Virtual Humans – Winter 23/24

Lecture 9_1 – Neural Implicits and Point Based Clothing Models

Prof. Dr.-Ing. Gerard Pons-Moll
University of Tübingen / MPI-Informatics

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

PART1: Neural Implicits for 3D Shapes

PART2: Neural Implicits for Humans

PART3: Neural Implicits – Generative Models

PART4: Point-based Clothing Models

PART1: Neural Implicits for 3D Shapes

PART2: Neural Implicits for Humans

PART3: Neural Implicits – Generative Models

PART4: Point-based Clothing Models

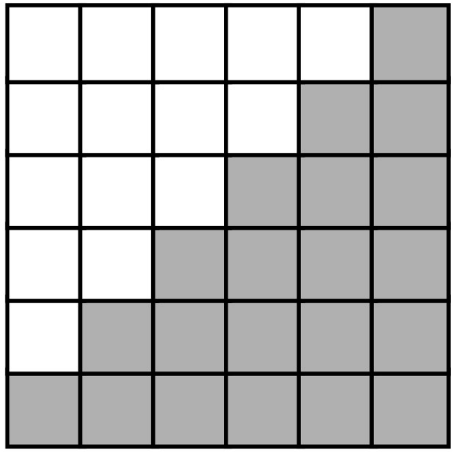
What is a good representation for 3D data?

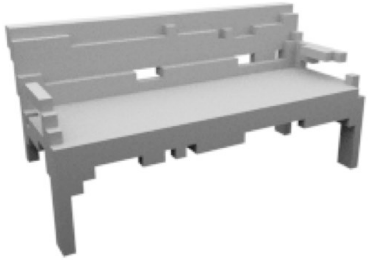
What is a good representation for 3D data?

- Compatible with neural networks.
- Flexible
- High fidelity

Voxels

- Discretization of 3D space into grid.
- Easy to process with neural networks.
- Cubic memory $\mathcal{O}(n^3) \longrightarrow$ limited resolution.



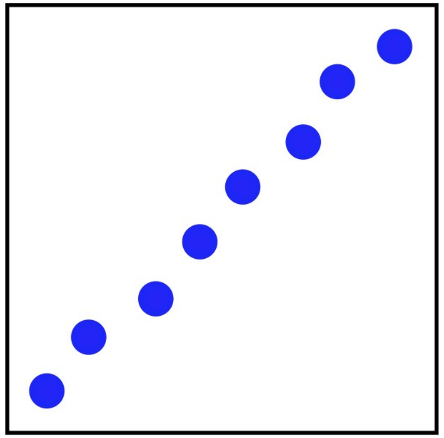


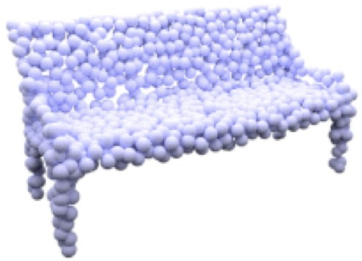
[Liao et al. CVPR'18]
[Chov et al. ECCV'16]

Image credit: Mescheder et al. CVPR'19

Pointclouds

- Discretization of 3D space into 3D points.
- Does not model connectivity/topology.
- Limited number of points.



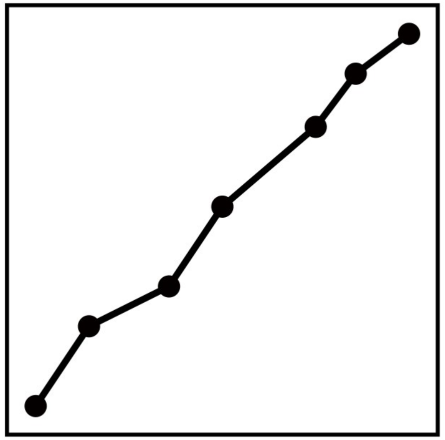


[Liao et al. CVPR'18]
[Chov et al. ECCV'16]

Image credit: Mescheder et al. CVPR'19

Meshes

- Discretization into vertices and faces.
- Limited number of vertices/granularity.
- Requires class specific template.
- Leads to self-intersections.

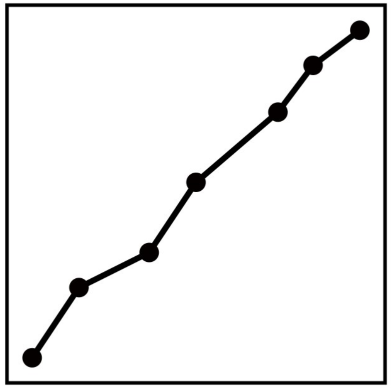


[Wang et al. ECCV'18]

Image credit: Mescheder et al. CVPR'19

Meshes

- Discretization into vertices and faces.
- Limited number of vertices/granularity.
- Requires class specific template.
- Leads to self-intersections.



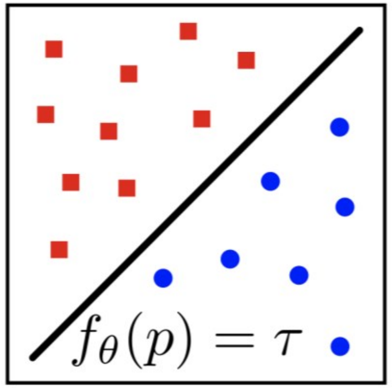
We have covered mesh-based human/clothing models.

[Wang et al. ECCV'18]

Image credit: Mescheder et al. CVPR'19

Implicit representation

- Implicit representation $\rightarrow$ No discretization.
- Arbitrary topology and resolution.
- Low memory footprint.
- Not restricted to specific class.



[Mescheder et al. CVPR'19]
[Chen et al. CVPR'19]
[Park et al. CVPR'19]

Image credit: Mescheder et al. CVPR'19

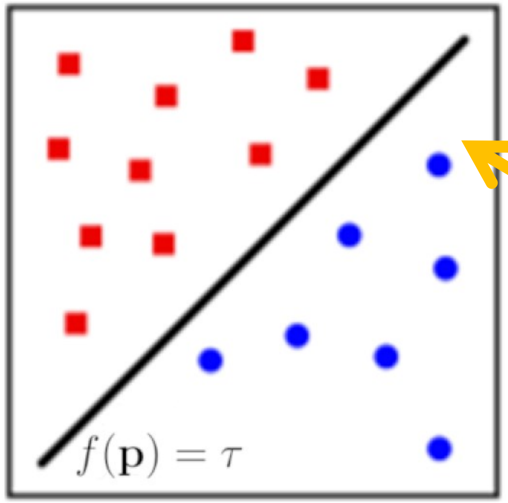
Surfaces as an Implicit Function

A function tells us whether a point is inside or outside an object

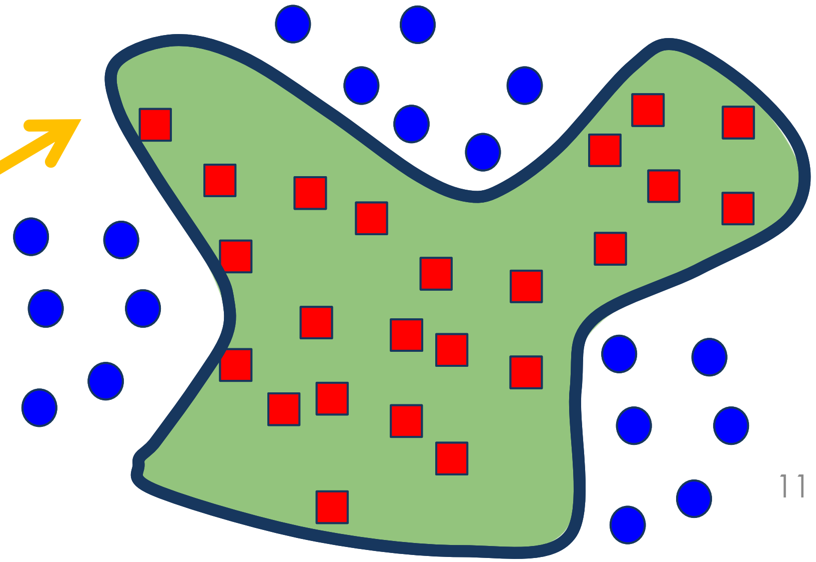
$$f(\mathbf{p}) = \begin{cases} 0, & \text{if} \quad \mathbf{p} \in \text{outside} \\ 1, & \text{if} \quad \mathbf{p} \in \text{inside} \end{cases}$$

$$\mathbf{p} = (x, y, z) \in \mathbb{R}^3$$

If the function is continuous, a levelset of it defines a surface $\mathcal{S}$



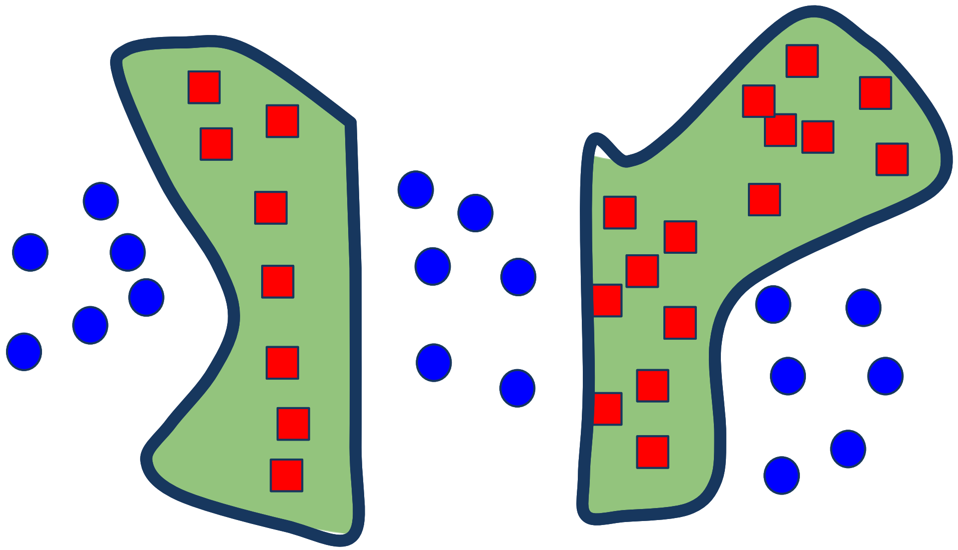
$$\mathcal{S} = \{\mathbf{p},\ f(\mathbf{p}) = \tau\}$$

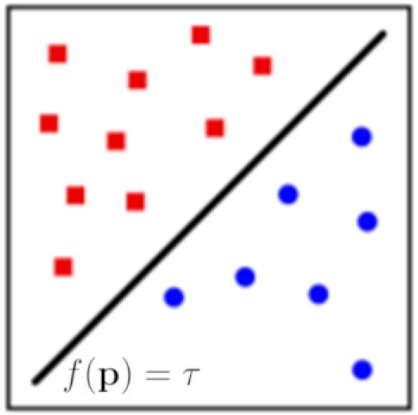


Surfaces as an Implicit Function

$$\mathbf{p} = (x, y, z) \in \mathbb{R}^3$$

$$f(\mathbf{p}) = \begin{cases} 0, & \text{if} \quad \mathbf{p} \in \text{outside} \\ 1, & \text{if} \quad \mathbf{p} \in \text{inside} \end{cases}$$





$$\mathcal{S} = \{\mathbf{p},\ f(\mathbf{p}) = \tau\}$$

✓ With implicit functions, Topology changes only require changing $f(\mathbf{p})$

✗ Mesh based representations would struggle

Neural Implicits for common objects

Neural Implicits for common objects

Work well for rigid objects: ✓ Continuous ✓ Multiple topologies

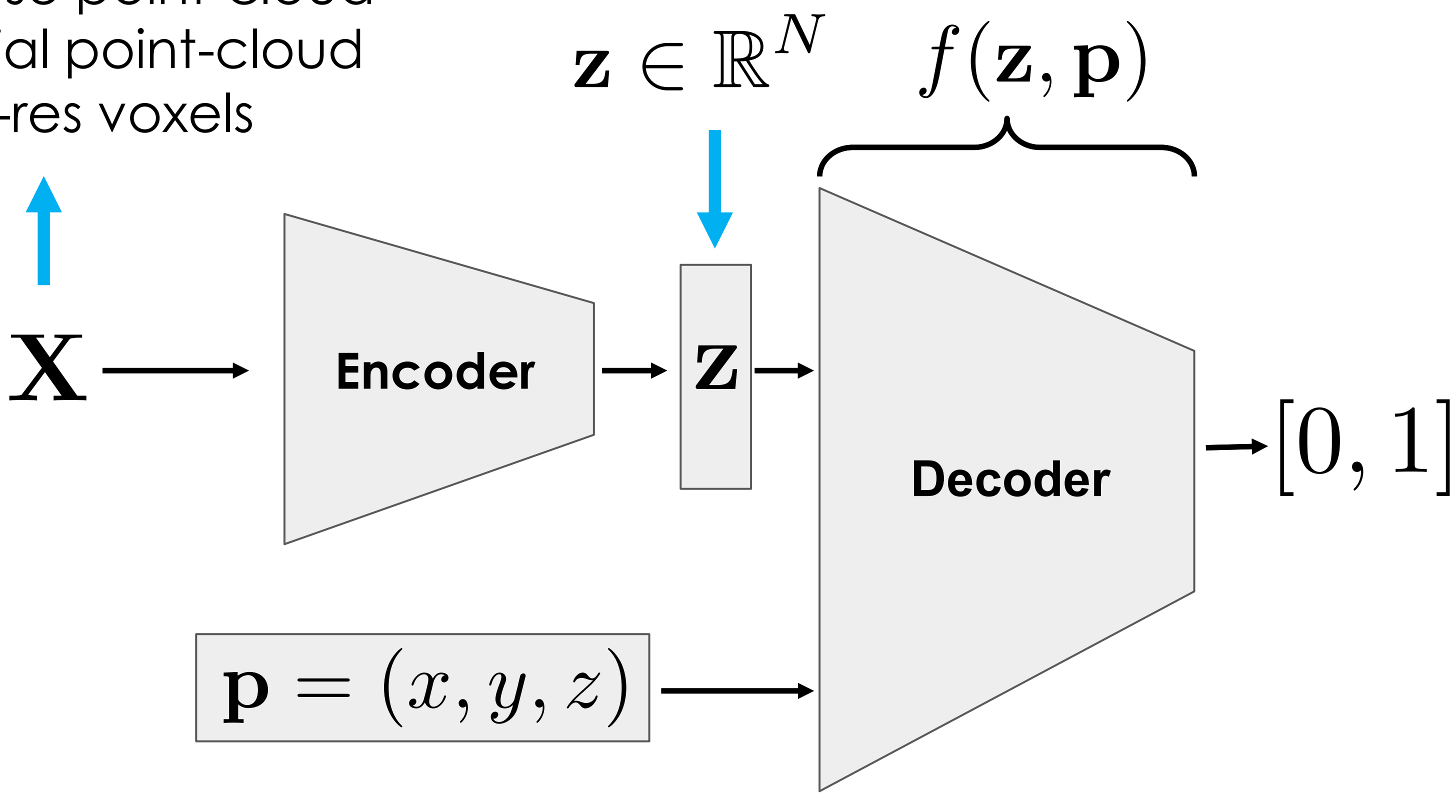
[Park et al. CVPR’19]

[Chen et al. CVPR’19]

[Mescheder et al. CVPR’19]

Previous Implicit Function Learning Architecture

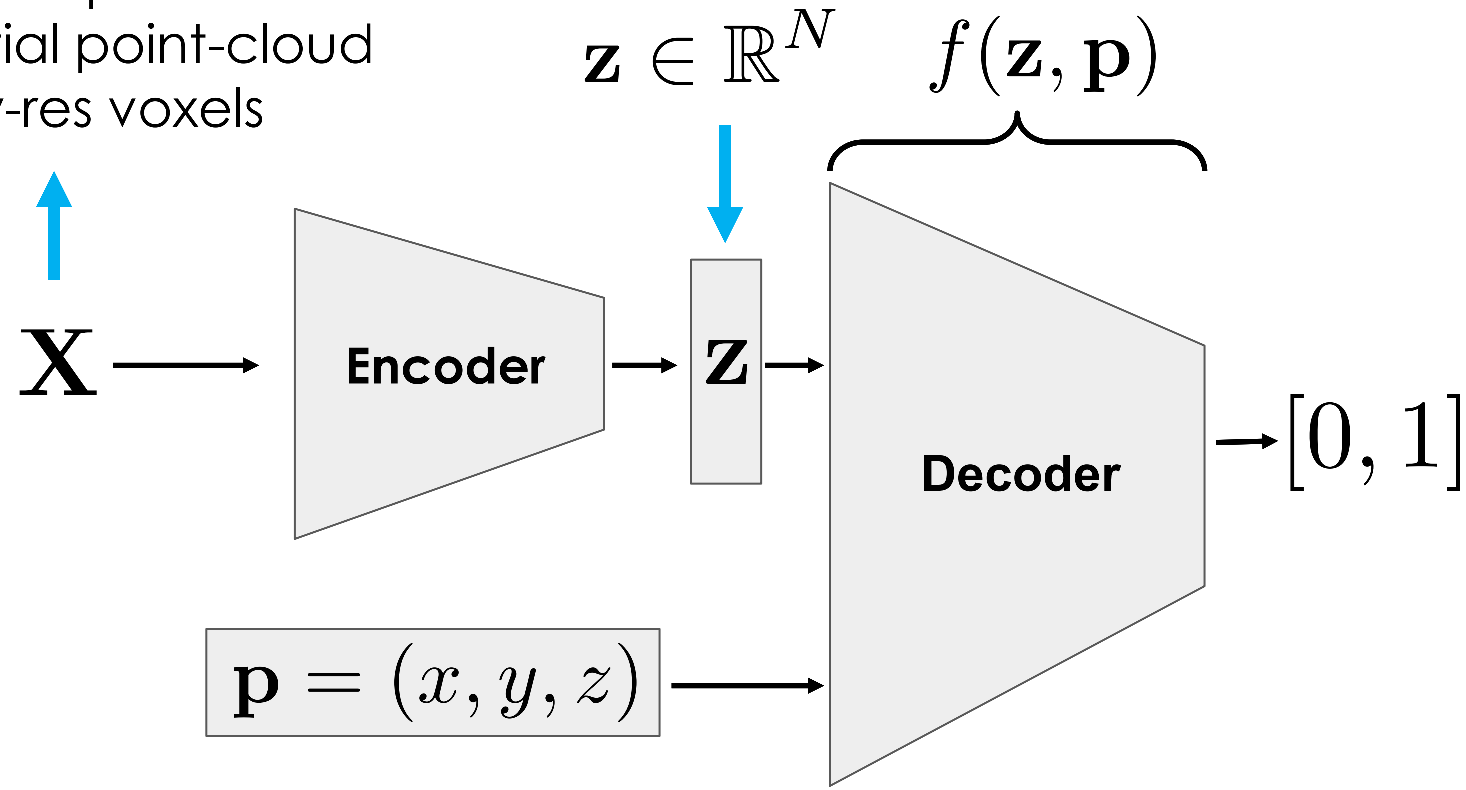
$\mathbf{X}$ is a shape observation:
- Sparse point-cloud
- Partial point-cloud
- Low-res voxels

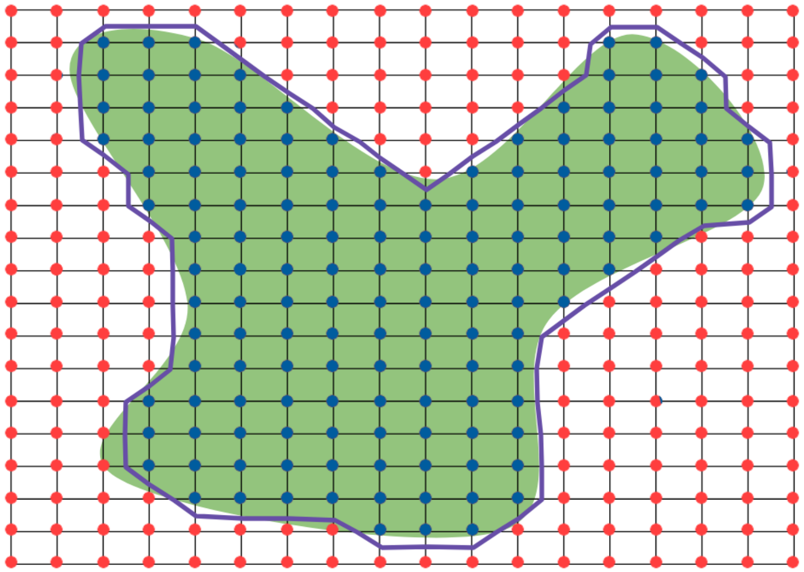


[Mescheder et al. CVPR'19
Chen et al. CVPR'19
Park et al. CVPR'19]

Previous Implicit Function Learning Architecture

$\mathbf{X}$ is a shape observation:
- Sparse point-cloud
- Partial point-cloud
- Low-res voxels



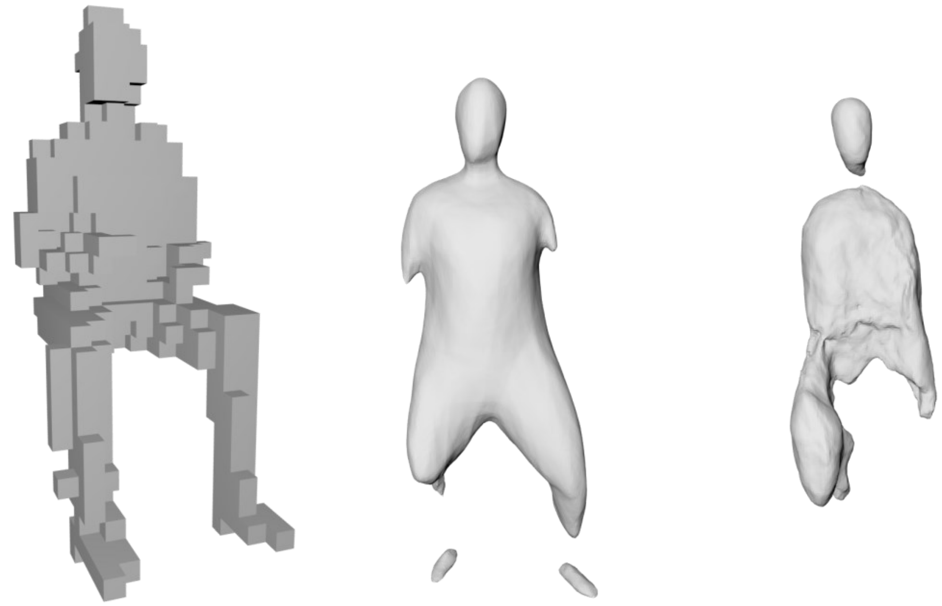


[Mescheder et al. CVPR'19
Chen et al. CVPR'19
Park et al. CVPR'19]

Problem with Previous Work

✗ Reconstruct Articulations

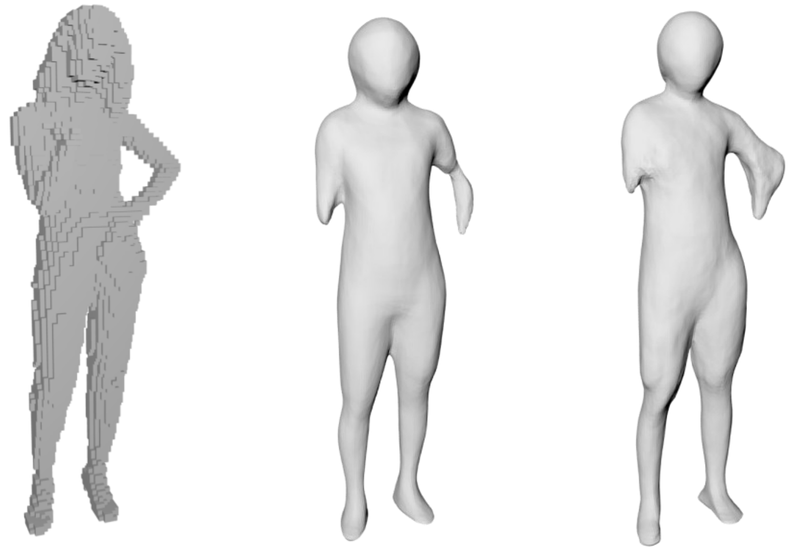
✗ Retain Details



X

[Chen et al. CVPR'19]

[Mescheder et al. CVPR'19]



X

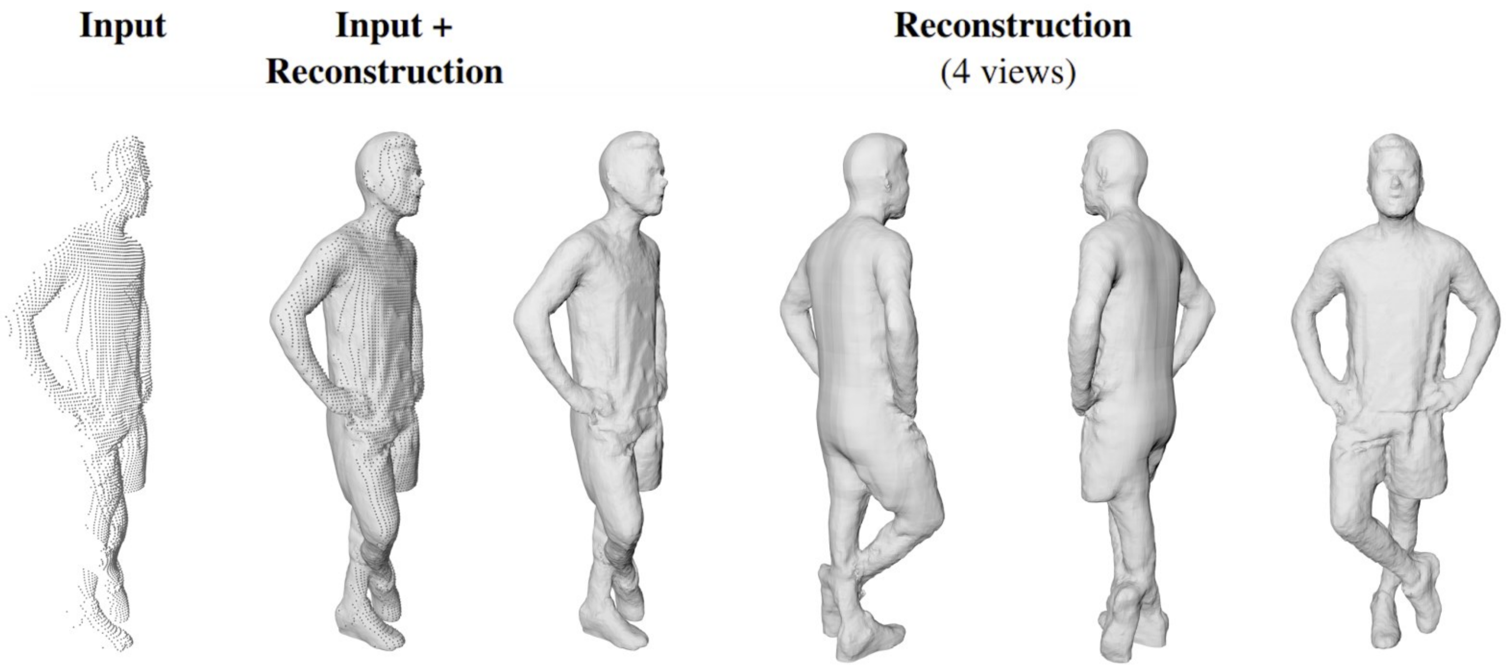
[Chen et al. CVPR'19]

[Mescheder et al. CVPR'19]

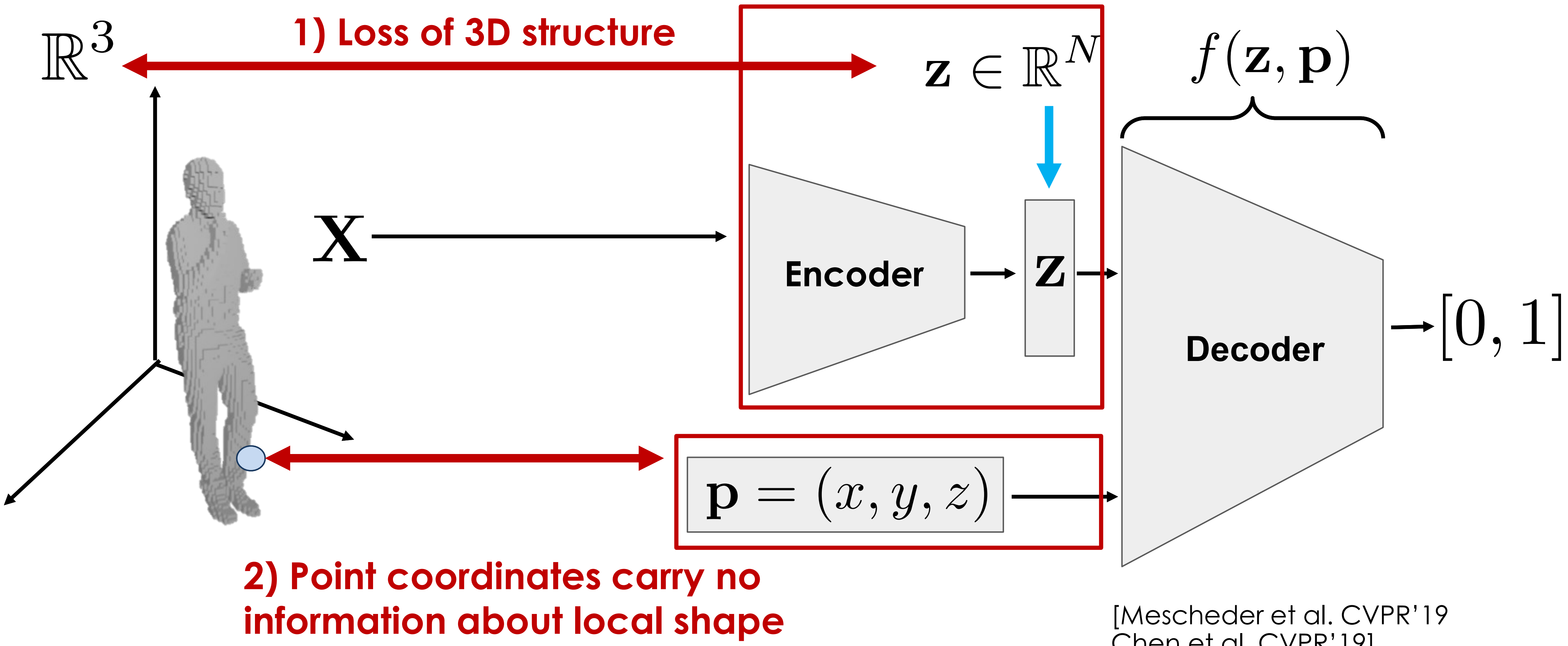
Implicit Functions in Feature Space for 3D Shape Reconstruction and Completion

Julian Chibane[1,2], Thiemo Alldieck[1,3], Gerard Pons-Moll[1]

CVPR 2020

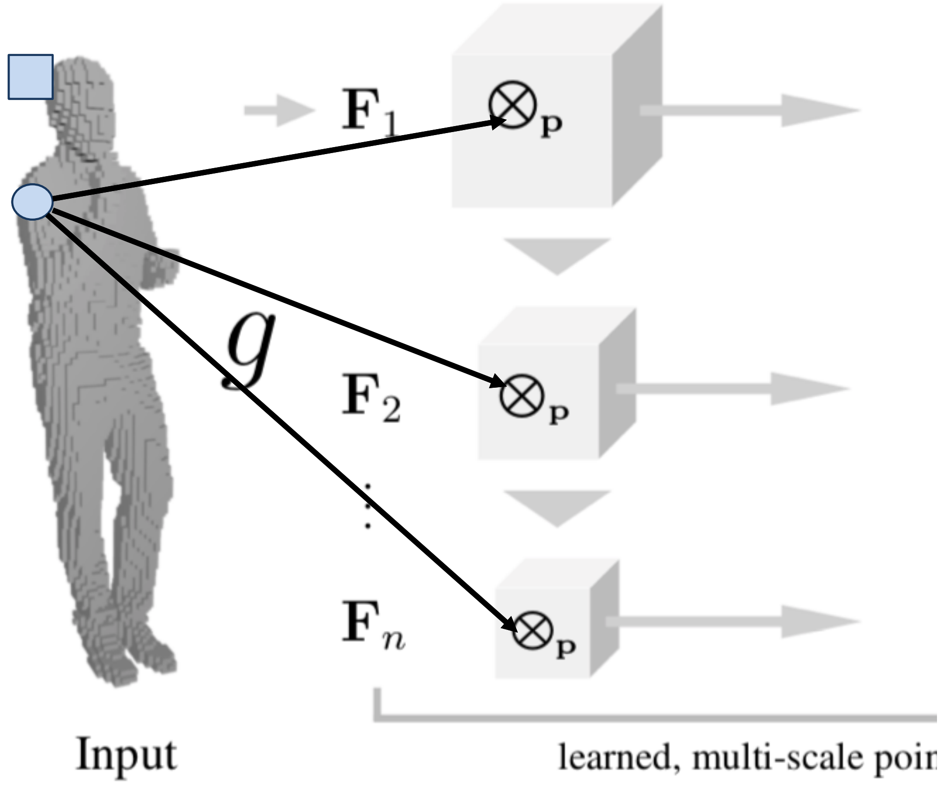


Problems with previous work



[Mescheder et al. CVPR'19
Chen et al. CVPR'19]

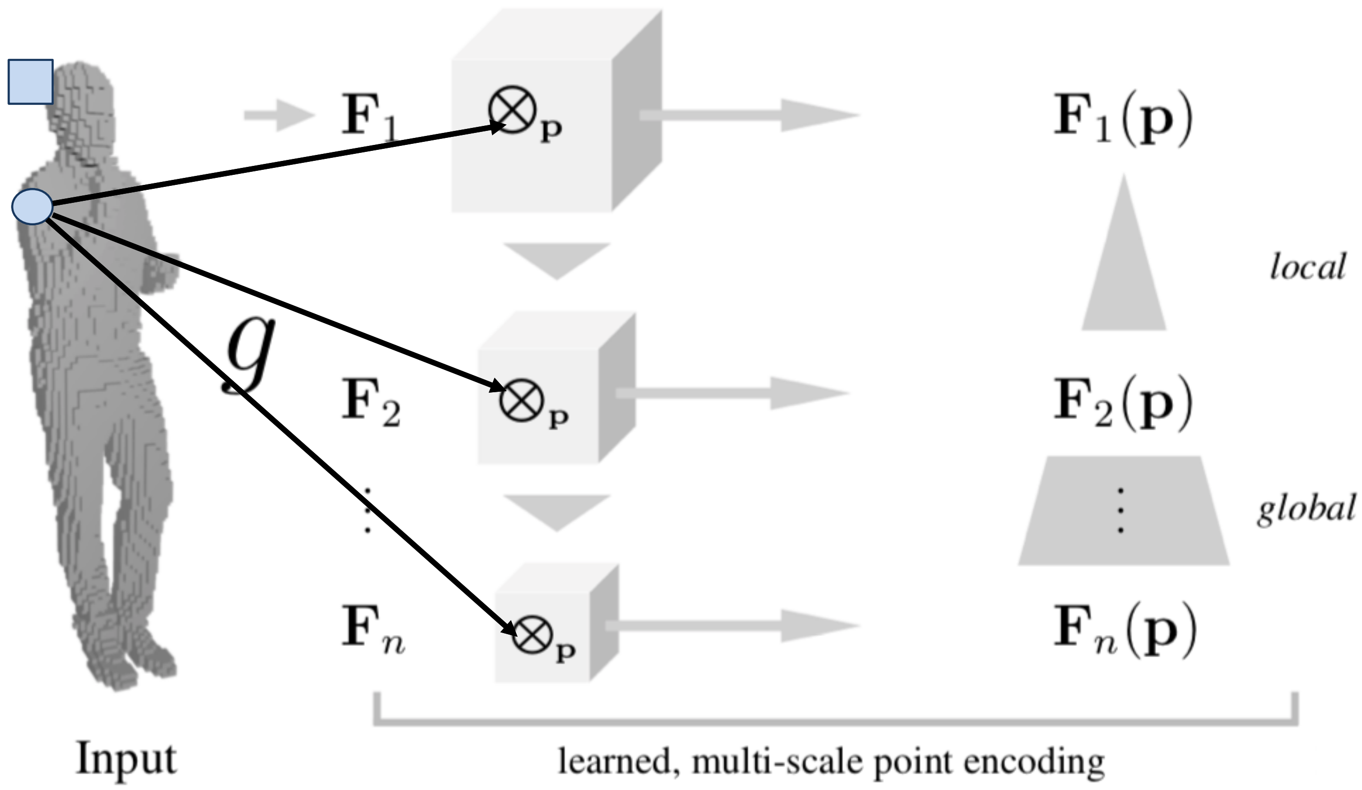
Implicit Feature Networks (IF-Nets)



3D Grid
$K \times K \times K$

Chibane et al. IF-Nets CVPR'20

Implicit Feature Networks (IF-Nets)



3D Grid

$K \times K \times K$

Chibane et al. IF-Nets CVPR'20

Representation of IF-Nets

Previous: $$f(\mathbf{z}, \mathbf{p}) \mapsto [0, 1]$$

$$\downarrow$$

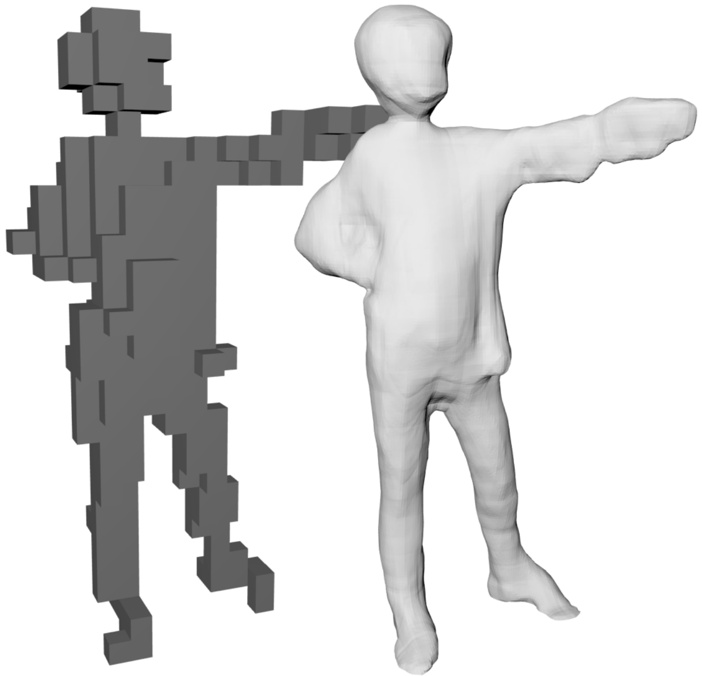
$$\mathbf{F}_1, \ldots, \mathbf{F}_n, \quad \mathbf{F} \in \mathcal{F}^{K \times K \times K}$$

Ours: $$f(\mathbf{F}_1(\mathbf{p}), \ldots, \mathbf{F}_n(\mathbf{p})) \mapsto [0, 1]$$

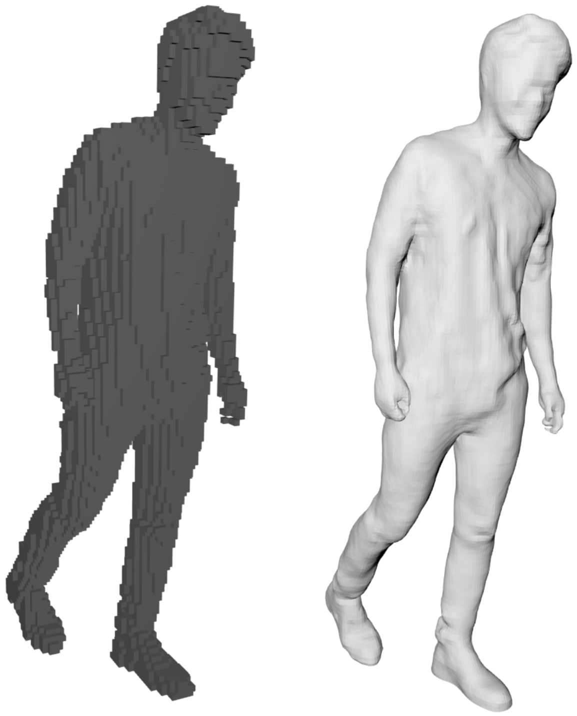
Chibane et al. IF-Nets CVPR'20

IF-Nets for 3D Shape Reconstruction and Completion

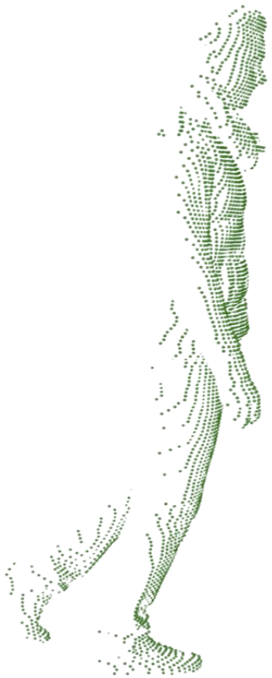
✓ Reconstruct Articulations



✓ Retain Details



✓ Complete Shape



IF-Nets for Texture completion

ECCV SHARP CHALLENGE

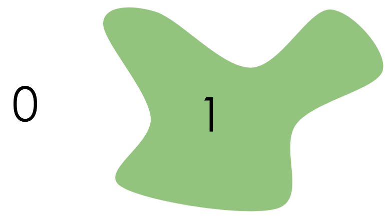
Input Prediction GT

Input Prediction GT

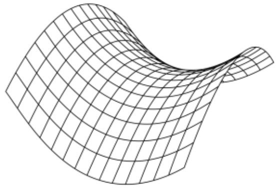
[Chibane and Pons-Moll, IF-Nets for texture. SHARP 2020
Chibane et al. IF-Nets CVPR'20]

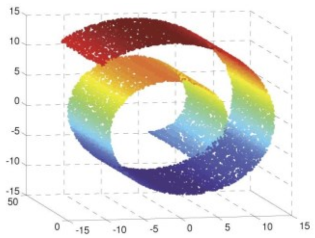
- Surfaces that do not divide the space in two regions can not be represented.

- We need a different **output representation.**

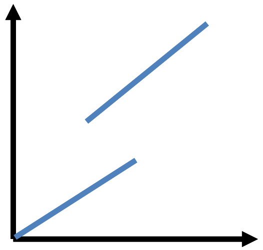


Only water-tight surfaces





Open surfaces and manifolds



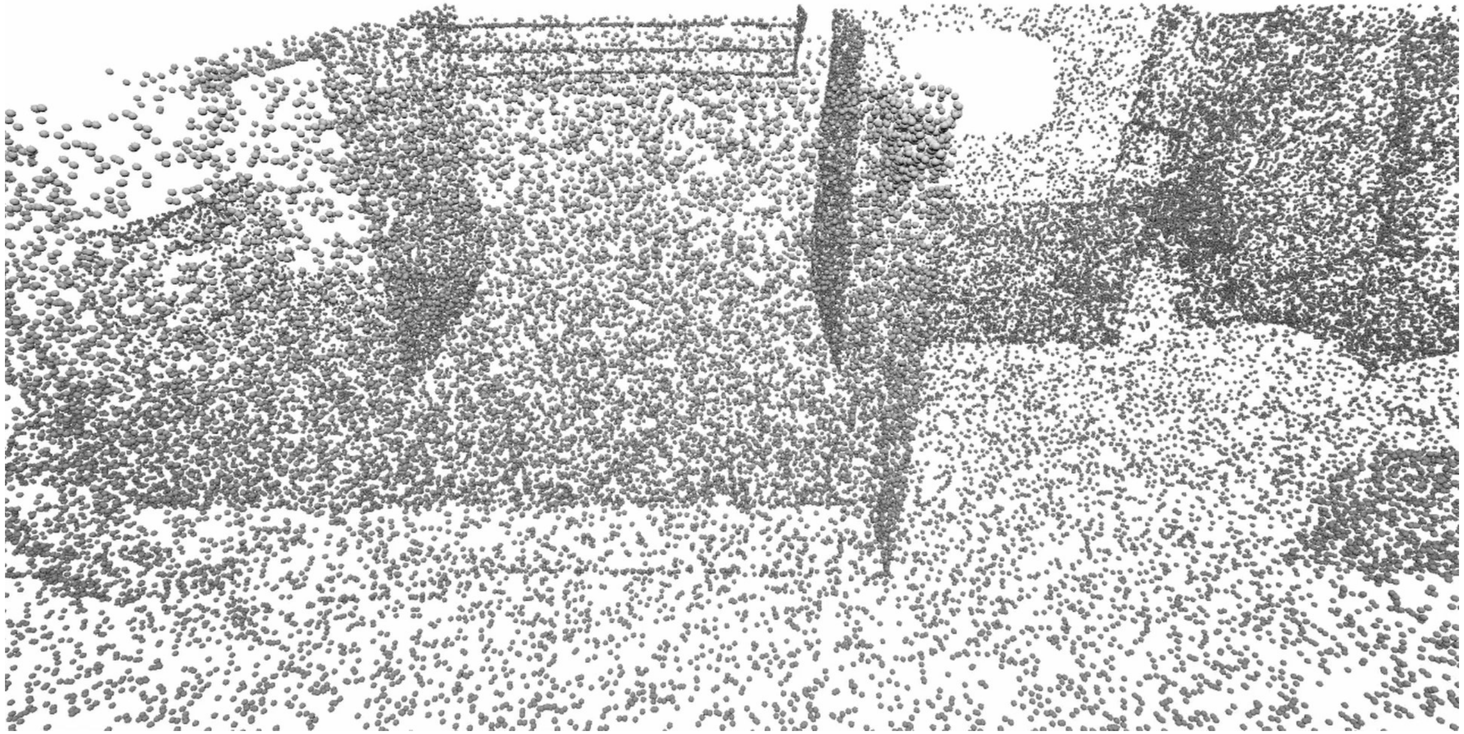
Functions

Complex shapes

Neural Unsigned Distance Fields for Implicit Function Learning

Julian Chibane, Aymen Mir, Gerard Pons-Moll

NeurIPS 2020



Chibane et al. **NDF**, NeurIPS 2020

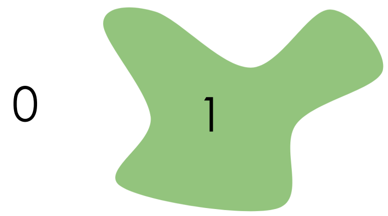
Our Solution

Change the output representation

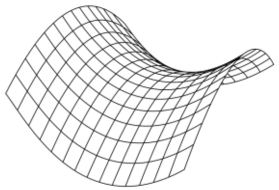
$$f(\mathbf{F}_1(\mathbf{p}), \ldots, \mathbf{F}_n(\mathbf{p})) \mapsto [0, 1] \qquad f(\mathbf{F}_1(\mathbf{p}), \ldots, \mathbf{F}_n(\mathbf{p})) \mapsto \mathbb{R}^+$$

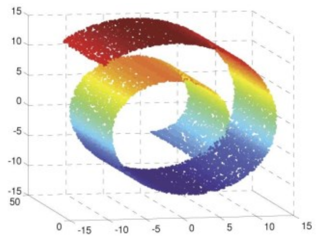
Unsigned distance:

$$\boxed{f(\mathbf{p}) = \min_{\mathbf{q} \in \mathcal{S}} \|\mathbf{p} - \mathbf{q}\|}$$

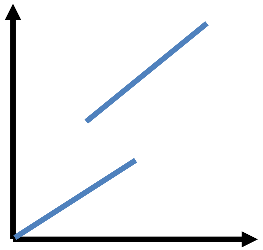


Only water-tight surfaces





Open surfaces and manifolds

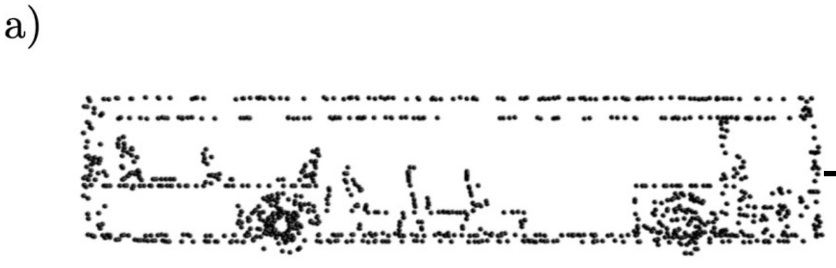


Functions

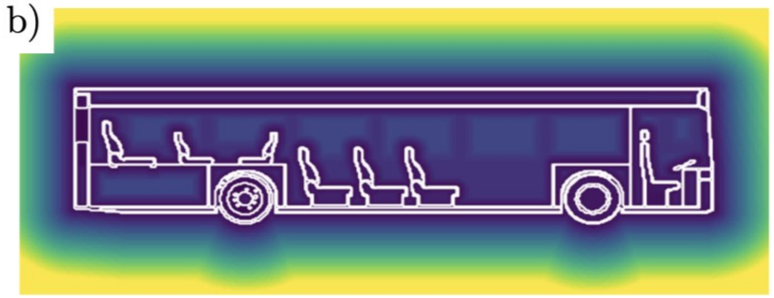
Complex shapes

Chibane et al. **NDF**, NeurIPS 2020

Neural Distance Fields

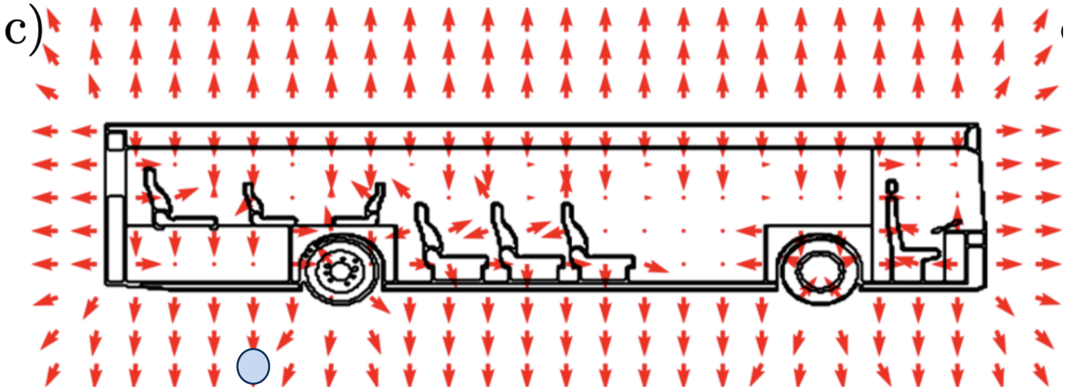


$$f(\mathbf{p}) = \min_{\mathbf{q} \in \mathcal{S}} \|\mathbf{p} - \mathbf{q}\|$$



$\mathbf{X}$

$$\mathcal{S} = \{\mathbf{p} \in \mathbb{R}^d \mid f(\mathbf{p}) = 0\}$$



$$\mathbf{q} = \mathbf{p} - f(\mathbf{p}) \nabla_{\mathbf{p}} f(\mathbf{p})$$

Chibane et al. **NDF**, NeurIPS 2020

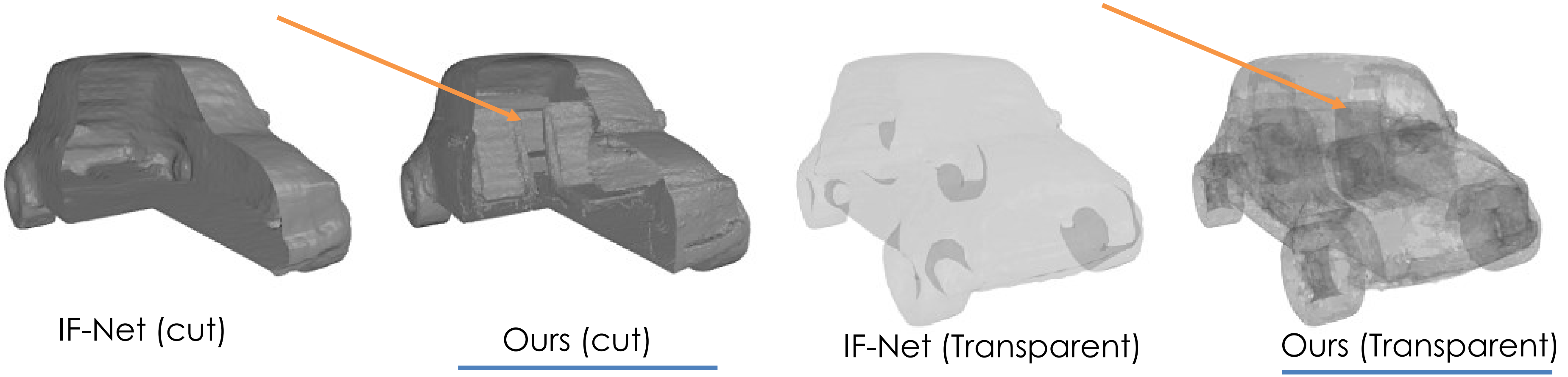
Neural processing of arbitrary surfaces

Next, we illustrate the capabilities of NDF to neurally process arbitrary surfaces, not representable by prior learned implicit work:

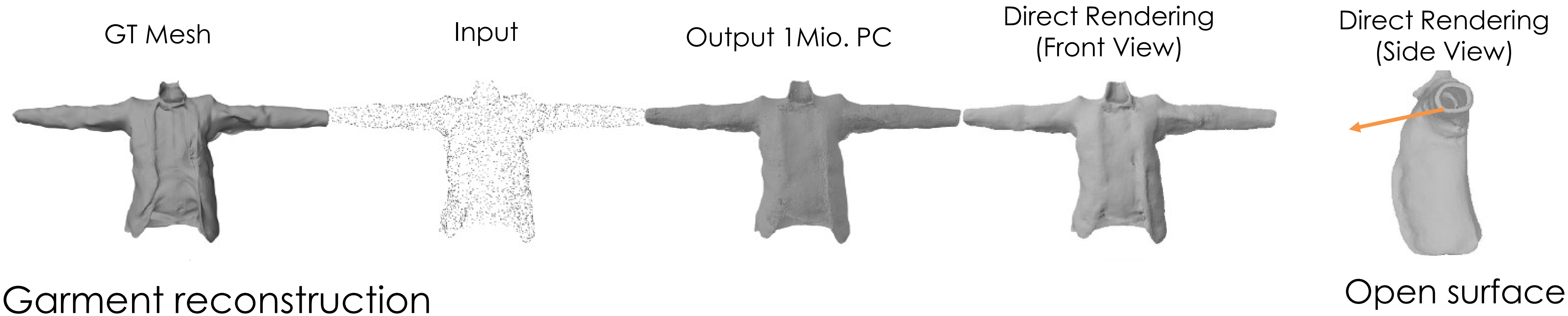
- **Mathematical Functions and Manifolds** – We train a single NDF on a dataset consisting of 1000 functions per type: linear function, parabola, sinusoids and spirals.
- **Garments** – Open Surfaces, without thickness. Training on ~300 garments of five types from [Bhatnagar et al. ICCV‘19].
- **Scenes** – Open surfaces with holes and no thickness. Training on 34 real world scenes captured by RGBD Sensors from [Xia et al. CVPR‘18].

Chibane et al. **NDF**, NeurIPS 2020

NDF results



1) Comparison with IF-Nets



2) Garment reconstruction

Chibane et al. **NDF**, NeurIPS 2020

Direct Rendering of NDF

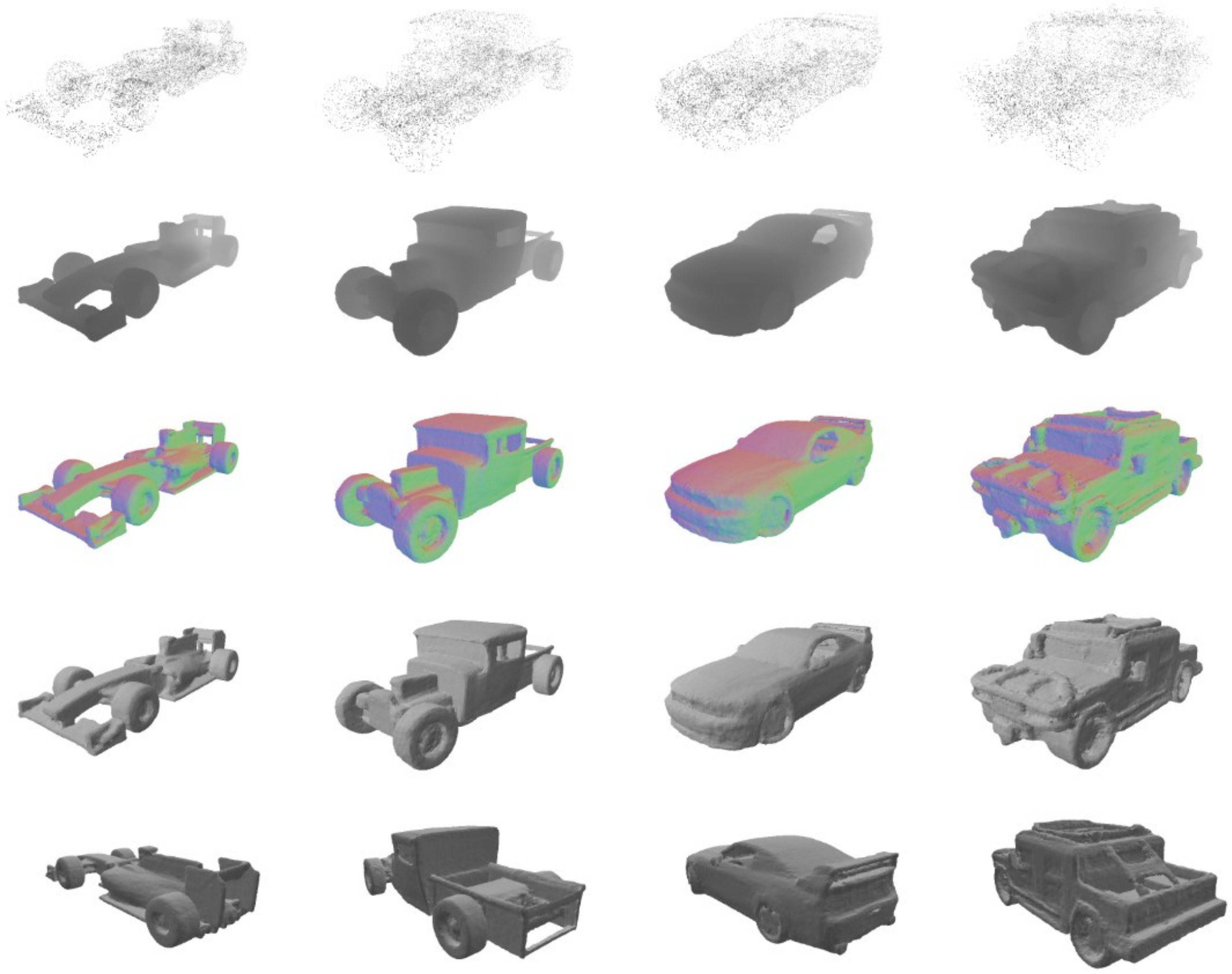
Input

Depth

Normals

Shaded

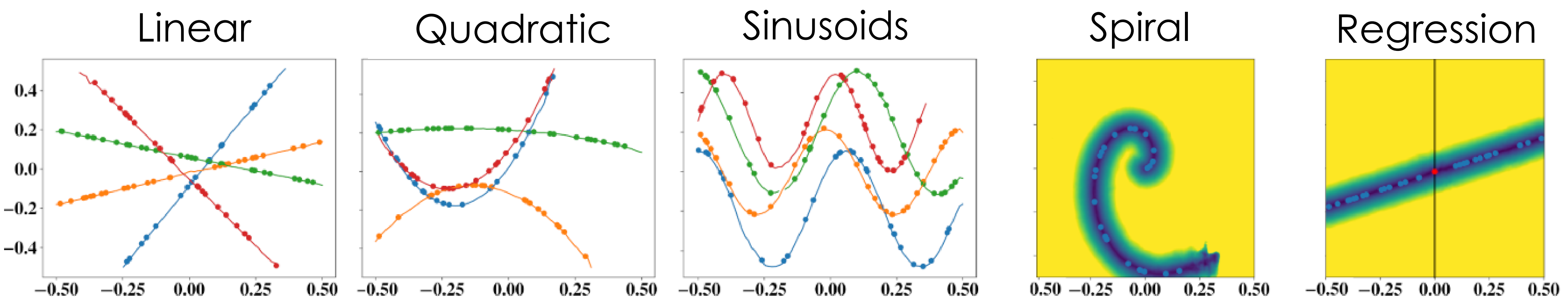
Back



Sphere marching

Chibane et al. **NDF**, NeurIPS 2020

Representation and Regression of Functions



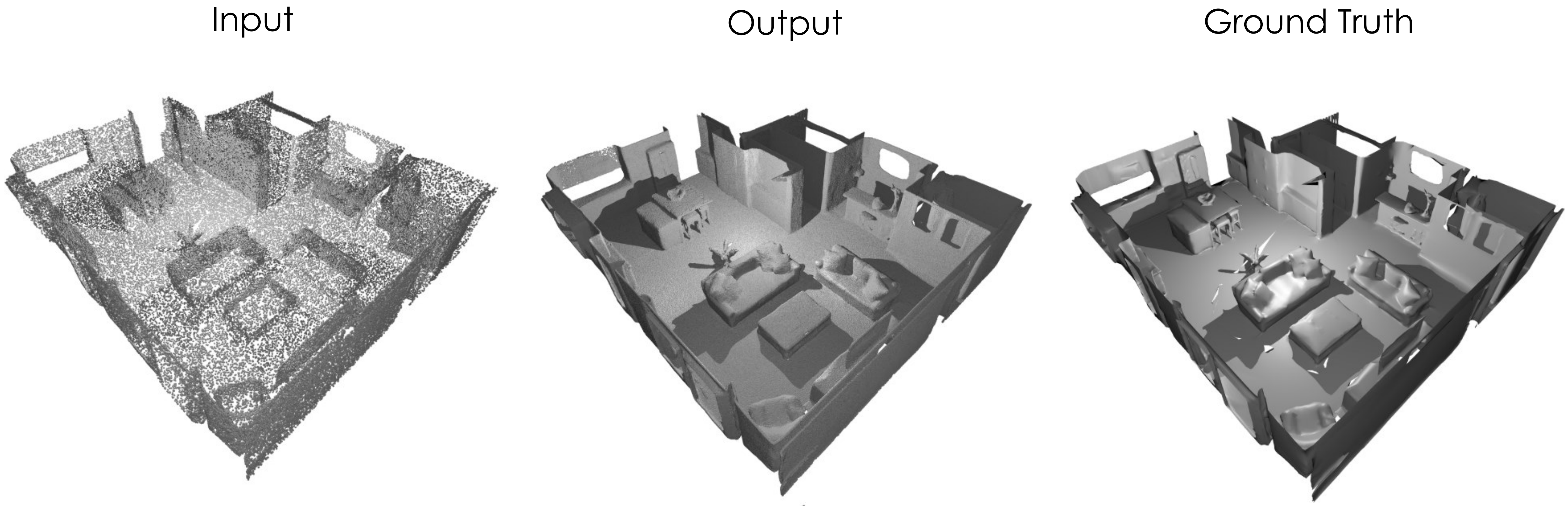
$$y = h(x_1, \ldots, x_n) \longrightarrow f(x_1, \ldots, x_n, y) = 0$$

$$\mathbf{p}(\lambda) = (x_1, \ldots, x_n, 0) + \lambda(0, \ldots, 0, 1) \longrightarrow f(\mathbf{p}(\lambda)) = 0$$

Classical regression using NDFs and an adapted sphere tracing (ray tracing method)

Chibane et al. **NDF**, NeurIPS 2020

Representation and Completion of Scenes



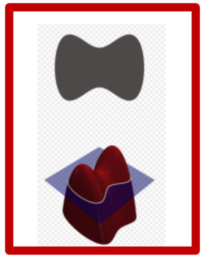
Chibane et al. **NDF**, NeurIPS 2020

Meshes vs Implicits

1) Parametric Meshes

[Alldieck et al. CVPR'18
Bhatnagar et al. ICCV'19, ECCV'20
Tiwari et al. ECCV'20]

2) Implicit Functions



General objects and humans

[Chibane et al. CVPR'20
Chibane et al. NeurIPS'20]

	Control /Meaning	Topology	Details
1)	✓	✗	✗
2)	✗	✓	✓

2) Compatible with learning

PART1: Neural Implicits for 3D Shapes

PART2: Neural Implicits for Humans

PART3: Neural Implicits – Generative Models

PART4: Point-based Clothing Models

Human and Clothing Models

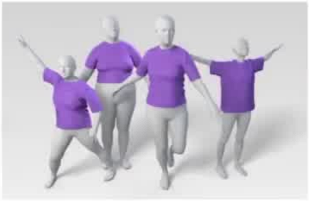
Prior works $\longrightarrow$ mesh based

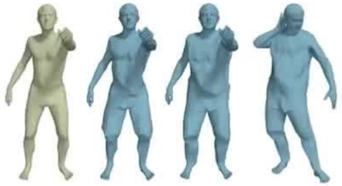

Guan et al., 2012

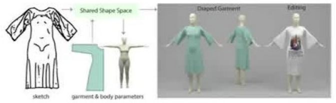

Danerek et al., 2016

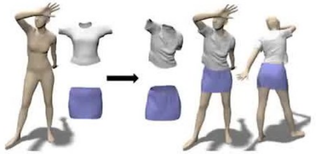

Lähner et al., 2018

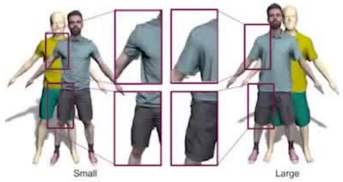

Gundogdu et al., 2019


Santesteban et al., 2019


Ma et al., 2020


Wang et al., 2018


Patel et al., 2020



Tiwari et al., 2020

Human and Clothing Models

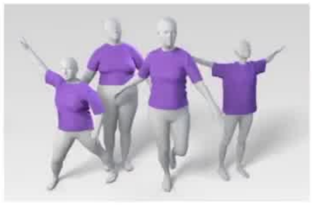
Prior works ⟶ mesh based

Guan et al., 2012

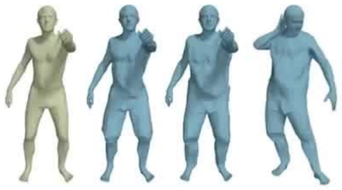
Danerek et al., 2016

Lähner et al., 2018

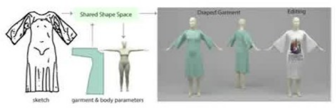
Gundogdu et al., 2019



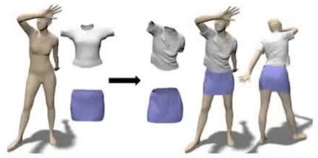
Santesteban et al., 2019



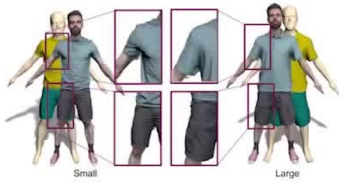
Ma et al., 2020



Wang et al., 2018



Patel et al., 2020



Tiwari et al., 2020

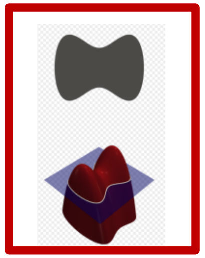
- ✗ **Fixed** topology
- ✗ Topology has to be **manually** predefined
- ✗ **Limited resolution**

Meshes vs Implicits

1) Parametric Meshes

[Alldieck et al. CVPR'18
Bhatnagar et al. ICCV'19, ECCV'20
Tiwari et al. ECCV'20]

2) Implicit Functions



General objects and humans

[Chibane et al. CVPR'20
Chibane et al. NeurIPS'20]

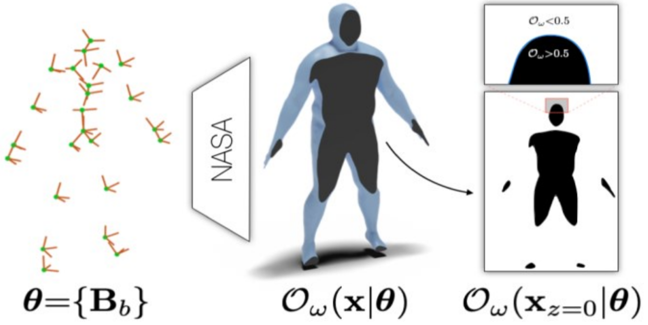
	Control /Meaning	Topology	Details
1)	✓	✗	✗
2)	✗	✓	✓

2) Compatible with learning

Human and clothing model using Neural Implicits

Corona et al. CVPR' 21

Tiwari et al. ICCV'21



Deng et al. ECCV'20

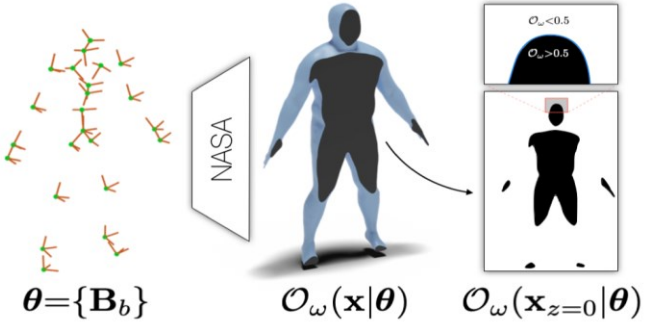
Saito et al. CVPR'21

- High fidelity
- Flexible topology
- Pose/Shape/Style controllable
- Learned directly from scans

Human and clothing model using Neural Implicits

Corona et al. CVPR' 21

Tiwari et al. ICCV'21



Deng et al. ECCV'20

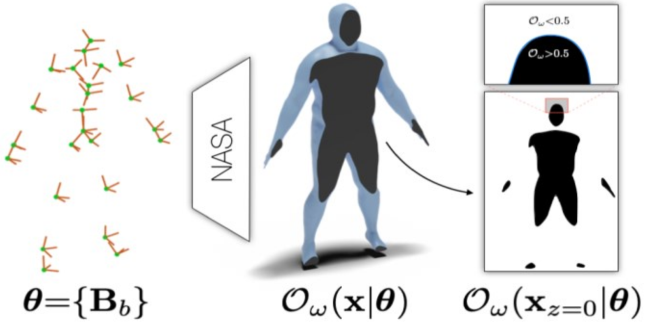
Saito et al. CVPR'21

- <u>High fidelity</u>
- <u>Flexible topology</u>
- Pose/Shape/Style controllable
- Learned directly from scans

Human and clothing model using Neural Implicits

Corona et al. CVPR' 21

Tiwari et al. ICCV'21



Deng et al. ECCV'20

Saito et al. CVPR'21

- High fidelity
- Flexible topology
- <u>Pose/Shape/Style controllable</u>
- <u>Learned directly from scans</u>

Controllable Neural Implicits for Human

Vertex based human model: SMPL

$$M(\boldsymbol{\theta}, \boldsymbol{\beta}) : \boldsymbol{\theta} \times \boldsymbol{\beta} \rightarrow \mathbf{V} \in \mathbb{R}^3$$

Neural Implicit for common objects:

$$\boldsymbol{f}(\mathbf{p}, z) : \mathbb{R}^3 \times \mathbb{R}^d \rightarrow \boldsymbol{d} \in \mathbb{R}$$

$$\mathcal{S} = \{\mathbf{p},\ f(\mathbf{p}) = \tau\}$$

Controllable Neural Implicits for Human

Vertex based human model: SMPL

$$M(\boldsymbol{\theta}, \boldsymbol{\beta}) : \boldsymbol{\theta} \times \boldsymbol{\beta} \rightarrow \mathbf{V} \in \mathbb{R}^{\mathbf{3}}$$

Controllable Neural Implicit humans:

$$f(\mathbf{p}, \boldsymbol{\theta}, \boldsymbol{\beta}) : \mathbb{R}^3 \times \boldsymbol{\theta} \times \boldsymbol{\beta} \rightarrow d \in \mathbb{R}$$

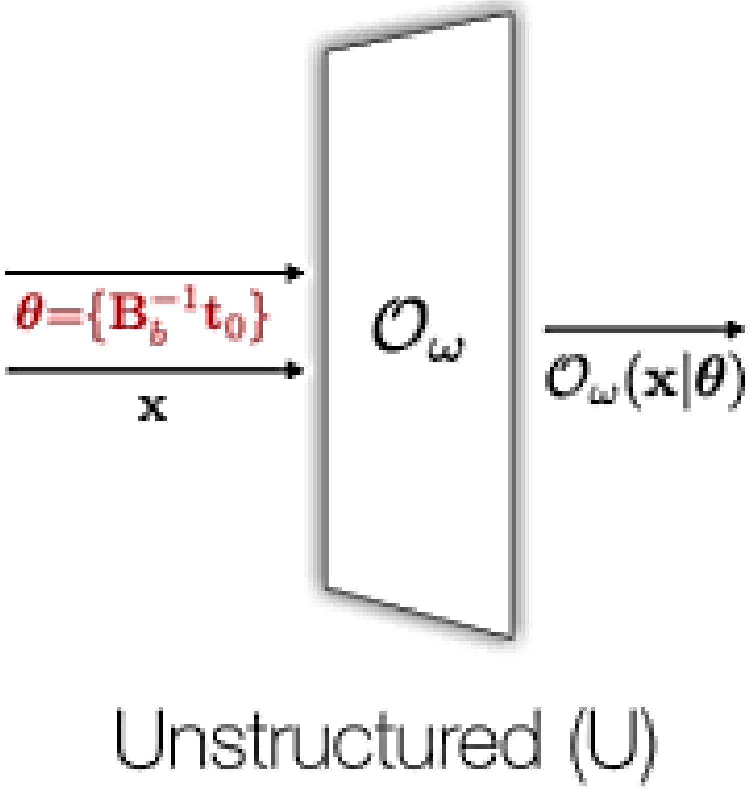
$$\mathcal{S} = \{\mathbf{p},\ f(\mathbf{p}) = \tau\}$$

Learning pose-conditioned occupancy

- Naïve solution(Unstructured)

$$\mathcal{O}(\mathbf{p}|\boldsymbol{\theta}) = f_w(\mathbf{p}, \{\mathbf{B}_b^{-1}\mathbf{t}_0\})$$

$\mathbf{B}_b$ Bone transformations
$\mathbf{t}_0$ Root translation
$\mathbf{p}$ Query point

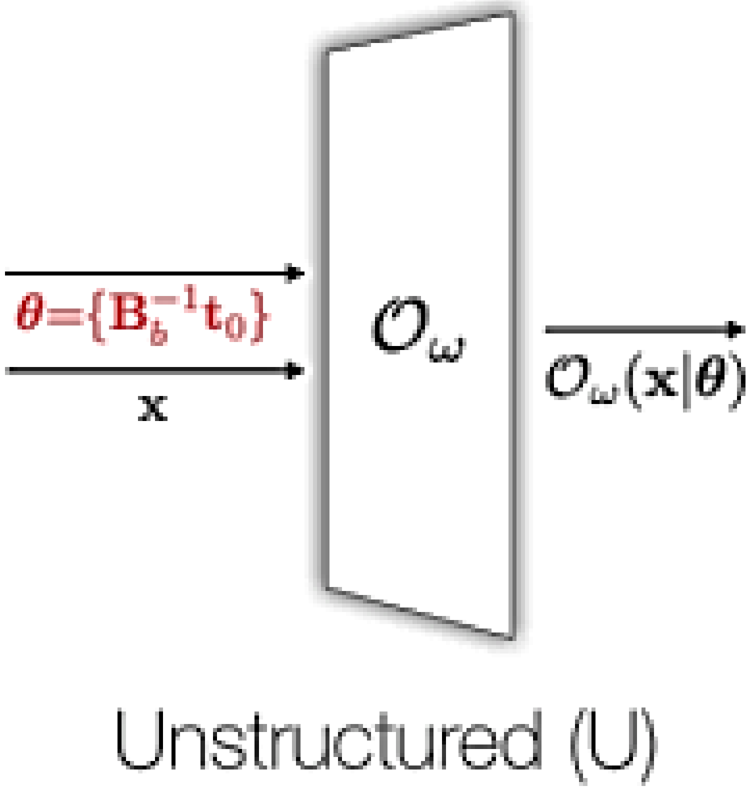


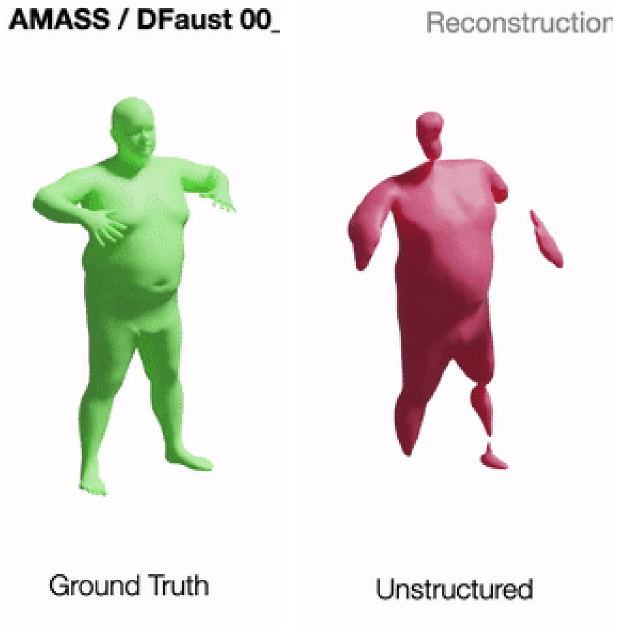
Deng et al. **NASA**, ECCV 2020

Learning pose-conditioned occupancy

- Naïve solution(Unstructured)

$$\mathcal{O}(\mathbf{p}|\boldsymbol{\theta}) = f_w(\mathbf{p}, \{\mathbf{B}_b^{-1}\mathbf{t}_0\})$$





Deng et al. **NASA**, ECCV 2020

Incorporating prior knowledge about human models

Vertex based human model: SMPL

$$M(\boldsymbol{\theta}, \boldsymbol{\beta}) : \boldsymbol{\theta} \times \boldsymbol{\beta} \rightarrow \mathbf{V} \in \mathbb{R}^3$$

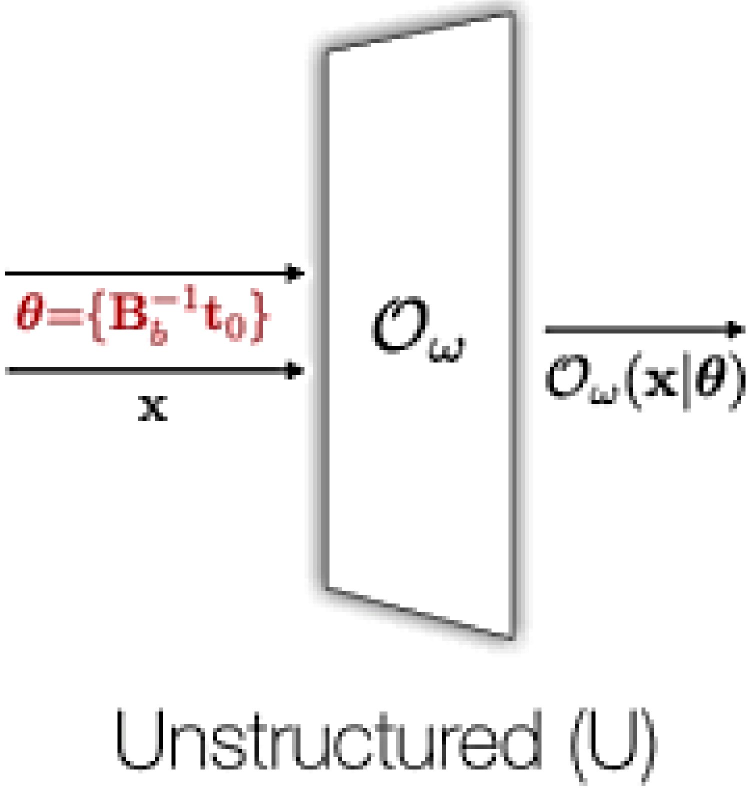
$$T(\boldsymbol{\theta}, \boldsymbol{\beta}) = \mathbf{T} + B_s(\boldsymbol{\beta}) + B_p(\boldsymbol{\theta})$$

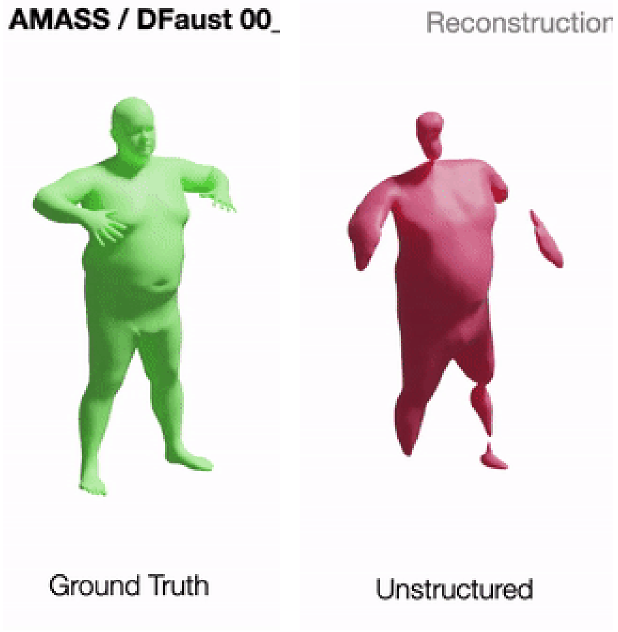
Learning pose/shape conditioned neural implicits using part composition

NASA

- Naïve solution(Unstructured)

$$\mathcal{O}(\mathbf{p}|\boldsymbol{\theta}) = f_w(\mathbf{p}, \{\mathbf{B}_b^{-1}\mathbf{t}_0\})$$



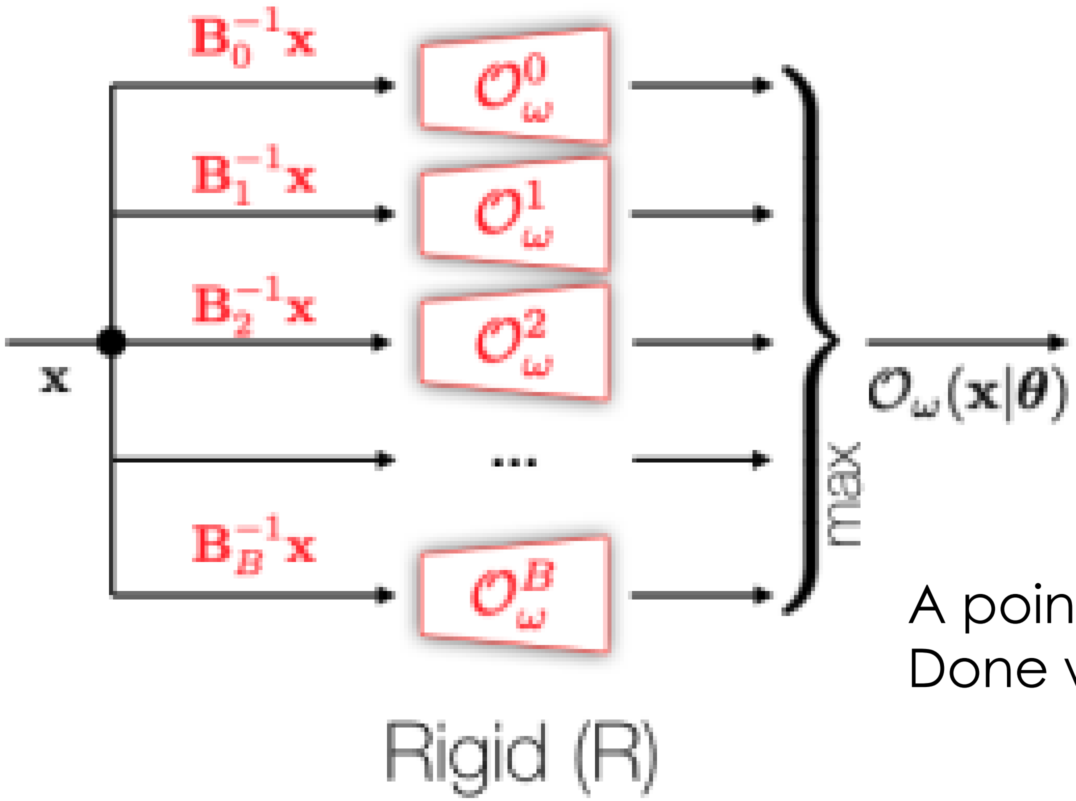


Deng et al. **NASA**, ECCV 2020

NASA

- Piecewise-rigid model

$$\mathcal{O}(\mathbf{p}|\boldsymbol{\theta}) = \max\{f_w(\mathbf{B}_b^{-1}\mathbf{p})\}$$



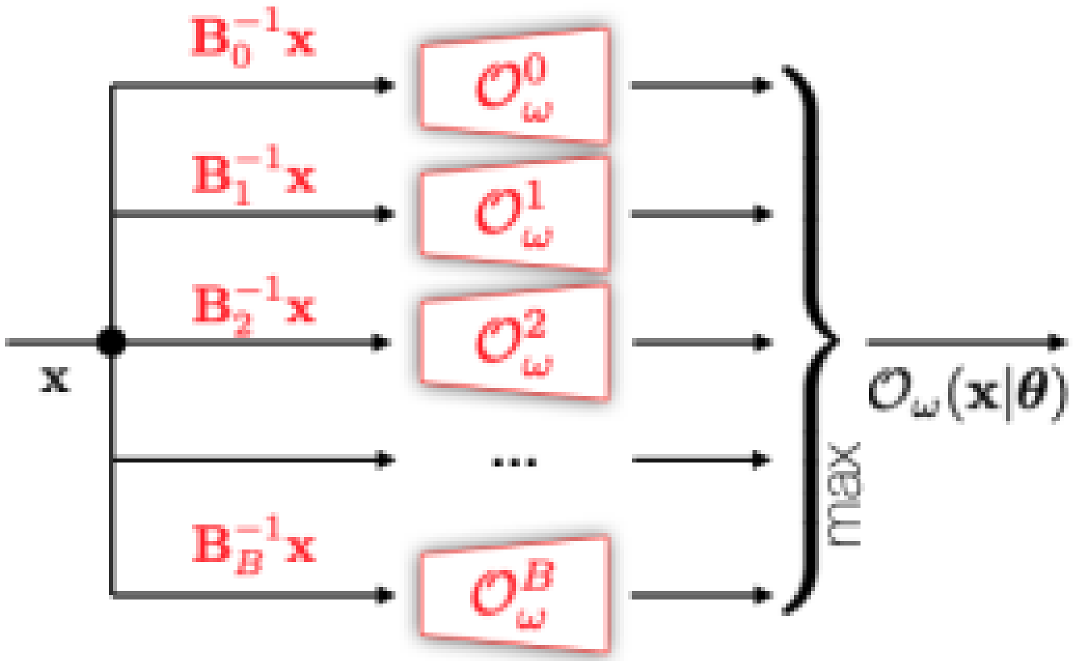
A point is occupied if it is occupied by *any* of the parts.
Done with max operator

Deng et al. **NASA**, ECCV 2020

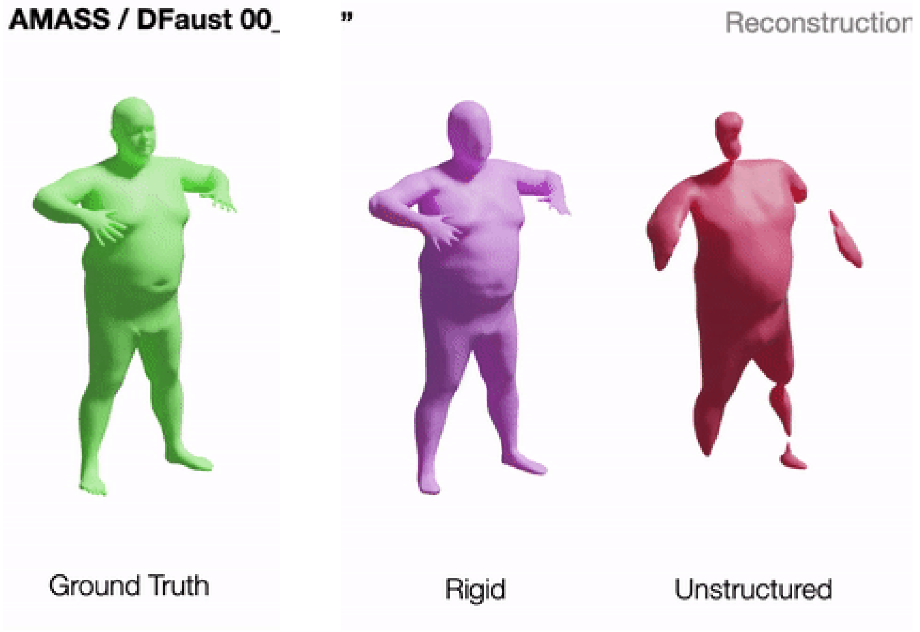
NASA

- Piecewise-rigid model

$$\mathcal{O}(\mathbf{p}|\boldsymbol{\theta}) = \max\{f_w(\mathbf{B}_b^{-1}\mathbf{p})\}$$



Rigid (R)

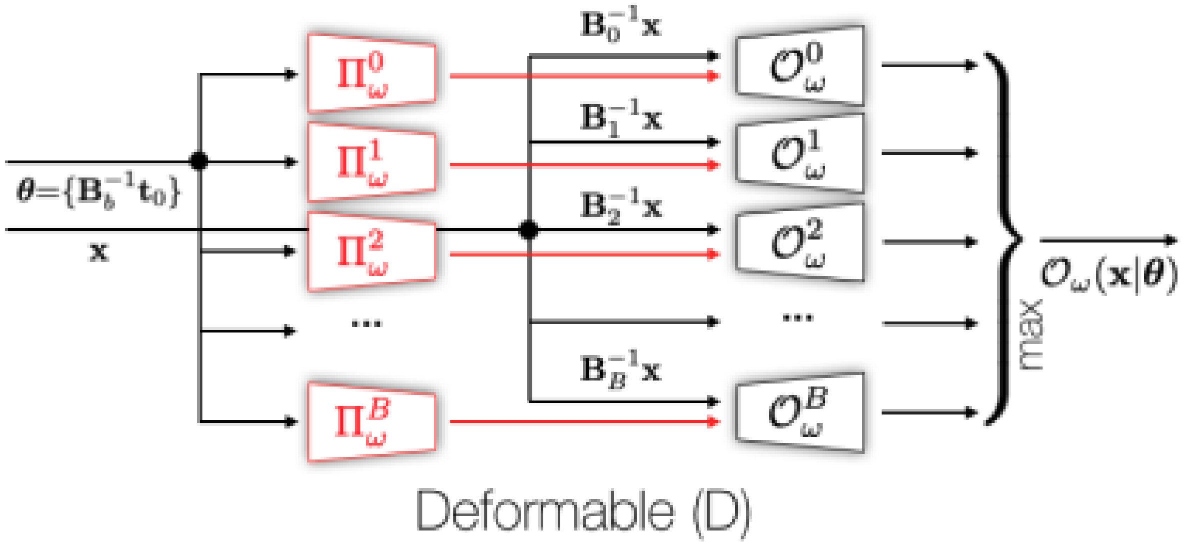


Deng et al. **NASA**, ECCV 2020

NASA

- Piecewise-deformable model

$$\mathcal{O}(\mathbf{p}|\boldsymbol{\theta}) = \max\{f_w(\mathbf{B}_b^{-1}\mathbf{p}|\boldsymbol{\theta})\}$$

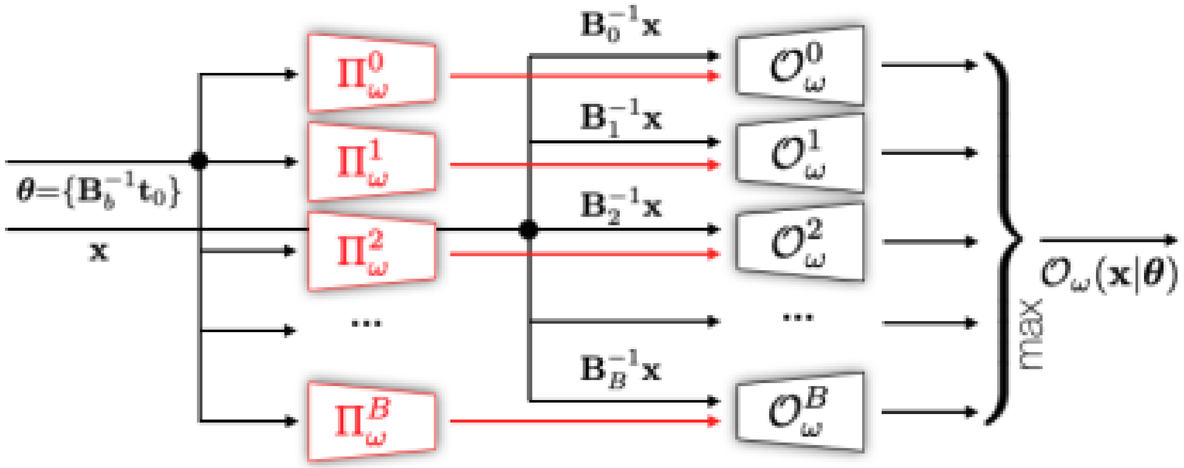


Deng et al. **NASA**, ECCV 2020

NASA results

- Piecewise-deformable model

$$\mathcal{O}(\mathbf{p}|\boldsymbol{\theta}) = \max\{f_w(\mathbf{B}_b^{-1}\mathbf{p}|\boldsymbol{\theta})\}$$



Deformable (D)

Deng et al. **NASA**, ECCV 2020

NASA:Neural Articulated Shape Approximation

- Limitations of NASA:
  - Part-based artefacts
    - No information about neighbouring body parts
  - Limited pose generalization
    - Low-dimensional pose encoding does not fully remove long-range spurious correlations

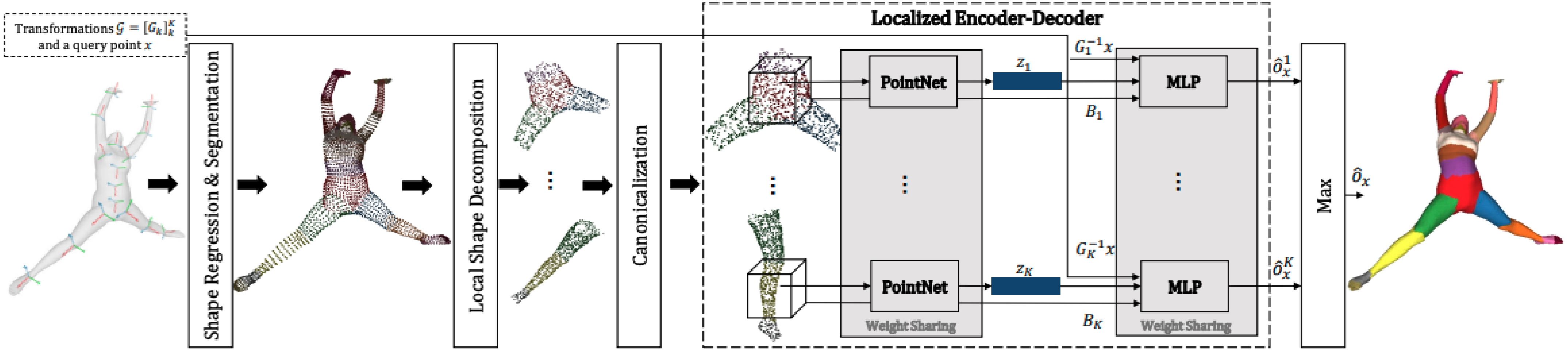
Deng et al. **NASA**, ECCV 2020

COAP: **Co**mpositional **A**rticulated Occupancy of **P**eople

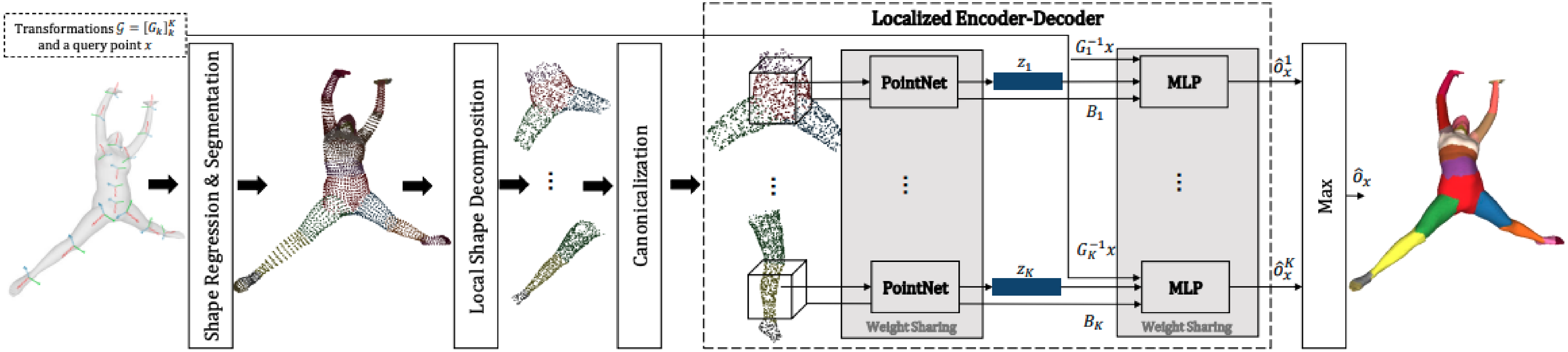
Marko Mihajlovic[1] Shunsuke Saito[2] Aayush Bansal[2] Michael Zollhoefer[2] Siyu Tang[1]
[1]ETH Zurich [2]Reality Labs Research at Meta

CVPR 2022

COAP: **Co**mpositional **A**rticulated Occupancy of **P**eople

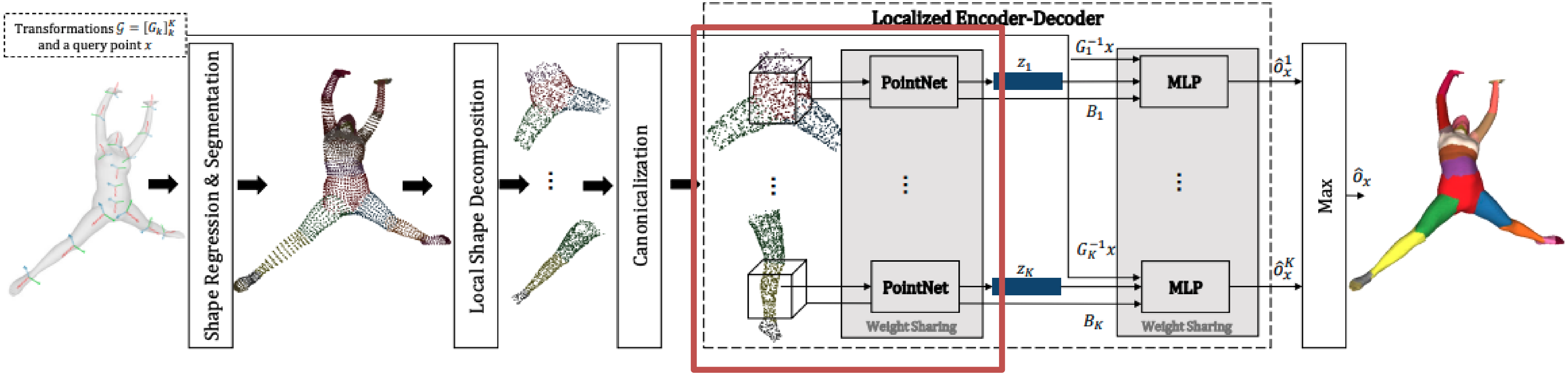


How is COAP different from NASA?



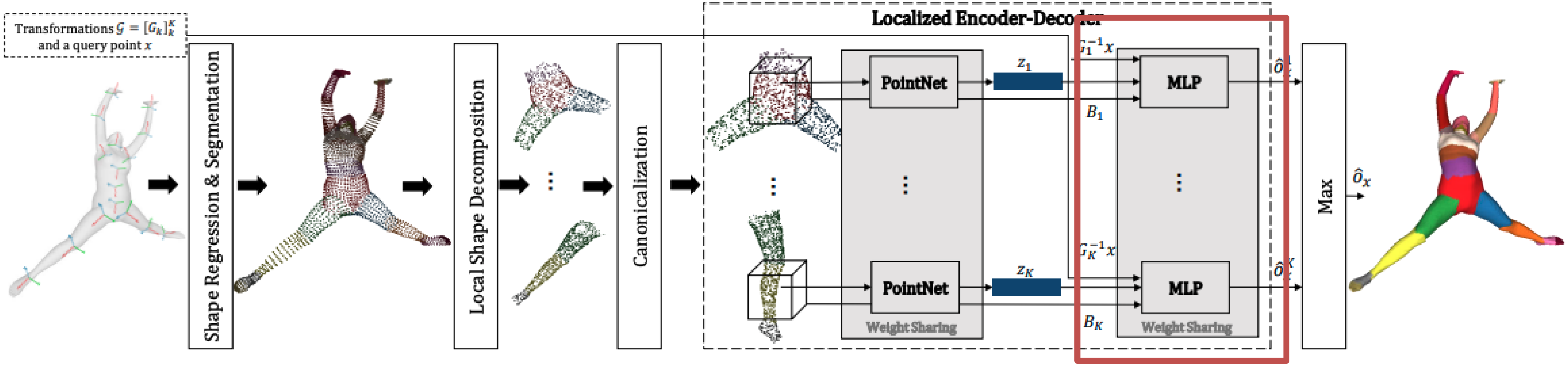
COAP is not subject-specific model

How is COAP different from NASA?



Per-part features = body part + few points from neighbouring parts

How is COAP different from NASA?

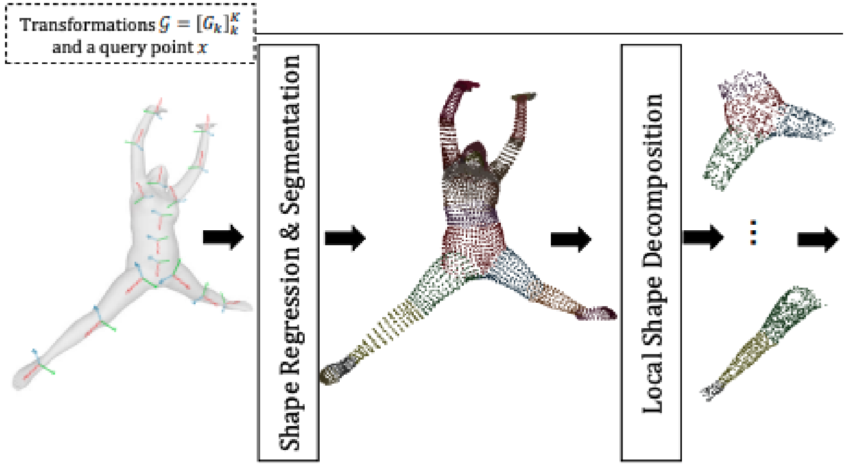


Shared Occupancy networks

A

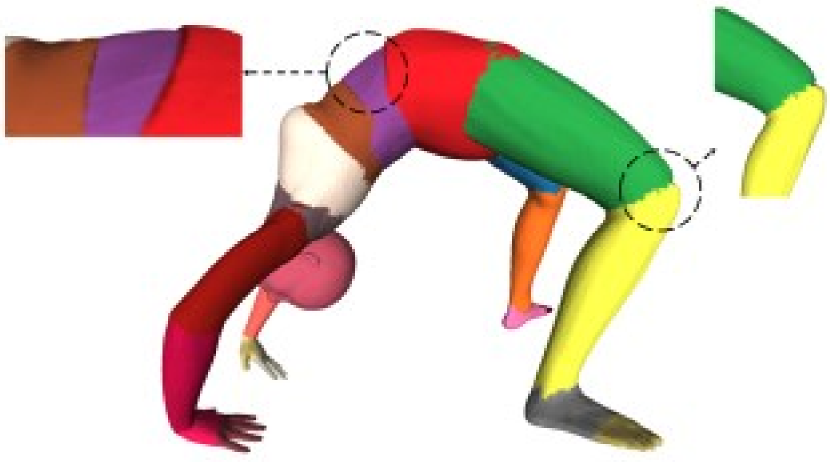
Part-based model

- Skinning weights are needed for part-decomposition
- Part-artefacts are prominent for out-of-distribution poses.
- Cannot model clothing/ loose clothing.



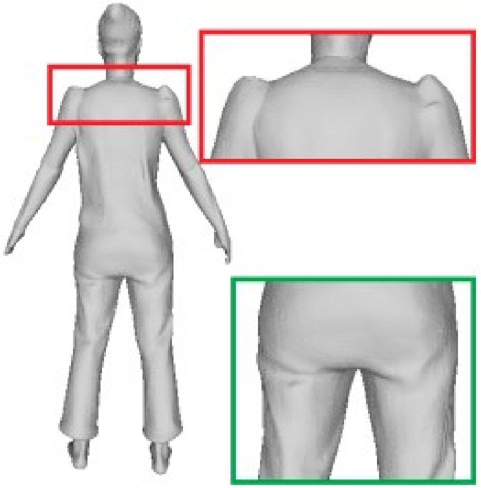
Part-based model limitations

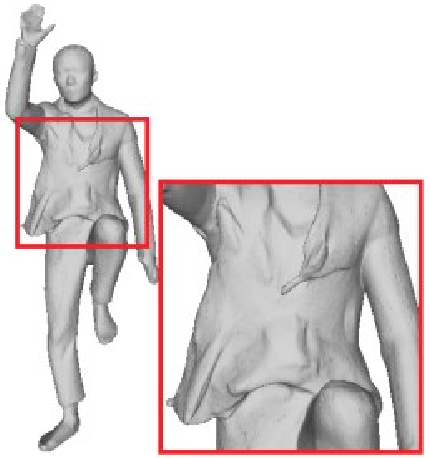
– Skinning weights are needed for part-decomposition
– Part-artefacts are prominent for out-of-distribution poses.
– Cannot model clothing/ loose clothing.



Part-based model

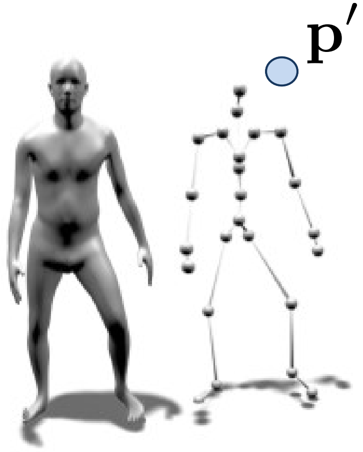
– Skinning weights are needed for part-decomposition
– Part-artefacts are prominent for out-of-distribution poses.
– Cannot model clothing/ loose clothing.





Learning pose/shape conditioned neural implicits using learned LBS and canonical shape

Learning pose/shape conditioned neural implicits using learned LBS and canonical shape



Given an input pose/shape and 3D query point

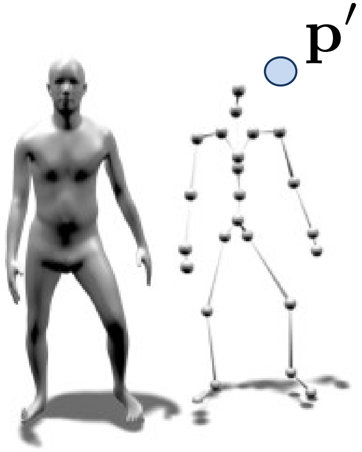
$$\mathbf{p}' \in \mathbb{R}^3$$

Predict pose/shape dependent deformation field in cananocial space

$$\Delta\mathbf{p} = f_{\text{d}}(\mathbf{p}, \boldsymbol{\theta}, \boldsymbol{\beta})$$

Get the final posed mesh

Learning pose/shape conditioned neural implicits using learned LBS and canonical shape



Given an input pose/shape and 3D query point

Map query point to canonical space? Using learned LBS?

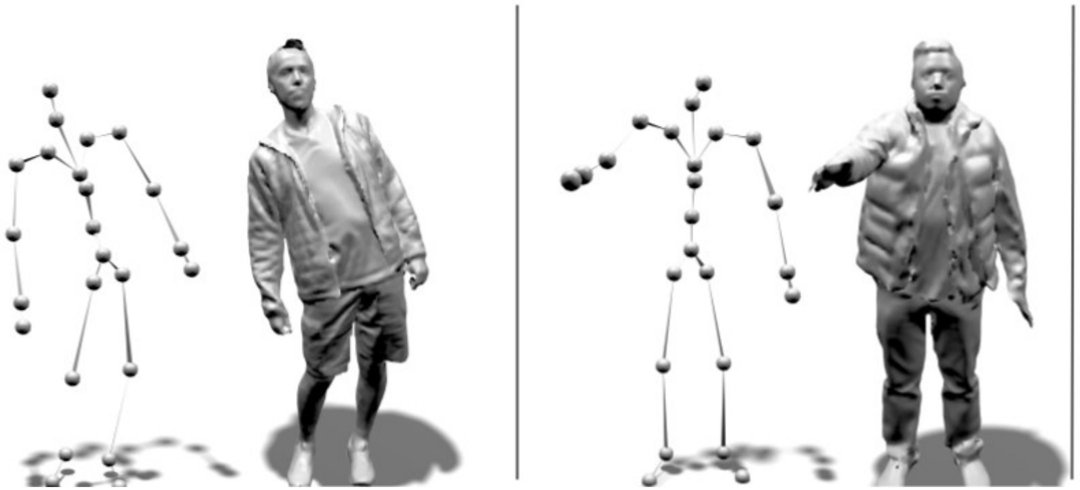
Predict pose/shape dependent deformation field in cananocial space

Get the final posed mesh

Neural-GIF: Neural Generalized Implicit Functions for Animating People in Clothing

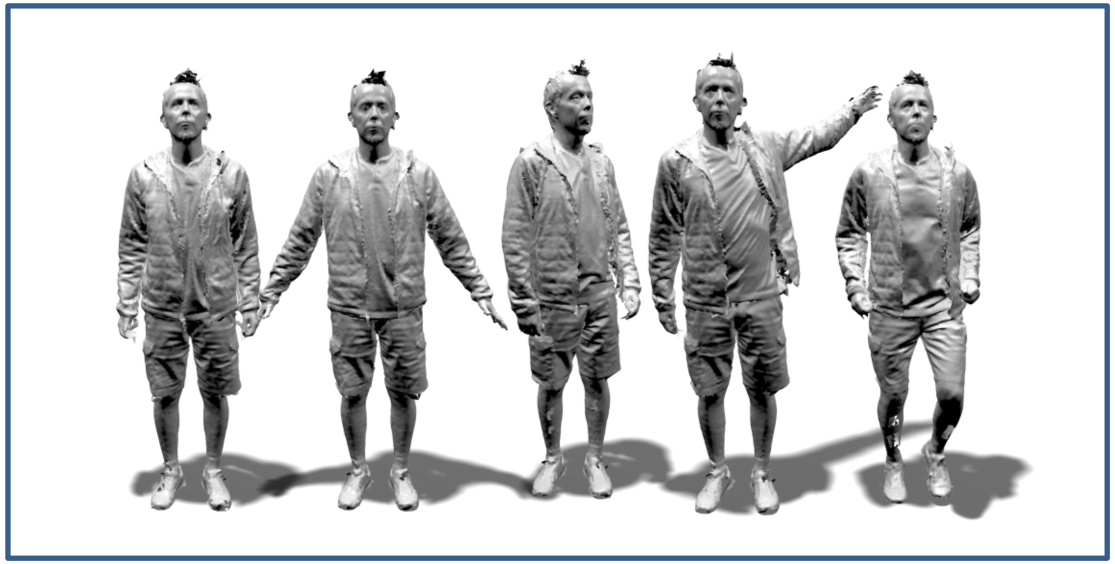
Garvita Tiwari Nikolaos Sarafianos Tony Tung Gerard Pons-Moll
University of Tuebingen MPI for Informatics, Saarland Informatics Campus, Germany Facebook Reality Labs, Sausalito, USA

ICCV 2021

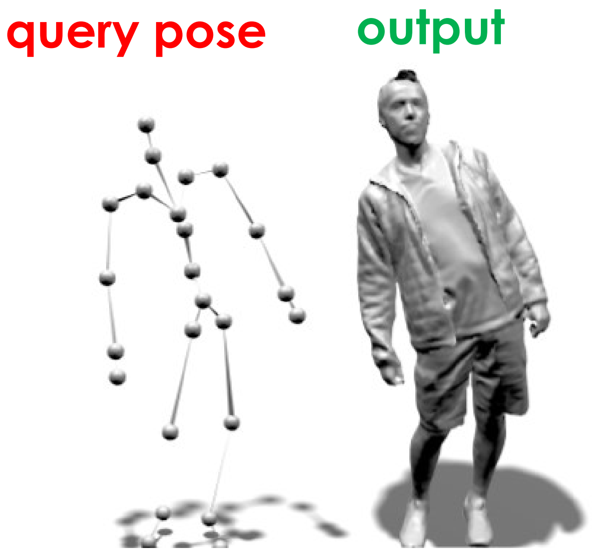


NeuralGIF

- A generalized framework to animate people in clothing(or clothing), which learns directly from scans



NeuralGIF is trained on set of raw scans for a given subject



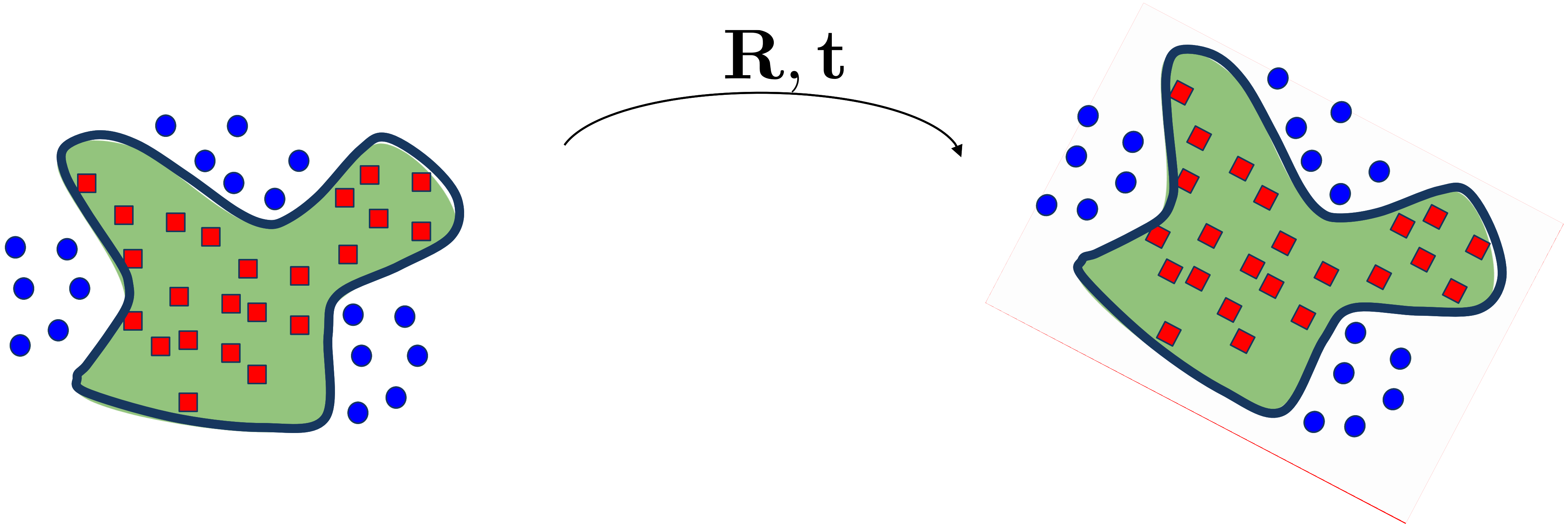
Given a query pose(left),
NeuralGIF animates the subject(right)

Tiwari et al. ICCV'21

Generalized Implicit Function

$\mathbf{p} = (x, y, z) \in \mathbb{R}^3$

$\mathbf{p}' = \mathbf{R}\mathbf{p} + \mathbf{t}$



$\mathbf{R}, \mathbf{t}$

$$\mathcal{S} = \{\mathbf{p},\ f(\mathbf{p}) = \tau\} \quad \mathcal{S}' = \{\mathbf{p}',\ f(\mathbf{R}^{-1}(\mathbf{p}' - \mathbf{t})) = \tau\}$$

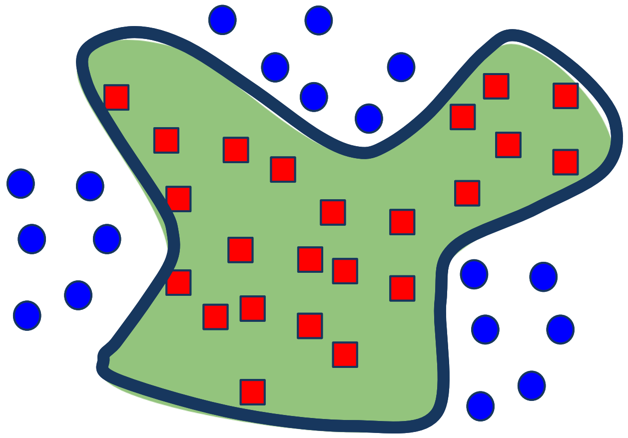
Sclaroff & Pentland Sigg'91
Tiwari et al. ICCV'21

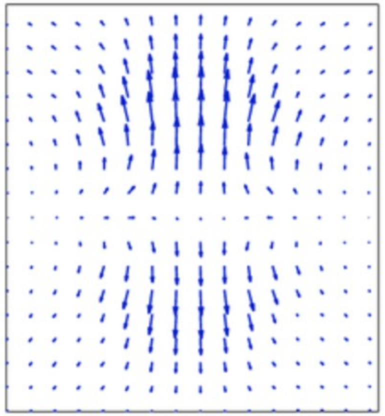
Generalized Implicit Function

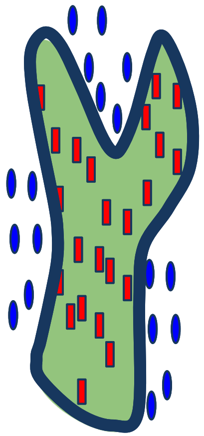
$$\mathbf{p} = (x, y, z) \in \mathbb{R}^3$$

$$\mathbf{p}' = \mathbf{p} + d(x, y, z)$$

$$d(x, y, z)$$





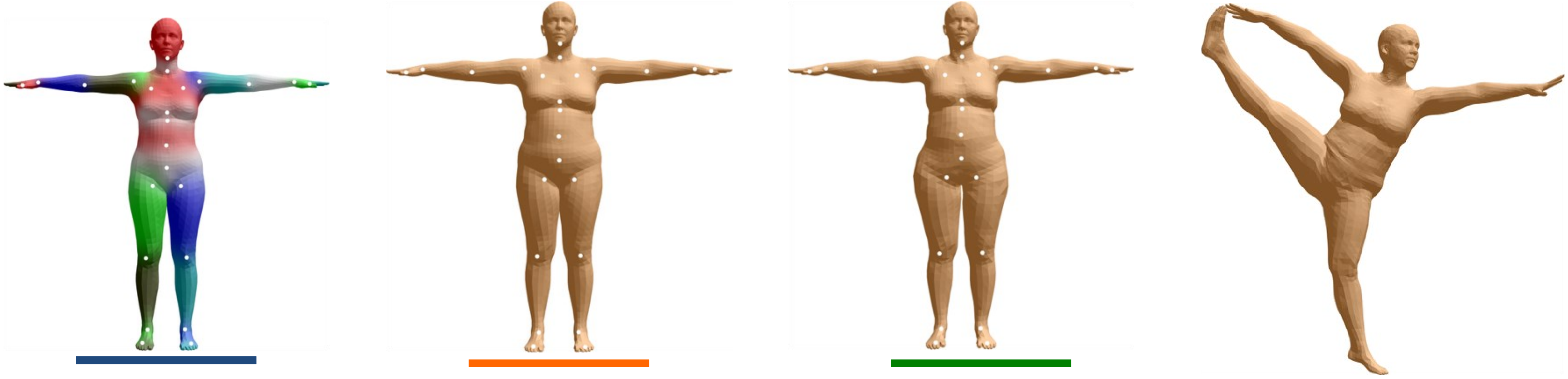


$$\mathcal{S} = \{\mathbf{p},\ f(\mathbf{p}) = \tau\} \quad \mathcal{S}' = \{\mathbf{p}',\ f(\mathbf{R}^{-1}(\mathbf{p}' - \mathbf{t})) = \tau\}$$

Tiwari et al. ICCV'21

SMPL model

$$\mathbf{p}' = \left(\sum_{i=1}^{K} \mathbf{w}_i \mathbf{B}_i \right) \mathbf{p}$$



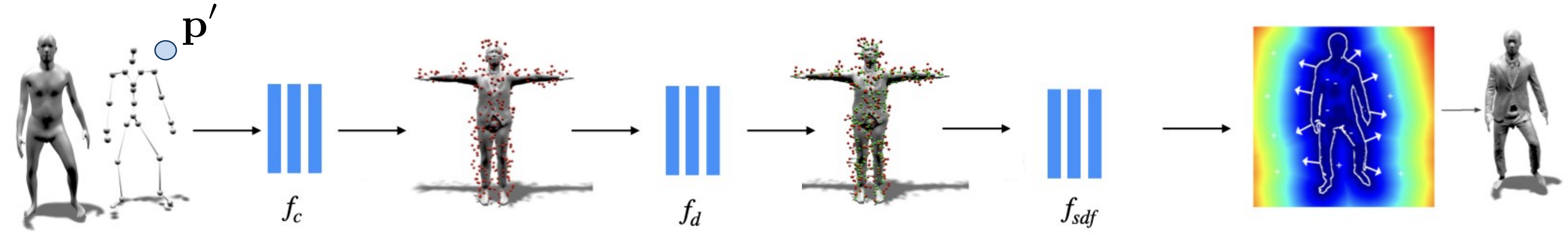
$$T(\boldsymbol{\theta}, \boldsymbol{\beta}) = \underline{\mathbf{T}_\mu} + \underline{B_s(\boldsymbol{\beta})} + \underline{B_p(\boldsymbol{\theta})}$$

$\downarrow$

Vertices in a 0-pose

Neural-GIF

How to predict the signed distance for a point in the posed space?



Query points Posed Space	Map to canonical space	Displacement field in canonical space	SDF prediction and reconstructed mesh
$\mathbf{p}' \in \mathbb{R}^3$	$\mathbf{p} = \left(\sum_{i=1}^{K} \mathbf{w}_i \mathbf{B}_i\right)^{-1} \mathbf{p}'$	$\Delta\mathbf{p} = f_{\mathrm{d}}(\mathbf{p}, \theta)$	$f_{\mathrm{sdf}}(\mathbf{p} + \Delta\mathbf{p})$

$$\mathbf{w} = f_c(\mathbf{p}', \theta) \qquad \mathbf{w} \in \mathbb{R}^{24}$$

$\mathbf{B}_i \rightarrow$ Joints transformation matrix

Tiwari et al. ICCV'21

Neural-GIF: Pose driven Animation

CAPE

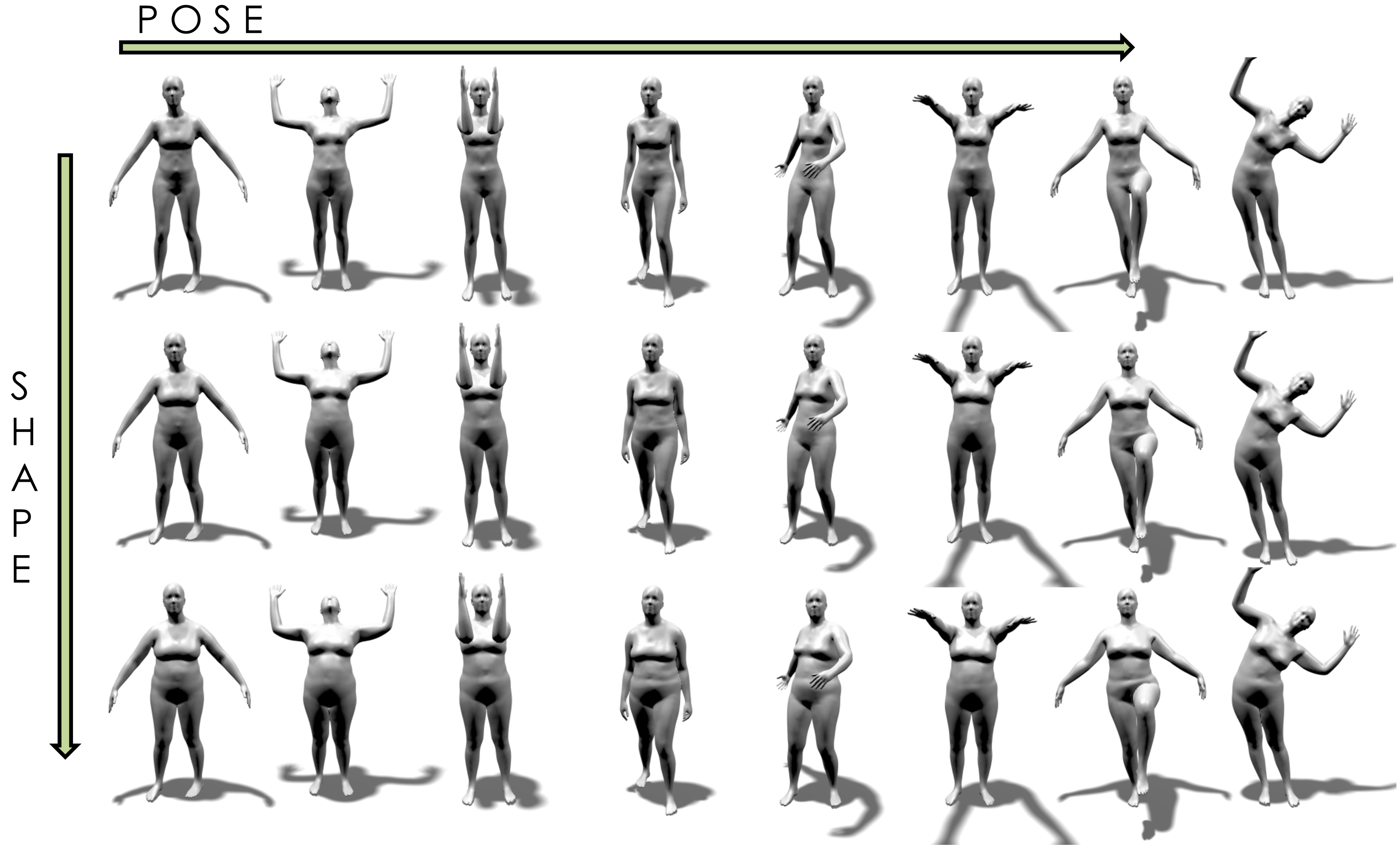
DFAUST

TailorNet-Shirt

TailorNet-Skirt

Tiwari et al. ICCV‘21

NeuralGIF as Multi-shape model



Tiwari et al. ICCV'21

Neural-GIF vs Scanimate

Scanimate

$$\mathcal{S} = \{\mathbf{p}, f_\theta(\mathbf{p};\theta)) = \tau\}$$

Neural-GIF

$$\mathcal{S} = \{\mathbf{p}', f(\mathbf{p} + \Delta\mathbf{p}(\theta))) = \tau\}$$

Advantadges of Neural-GIF

- A single Canonical f() is learned.
- More flexibility in toplogies
- Better detail
- Simpler model

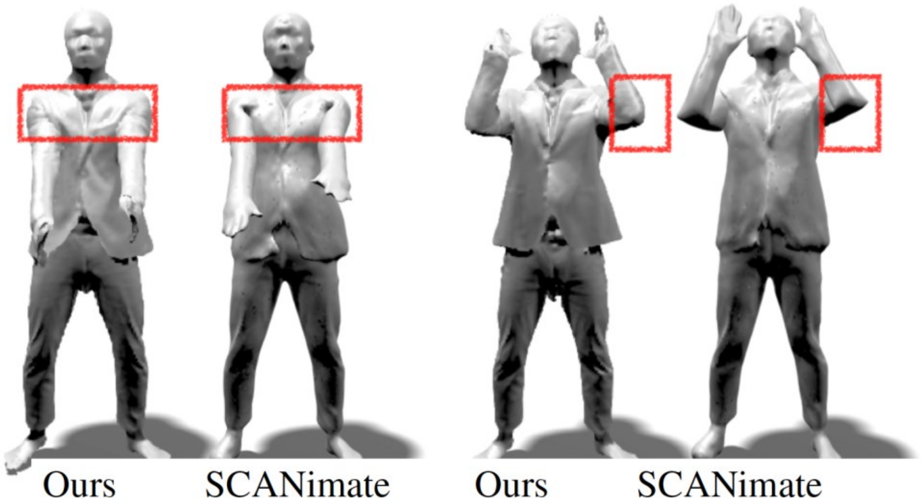


Figure 7. **Comparison with SCANimate:** We compare the results of our method on the CAPE dataset with SCANimate [51]. Our model preserves more details and does not have posing artifacts.

Tiwari et al. ICCV'21

Comparison with State-of-the-art methods

Dataset \ Model	NASA [18]			SCANimate [51]			**Ours (Neural-GIF)**		
	Point2Surface ↓	IoU ↑	F-Score ↑	Point2Surface ↓	IoU ↑	F-Score ↑	Point2Surface ↓	IoU ↑	F-Score ↑
CAPE [31]	10.67	0.918	94.32	**5.82**	**0.957**	**98.51**	**5.86**	**0.957**	**98.53**
ClothSeq	23.26	0.780	57.29	7.32	0.953	97.32	**4.73**	**0.967**	**99.15**
DFAUST [13]	10.52	0.939	95.48	3.79	0.971	99.50	**3.21**	**0.972**	**99.56**

Comparison with NASA [1] and SCANimate [2]. We report point to surface distance (in mm) and IoU and F-Scores(%) for comparison

1. NASA: Neural Articulated Shape Approximation, Deng et al., ECCV2020
2. SCANimate: Weakly Supervised Learning of Skinned Clothed Avatar Networks, Saito et al., CVPR2021

Most of the improvement is in modelling fine geometric details.

Tiwari et al. ICCV'21

NeuralGIF as Multi-shape model

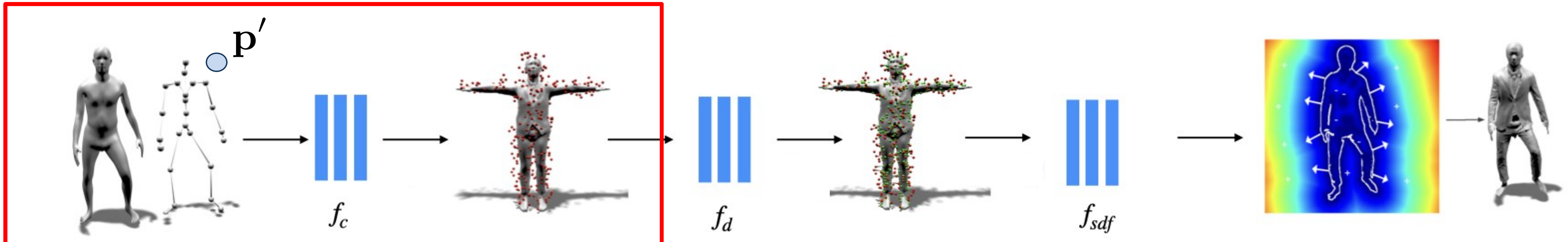
Dataset \ Model	LEAP [36]		Ours (Neural-GIF)	
	Point2Surface ↓	IoU ↑	Point2Surface ↓	IoU ↑
DFAUST [13]	3.42	0.958	**3.35**	**0.963**
MoVi [19]	**3.19**	**0.969**	3.20	0.969
SMPL	3.26	0.968	**3.18**	**0.971**

We quantitatively compare the results of our method with LEAP[1] on various datasets. We report point to surface distance (in mm) and IoU for comparison.

1. Learning Articulated Occupancy of People, Mihajlovic et al., CVP2021

Neural-GIF

Pose space to unpose space using skinning weights



Query points
Posed Space

Map to
canonical space

Displacement field in
canonical space

$\mathbf{p}' \in \mathbb{R}^3$

$$\mathbf{p} = \left(\sum_{i=1}^{K} \mathbf{w}_i \mathbf{B}_i \right)^{-1} \mathbf{p}'$$

$$\mathbf{w} = f_c(\mathbf{p}', \theta) \qquad \mathbf{w} \in \mathbb{R}^{24}$$

$\mathbf{B}_i \rightarrow$ Joints transformation matrix

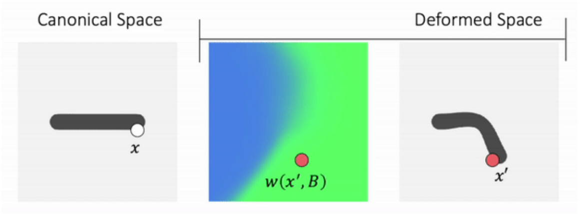
$$\Delta \mathbf{p} = f_{\mathrm{d}}(\mathbf{p}, \theta)$$

$$f_{\mathrm{sdf}}(\mathbf{p} + \Delta \mathbf{p})$$

Tiwari et al. ICCV'21

Backward and Forward skinning

Backward skinning

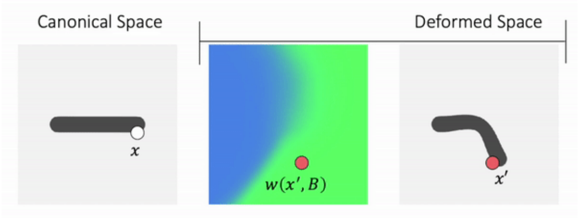


Posed space skinning field predicted

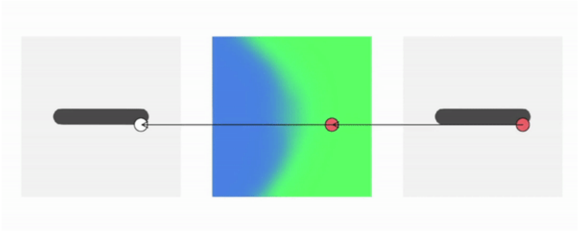
Source: https://autonomousvision.github.io/snarf/

Backward and Forward skinning

Backward skinning



Posed space skinning field predicted

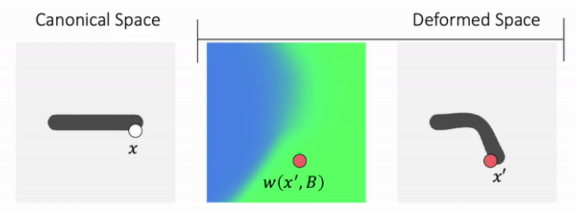


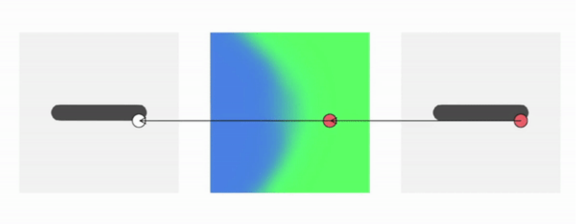
In backward skinning, w is predicted from the deformed point x' and the pose θ

Source: https://autonomousvision.github.io/snarf/

Backward and Forward skinning

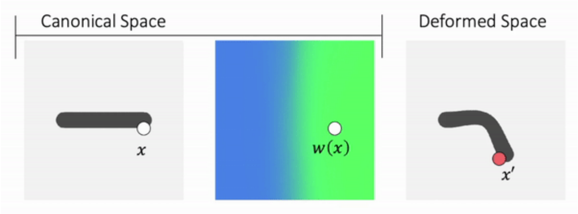
Backward skinning

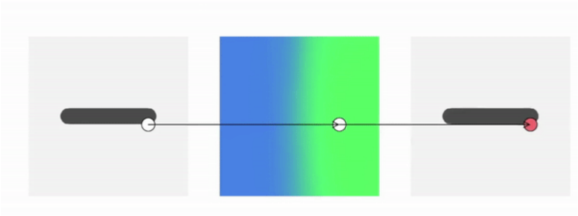




In backward skinning, w is predicted from the deformed point x' and the pose θ

Forward skinning

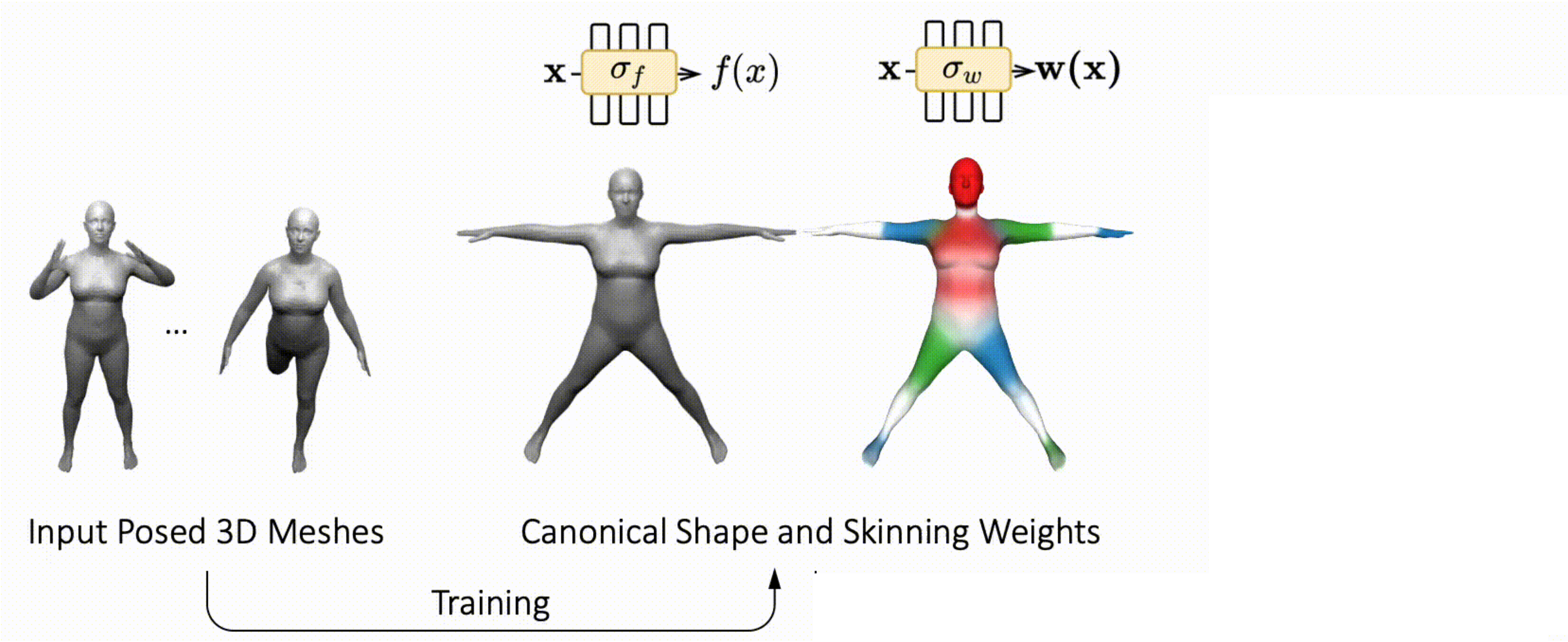




In forward skinning, w is predicted from the canonic point x^*

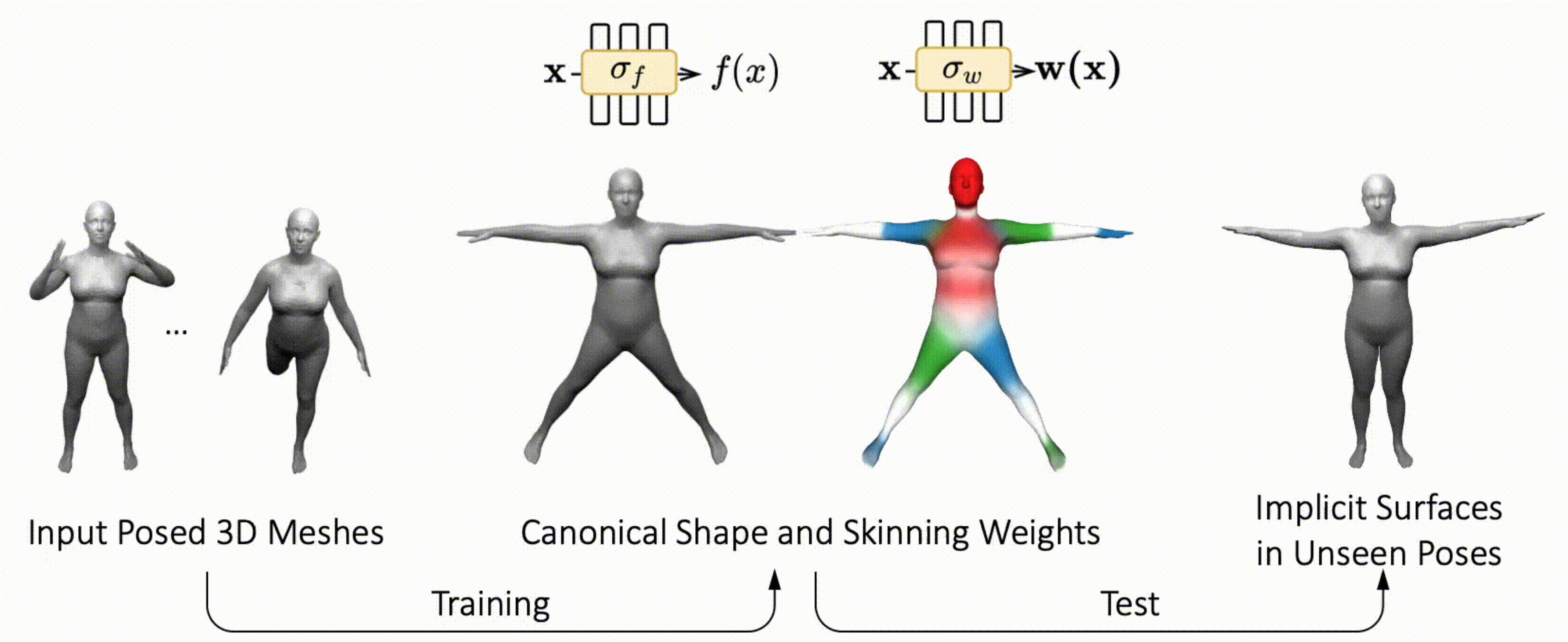
Source: https://autonomousvision.github.io/snarf/

SNARF: Differentiable Forward Skinning for Animating Non-Rigid Neural Implicit Shapes



Source: https://autonomousvision.github.io/snarf/

SNARF: Differentiable Forward Skinning for Animating Non-Rigid Neural Implicit Shapes



Source: https://autonomousvision.github.io/snarf/

SNARF: Differentiable Forward Skinning for Animating Non-Rigid Neural Implicit Shapes

SNARF is a forward skinning method:
Forward skinning explicitly defines

$$\mathbf{x} \rightarrow \mathbf{x}' \ \forall \mathbf{x}$$

Learning canonical shape from posed scans requires,

$$\mathbf{x}' \rightarrow \mathbf{x} \ \forall \mathbf{x}'$$

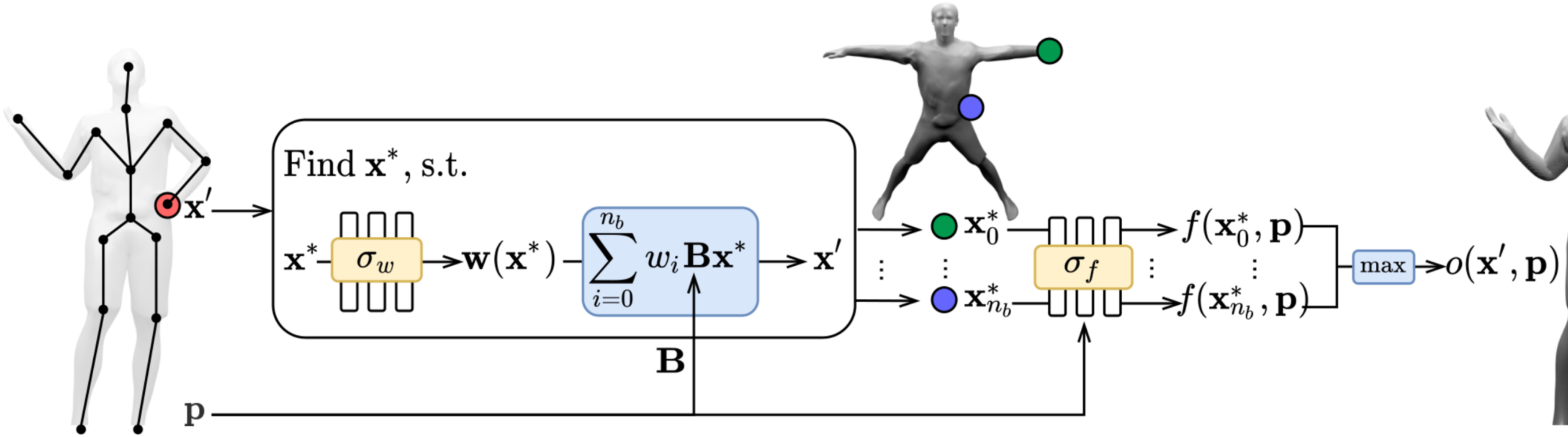
Given $\mathbf{x} \rightarrow \mathbf{x}'$, determine $\mathbf{x}' \rightarrow \mathbf{x}$

- Implicit relation, no closed form solution
- Non-bijective mapping, multiple solution may exists

Source: Chen et al., ICCV 2021

SNARF: Differentiable Forward Skinning for Animating Non-Rigid Neural Implicit Shapes

Key Idea: Differentiable Forward Skinning



Requires differentiating through the solution of a non-linear system

Source: Chen et al., ICCV 2021

SNARF: Understand the training objective

$$\text{find } \mathbf{x}^* \text{ such that } \mathbf{x}' - \sum_{i=0}^{n_b} \mathbf{B}_i \sigma_{w,i}(\mathbf{x}^*) = 0$$

Canonical Point (multiple solutions)

Posed point

Neural network which predicts skinning weights from canonic point $\sigma_{w,i}(\mathbf{x}^*) = \mathbf{w}(\mathbf{x}^*) \in \mathbb{R}^{nb}$

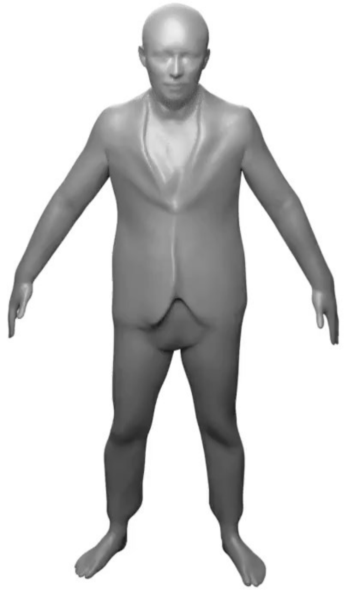
$$\mathcal{L}_w = \mathcal{L}(\max_b \{ f(\mathbf{x}_b^*(w)) \}_{b=1}^{n_b}, o(\mathbf{x}'))$$

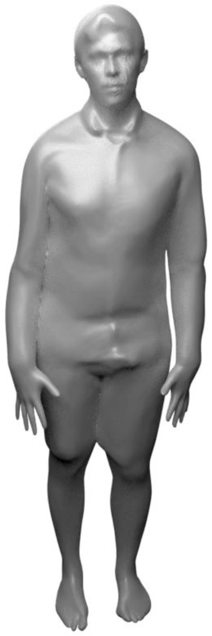
Cross-entropy loss

Ground truth occupancy

Challenge: compute $\frac{\partial \mathcal{L}}{\partial w}$ $\longrightarrow$ Possible to backprop iterative root finding

SNARF results

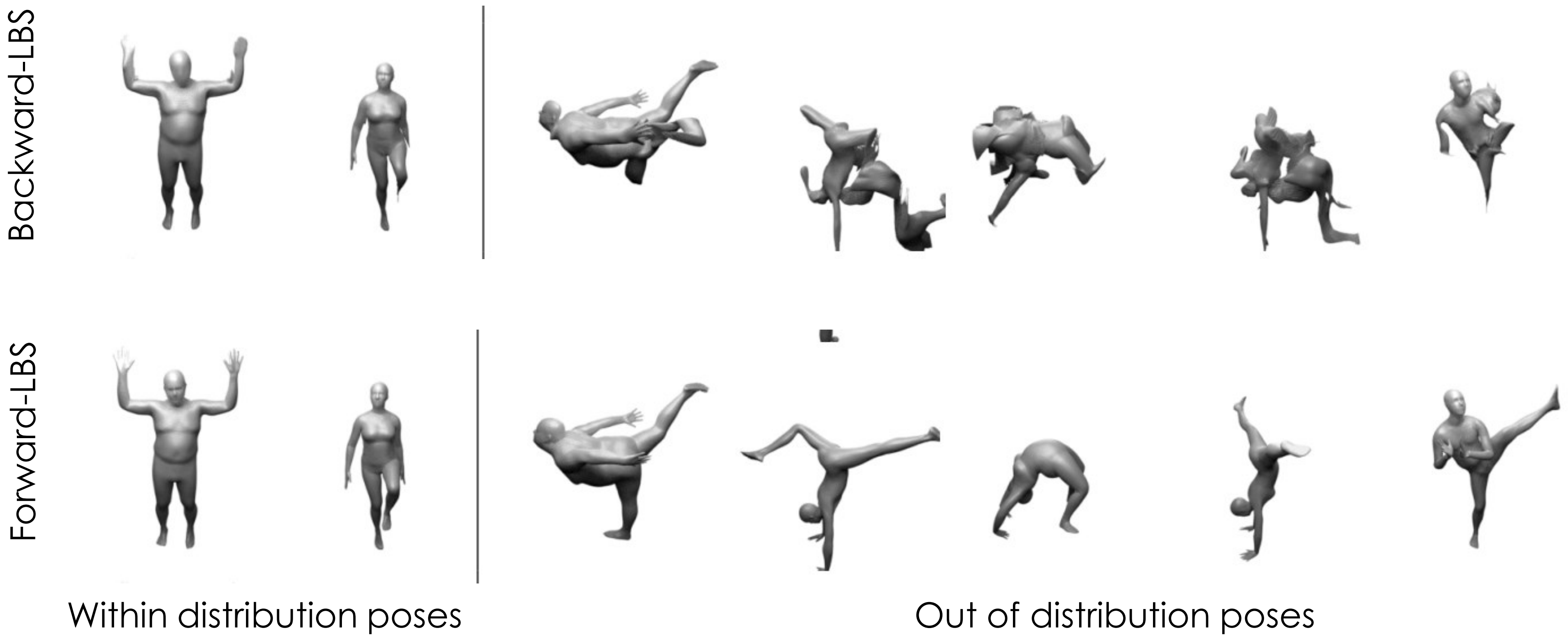




Source: Chen et al., ICCV 2021

Backward and Forward skinning

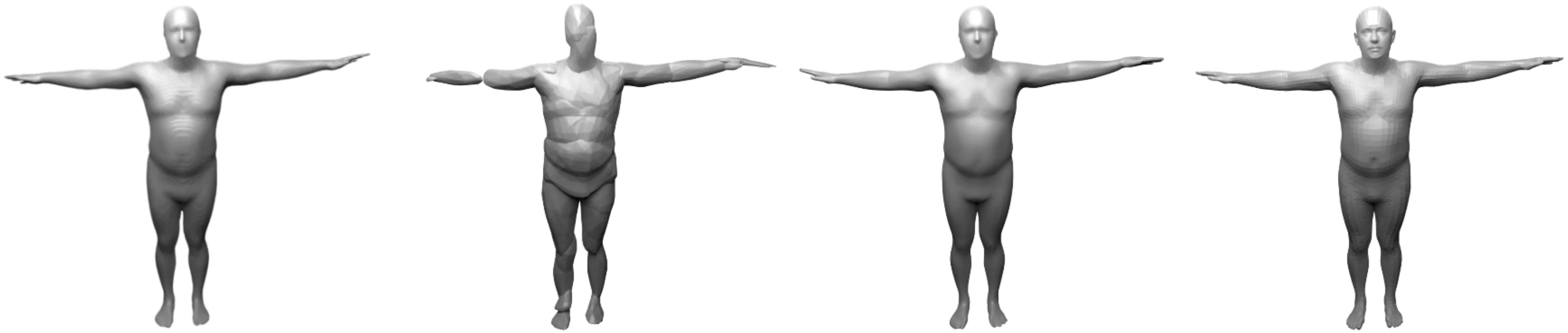
Forward skinning models have better generalization w.r.t. unseen pose.



Backward and Forward skinning

Forward skinning models have better generalization w.r.t. unseen pose.

Reconstructions of Novel Poses **outside** Training Distribution
(PosePrior)



Backward Skinning NASA Ours Ground Truth

Backward and Forward skinning

- **Forward** skinning models have **better generalization** w.r.t. unseen pose.
- **Backward** skinning models have **higher fidelity** (for distribution poses) and **more flexible** to model loose clothing.

Works like NASA, COAP, Neural-GIF, SNARF are not generative models

PART1: Neural Implicits for 3D Shapes

PART2: Neural Implicits for Humans

PART3: Neural Implicits – Generative Models

PART4: Point-based Clothing Models

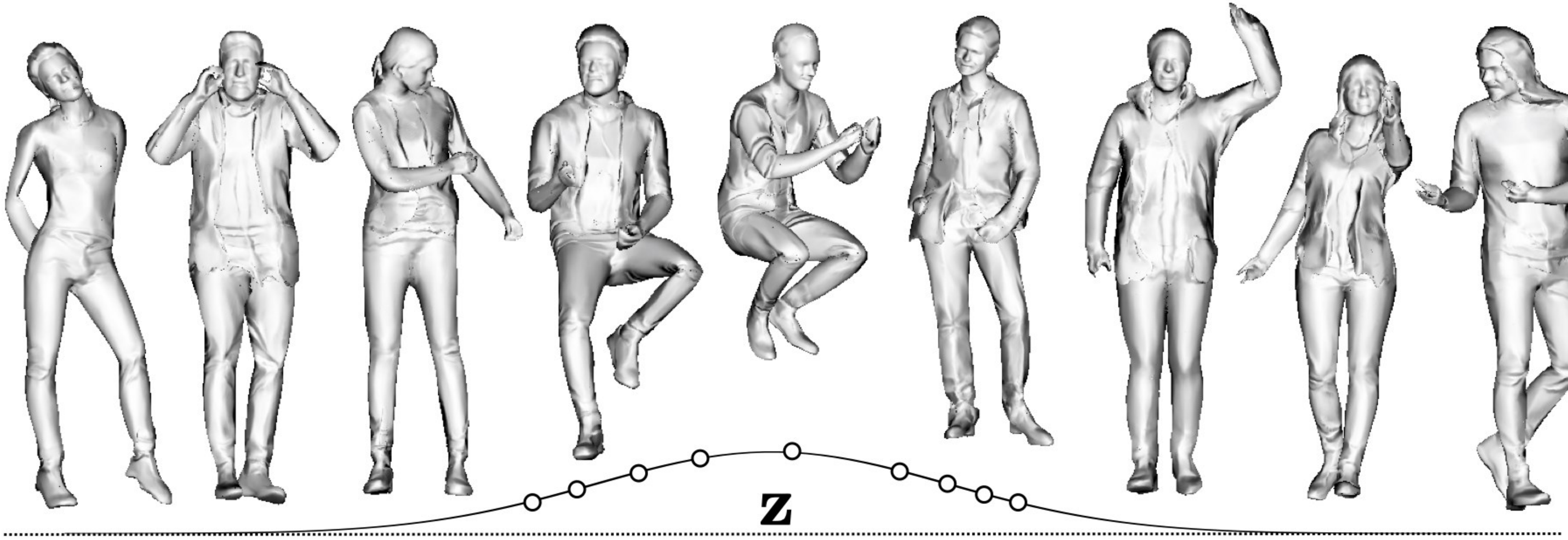
Works like NASA, COAP, Neural-GIF, SNARF are not generative models

What’s next??

Neural Implicit based generative model of people in clothing.

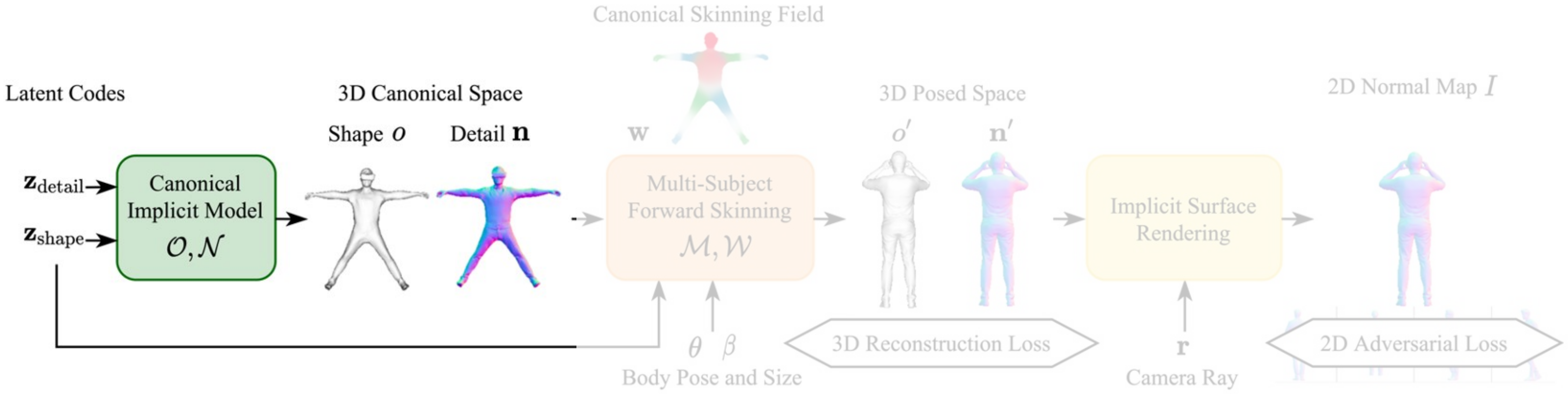
gDNA: Towards Generative Detailed Neural Avatars

Generative model of human in clothing using Neural Implicits



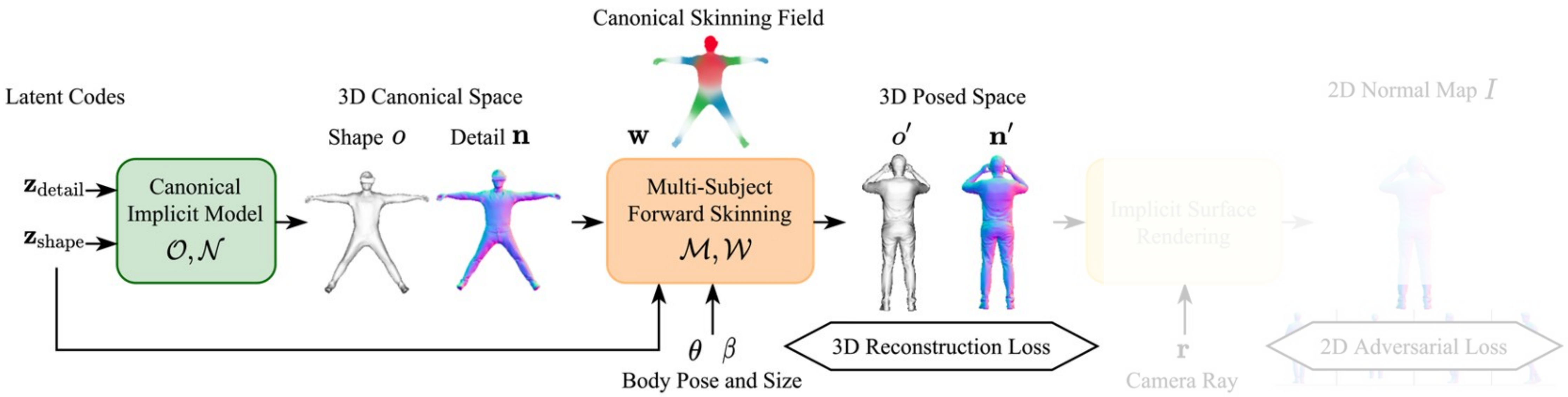
Chen et al., CVPR 2022

gDNA: Towards Generative Detailed Neural Avatars



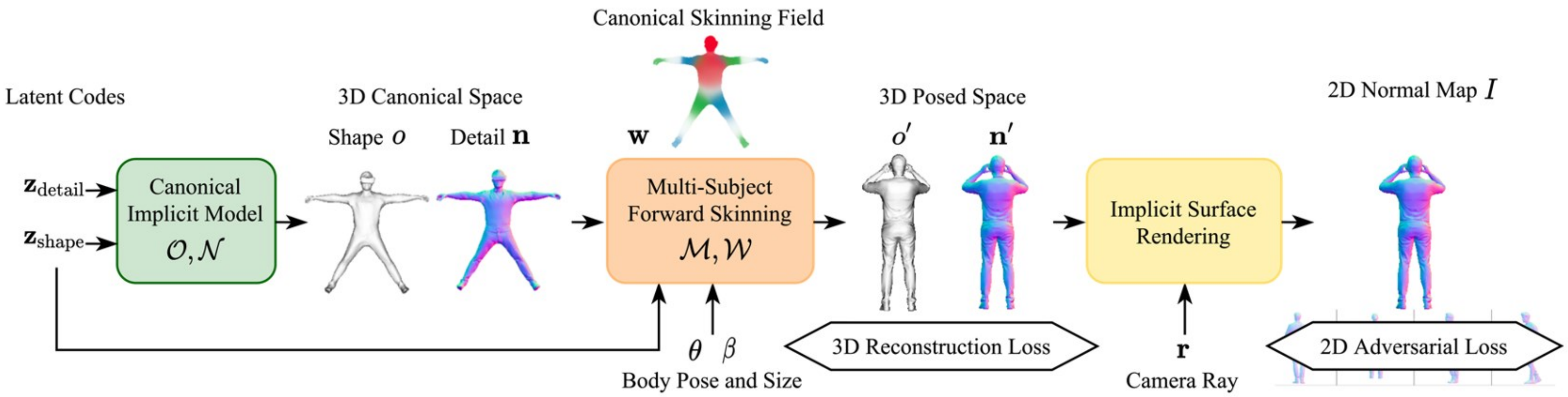
Chen et al., CVPR 2022

gDNA: Towards Generative Detailed Neural Avatars



Chen et al., CVPR 2022

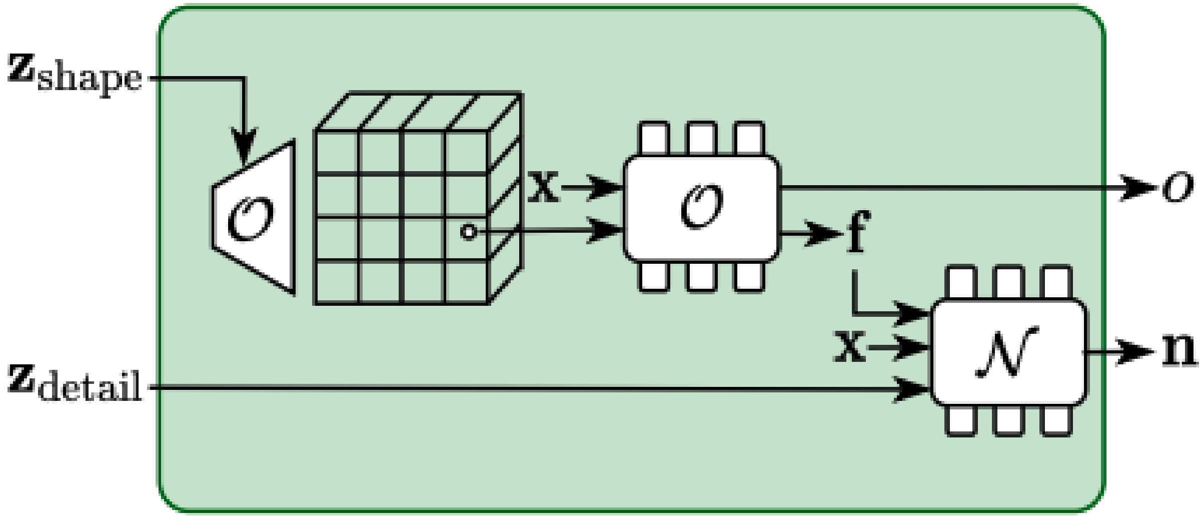
gDNA: Towards Generative Detailed Neural Avatars



Training is based on auto-decoders for the 3D shape and GANs for stochastic detail

Chen et al., CVPR 2022

Canonical Implicit model



$$\mathcal{S}(\mathbf{z}_{\text{shape}}) = \{\mathbf{x} \mid \mathcal{O}(\mathbf{x}, \mathbf{z}_{\text{shape}}) = \tau\}$$

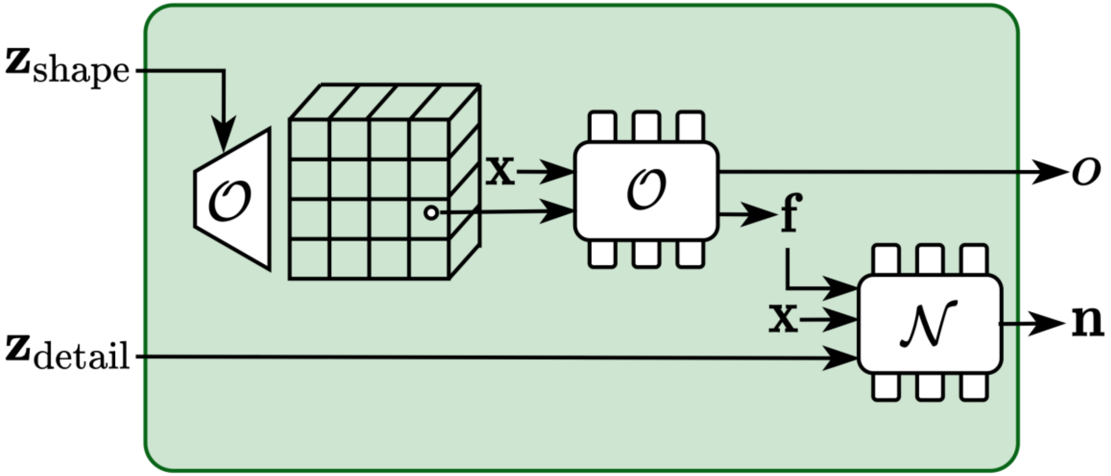
$$\mathcal{O} : \mathbb{R}^3 \times \mathbb{R}^{L_{\text{shape}}} \to [0,1] \times \mathbb{R}^{L_{\mathbf{f}}}$$

$$(\mathbf{x}, \mathbf{z}_{\text{shape}}) \mapsto (o, \mathbf{f})$$

$$\mathbf{z}_{\text{shape}} \in \mathbb{R}^{L_{\text{shape}}}$$

Chen et al., CVPR 2022

Canonical Implicit model



$$\mathcal{S}(\mathbf{z}_{\text{shape}}) = \{\mathbf{x} \mid \mathcal{O}(\mathbf{x}, \mathbf{z}_{\text{shape}}) = \tau\}$$

$$\mathcal{O} : \mathbb{R}^3 \times \mathbb{R}^{L_{\text{shape}}} \to [0,1] \times \mathbb{R}^{L_{\mathbf{f}}}$$

$$(\mathbf{x}, \mathbf{z}_{\text{shape}}) \mapsto (o, \mathbf{f})$$

$$\mathbf{z}_{\text{shape}} \in \mathbb{R}^{L_{\text{shape}}}$$

$$\mathcal{N} : \mathbb{R}^3 \times \mathbb{R}^{L_{\text{detail}}} \times \mathbb{R}^{L_{\mathbf{f}}} \to \mathbb{R}^3$$

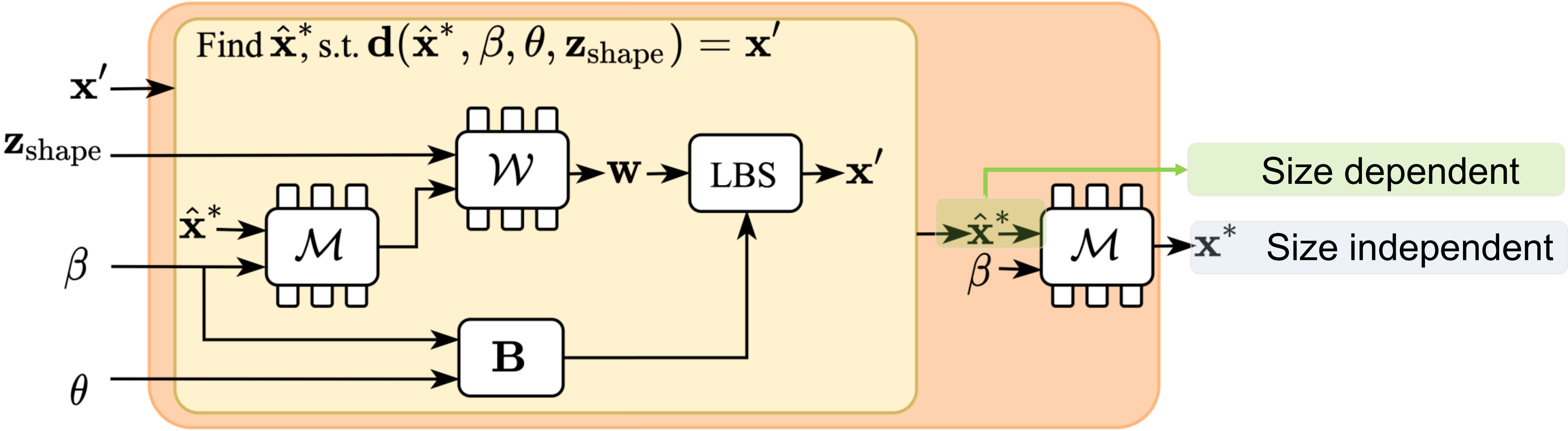
$$(\mathbf{x}, \mathbf{z}_{\text{detail}}, \mathbf{f}) \mapsto \mathbf{n}$$

$$\mathbf{z}_{\text{detail}} \in \mathbb{R}^{L_{\text{detail}}}$$

Chen et al., CVPR 2022

Multi-subject forward-skinning model

Based on SNARF



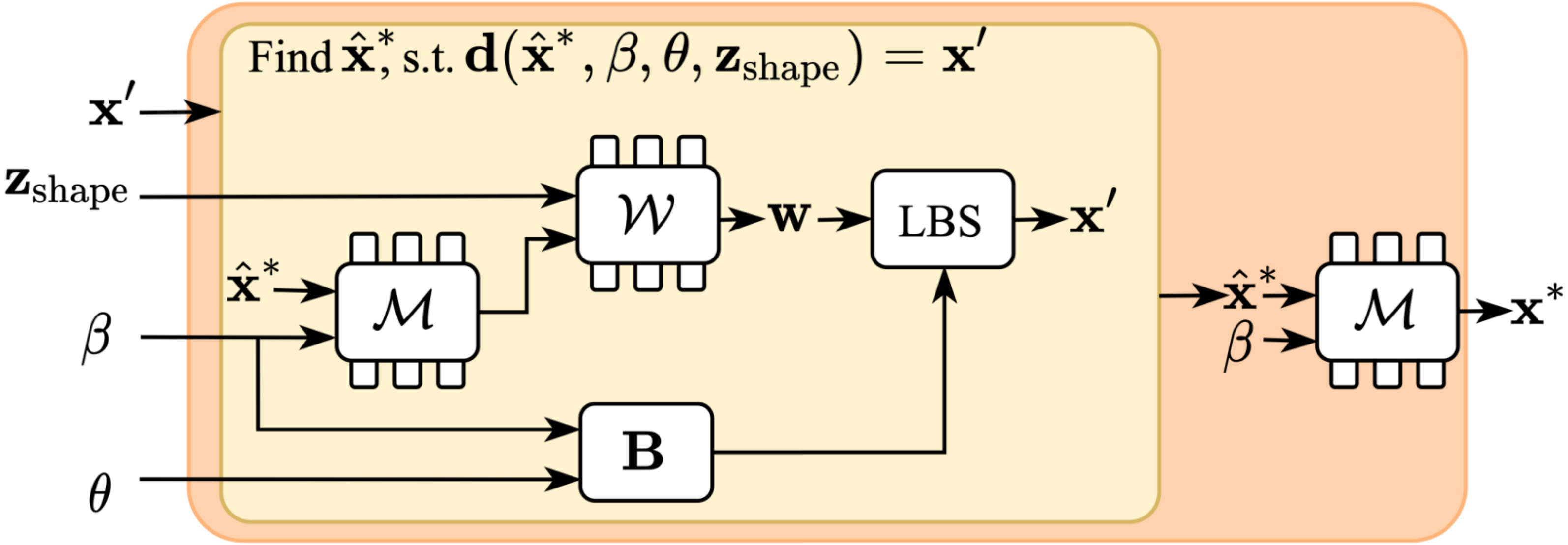
$$\mathcal{W} : \mathbb{R}^3 \times \mathbb{R}^{L_{\text{shape}}} \to \mathbb{R}^{n_b}$$
$$(\mathbf{x}, \mathbf{z}_{\text{shape}}) \mapsto \mathbf{w}$$

Skinning field in a body-shape-independent space

Chen et al., CVPR 2022

Multi-subject forward-skinning model

Based on SNARF



$$\mathcal{W} : \mathbb{R}^3 \times \mathbb{R}^{L_{\text{shape}}} \to \mathbb{R}^{n_b}$$
$$(\mathbf{x}, \mathbf{z}_{\text{shape}}) \mapsto \mathbf{w}$$

Skinning field in a body-shape-independent space

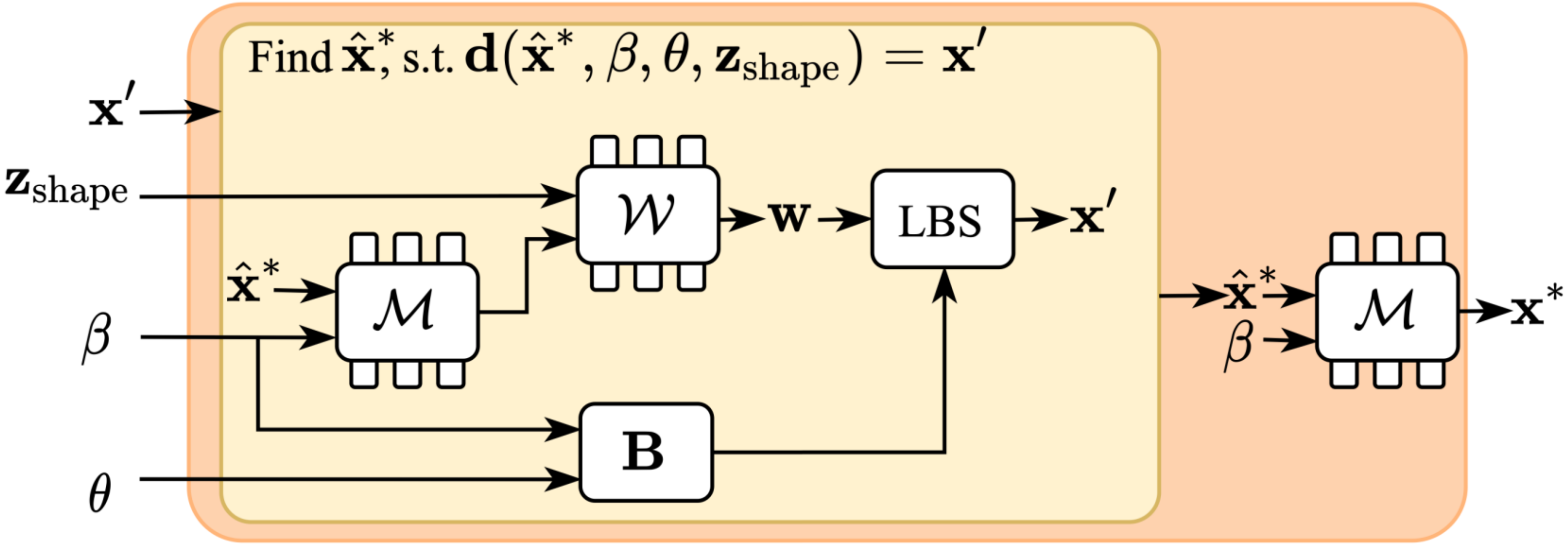
$$\mathcal{M} : \mathbb{R}^3 \times \mathbb{R}^{L_\beta} \to \mathbb{R}^3$$
$$(\hat{\mathbf{x}}, \beta) \mapsto \mathbf{x}$$

Body shape dependent warping field

Chen et al., CVPR 2022

Multi-subject forward-skinning model

Based on SNARF

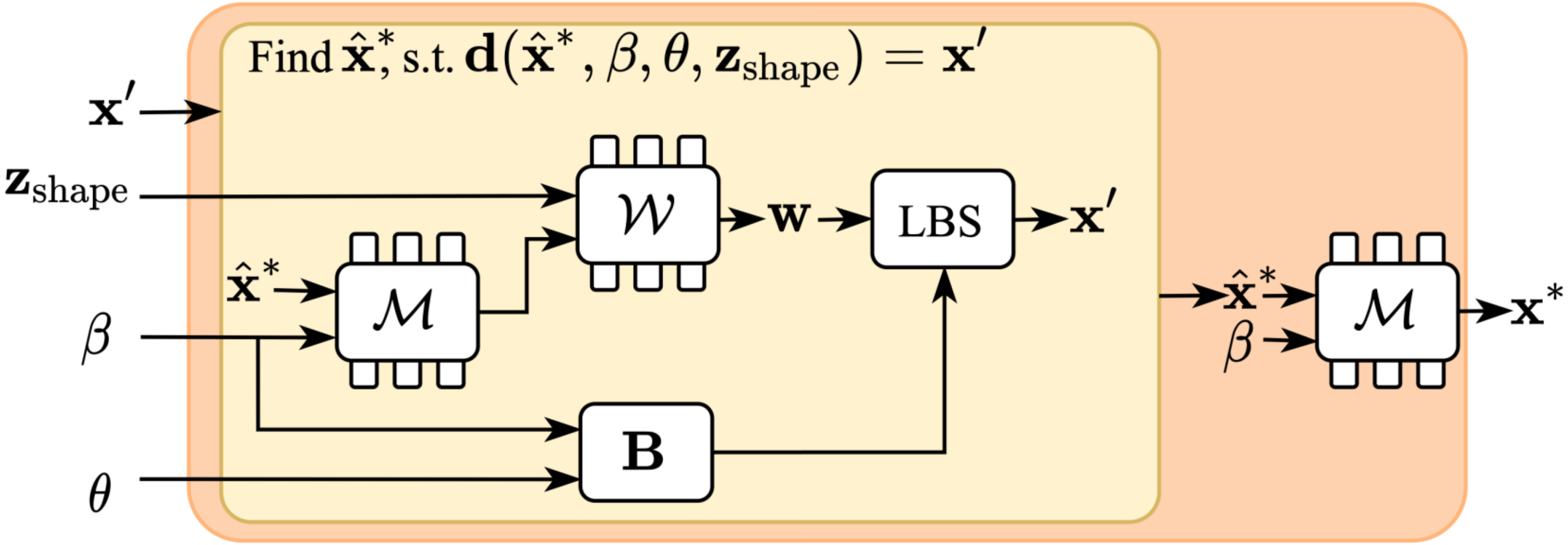


$$\mathbf{d}(\hat{\mathbf{x}}, \boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{z}_{\text{shape}}) - \mathbf{x}' = \mathbf{0},$$

Chen et al., CVPR 2022

Multi-subject forward-skinning model

Based on SNARF



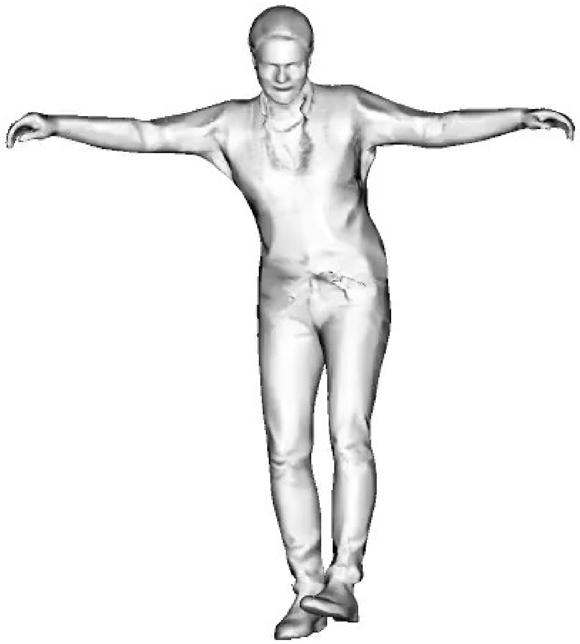
$$\mathbf{d}(\hat{\mathbf{x}}, \boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{z}_{\text{shape}}) - \mathbf{x}' = \mathbf{0},$$

$$\mathbf{x}^* = \mathcal{M}(\hat{\mathbf{x}}^*, \boldsymbol{\beta})$$

Chen et al., CVPR 2022

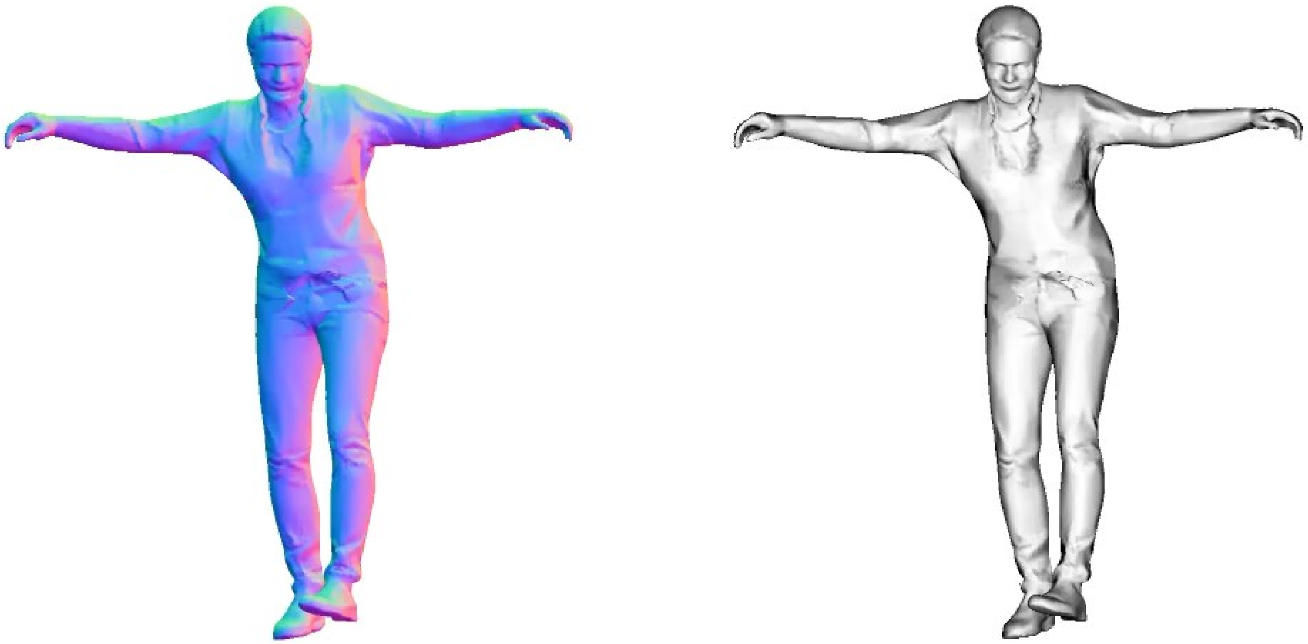
gDNA: Towards Generative Detailed Neural Avatars

Pose conditioned



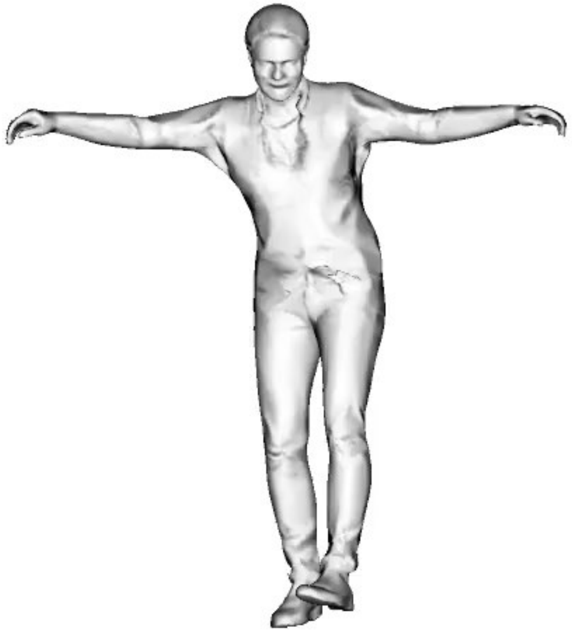
gDNA: Towards Generative Detailed Neural Avatars

Body shape conditioned



gDNA: Towards Generative Detailed Neural Avatars

Clothing style/shape conditioned

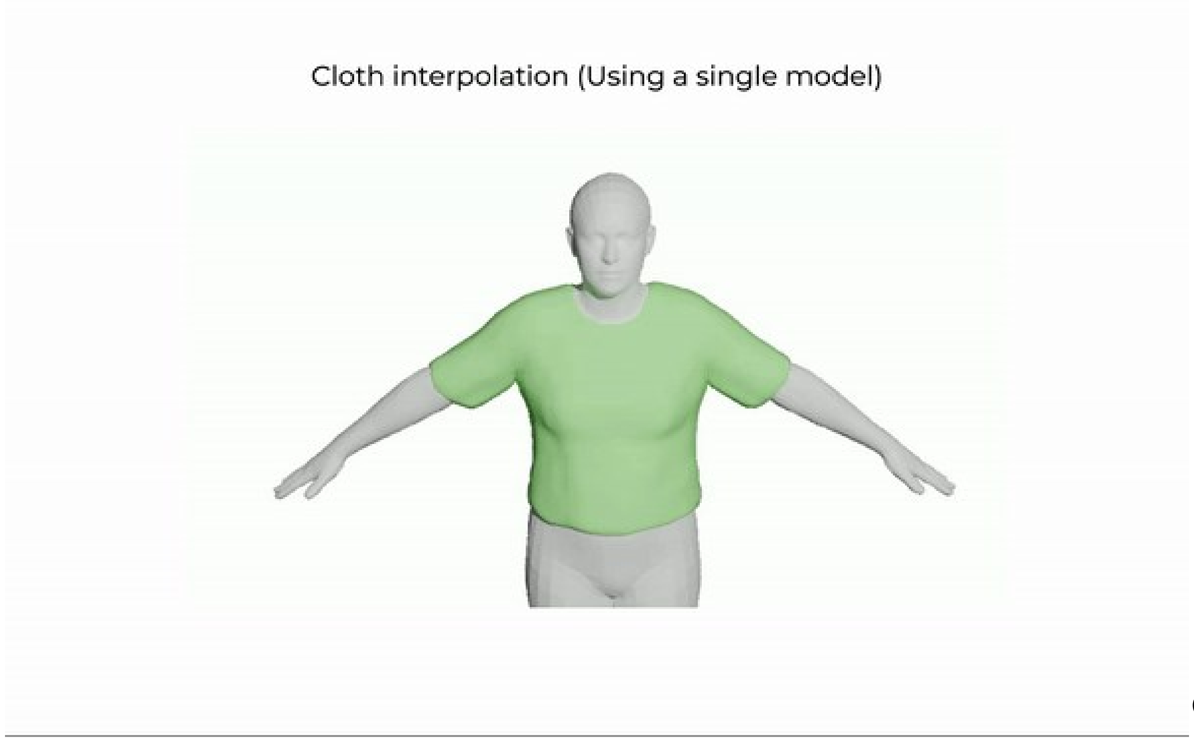


SMPLicit: Topology-aware Generative Model for Clothed People

Enric Corona Albert Pumarola Guillem Alenya Gerard Pons-Moll Francesc Moreno-Noguer

Institut de Robotica i Informatica Industrial, CSIC-UPC, Barcelona, Spain Max Planck Institute for Informatics

CVPR 2021



Corona et al., **SMPLicit,** CVPR'21

Vertex-based SMPL to Implicit SMPL(SMPLicit)

Vertex based Clothing model

E.g, TailorNet predicts vertex displacement D as a function of pose, shape and clothing style (requires multiple cloth templates)

$$M(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{D}) : \boldsymbol{\theta} \times \boldsymbol{\beta} \times \mathbf{D} \rightarrow \mathbf{V} \in \mathbb{R}^3$$
$$T(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{D}) = \mathbf{T} + B_s(\boldsymbol{\beta}) + B_p(\boldsymbol{\theta}) + \mathbf{D}$$

Neural implicit clothing model

Predicts the unsigned distance of the surface as a function of pose, shape, clothing cut, and style

$$C(\mathbf{p}, \boldsymbol{\theta}, \mathbf{z}_{\text{cut}}, \mathbf{z}_{\text{style}}) \rightarrow \mathbb{R}^+$$

Clothing cut controls how much clothing overlaps with the body (sleeve length, pant length)

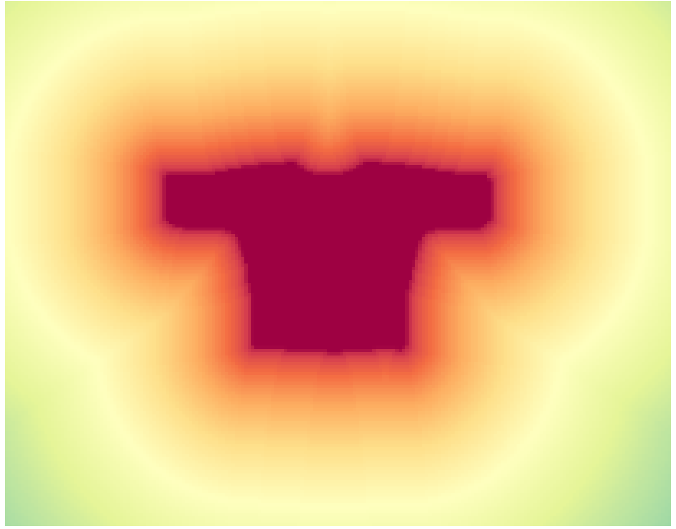
Clothing style controls the size, and fit

Corona et al., **SMPLicit,** CVPR’21

Moving to new topologies: Implicit representations

Unsigned distance field

$$D(p) \leftarrow C(\mathbf{P}_{\boldsymbol{\beta}}, \mathbf{z})$$

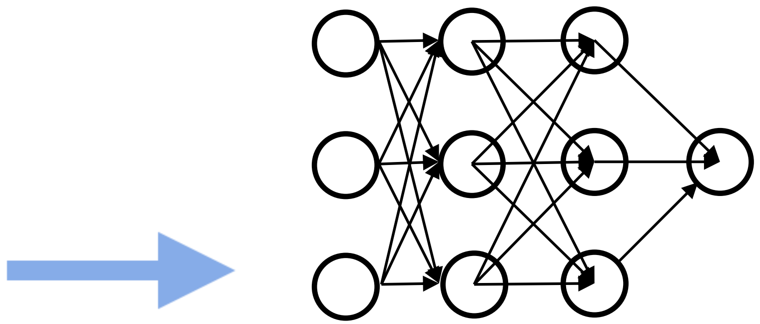


Corona et al., **SMPLicit,** CVPR'21

Moving to new topologies: Implicit representations

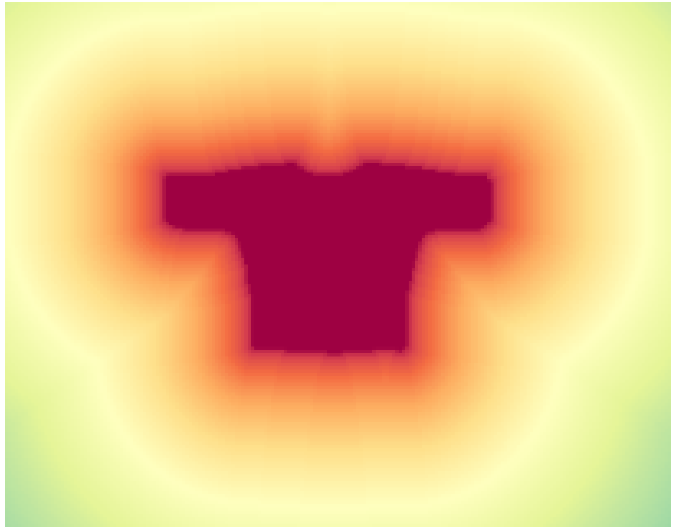
Input point $\mathbf{P}$

MLP



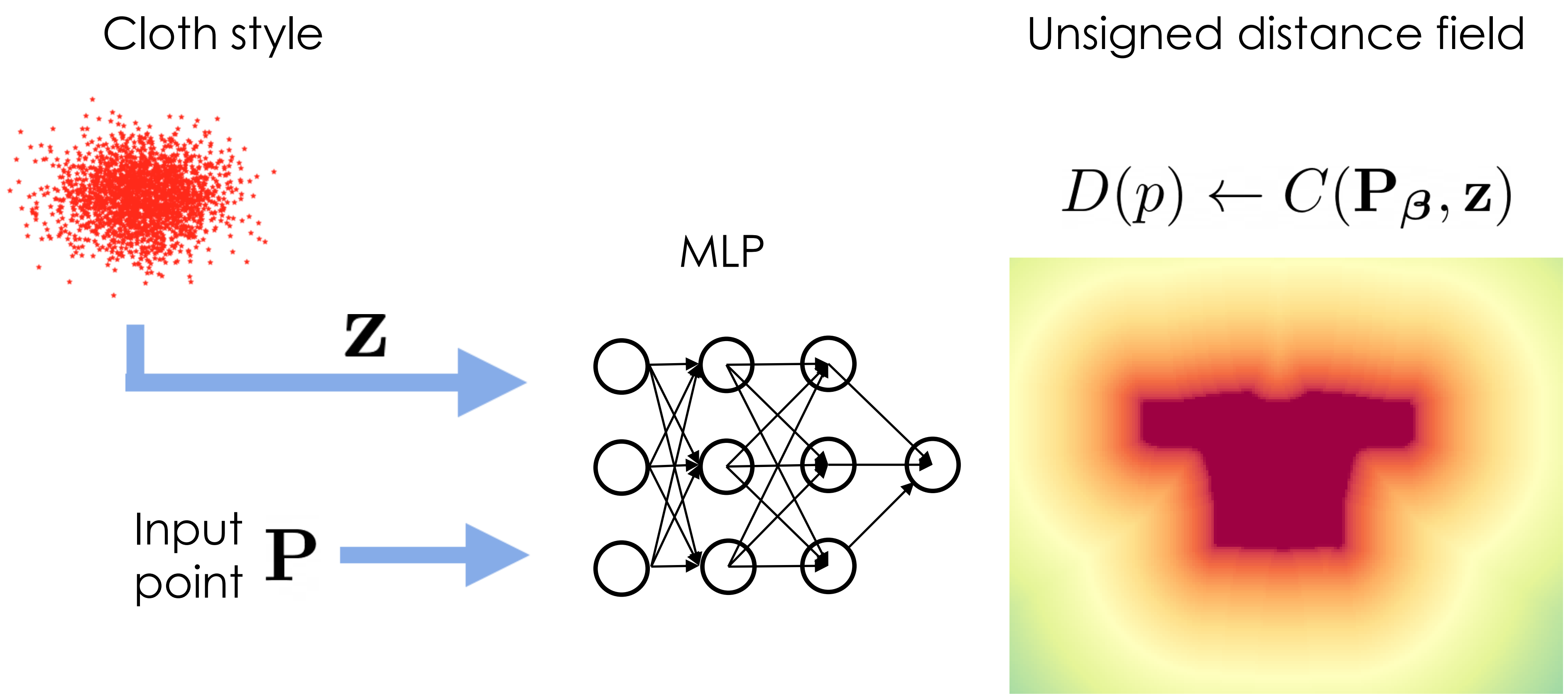
Unsigned distance field

$$D(p) \leftarrow C(\mathbf{P}_{\boldsymbol{\beta}}, \mathbf{z})$$



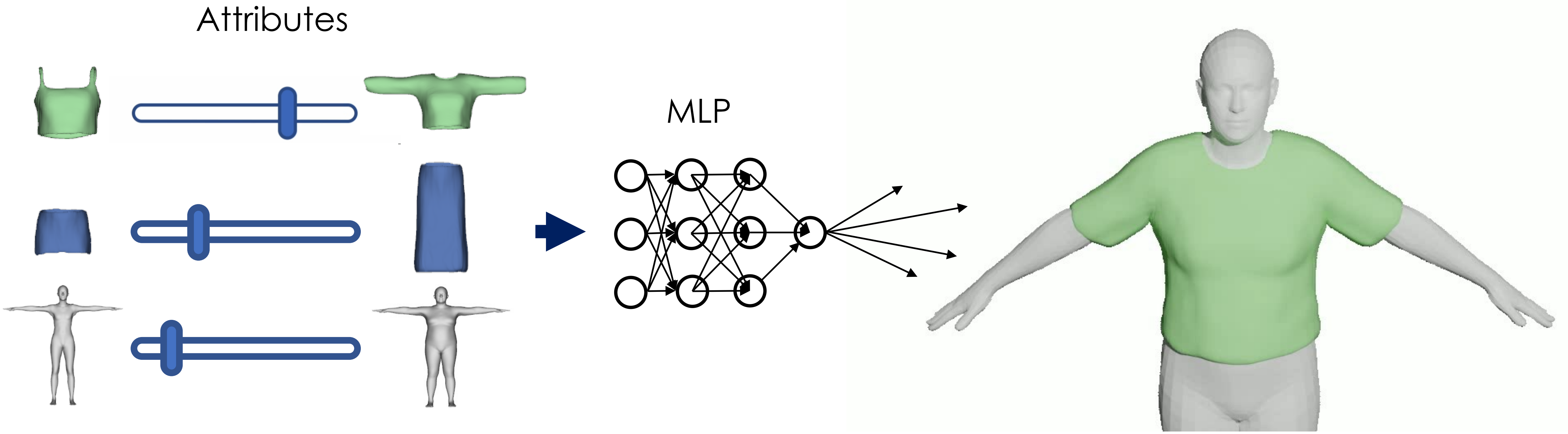
Corona et al., **SMPLicit,** CVPR'21

Moving to new topologies: Implicit representations



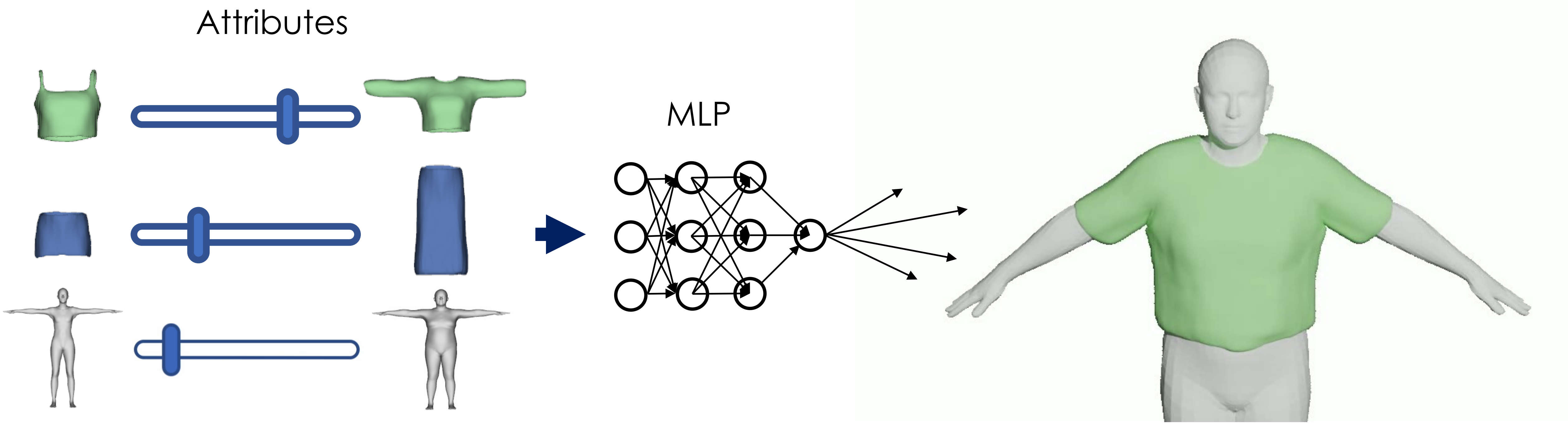
Corona et al., **SMPLicit,** CVPR'21

Dressing humans



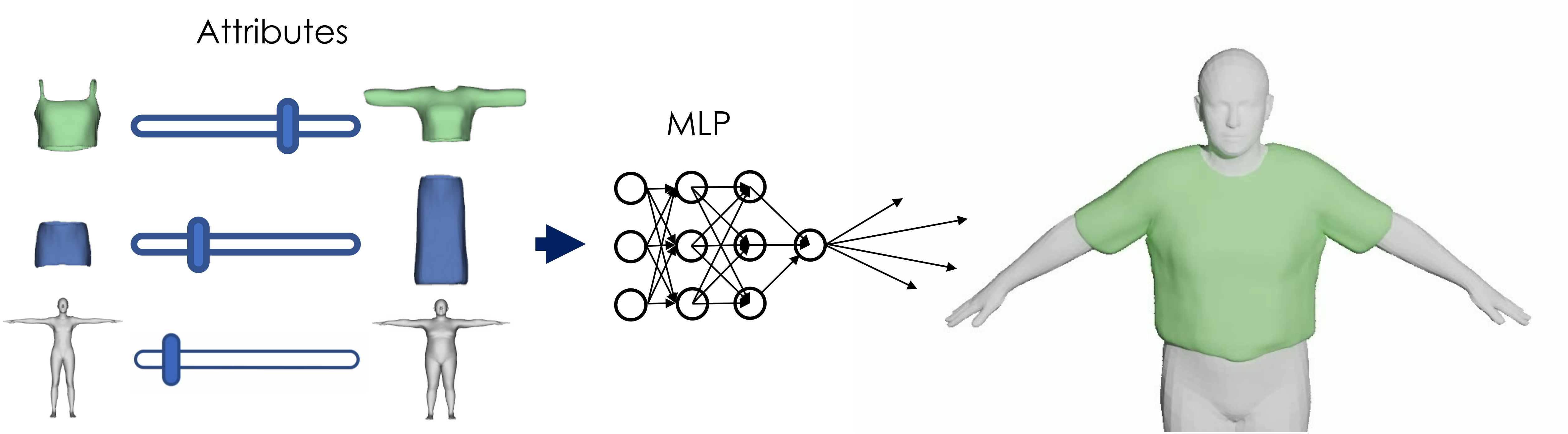
Corona et al., **SMPLicit,** CVPR’21

Dressing humans



Corona et al., **SMPLicit,** CVPR'21

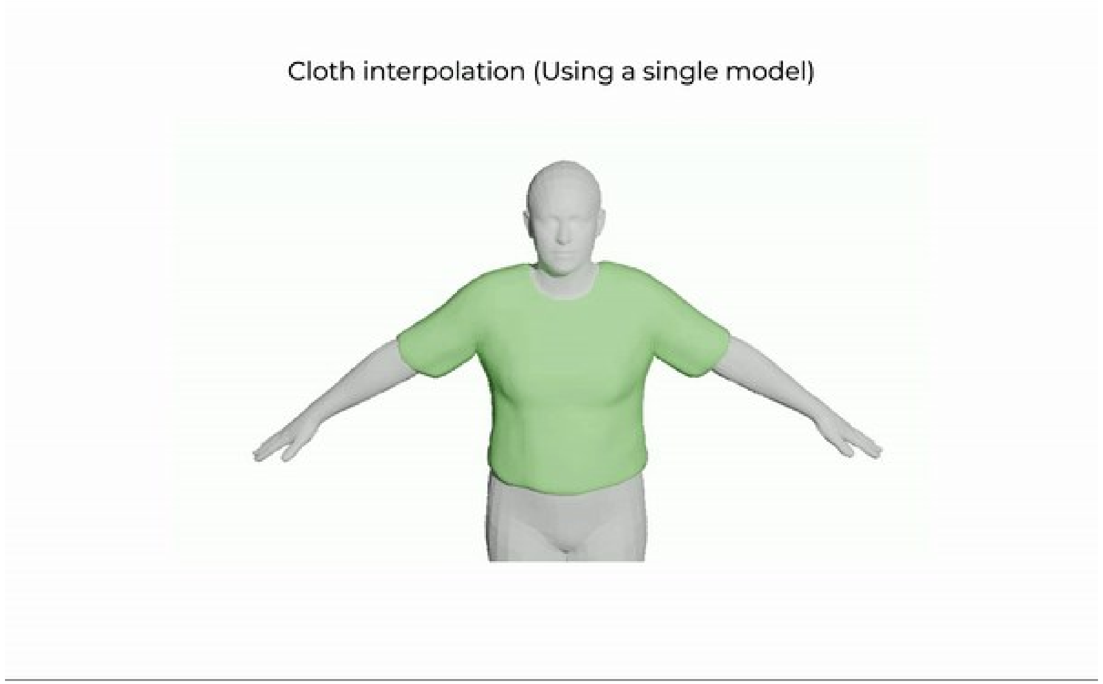
Interpolation in latent space



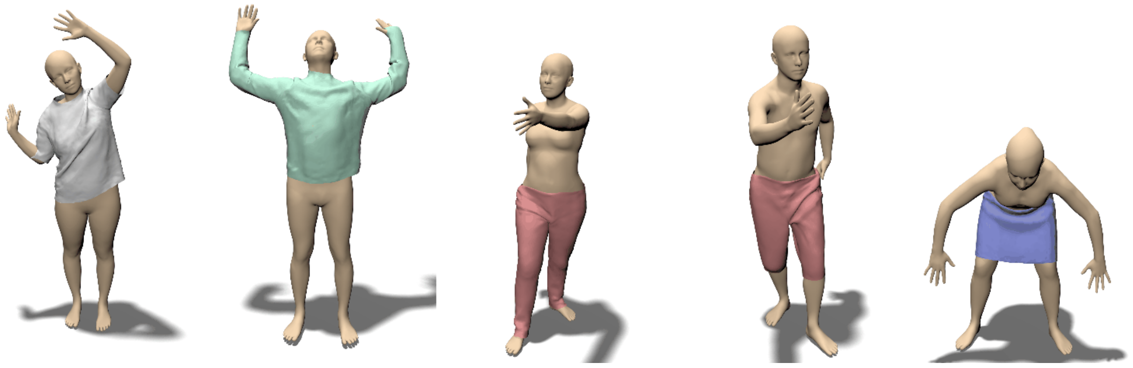
Corona et al., **SMPLicit,** CVPR'21

Key advantages of SMPLicit

Interpolation



Interpolating clothing of different topolgy using single model with SMPLicit

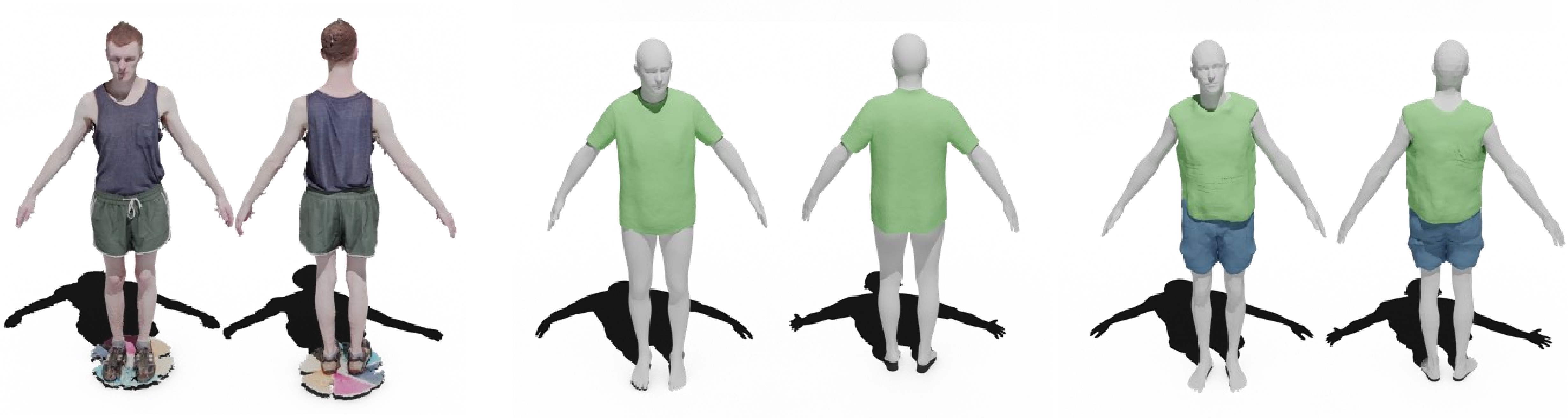


TailorNet(vertex-based model) uses one model/garment type

Corona et al., **SMPLicit,** CVPR'21

Key advantages of SMPLicit

Fitting to scans



Reference person **[TailorNet, CVPR 2020]** **SMPLicit**

✓ Represents multiple topologies in one network

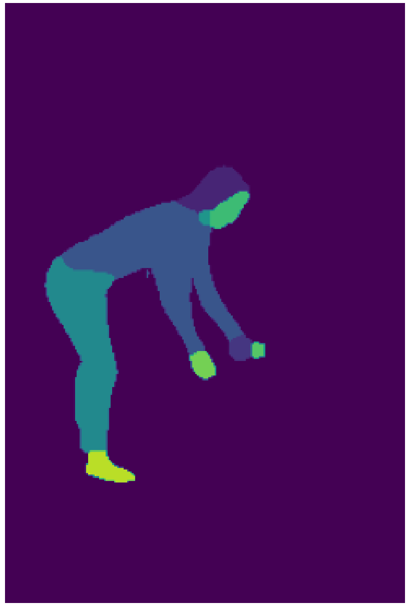
✓ No need to pre-define clothes and train independently per template

Corona et al., **SMPLicit,** CVPR'21

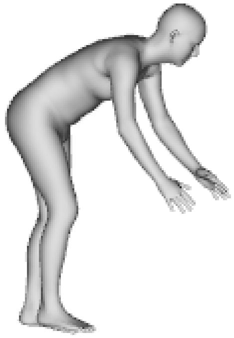
Model fitting with SMPLicit

Fitting SMPLicit by minimizing projection error

Input image



Cloth Segmentation
[RP-R-CNN, ECCV 2020]



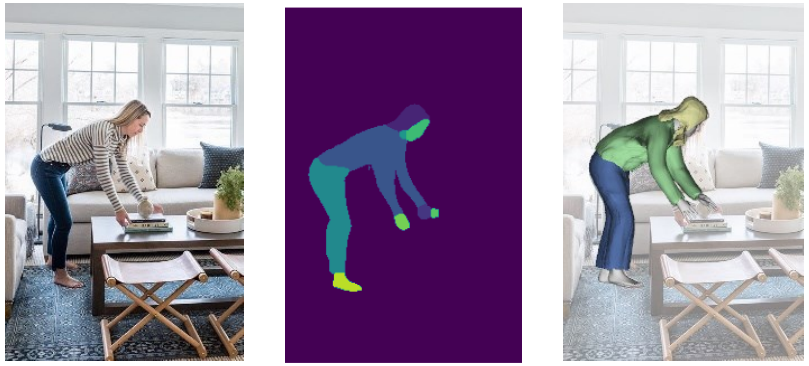
Body Estimation
[FrankMocap, ICCVW 2021]

3D Reconstruction

Semantic Labels

Corona et al., **SMPLicit,** CVPR'21

Model fitting with SMPLICIT



$$\mathcal{L}_I(\mathbf{z}) = \begin{cases} |C(\mathbf{P}_{\boldsymbol{\beta}}, \mathbf{z}) - \mathbf{d}_{\max}|, & \text{if } s_{\mathbf{p}} = 0 \\ \min_i |C(\mathbf{P}^i_{\boldsymbol{\beta}}, \mathbf{z})|, & \text{if } s_{\mathbf{p}} = 1 \end{cases}$$

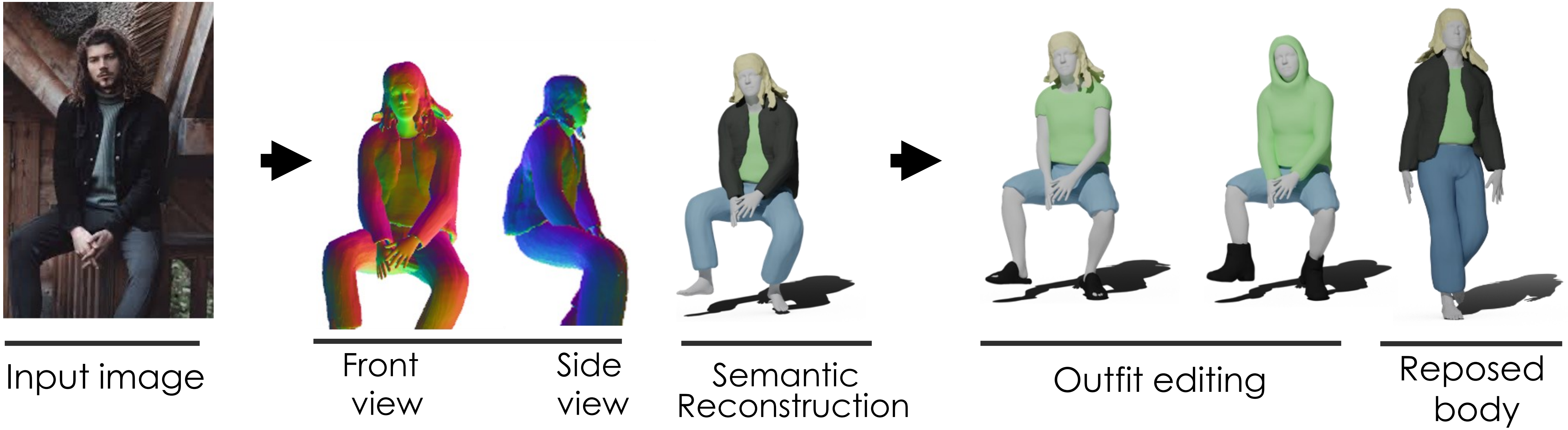
Min over points along the ray

$\mathbf{P}_\beta$ Body relative representation of a sampled point in canonic space

$\mathbf{s}_p = 0$ Point projects outside segmentation mask → force to predict maximum distance or off-surface

$\mathbf{s}_p = 1$ Point projects inside segmentation mask → force to predict 0 distance (on-surface)

Combining the flexibility of implicit representations with the control of explicit parametric models



3D Reconstruction

✓ Represents multiple topologies

✓ Automatic training from a general dataset

✗ High-resolution details

Meshes vs Implicits

1) Parametric Meshes

[Alldieck et al. CVPR'18
Bhatnagar et al. ICCV'19, ECCV'20
Tiwari et al. ECCV'20]

2) Implicit Functions

General objects and humans

[Chibane et al. CVPR'20
Chibane et al. NeurIPS'20]

	Control /Meaning	Topology	Details
1)	✓	✗	✗
2)	✗	✓	✓

2) Compatible with learning

More works on Human modeling using Neural Implicits

- SCANimate, Saito et al. CVPR'21
- LEAP, Mihajlovic et al. CVPR'21
- imGHUM, Alldieck et al. ICCV'21
- MetaAvatar, Weng et al. NeurIPS'21
- ICON, Xiu et al. CVPR'22
- PINA, Dong et al. CVPR'22
- AutoAvatar, Bai et al. ECCV'22

And Many more.....

NeuralFields
https://neuralfields.cs.brown.edu/

PART1: Neural Implicits for 3D Shapes

PART2: Neural Implicits for Humans

PART3: Neural Implicits – Generative Models

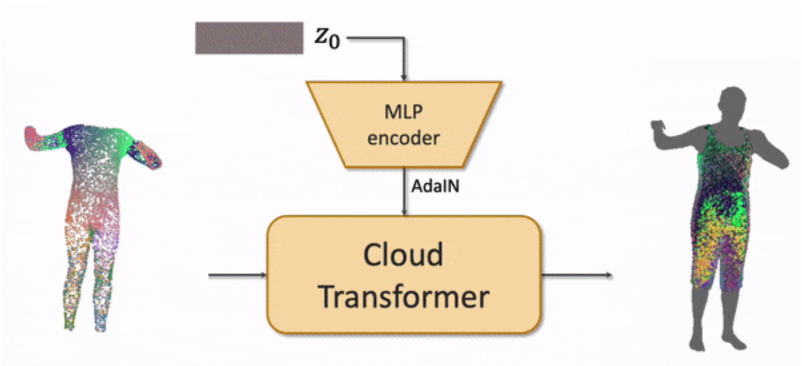
PART4: Point-based Clothing Models

Meshes vs Implicits vs PointClouds

	Control /Meaning	Topology	Details	Speed	Continuous
1) Meshes	✓	✗	✗	✓	✗
2) Implicits	✓	✓	✓	✗	✓
3) PointClouds	?	✓	?	✓	?

Human and clothing model using PointClouds

Ma et al. ICCV'21



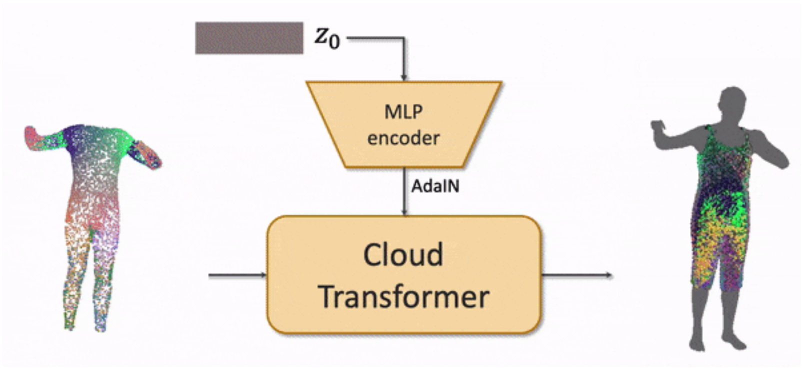
Zakharkin et al. ICCV'21

Ma et al. 3DV'22

- High fidelity
- Flexible topology
- Pose/Shape/Style controllable
- Learned directly from scans
- Fast rendering

Human and clothing model using PointClouds

Ma et al. ICCV'21



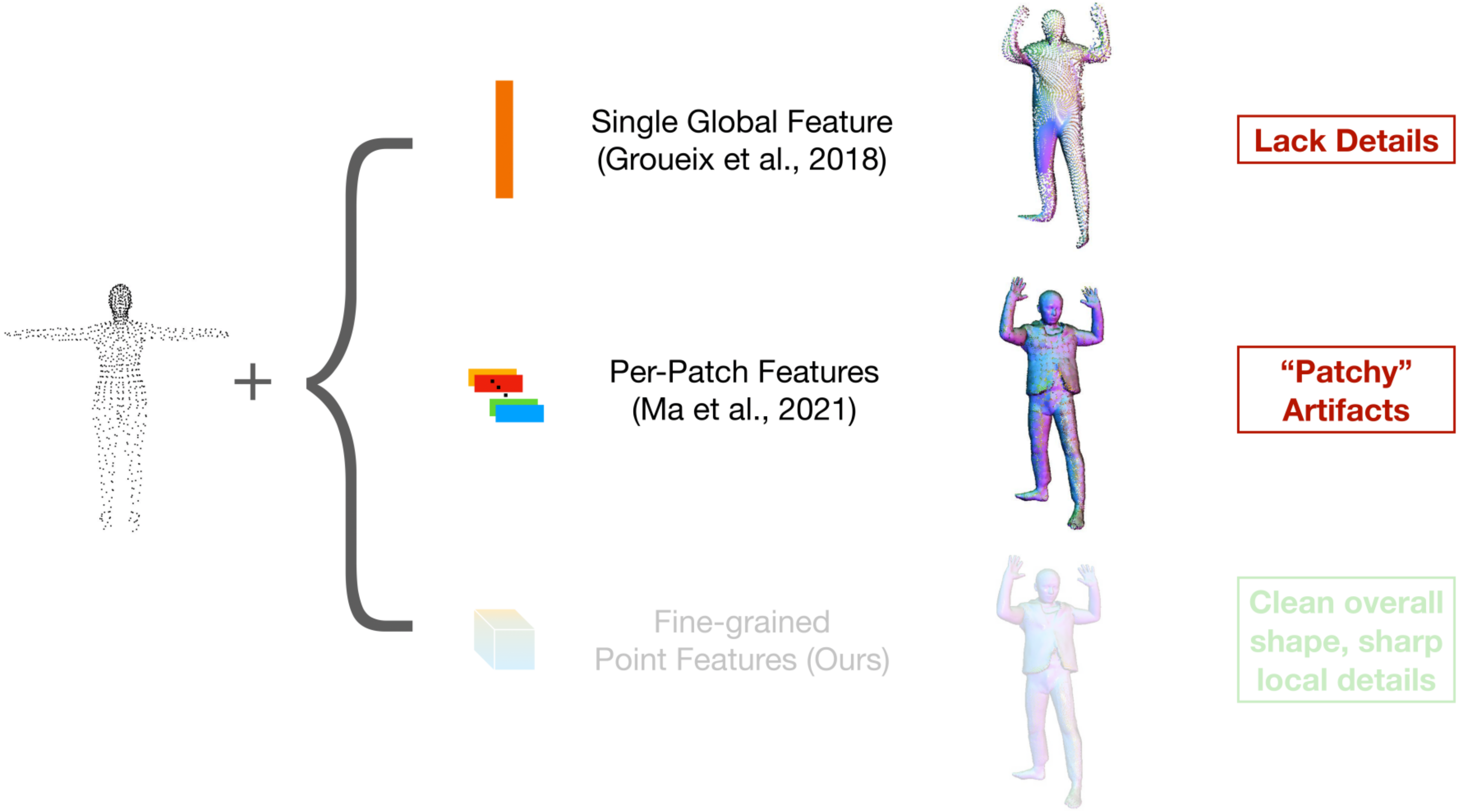
Zakharkin et al. ICCV'21

Ma et al. 3DV'22

- High fidelity
- Flexible topology
- Pose/Shape/Style controllable
- Learned directly from scans
- <u>Fast rendering</u>

The Power of Points for Modeling Humans in Clothing

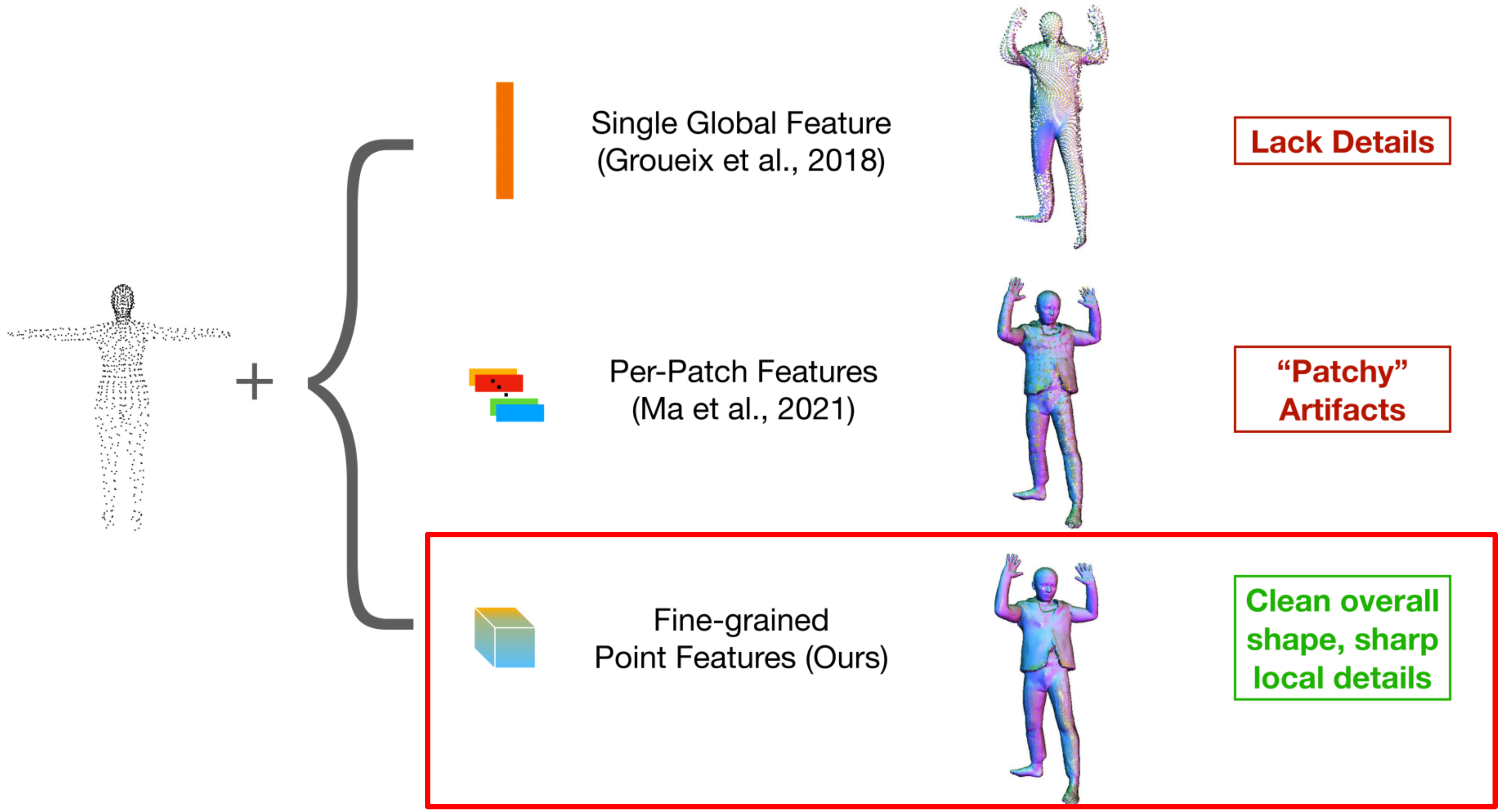
Using pointcloud for humans/clothing



Ma et al. ICCV'21

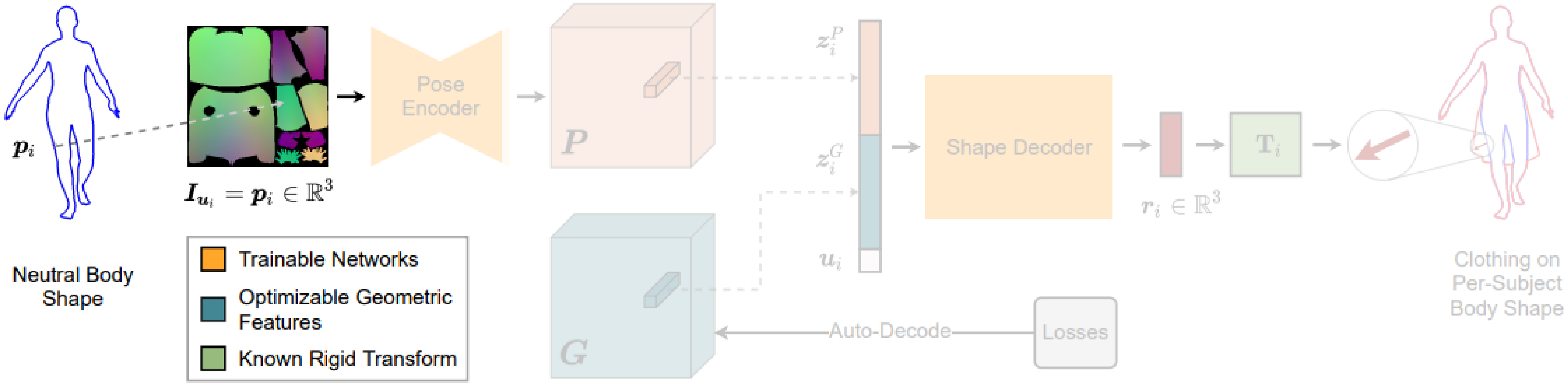
How is this different from prior Point-based works?

Using pointcloud for humans/clothing



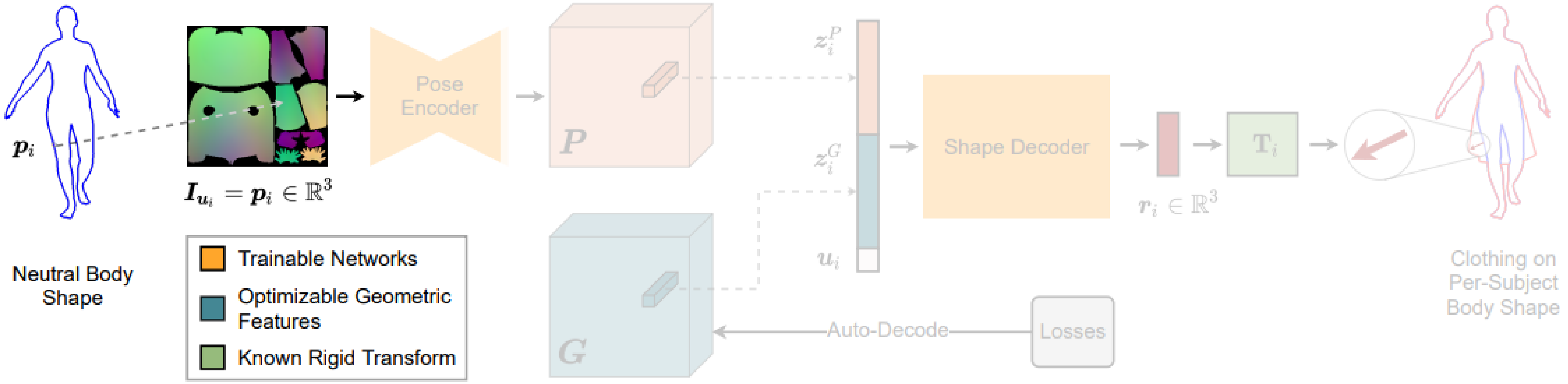
Ma et al. ICCV'21

The Power of Points for Modeling Humans in Clothing



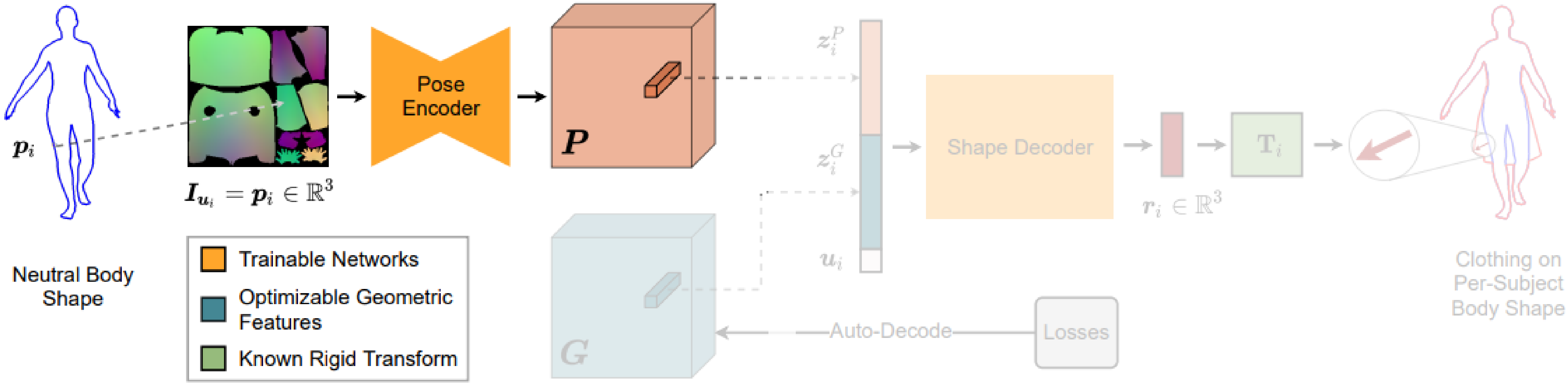
Ma et al. ICCV'21

The Power of Points for Modeling Humans in Clothing



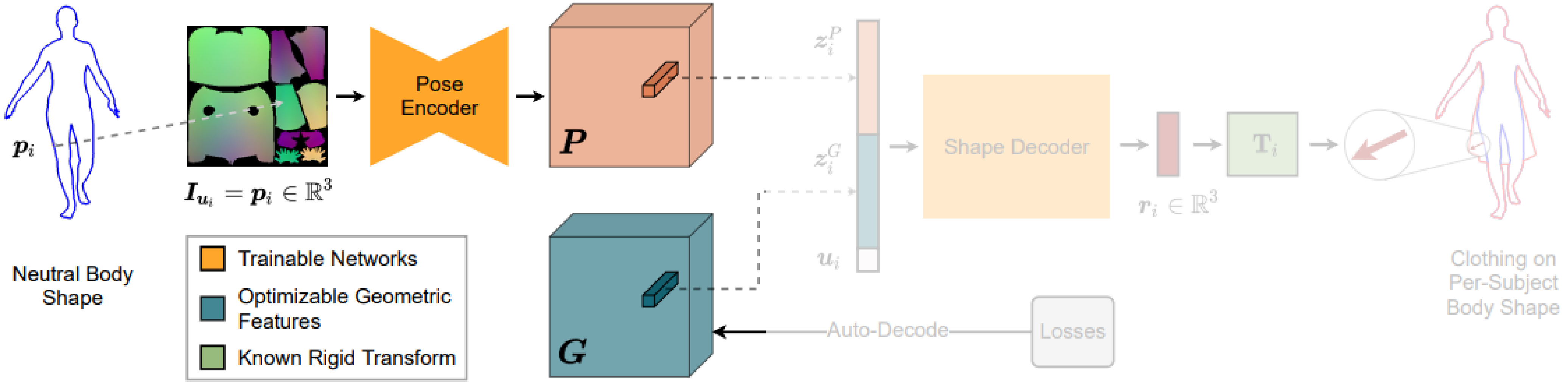
Ma et al. ICCV'21

The Power of Points for Modeling Humans in Clothing



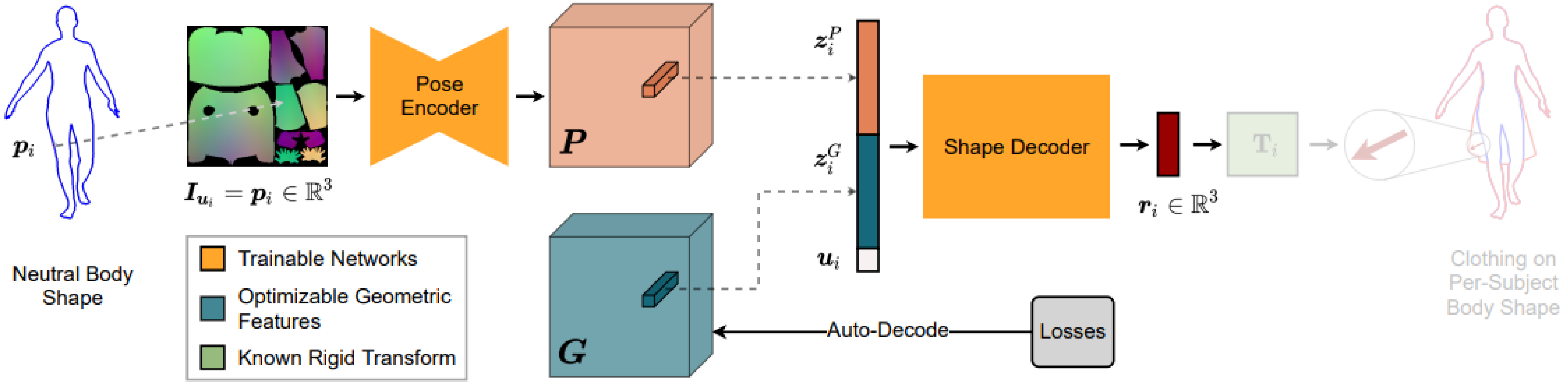
Ma et al. ICCV'21

The Power of Points for Modeling Humans in Clothing



Ma et al. ICCV'21

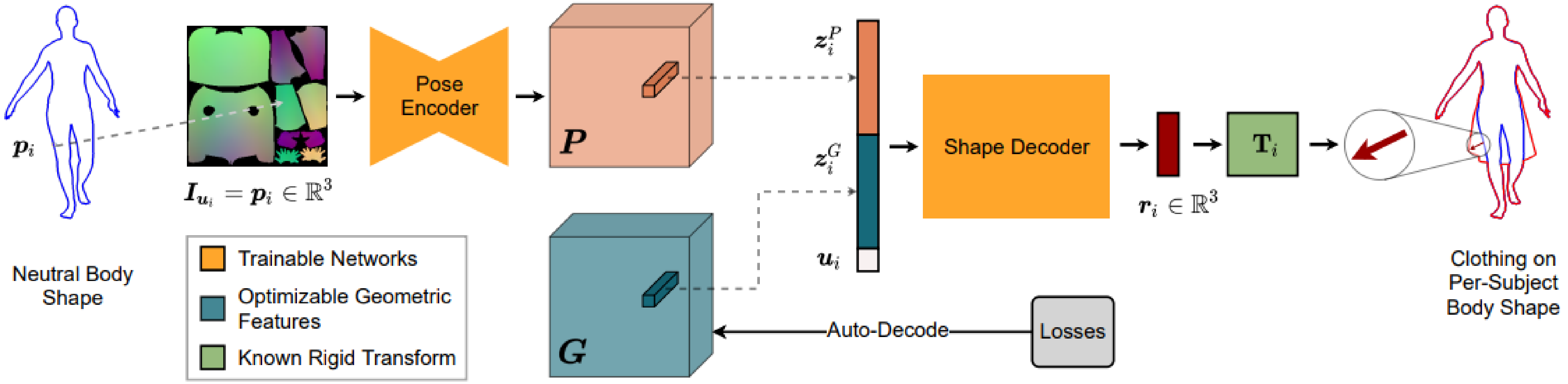
The Power of Points for Modeling Humans in Clothing



$$r_i = f_{\mathbf{w}}(u_i; z_i) : \mathbb{R}^2 \times \mathbb{R}^Z \rightarrow \mathbb{R}^3,$$

Ma et al. ICCV'21

The Power of Points for Modeling Humans in Clothing

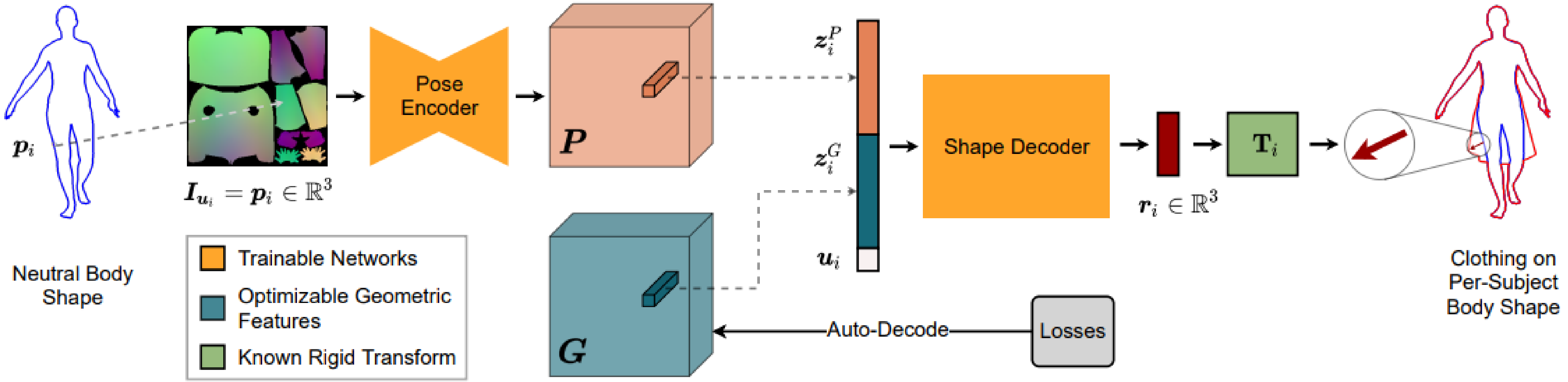


$r_i = f_{\mathbf{w}}(u_i; z_i) : \mathbb{R}^2 \times \mathbb{R}^Z \to \mathbb{R}^3,$ Displacement vector

$\mathbf{x}_i = \mathbf{T}_i \cdot r_i + p_i,$ Point in posed space

Ma et al. ICCV'21

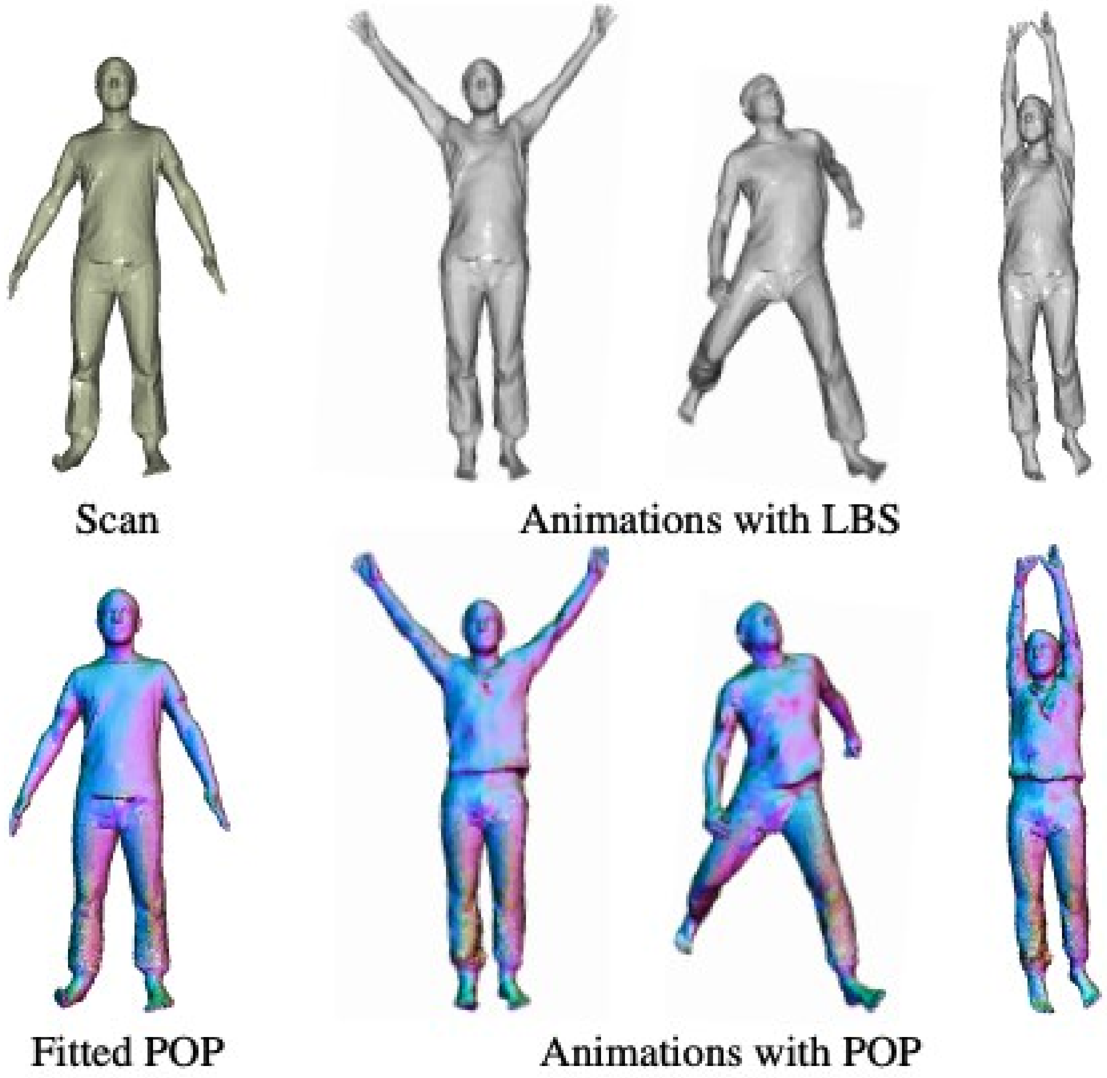
The Power of Points for Modeling Humans in Clothing



Ma et al. ICCV'21

The Power of Points for Modeling Humans in Clothing

Results: POP produces high-quality and fine-detailed results than LBS

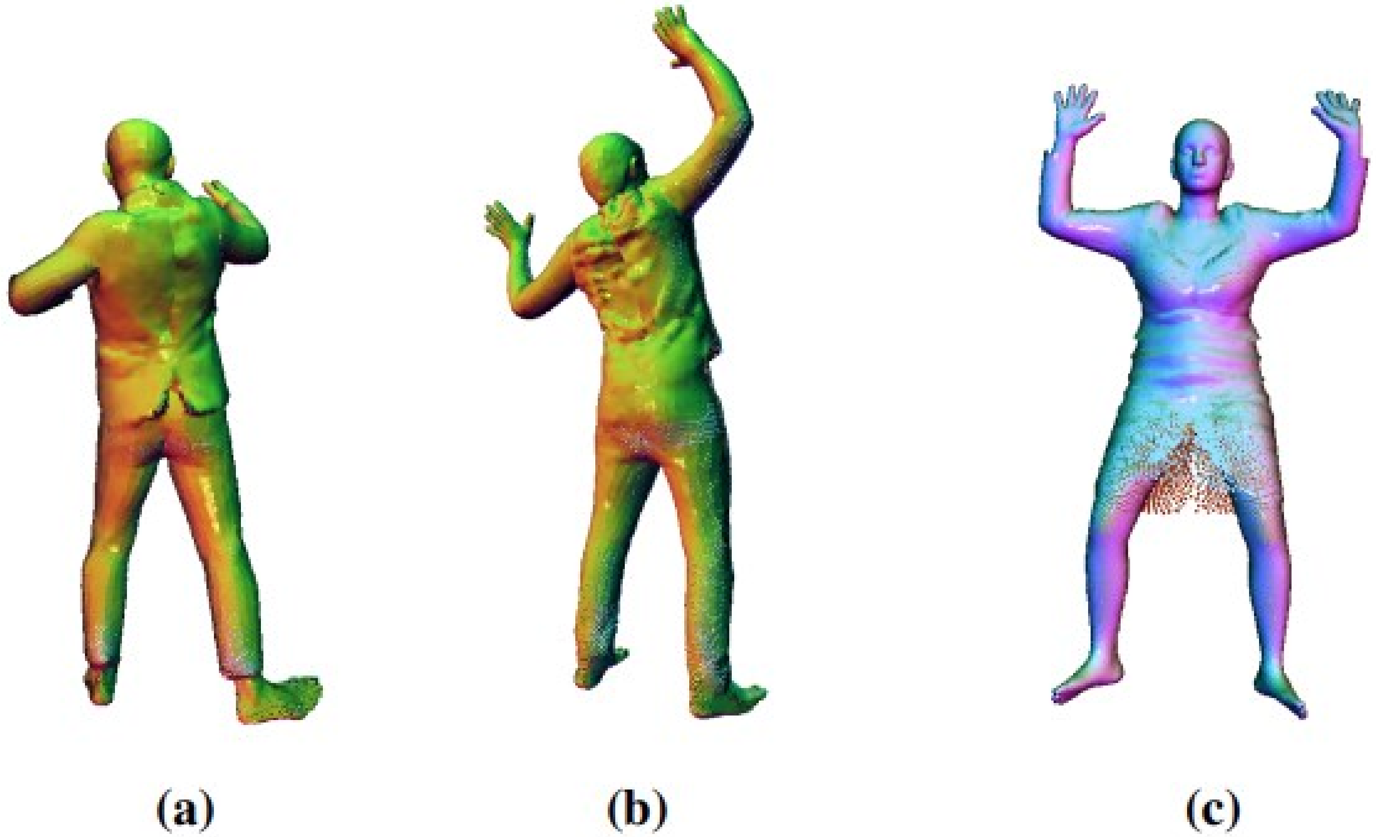


The Power of Points for Modeling Humans in Clothing

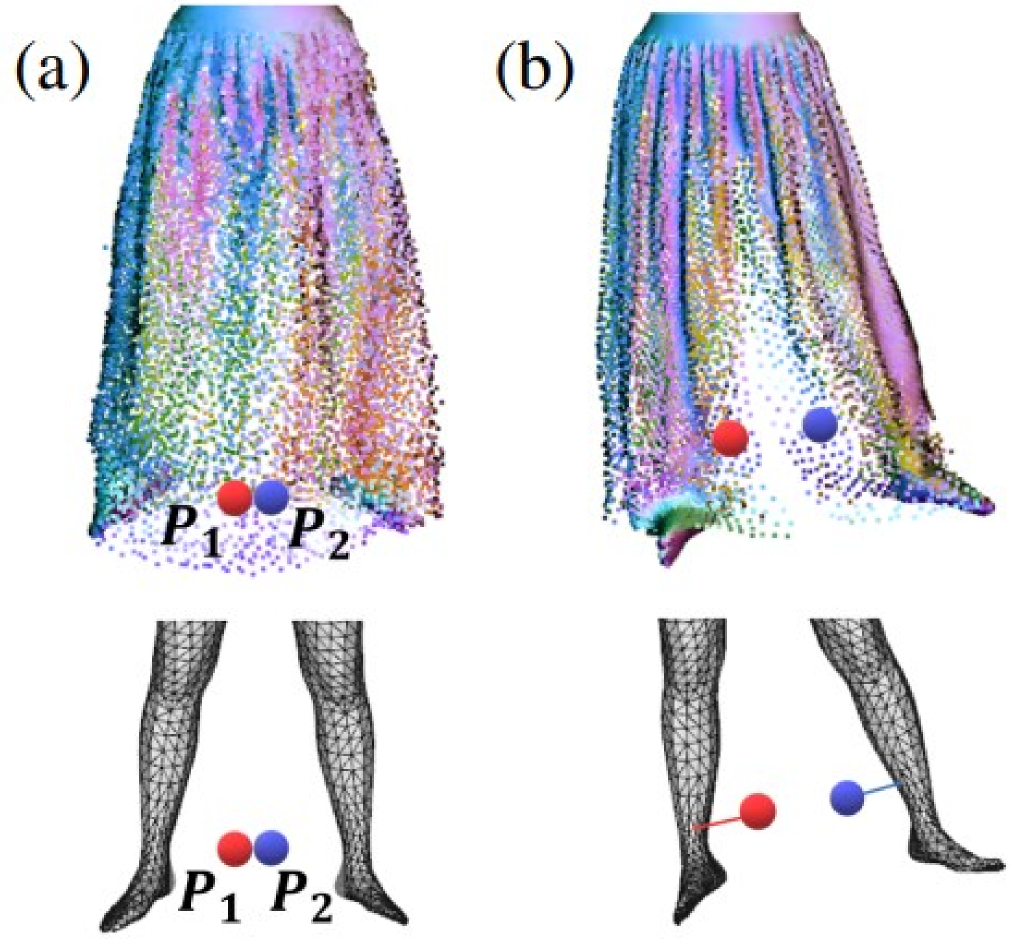
Limitations:

Discontinuity due to SMPL UV maps

- This results in visible “seams” between certain body parts.
- More significant for skirts.



Discontinuity in POP



POP

Ma et al., 3DV 2022

Meshes vs Implicits vs PointClouds

	Control /Meaning	Topology	Details	Speed	Continuous
1) Meshes	✓	✗	✗	✓	✗
2) Implicits	✓	✓	✓	✗	✓
3) PointClouds	✓	✓	✓	✓	✗

Slides credit and resources

Thanks to
Julian Chibane, Enric Corona and Qianli Ma
for providing materials.

[TUM AI Lecture Series - Neural Implicit Representations for 3D Vision]

(talk by Prof. Pons-Moll)