UniGe | MaLGa

Human in the image

Computational Vision, 05/05/2025

Matteo Moro

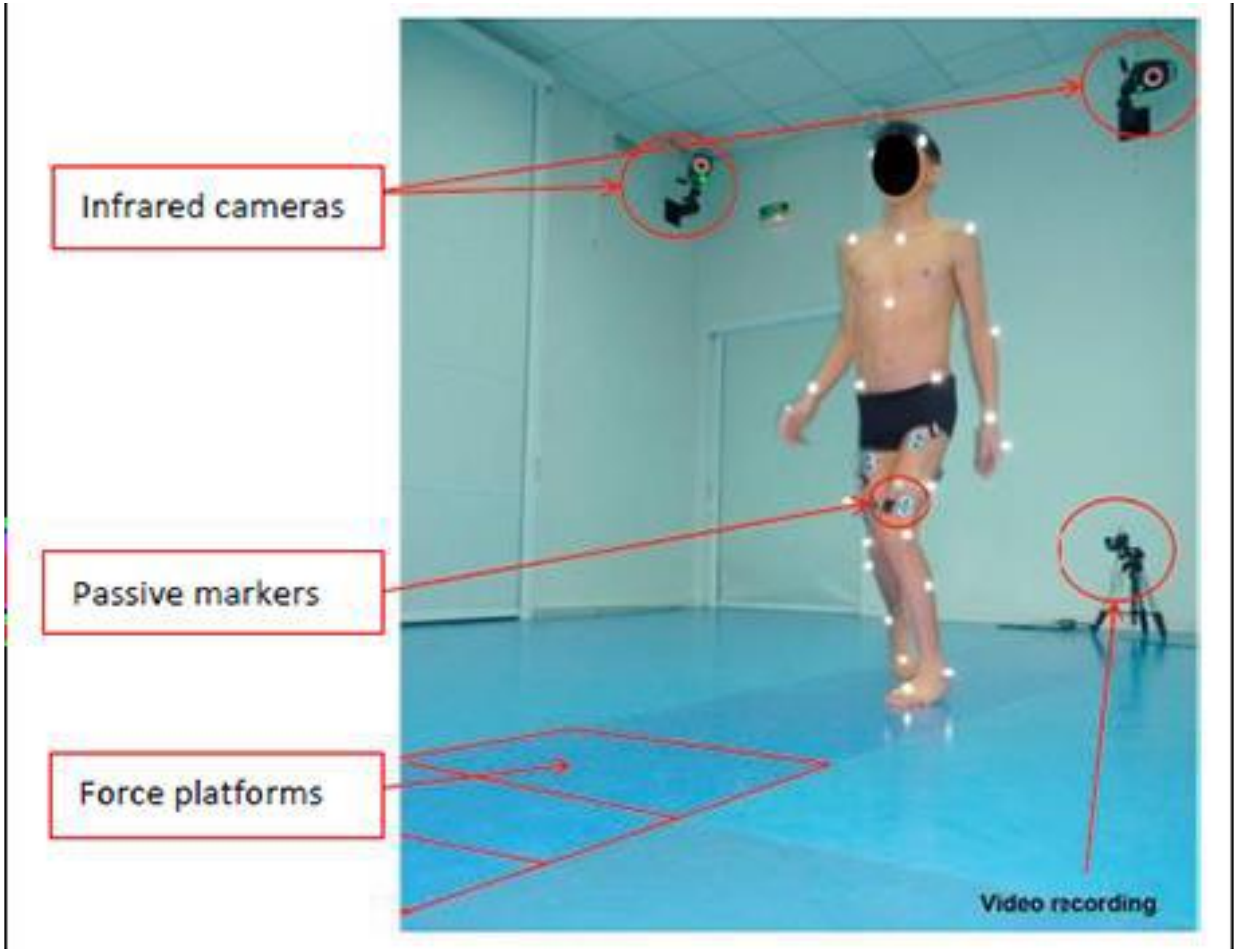
Human motion

Computer-generated Imagery

A typical application of MARKER-BASED approaches, where a precise 3D estimation of the pose is essential

Medical applications

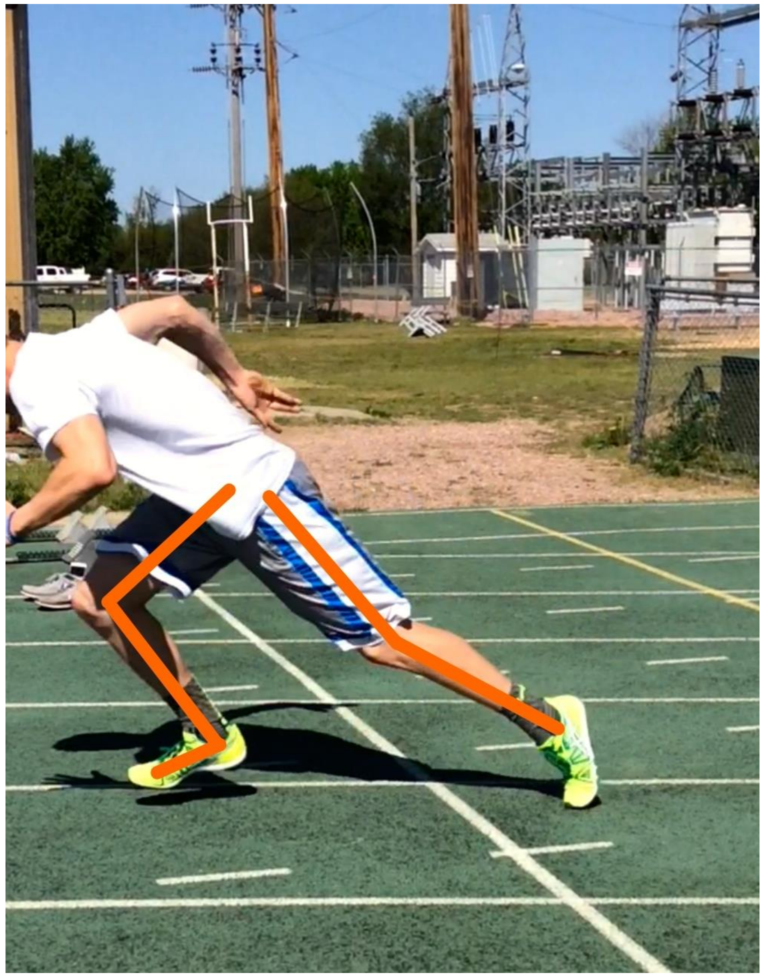
Neuro-motor evaluation and follow-up



Khouri, N., and E. Desailly. "Contribution of clinical gait analysis to single-event multi-level surgery in children with cerebral palsy." *Orthopaedics & Traumatology: Surgery & Research* 103.1 (2017): S105-S111.

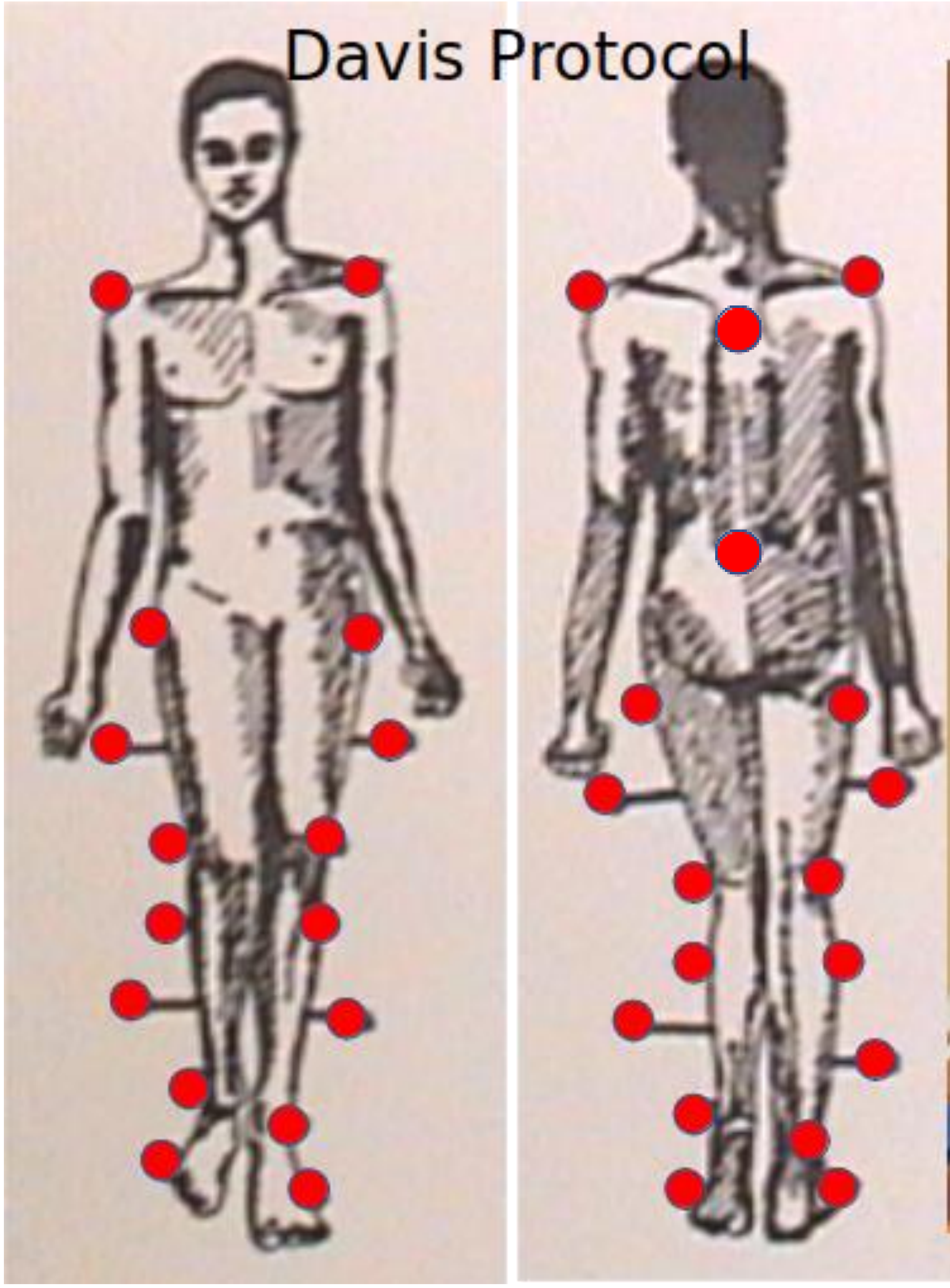
Unconstrained environments

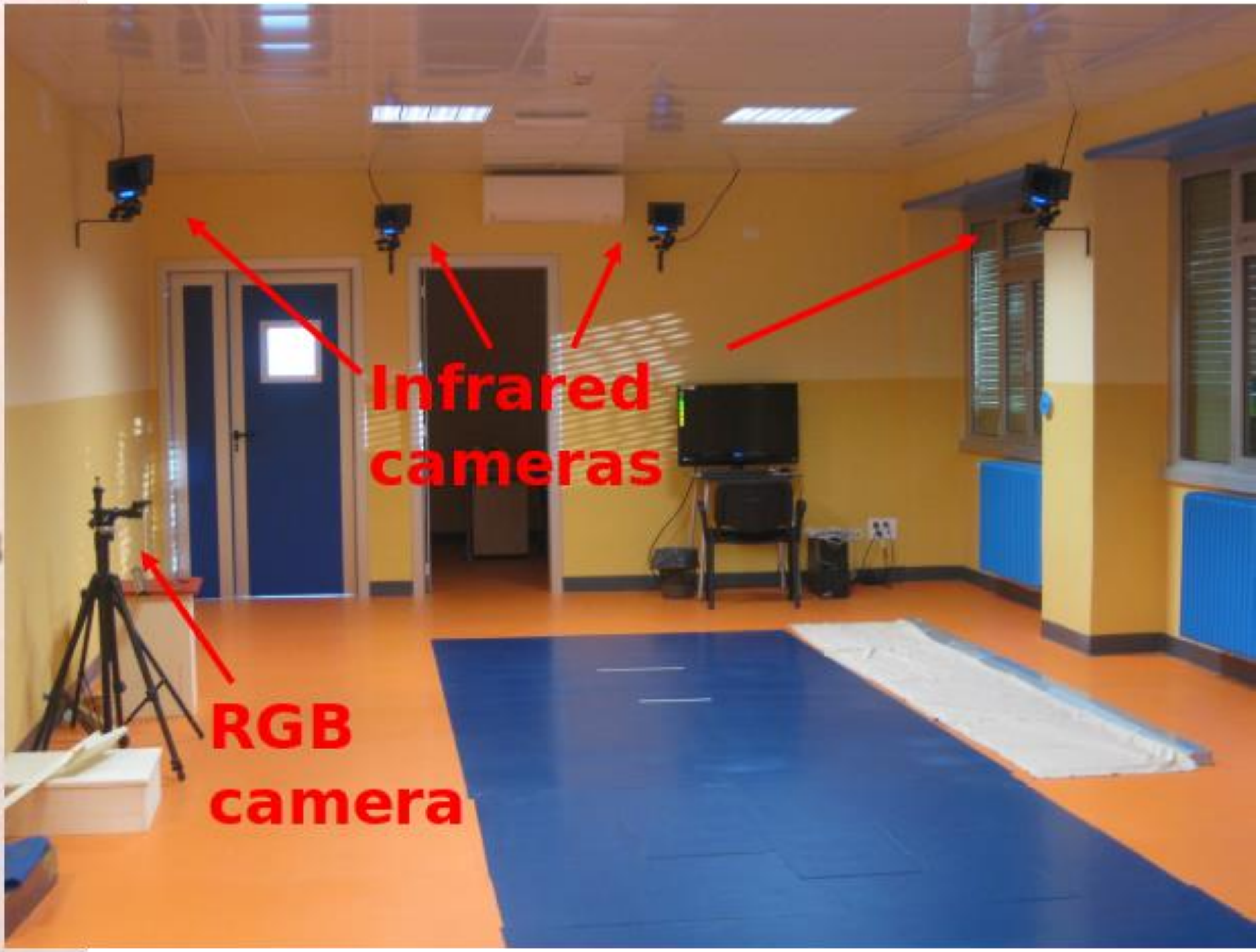
Fare clic per inserire testo



ZOTMAN
pizza pie

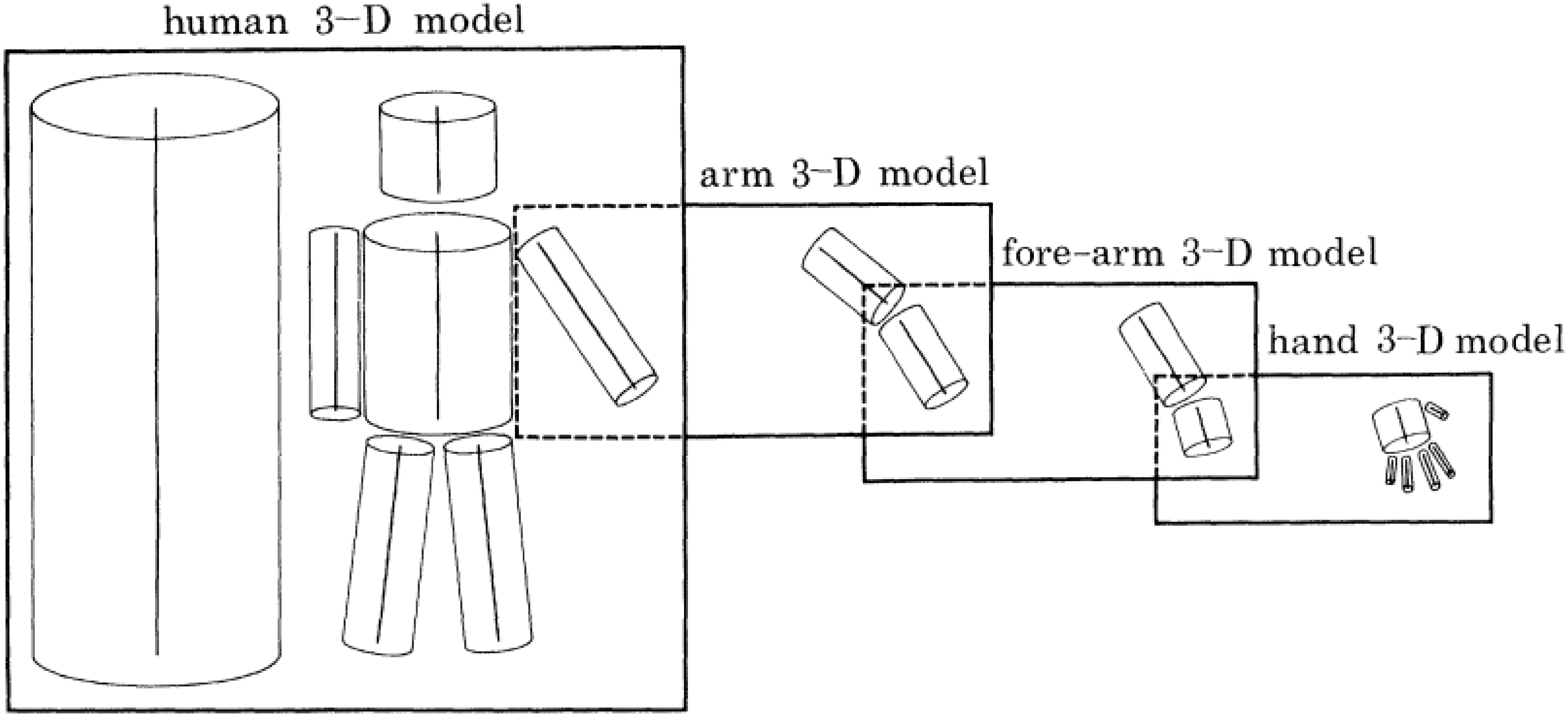
State of the art techniques





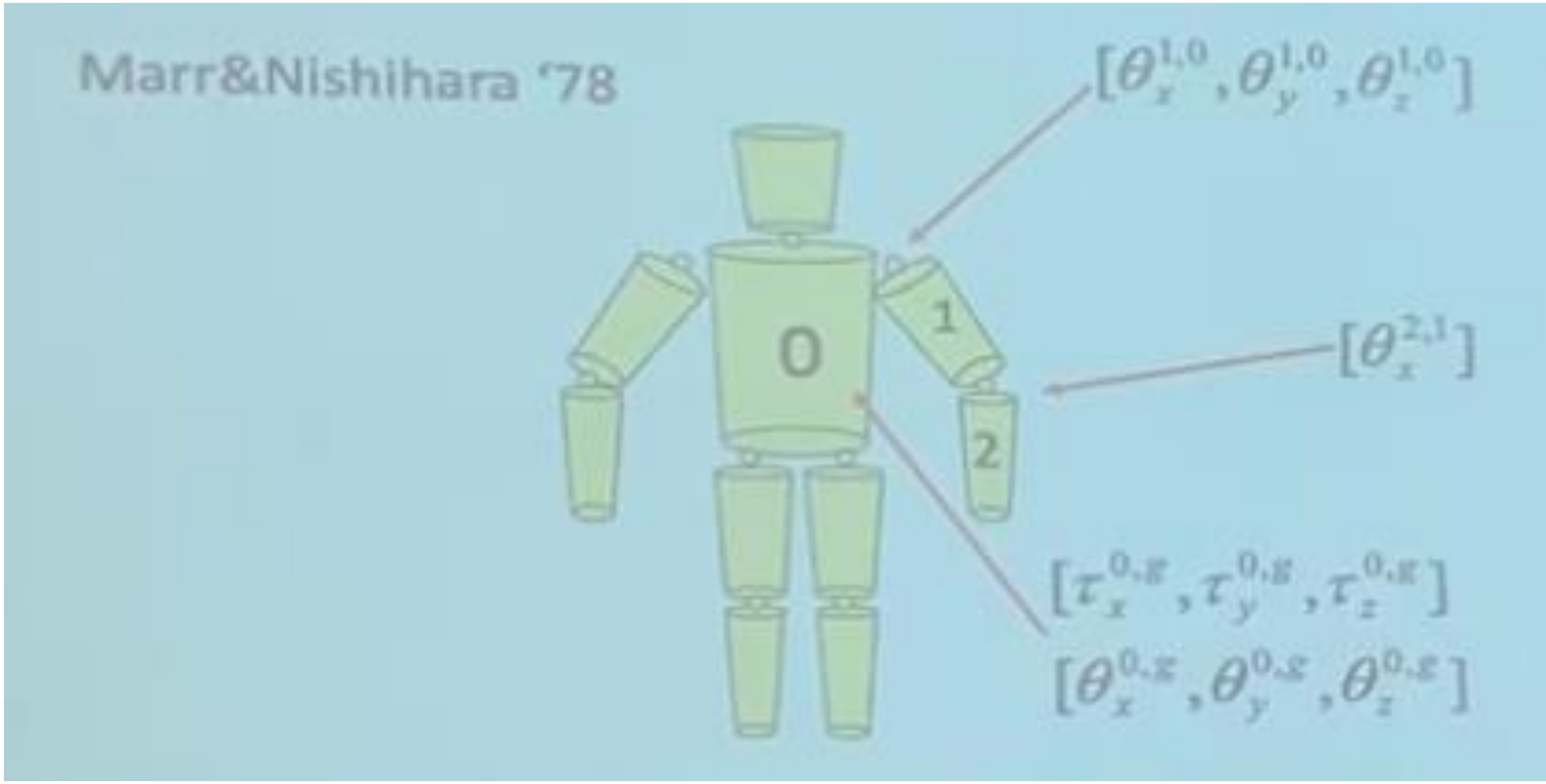
Deformable models

The early history was 3D



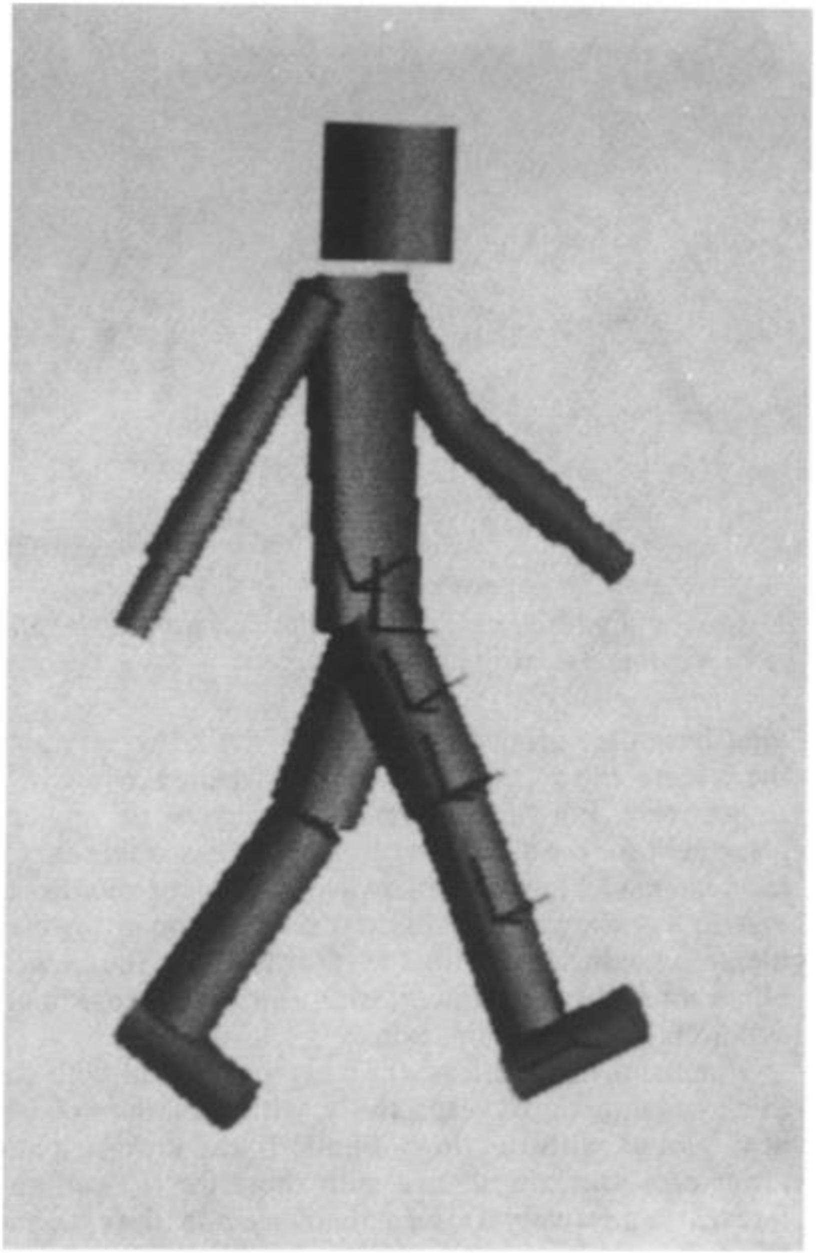
Marr, David, and Herbert Keith Nishihara. "Representation and recognition of the spatial organization of three-dimensional shapes." Proceedings of the Royal Society of London. Series B. Biological Sciences 200.1140 (1978): 269-294.

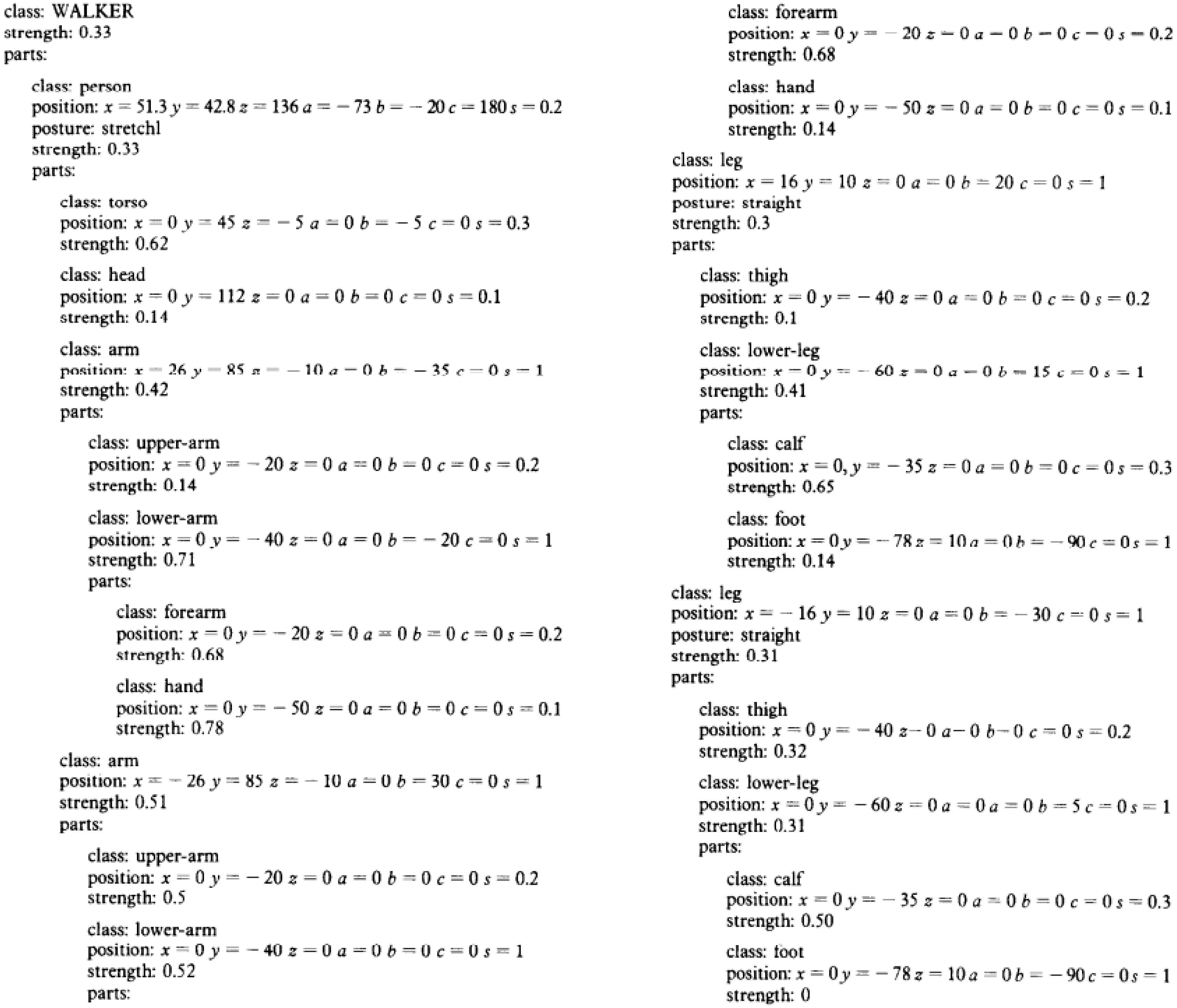
Kinematic tree


Marr&Nishihara '78
$[\theta_x^{1,0}, \theta_y^{1,0}, \theta_z^{1,0}]$
0
1
2
$[\theta_x^{2,1}]$
$[\tau_x^{0,g}, \tau_y^{0,g}, \tau_z^{0,g}]$
$[\theta_x^{0,g}, \theta_y^{0,g}, \theta_z^{0,g}]$

Marr, David, and Herbert Keith Nishihara. "Representation and recognition of the spatial organization of three-dimensional shapes." Proceedings of the Royal Society of London. Series B. Biological Sciences 200.1140 (1978): 269-294.

Rule-based approaches





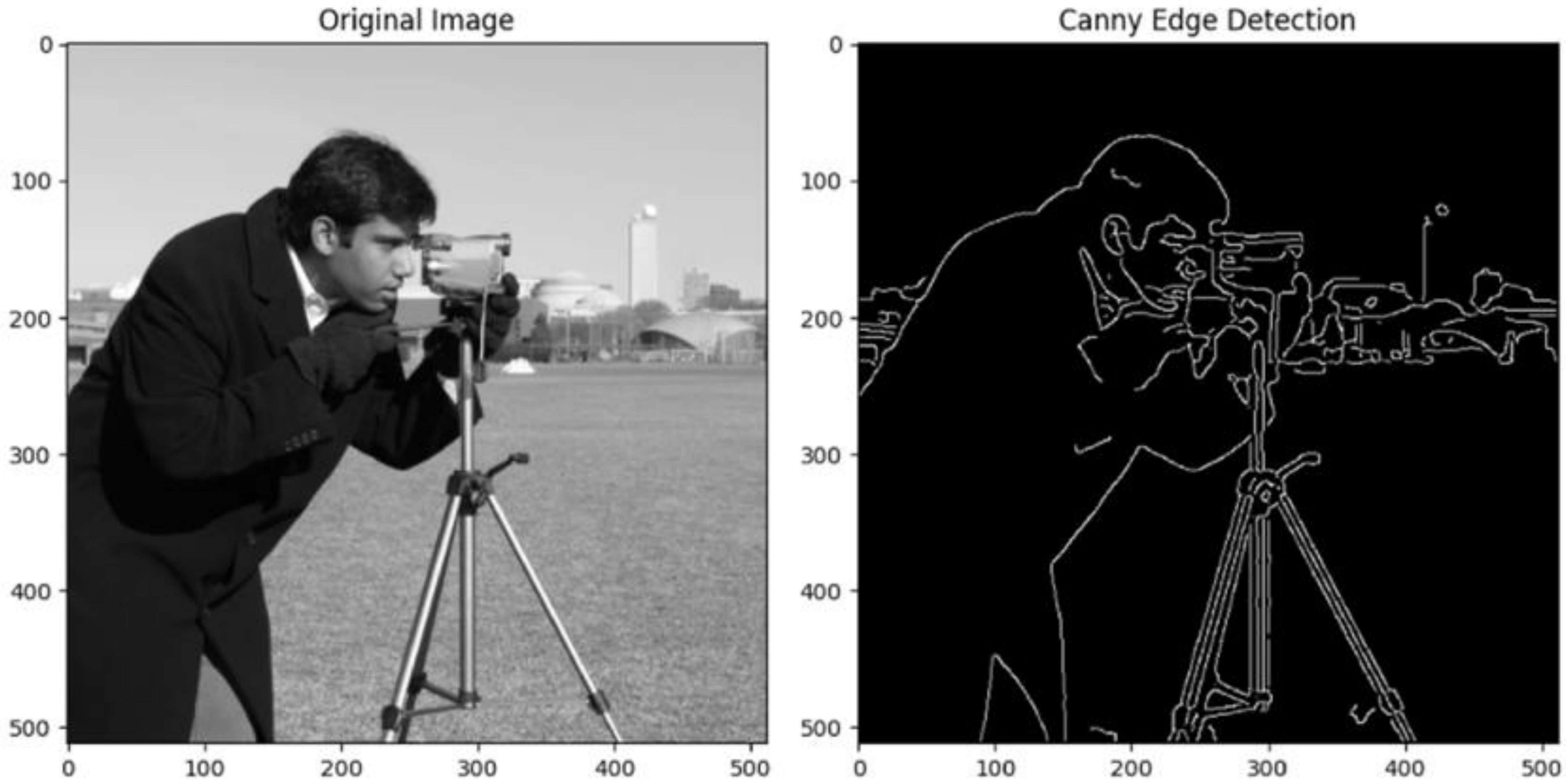
```
class: WALKER
strength: 0.33
parts:

    class: person
    position: x = 51.3 y = 42.8 z = 136 a = − 73 b = − 20 c = 180 s = 0.2
    posture: stretchl
    strength: 0.33
    parts:

        class: torso
        position: x = 0 y = 45 z = − 5 a = 0 b = − 5 c = 0 s = 0.3
        strength: 0.62

        class: head
        position: x = 0 y = 112 z = 0 a = 0 b = 0 c = 0 s = 0.1
        strength: 0.14

        class: arm
        position: x = 26 y = 85 z = − 10 a = 0 b = − 35 c = 0 s = 1
        strength: 0.42
        parts:

            class: upper-arm
            position: x = 0 y = − 20 z = 0 a = 0 b = 0 c = 0 s = 0.2
            strength: 0.14

            class: lower-arm
            position: x = 0 y = − 40 z = 0 a = 0 b = − 20 c = 0 s = 1
            strength: 0.71
            parts:

                class: forearm
                position: x = 0 y = − 20 z = 0 a = 0 b = 0 c = 0 s = 0.2
                strength: 0.68

                class: hand
                position: x = 0 y = − 50 z = 0 a = 0 b = 0 c = 0 s = 0.1
                strength: 0.78

        class: arm
        position: x = − 26 y = 85 z = − 10 a = 0 b = 30 c = 0 s = 1
        strength: 0.51
        parts:

            class: upper-arm
            position: x = 0 y = − 20 z = 0 a = 0 b = 0 c = 0 s = 0.2
            strength: 0.5

            class: lower-arm
            position: x = 0 y = − 40 z = 0 a = 0 b = 0 c = 0 s = 1
            strength: 0.52
            parts:

                class: forearm
                position: x = 0 y = − 20 z = 0 a = 0 b = 0 c = 0 s = 0.2
                strength: 0.68

                class: hand
                position: x = 0 y = − 50 z = 0 a = 0 b = 0 c = 0 s = 0.1
                strength: 0.14

        class: leg
        position: x = 16 y = 10 z = 0 a = 0 b = 20 c = 0 s = 1
        posture: straight
        strength: 0.3
        parts:

            class: thigh
            position: x = 0 y = − 40 z = 0 a = 0 b = 0 c = 0 s = 0.2
            strength: 0.1

            class: lower-leg
            position: x = 0 y = − 60 z = 0 a = 0 b = 15 c = 0 s = 1
            strength: 0.41
            parts:

                class: calf
                position: x = 0, y = − 35 z = 0 a = 0 b = 0 c = 0 s = 0.3
                strength: 0.65

                class: foot
                position: x = 0 y = − 78 z = 10 a = 0 b = − 90 c = 0 s = 1
                strength: 0.14

        class: leg
        position: x = − 16 y = 10 z = 0 a = 0 b = − 30 c = 0 s = 1
        posture: straight
        strength: 0.31
        parts:

            class: thigh
            position: x = 0 y = − 40 z− 0 a− 0 b− 0 c = 0 s = 0.2
            strength: 0.32

            class: lower-leg
            position: x = 0 y = − 60 z = 0 a = 0 a = 0 b = 5 c = 0 s = 1
            strength: 0.31
            parts:

                class: calf
                position: x = 0 y = − 35 z = 0 a = 0 b = 0 c = 0 s = 0.3
                strength: 0.50

                class: foot
                position: x = 0 y = − 78 z = 10 a = 0 b = − 90 c = 0 s = 1
                strength: 0
```

Hogg, David. "Model-based vision: a program to see a walking person." Image and Vision computing 1.1 (1983): 5-20.

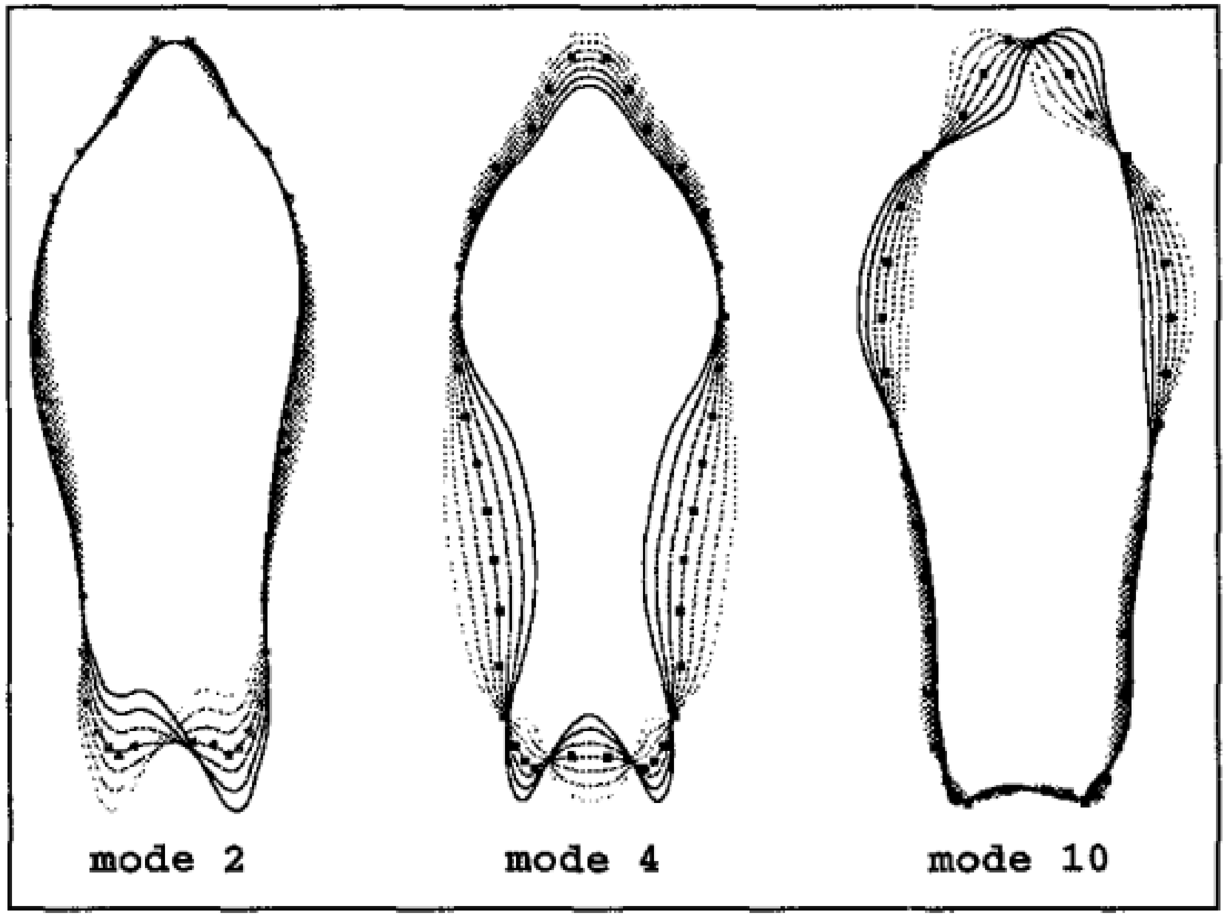
UniGe | MaLGa

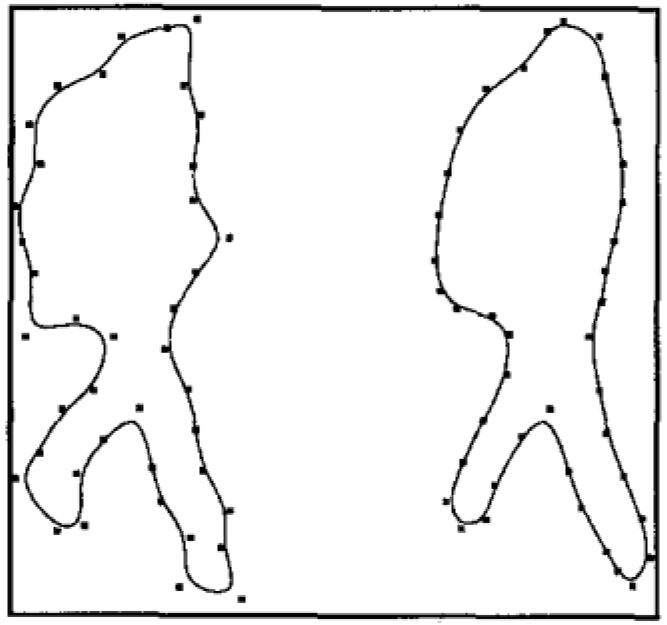
Contour-based 2D models

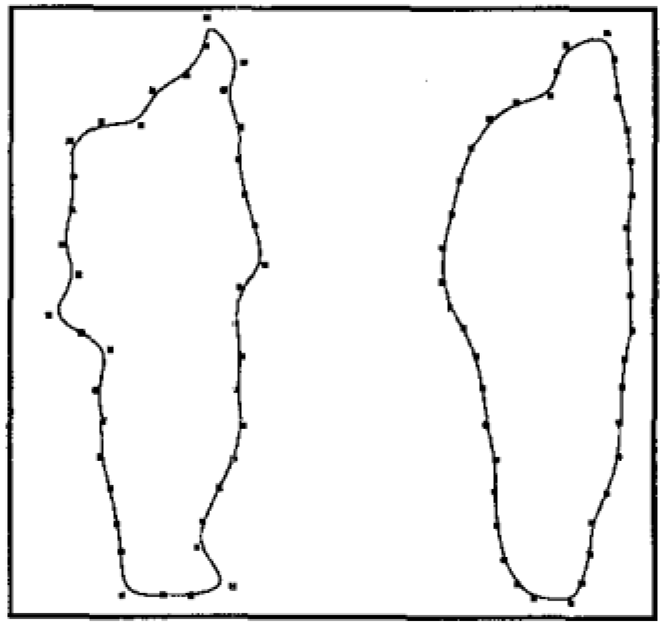
Edge detection



First learned body model







Baumberg, Adam, and David Hogg. "Learning flexible models from image sequences." Computer Vision—ECCV'94: Third European Conference on Computer Vision Stockholm, Sweden, May 2–6, 1994 Proceedings, Volume I 3. Springer Berlin Heidelberg, 1994.

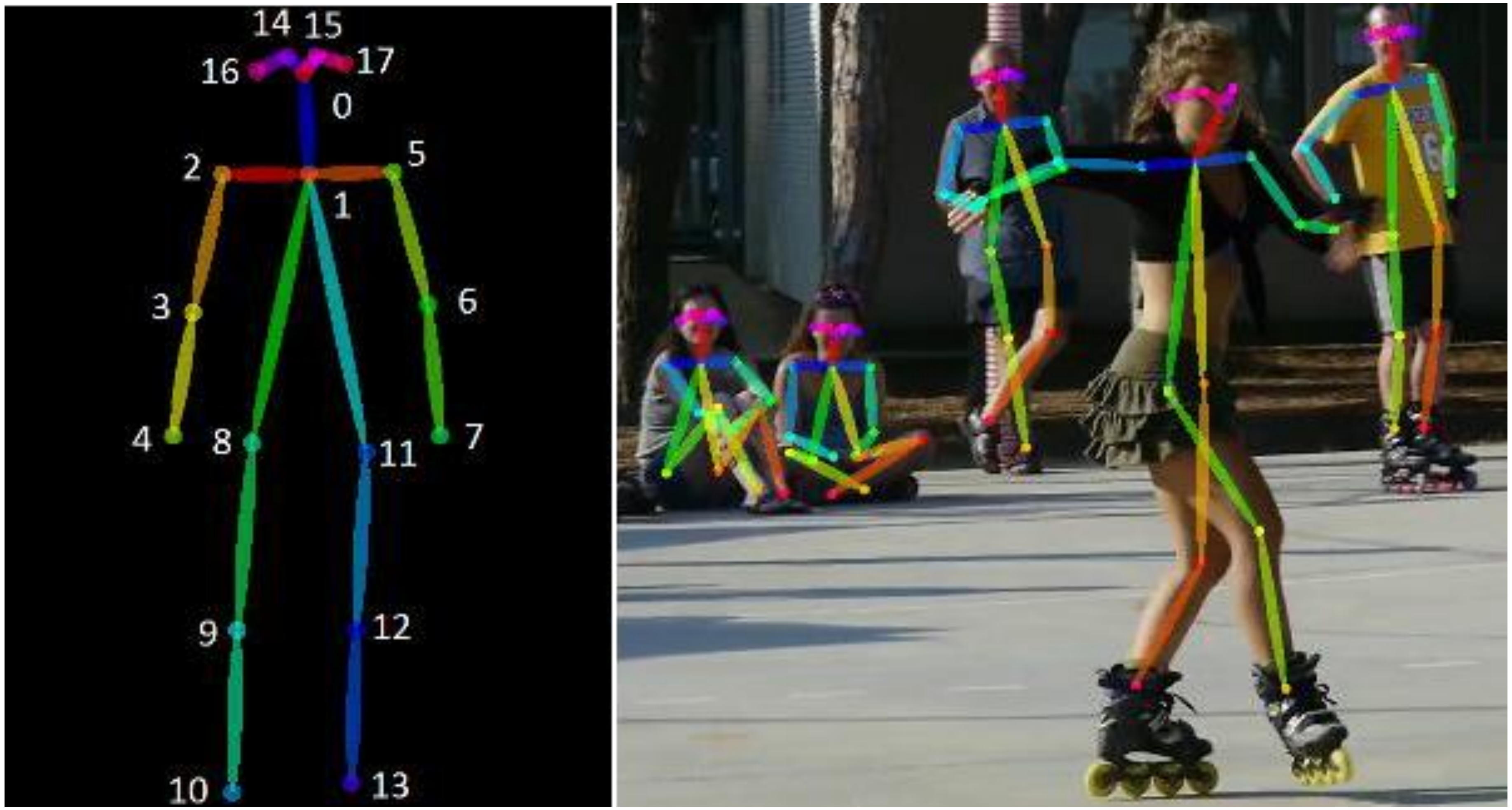
UniGe | MaLGa

2D Skeleton-based human pose estimation

Human Pose Estimation

Problem: estimating the configuration of the body (pose)

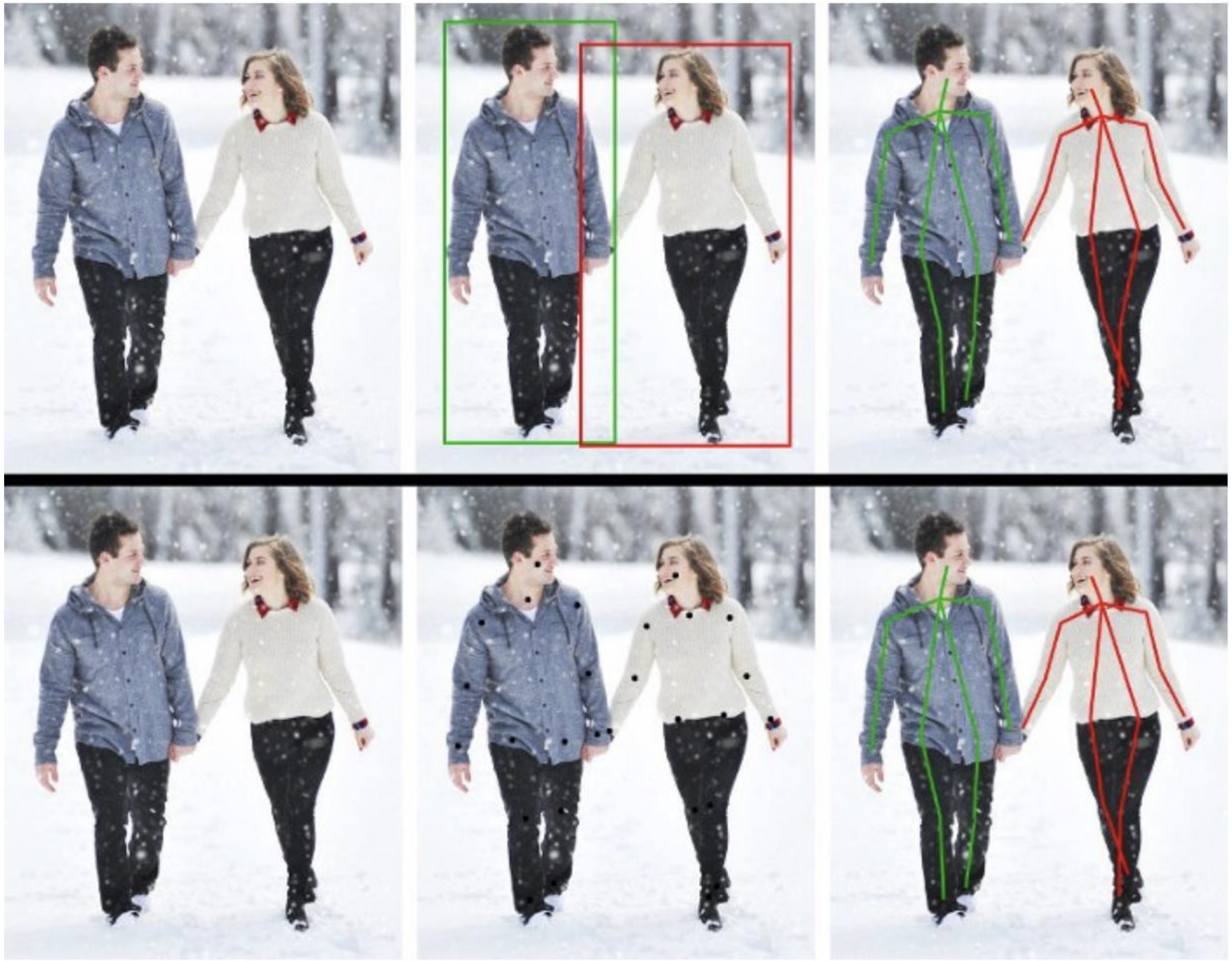
While classical methods adopt wearable sensors or combinations of markers and motion capture (MoCap) systems, in Computer Vision this task is addressed using images as Inputs.



Human Pose Estimation: variants

Different methods are based on different assumptions on how many people can be jointly analyzed: **single person** or **multi- persons** algorithms exist

Top-down and bottom-up approaches



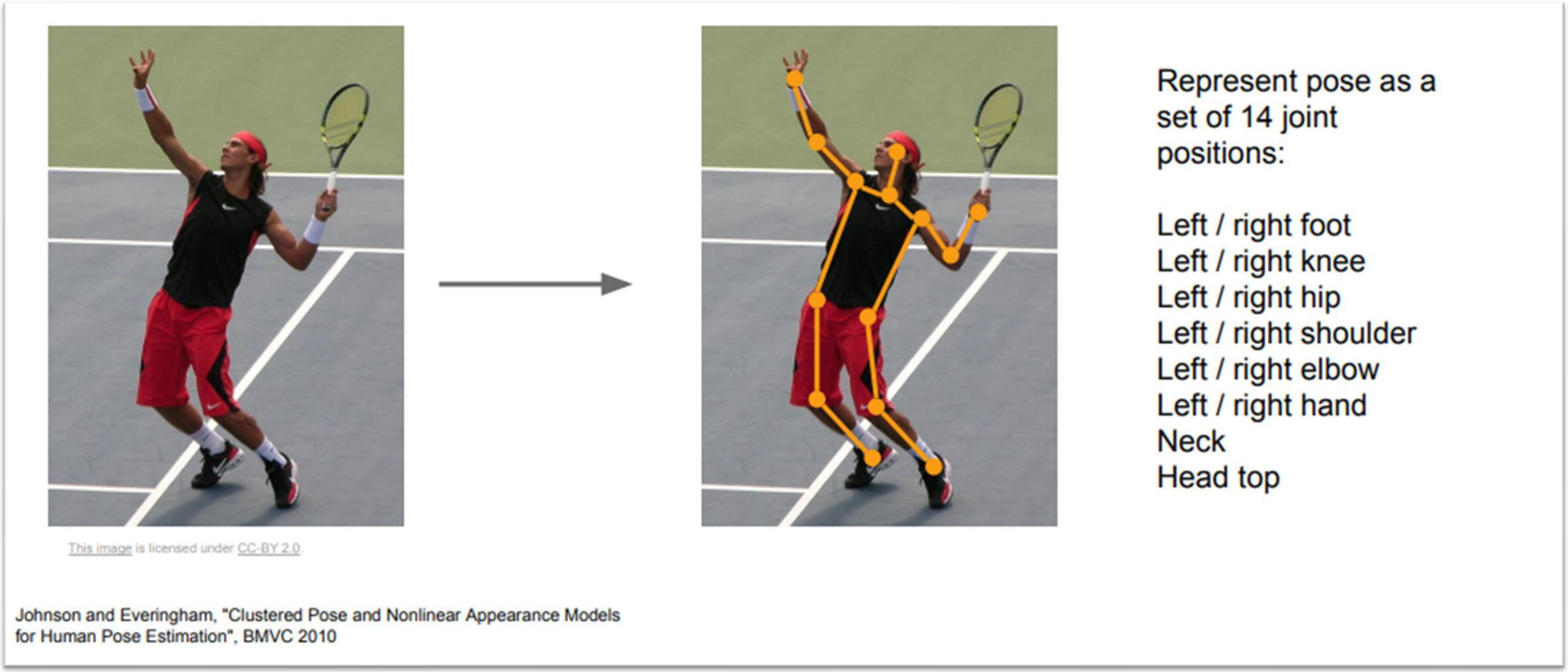
Top down

Bottom up

Notice: *bottom-up* approaches can be used as a “richer” alternative to people detectors

HPE following detection (top-down)

-*Top-down* approaches start from the output of a person detector.



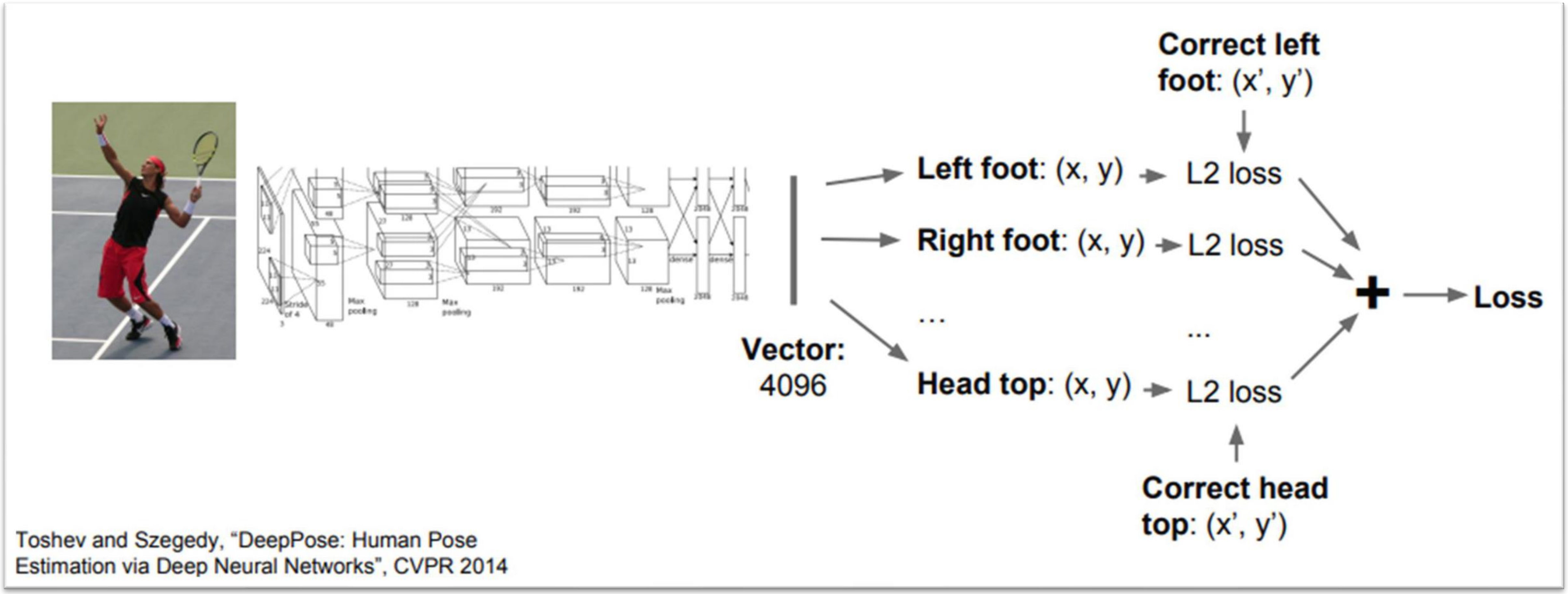
Represent pose as a set of 14 joint positions:

Left / right foot
Left / right knee
Left / right hip
Left / right shoulder
Left / right elbow
Left / right hand
Neck
Head top

Johnson and Everingham, "Clustered Pose and Nonlinear Appearance Models for Human Pose Estimation", BMVC 2010

HPE following detection (top-down)

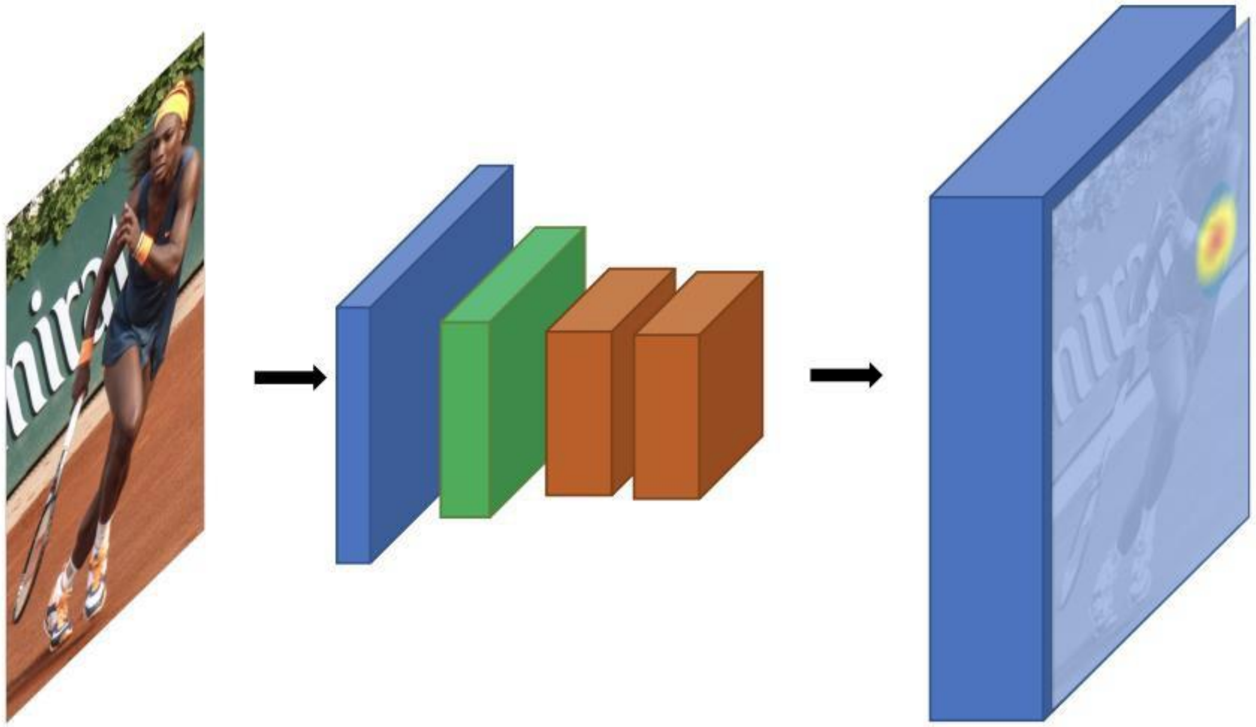
-A possible strategy to address the task is to define the localization of each feature as a regression task (as we did for object localization)



Toshev and Szegedy, "DeepPose: Human Pose Estimation via Deep Neural Networks", CVPR 2014

HPE following detection (towards bottom-up)

-An alternative approach is the estimation of heat maps

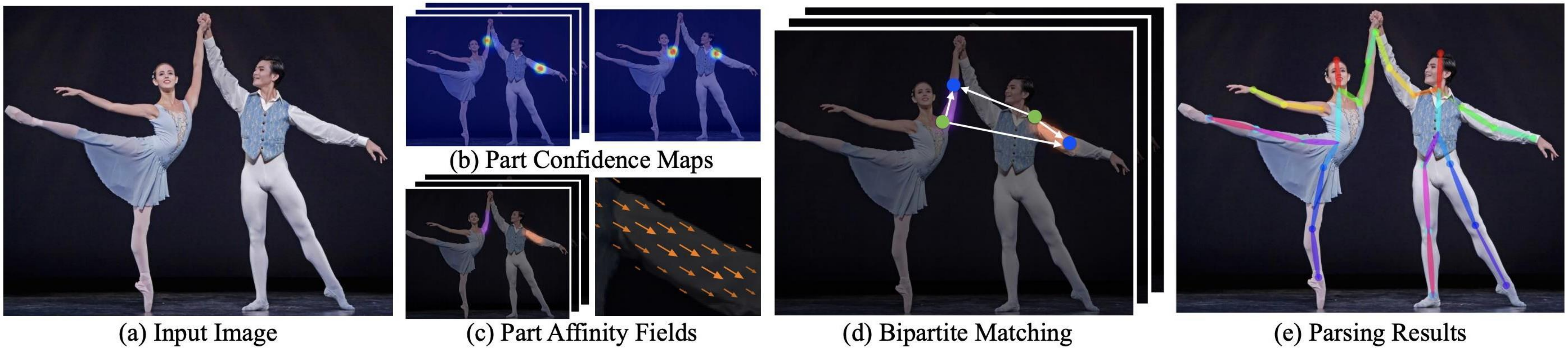


Some of the challenges for HPE (multi- person)

- Unknown number of people
- People can appear at any pose or scale
- People contact and overlapping
- Runtime complexity grows with the number of people

Bottom-up detection: OpenPose

It first detects parts (keypoints) belonging to every person in the image, followed by assigning parts to distinct individuals. Shown below is the architecture of the OpenPose model.



(a) Input Image (b) Part Confidence Maps (c) Part Affinity Fields (d) Bipartite Matching (e) Parsing Results

Figure 2. Overall pipeline. Our method takes the entire image as the input for a two-branch CNN to jointly predict confidence maps for body part detection, shown in (b), and part affinity fields for parts association, shown in (c). The parsing step performs a set of bipartite matchings to associate body parts candidates (d). We finally assemble them into full body poses for all people in the image (e).

https://arxiv.org/pdf/1611.08050.pdf

OpenPose pipeline

(a) Take the entire image as the input for a CNN

(b) Predict confidence maps for body parts detection

(c) Predict PAFs for part association

(d) Perform a set of bipartite matching

(e) Assemble into a full body pose

Open Pose: Schematic architecture (1)

- An input RGB image is fed to a two branch multi-stage CNN;

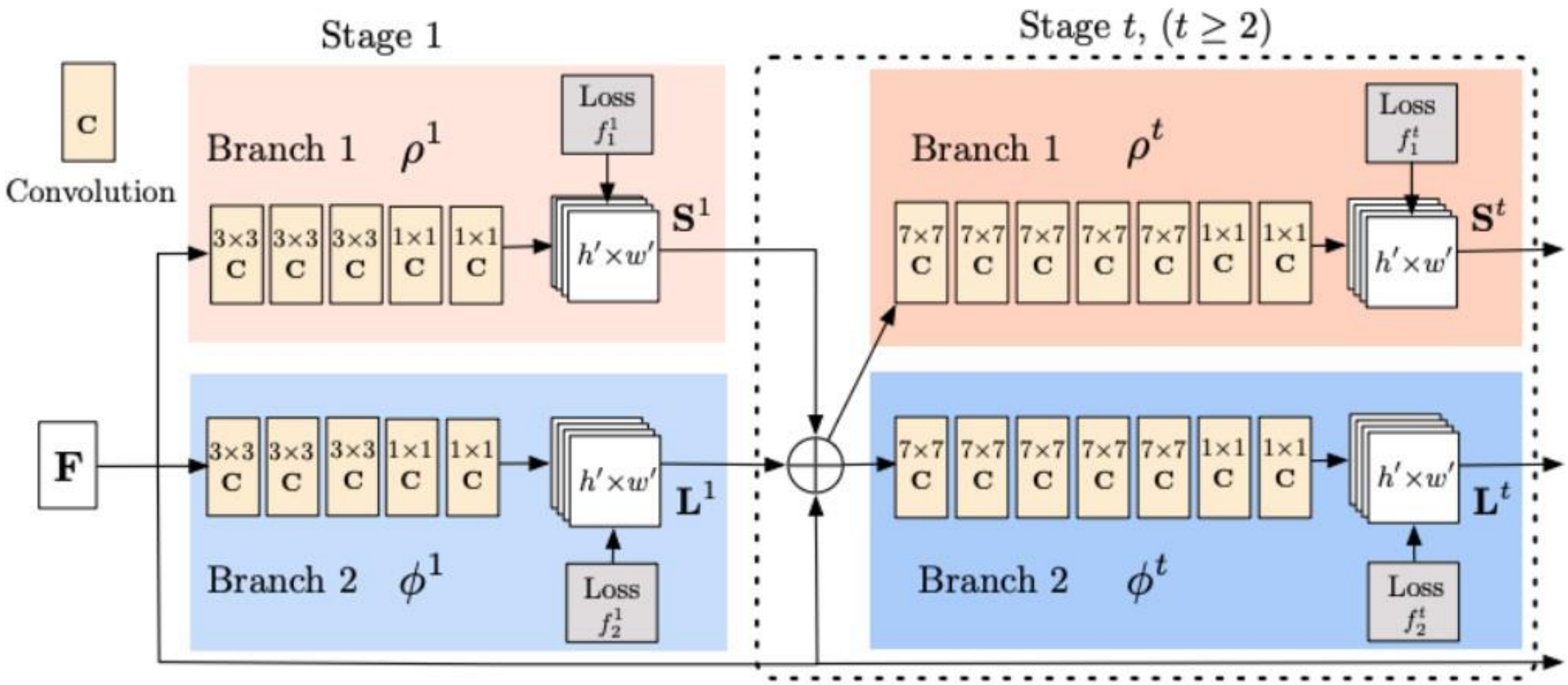


Fig 2. Architecture of the two-branch multi-stage CNN. Image taken from "Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields".

Open Pose: Schematic architecture (2)

Two-branches: The top branch, shown in beige, predicts the confidence maps of different body parts location such as the right eye, left eye, right elbow and others. The bottom branch, shown in blue, predicts the part affinity fields, which represents a degree of association between different body parts.

Multi-stage: At step 0 an initial estimation of confidence maps and part affinity fields. In the next steps, the image is concatenated with the initial estimations to obtain refined estimations.

Finally, a greedy inference algorithm produces the final output, with the identification of the pose.

Further details on architecture for OpenPose

The image is first analyzed by a pre-trained convolutional neural network such as the first 10 layers of VGG-19, to produce a set of feature maps **F**

Stage 1

$$\mathbf{S}^1 = \rho^1(\mathbf{F})$$

$$\mathbf{L}^1 = \phi^1(\mathbf{F})$$

$$\begin{aligned}\mathbf{S}^t &= \rho^t(\mathbf{F}, \mathbf{S}^{t-1}, \mathbf{L}^{t-1}), \ \forall t \geq 2, \\ \mathbf{L}^t &= \phi^t(\mathbf{F}, \mathbf{S}^{t-1}, \mathbf{L}^{t-1}), \ \forall t \geq 2,\end{aligned}$$

Stage >=2

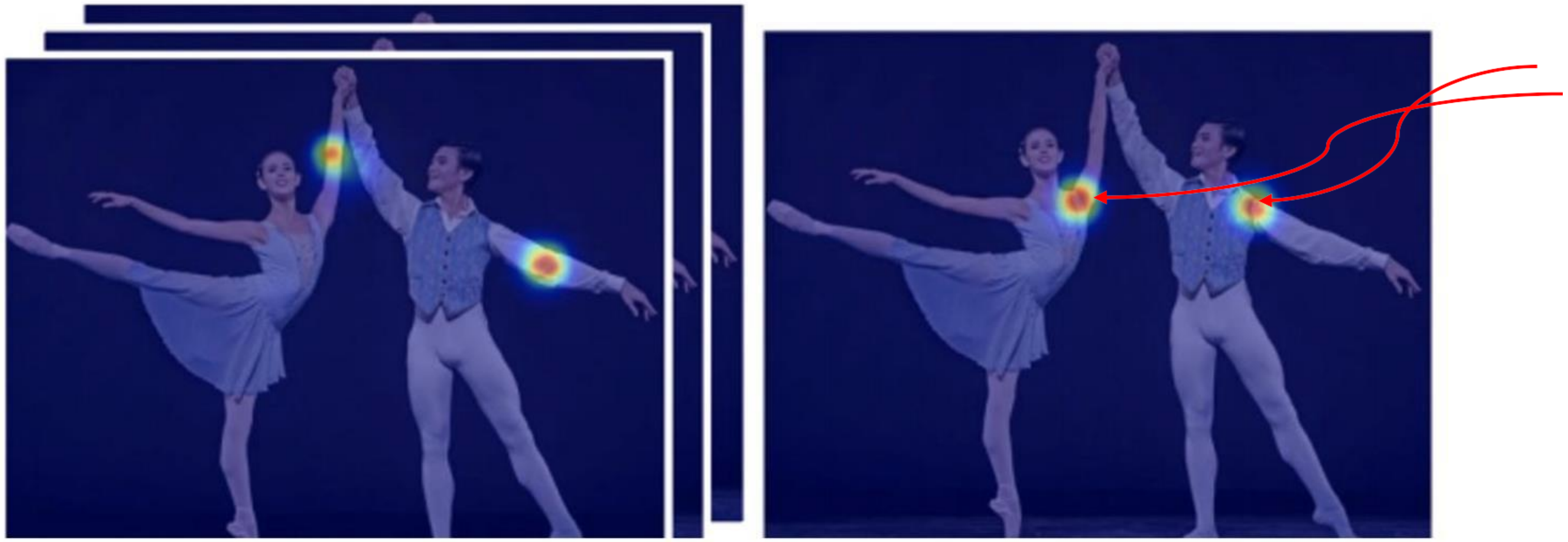
What about the loss function?

L2 losses;

Total loss is the sum of the two L2 losses for confidence maps and part affinity fields

$$
\begin{aligned}
f_{\mathbf{S}}^{t} &= \sum_{j=1}^{J} \sum_{\mathbf{p}} \mathbf{W}(\mathbf{p}) \cdot \|\mathbf{S}_j^t(\mathbf{p}) - \mathbf{S}_j^*(\mathbf{p})\|_2^2, \\
f_{\mathbf{L}}^{t} &= \sum_{c=1}^{C} \sum_{\mathbf{p}} \mathbf{W}(\mathbf{p}) \cdot \|\mathbf{L}_c^t(\mathbf{p}) - \mathbf{L}_c^*(\mathbf{p})\|_2^2,
\end{aligned}
$$

OpenPose: confidence map



HIGH CONFIDENCE AREAS

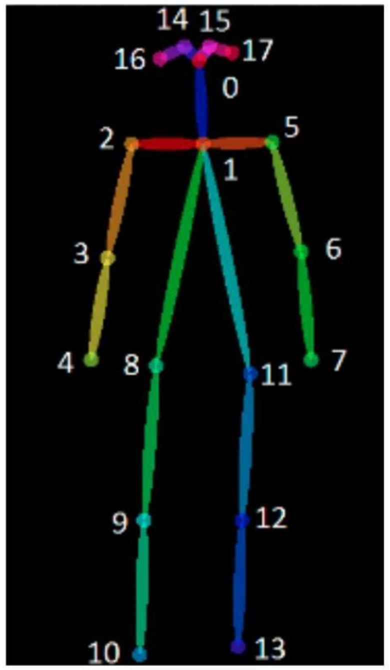
(SHOULDER)

Confidence map is the 2D representation of the belief that a particular body part can be located .A single body part will be represented on a single map. So, the number of maps is the same as the total number of the body parts.

$$S = (S_1, S_2, S_3 \ ... \ S_j)$$

$$S_j \in \mathbb{R}^{w \times h}$$

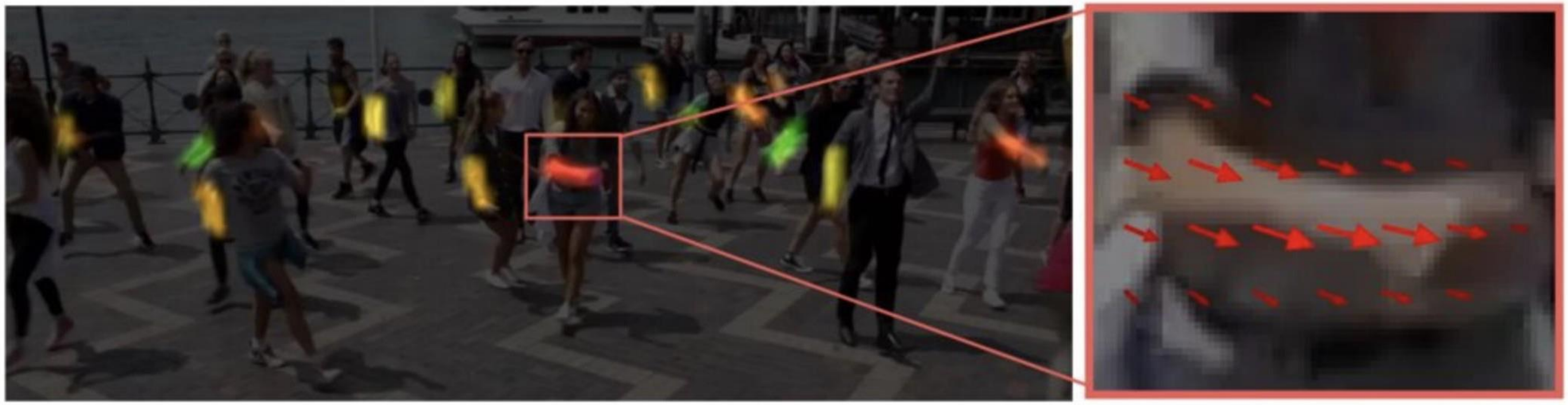
$j \in \{1 \ ... \ \mathrm{J}\}$ where J is the total number of body parts



OpenPose: skeleton derivation

Part Affinity Fields

Vector fields that encode the location and the orientation of limbs in the image



$\mathrm{L} = (L_1, L_2, L_3 \ \dots L_c)$

$L_c \in \mathbb{R}^{w \times h \times 2}$

c ∈ {1 ... C} where C is the total number of limbs

Evaluation Metrics – some examples

Defining with x_p the predicted keypoint and with x_{GT} the ground truth:

- **Percentage of Correct Parts – PCP**: the distance between two predicted joints locations (x^1_p and x^2_p) and the true limb joint locations is less than half of the limb length (L)

$$|(x^1_p - x^2_p) - (x^1_{GT} - x^2_{GT})| < L/2$$

Evaluation Metrics – some examples

Defining with x_p the predicted joint and with x_{GT} the ground truth:

- **Percentage of Correct Key-points – PCK**: A detected joint is considered correct if the distance between the predicted (x_p) and the true joint (x_{GT}) is within a certain threshold (*thr*)

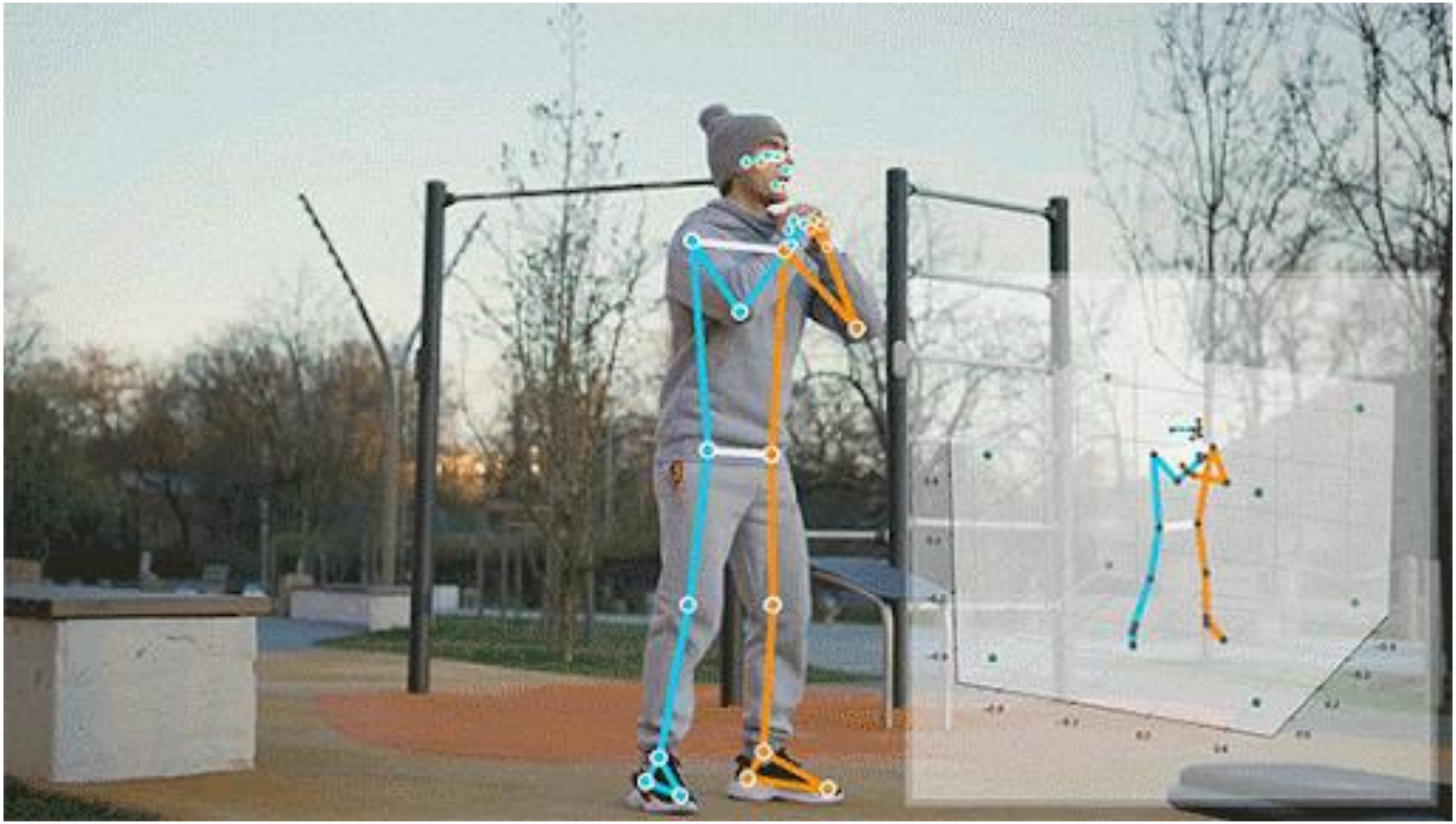
$$|x_p - x_{GT}| < thr$$

- **Mean Average Precision (mAP):** percentage of correctly estimated joints

UniGe | MaLGa

3D monocular human pose estimation

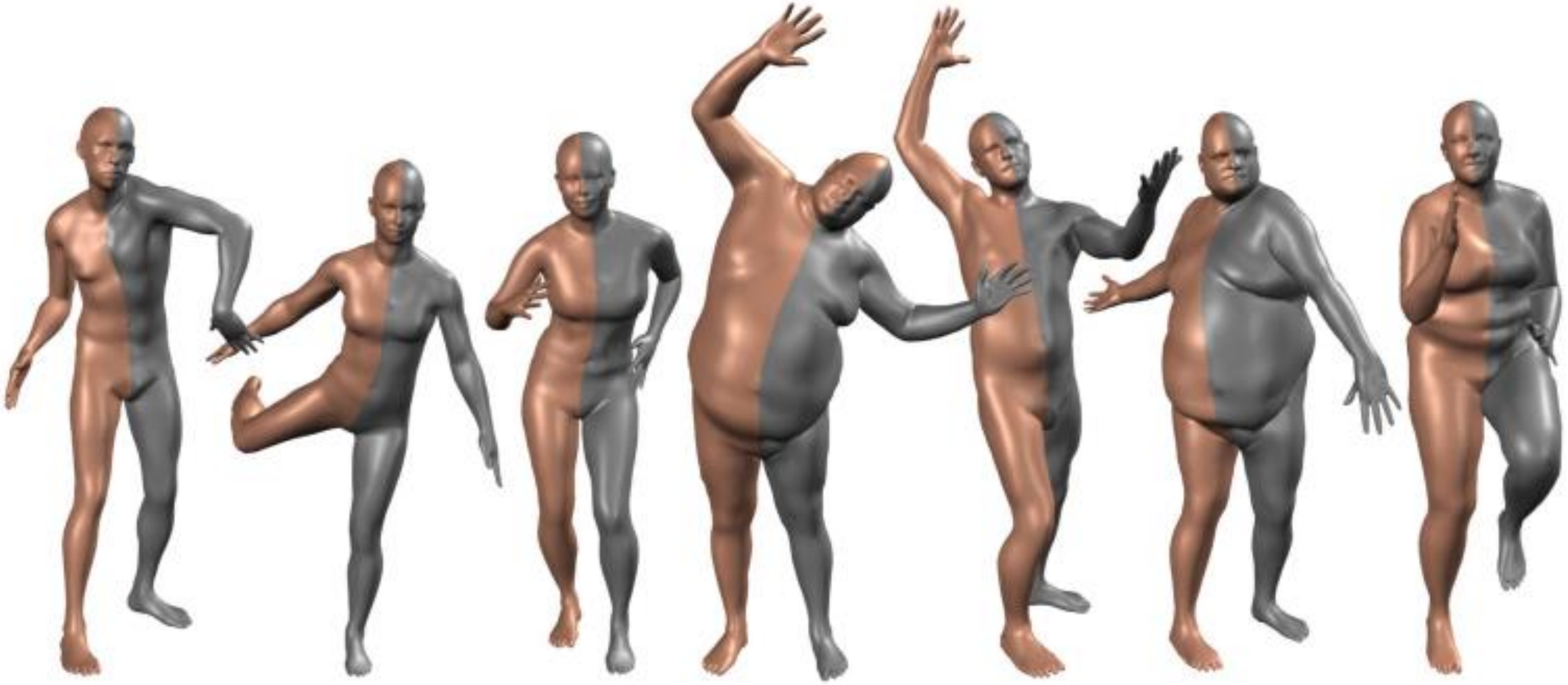
Mediapipe and TensorFlow.js



https://blog.tensorflow.org/2021/08/3d-pose-detection-with-mediapipe-blazepose-ghum-tfjs.html

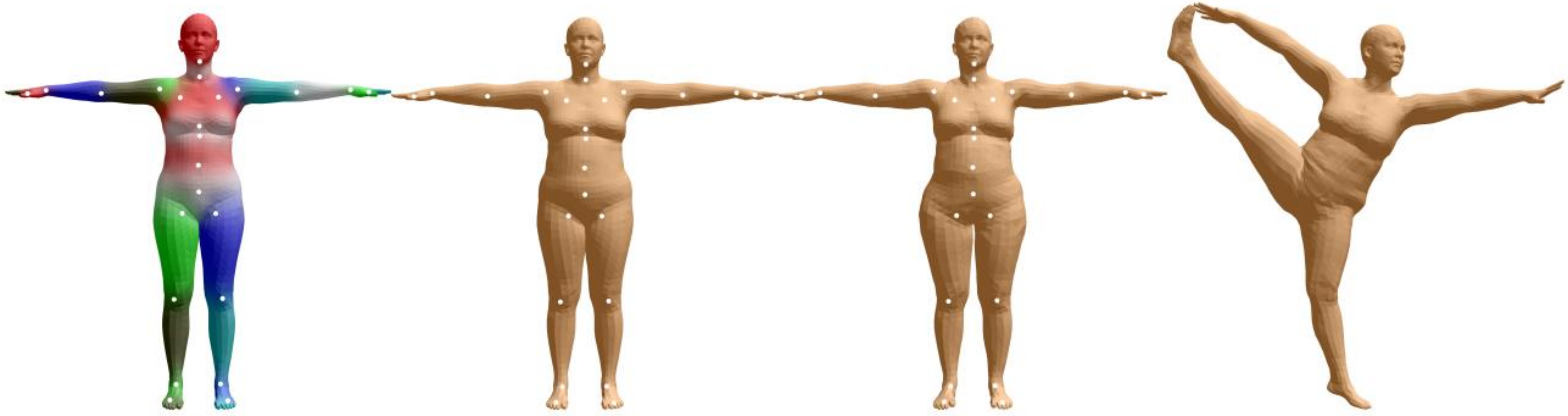
Skinned multi-person linear mode (SMPL)

SMPL



Loper, Matthew, et al. "SMPL: A skinned multi-person linear model." Seminal Graphics Papers: Pushing the Boundaries, Volume 2. 2023. 851-866.

SMPL



(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)** $W(T_P(\vec{\beta},\vec{\theta}), J(\vec{\beta}), \vec{\theta}, \mathcal{W})$

Loper, Matthew, et al. "SMPL: A skinned multi-person linear model." Seminal Graphics Papers: Pushing the Boundaries, Volume 2. 2023. 851-866.

UniGe

MaLGa