geometric Learning, Graph Learning, 3D perception, Computer Vision
  - Applications to robotics, automated measurements systems, optimization problems

Outline

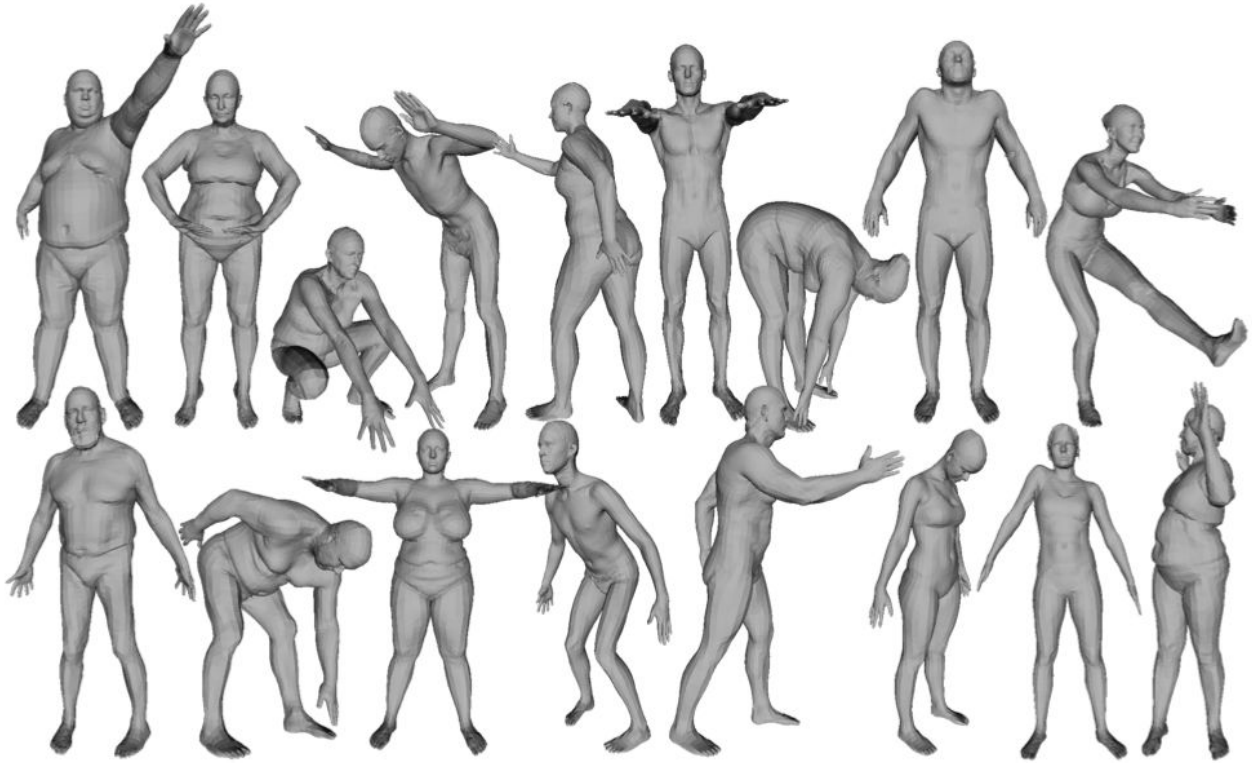
- Introduction
  - Definition of the problem
  - Challenges
  - Common approaches
- Statistical body models
  - Definition of the setup
  - Aim of the methods
  - Typical solutions
  - **SMPL**
- 3D human pose
  - Challenges
  - 2D solutions
  - **SMPLify**
- Conclusions

The data

SHAPE



POSE



The Model

$$M(\vec{\beta},\vec{\theta};\Phi) : \mathbb{R}^{|\vec{\theta}|\times|\vec{\beta}|} \mapsto \mathbb{R}^{3N} \longrightarrow W\left(T_P(\vec{\beta},\vec{\theta};\bar{\mathbf{T}},\mathcal{S},\mathcal{P}), J(\vec{\beta};\mathcal{J},\bar{\mathbf{T}},\mathcal{S}),\vec{\theta},\mathcal{W}\right)$$

Shape linear coefficients

Vector β

multiplies the principal components of the learned shape parameters that modify the N vertices of the mesh

Pose vector

Vector θ

represents the 3D angles between the kinematic tree of the K=23 joints in the skeleton

$$\Phi = \{\bar{\mathbf{T}},\mathcal{W},\mathcal{S},\mathcal{J},\mathcal{P}\}$$

Parameters of the model related to pose are trained first:

$$\{\mathcal{J},\mathcal{W},\mathcal{P}\}$$

Parameters of the model related to shape are trained after:

$$\{\bar{\mathbf{T}},\mathcal{S}\}$$

Joint positions are defined as a function of the body shape:

$$J(\vec{\beta};\mathcal{J},\bar{\mathbf{T}},\mathcal{S}) = \mathcal{J}(\bar{\mathbf{T}} + B_S(\vec{\beta};\mathcal{S}))$$

SMPL

SMPL stands for Skinned Multi-Person Linear model.

It is a “statistical” body model, which means it is learned from data.

It captures 2 aspects of the human body:

1. Shape
2. Pose

SHAPE

The shape of a male and female body is learned from 3D data in the CAESAR dataset

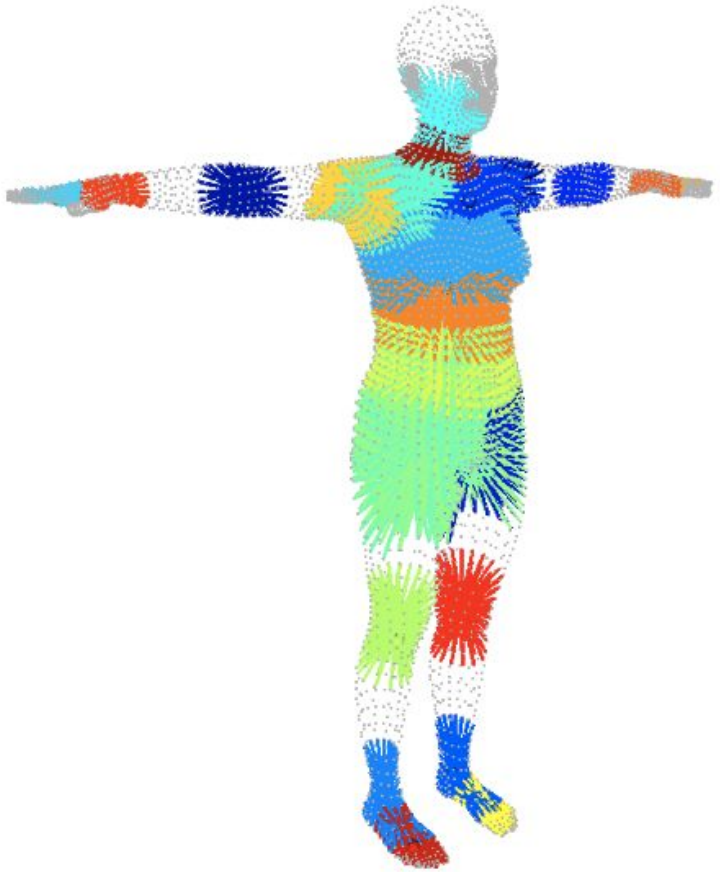
Shape is represented by a 3D mesh

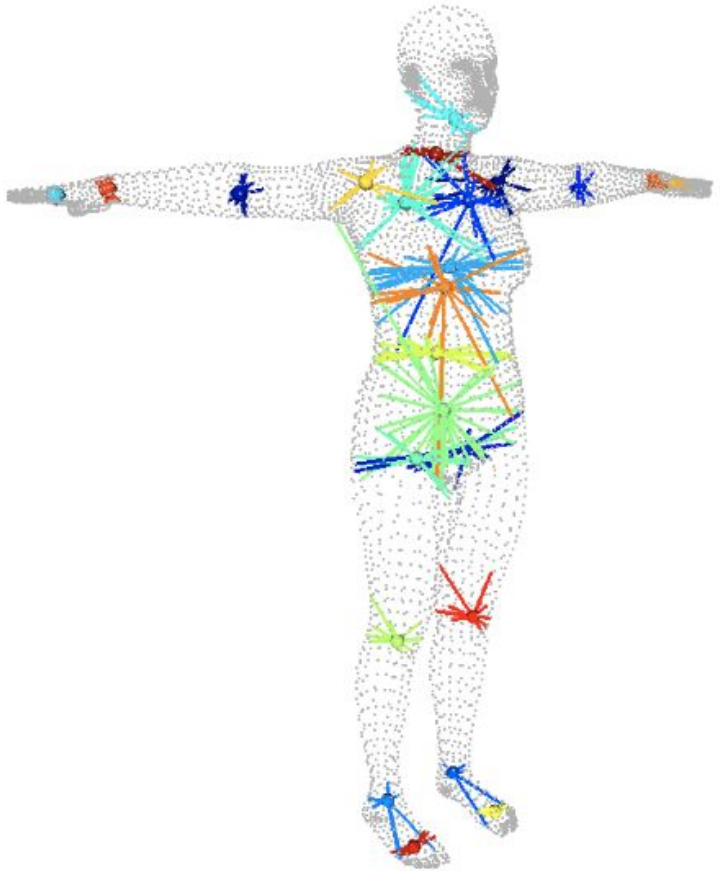
POSE

The pose of a male and female body is learned from 3D data in the FAUST dataset

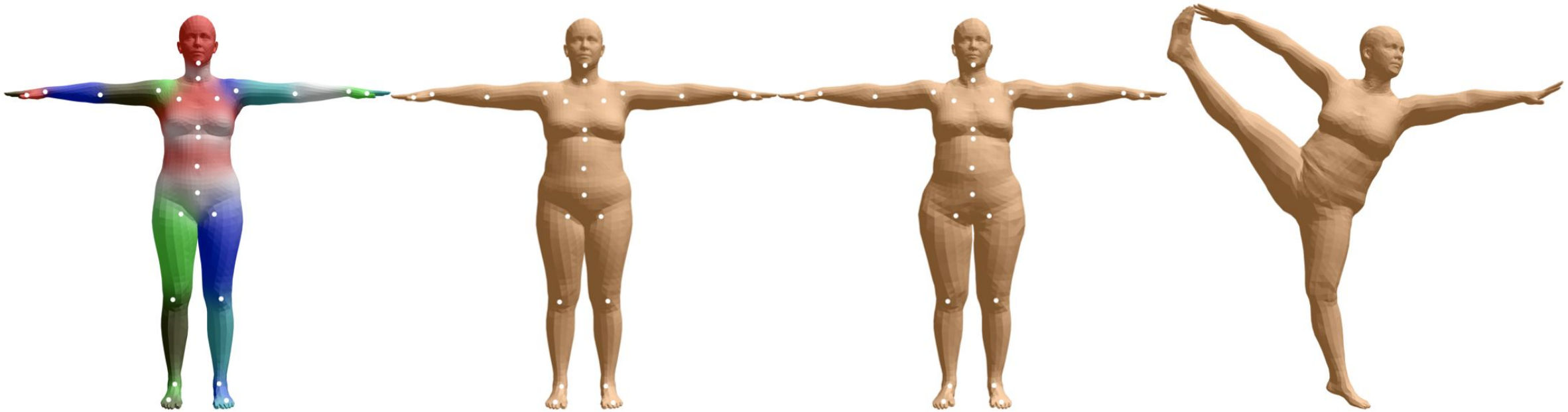
Pose is represented by a skeleton.

Example: Joint regression





Example: Inference



(a) $\bar{\mathbf{T}}, \mathcal{W}$ (b) $\bar{\mathbf{T}} + B_S(\vec{\beta}), J(\vec{\beta})$ (c) $T_P(\vec{\beta}, \vec{\theta}) = \bar{\mathbf{T}} + B_S(\vec{\beta}) + B_P(\vec{\theta})$ (d) $W(T_P(\vec{\beta}, \vec{\theta}), J(\vec{\beta}), \vec{\theta}, \mathcal{W})$

Start from a mesh template T and blend weights W The weights represent how much vertices are affected by joint movement	Modify the template T using the shape vector and apply the joint regressor based on the shape vector. The mesh and joints change with shape.	Add the pose vector and the pose blend shapes that will modify the mesh vertices based on the location of the joints	Apply Linear Blend Skinning with the new template, joints and weights.

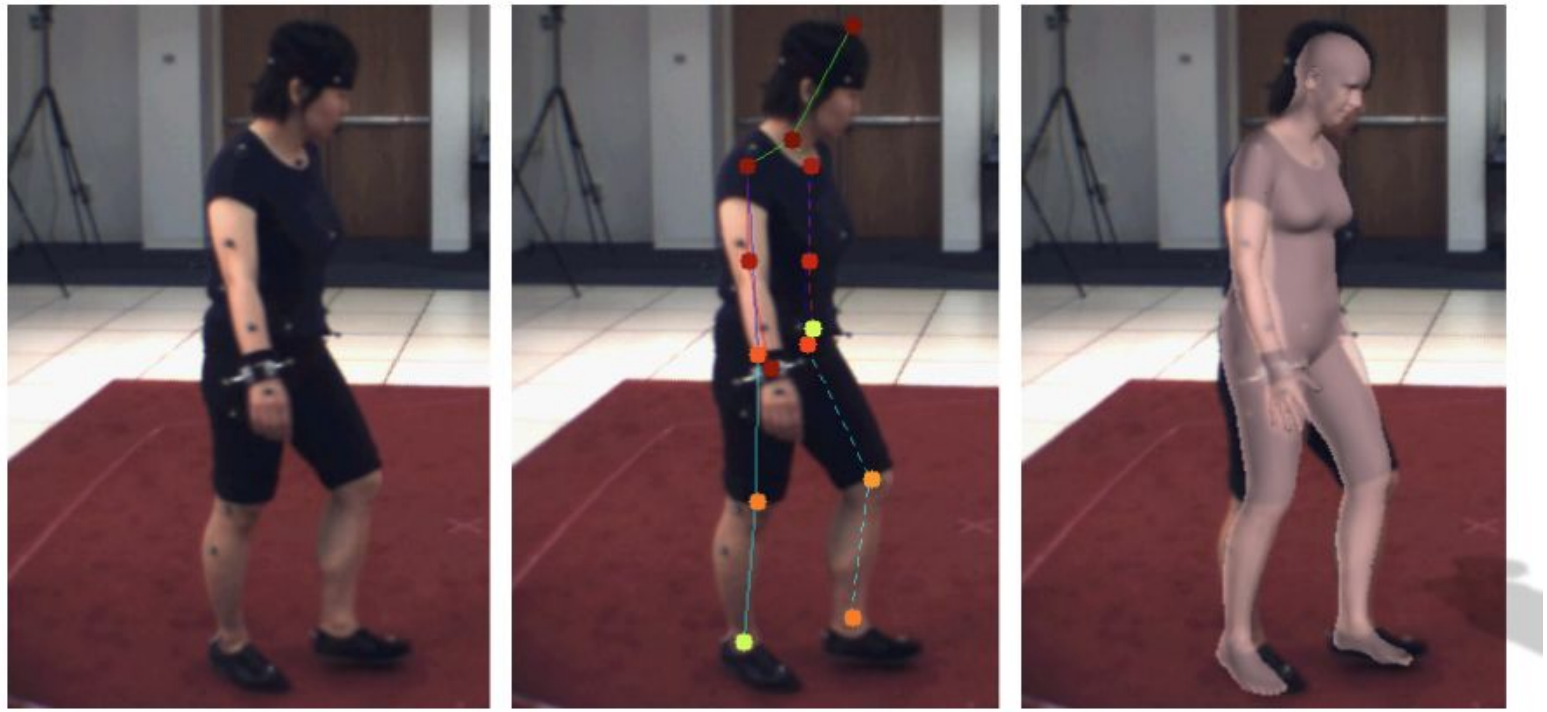
SMPLify

The goal of **SMPLify** is to automatically estimate **both** 3D **pose** and 3D body **shape** from a **single** RGB **image**.

Remember:

3D pose from 2D coordinates is an ambiguous ill-defined problem.

The missing depth information makes it difficult to place the joints in the perpendicular direction.

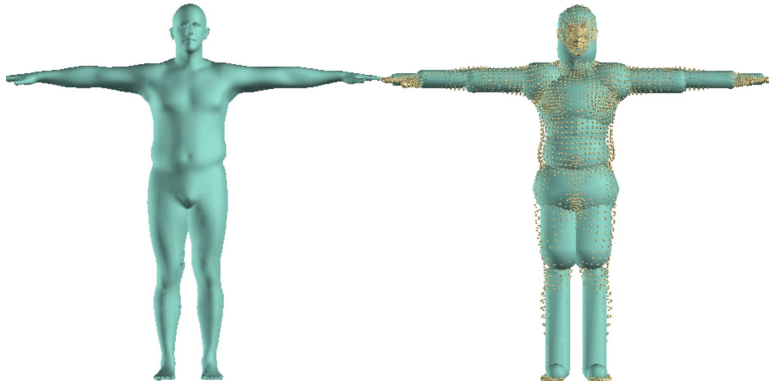


Ingredients

Input: RGB image (pixel coordinates and pixel values) 2D

Intermediate representations:

- **Joint locations: 2D coordinates of 23 joints**
- **Body model: SMPL, it is just a function**
- **Capsules: each bone is replaced by a cylinder**



Output: Posed body model (mesh vertices) 3D

Method: non-linear least-squares optimization using Powell's dogleg

Objective functions

$$E_J(\boldsymbol{\beta}, \boldsymbol{\theta}; K, J_{\text{est}}) + \lambda_\theta E_\theta(\boldsymbol{\theta}) + \lambda_a E_a(\boldsymbol{\theta}) + \lambda_{sp} E_{sp}(\boldsymbol{\theta}; \boldsymbol{\beta}) + \lambda_\beta E_\beta(\boldsymbol{\beta})$$

Joint-based error function: to minimize this term we require that the joint position in 3D is close to the estimated joint position in the image when projected to 2D with a perspective camera projection with parameters K.

$$E_J(\boldsymbol{\beta}, \boldsymbol{\theta}; K, J_{\text{est}}) = \sum_{\text{joint } i} w_i \rho(\Pi_K(R_\theta(J(\boldsymbol{\beta})_i)) - J_{\text{est},i})$$

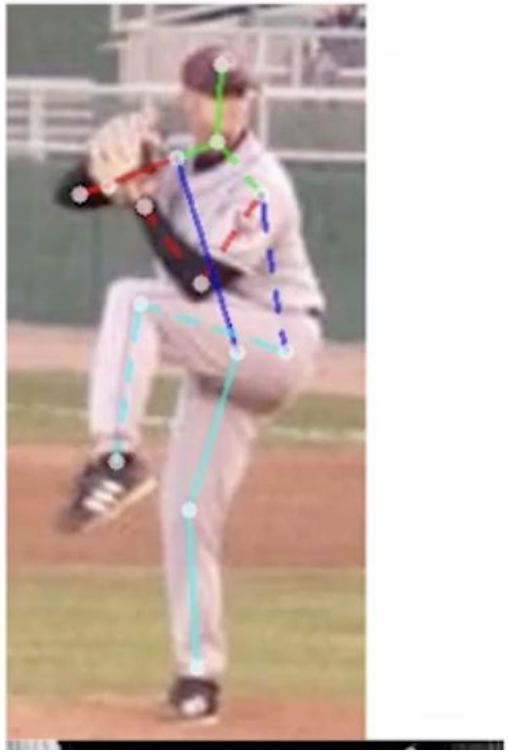
Three pose priors that minimize the following: unnatural bending of knees and elbows, improbable poses, and interpenetration of capsules (avoid intersecting body parts)

A shape prior to keep the shape similar to the description given by the principal components used in the SMPL model

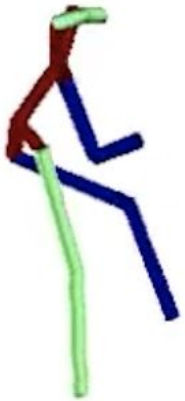
$$E_\beta(\boldsymbol{\beta}) = \boldsymbol{\beta}^T \Sigma_\beta^{-1} \boldsymbol{\beta}$$

Example: comparison

Input



Other approaches without body shape



Output

References

Models

- SMPL
  - Project website: https://smpl.is.tue.mpg.de/
- SMPLify
  - Project website: http://smplify.is.tue.mpg.de/

Datasets

- FAUST
  - Project website: http://faust.is.tue.mpg.de/
- CAESAR
  - Project website: http://store.sae.org/caesar/

My ongoing notes on Notion:

https://www.notion.so/erinaldi/Fundamentals-of-rigging-eb8acd313a444b6999ef1b0a7a13b892

Future discussion

A few links for modern motion capture papers:

- **DeepCap:** DeepCap: Monocular Human Performance Capture Using Weak Supervision, CVPR 2020
- **EventCap:** EventCap: Monocular 3D Capture of High-Speed Human Motions using an Event Camera, CVPR 2020
- **HandCap:** Monocular Real-time Hand Shape and Motion Capture using Multi-modal Data, CVPR 2020
- **StyleRig:** http://gvv.mpi-inf.mpg.de/projects/StyleRig/
- **XNect:** XNect: Real-time Multi-Person 3D Motion Capture with a Single RGB Camera, SIGGRAPH 2020
- **LiveCap:** LiveCap: Real-time Human Performance Capture from Monocular Video, ToG 2019
- **MonoPerfCap:** MonoPerfCap: Human Performance Capture from Monocular Video, TOG 2018

Arithmer 株式会社
〒106-6040
東京都港区六本木一丁目６番１号 泉ガーデンタワー 38/40F(受付)
03-5579-6683
https://arithmer.co.jp/

Arithmer