



# Occupancy Flow

## 4D Reconstruction by Learning Particle Dynamics

Michael Niemeyer<sup>1,2</sup> Lars Mescheder<sup>1,2</sup> Michael Oechsle<sup>1,2,3</sup> Andreas Geiger<sup>1,2</sup>

<sup>1</sup>MPI for Intelligent Systems <sup>2</sup>University of Tübingen <sup>3</sup>ETAS GmbH, Stuttgart

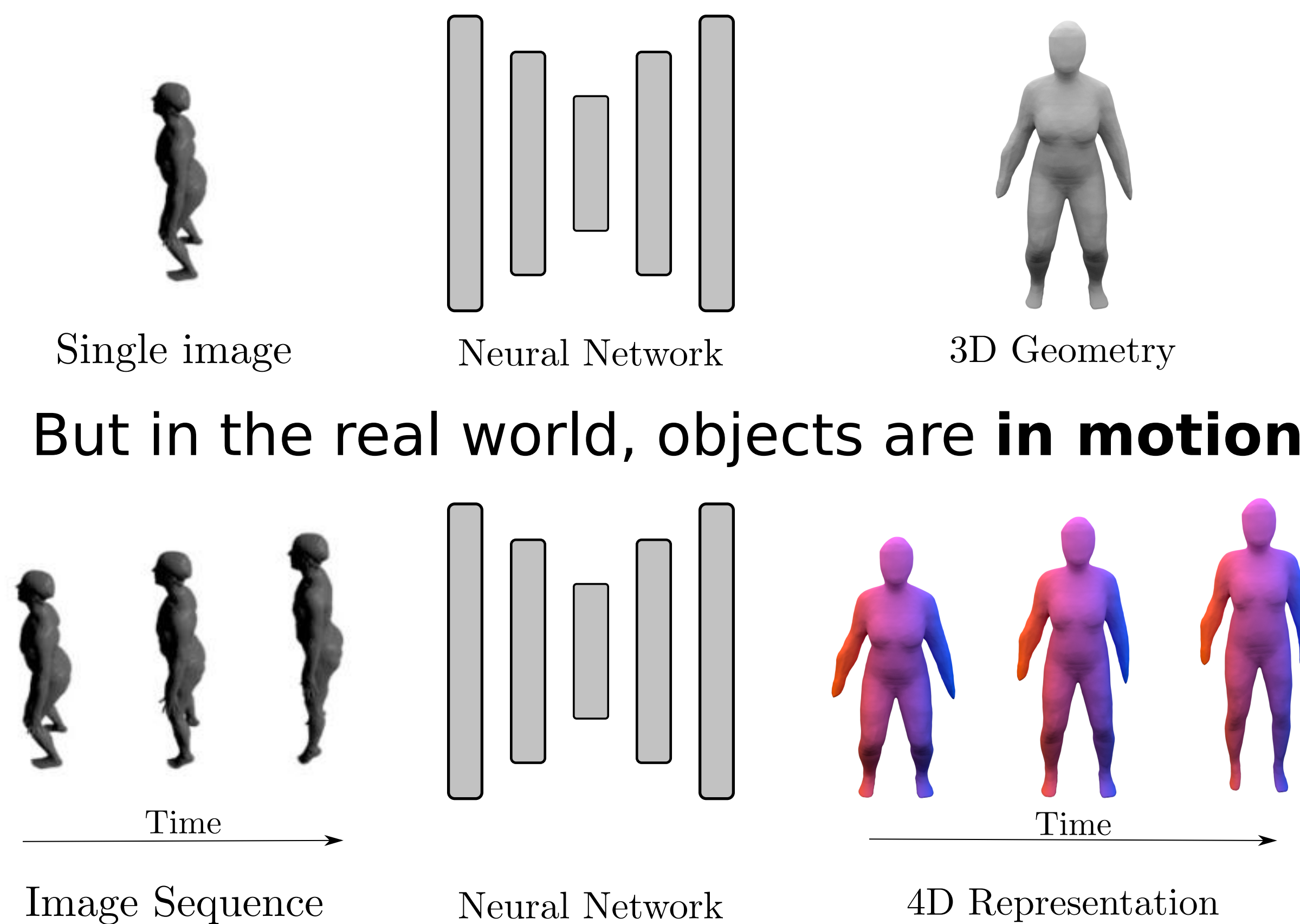


### Motivation

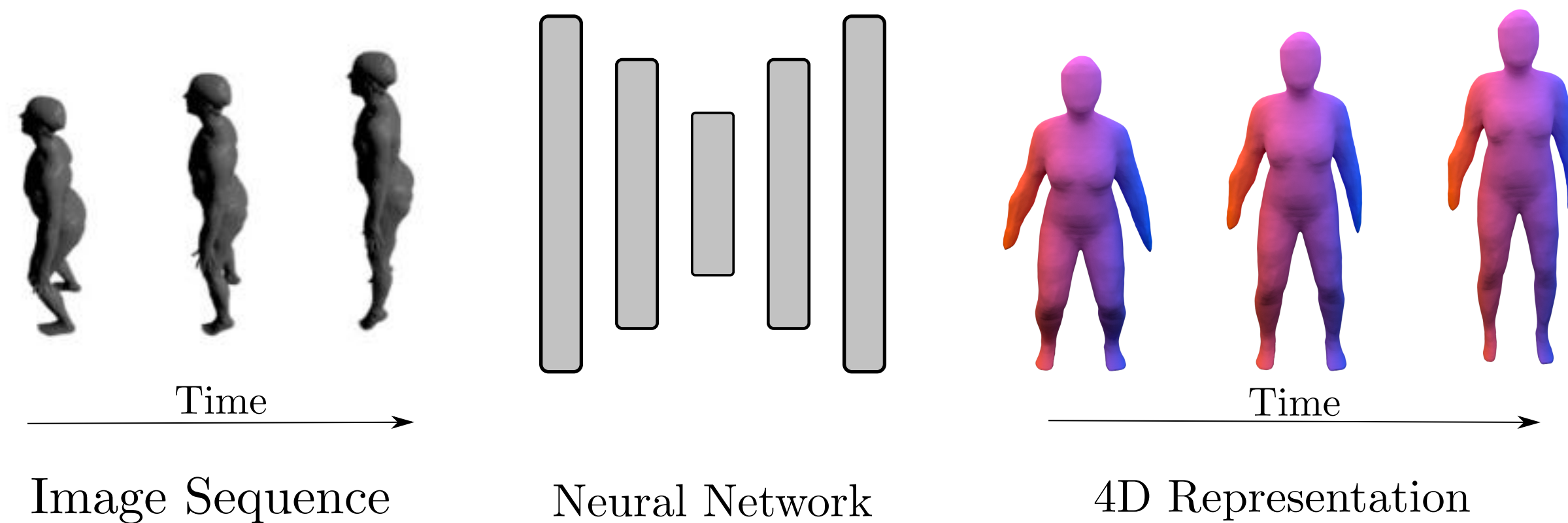
- Our world is full of **3D objects in motion**
- To act and reason autonomously, machines need an adequate **model of time-varying 3D geometry**
- Inferring such a model from **sparse sensory inputs** requires **knowledge of the world**
- Can we find a **4D representation** which can be **learned from observations**?

### Learning-based 3D Reconstruction

- Successful because **rich prior knowledge** can be used, e.g. to resolve ambiguities



- But in the real world, objects are **in motion**

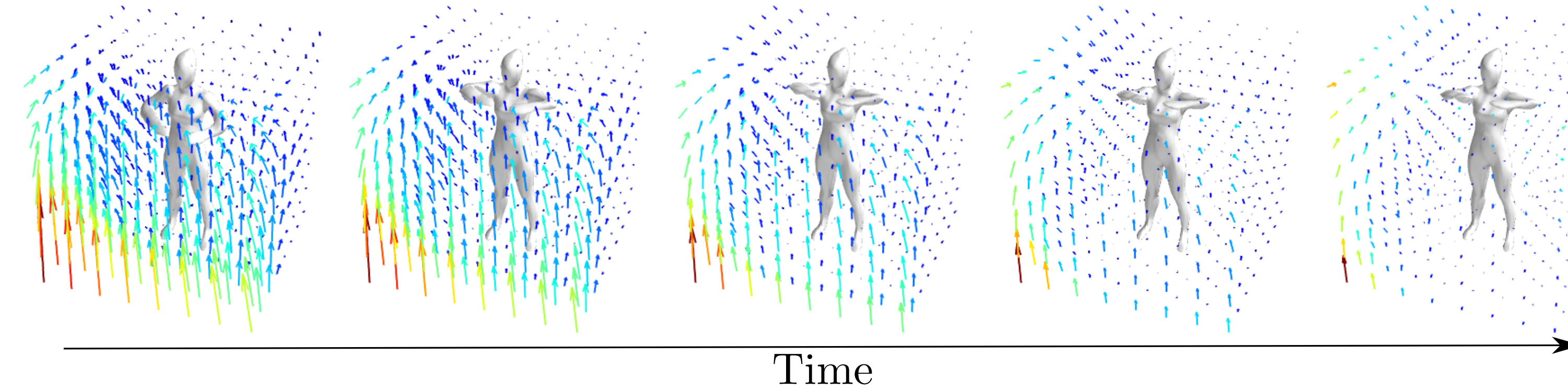


### How can we extend the 3D models to 4D?

- Naïvely discretizing the temporal domain leads to
  - Sparsity in time
  - No correspondences
  - Slow inference

### Our Representation

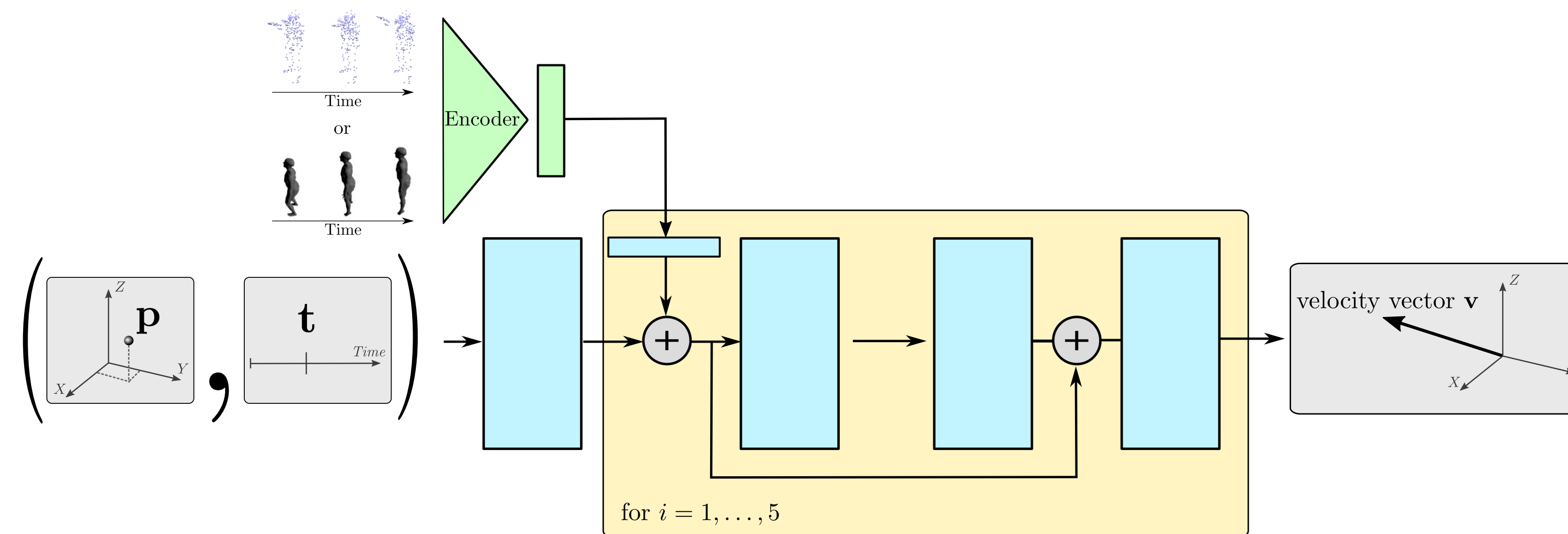
Represent **motion** by a temporally and spatially **continuous vector field**  $v_\theta : \mathbb{R}^3 \times [0, T] \rightarrow \mathbb{R}^3$



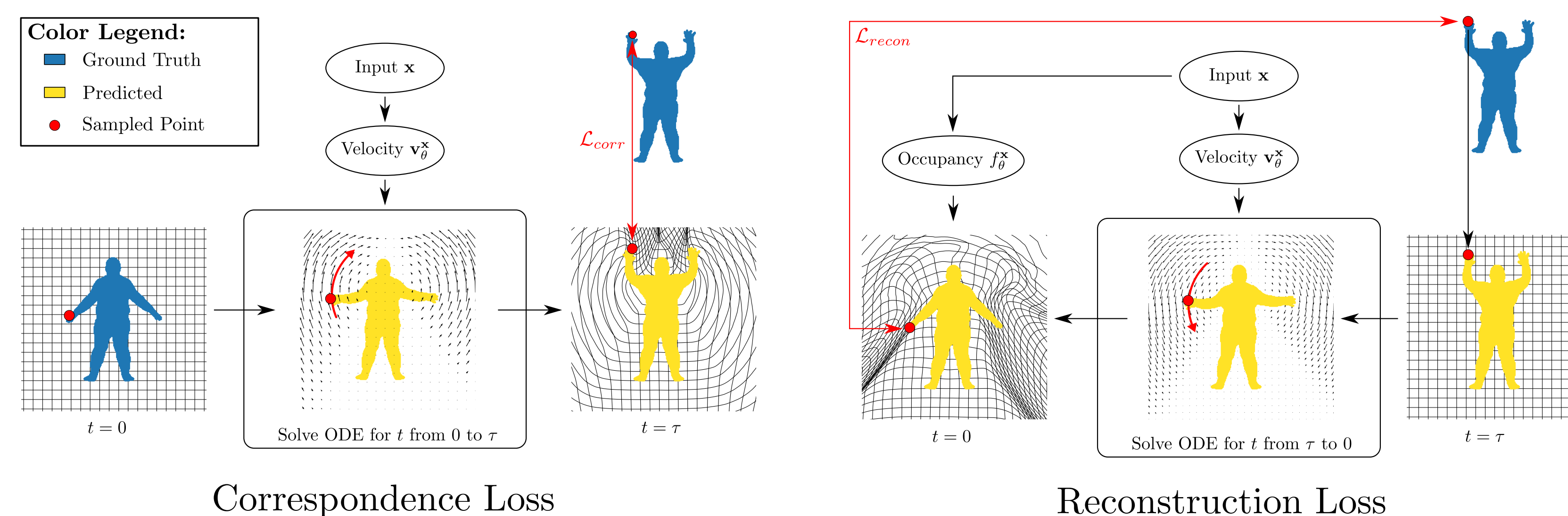
Represent **3D shape** at time 0 as the **continuous decision boundary** of a binary classifier  $f_\theta : \mathbb{R}^3 \rightarrow [0, 1]$

- + Spatially and temporally continuous
- + Implicit correspondences over time
- + Fast inference

### Velocity Field Architecture

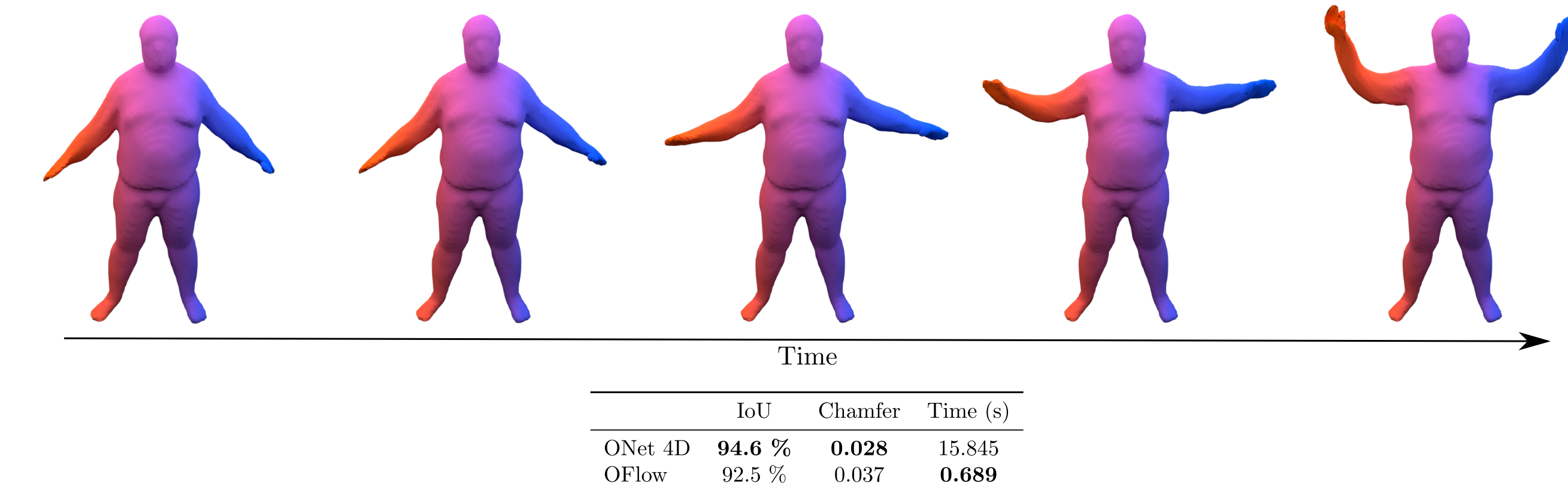


### Loss Formulation

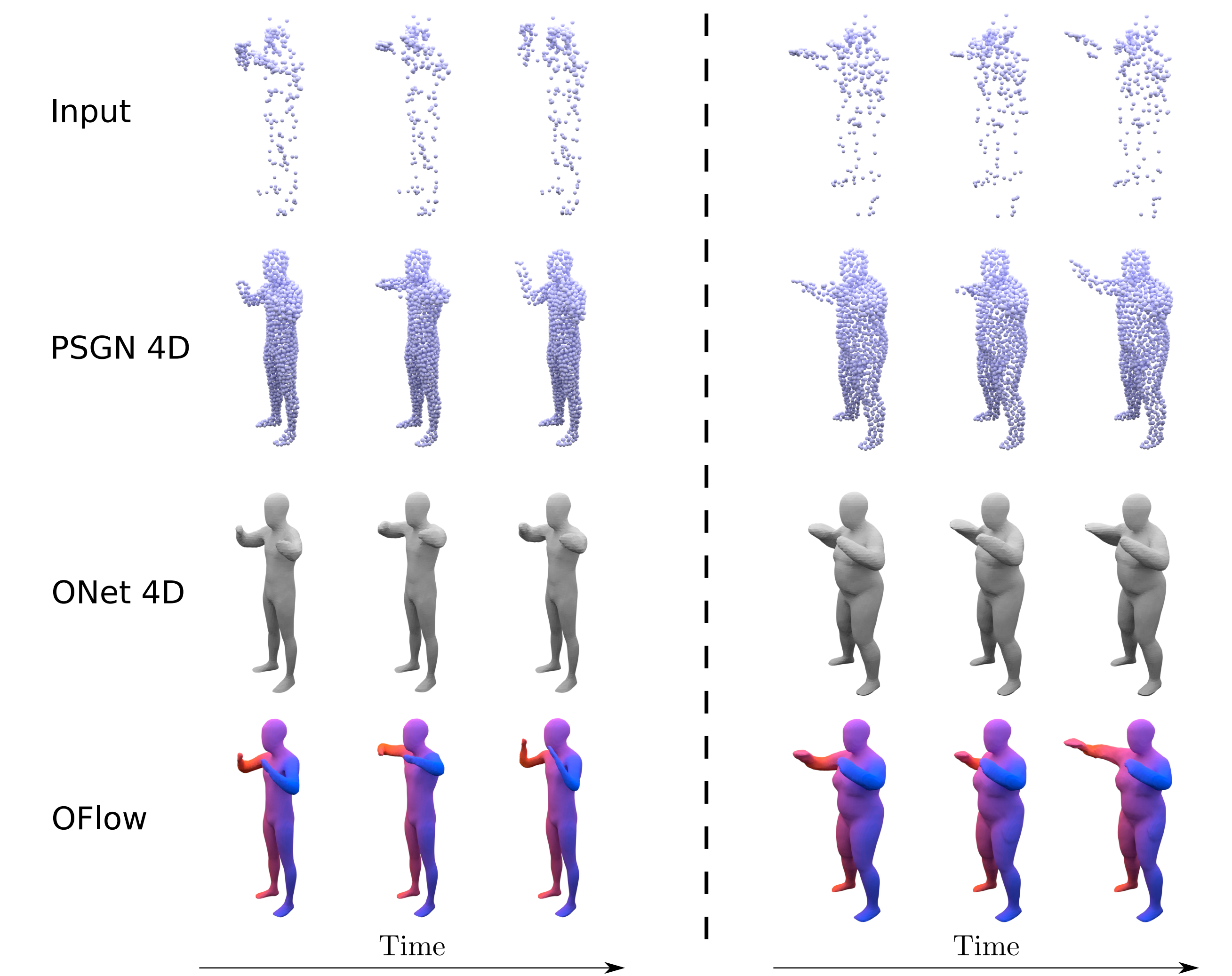


### Experiments

#### Representation Power



#### 4D Point Cloud Completion



	IoU	Chamfer	Correspond.
PSGN 4D	-	0.258	2.576
PSGN 4D (w/ cor.)	-	0.265	2.580
ONet 4D	44.0 %	0.348	-
OFlow	56.6 %	0.193	0.292
OFlow (w/ cor.)	59.6 %	0.166	0.226

	IoU	Chamfer	Correspond.
PSGN 4D	-	0.108	3.234
PSGN 4D (w/ cor.)	-	0.101	0.102
ONet 4D	77.9 %	0.084	-
OFlow	79.1 %	0.077	0.129
OFlow (w/ cor.)	81.2 %	0.066	0.096

4D Reconstruction from Image Sequences (D-FAUST)

4D Point Cloud Completion (D-FAUST)

#### Shape Interpolation

