Motion2VecSets: 4D Latent Vector Set Diffusion for Non-rigid Shape Reconstruction and Tracking – Supplementaty Material –

Wei Cao[1*‡] Chang Luo[1*] Biao Zhang[2] Matthias Nießner[1] Jiapeng Tang[1†]

[1]Technical University of Munich [2]King Abdullah University of Science and Technology

In this supplementary material, we complement our main paper with additional details. It begins with an introduction to the notations used in our paper in Sec. 1. The follow wing section, Sec. 2, discusses our network architectures, highlighting key design choices and their impact. This is complemented by Sec. 3, which provides essential implementation details. Then, we provide more comparisons between our approach and state-of-the-art methods in Sec. 4, offering insights into our model's strengths and improvements. We also provide an additional ablation study to verify the usefulness of latent set diffusion on the setting of 4D shape reconstruction from sparse and noisy point clouds. Finally, in Sec. 5, we demonstrate the real-world applicability of our model, underscoring its practical effectiveness.

1. Notations

In this paper, we use the following notations, as summarized in Tab. 1. The symbol T represents the sequence length, with the superscript t indicating the time step of a frame in the sequence. Our network's input is a sparse or partial point cloud, denoted as $\mathcal{P} = \{\mathbf{P}^t\}_{t=0}^{T-1}$, where each frame $\mathbf{P}^t$ comprises L points, serving as conditions in reference time. The sets $\mathcal{S}, \mathcal{D}, \mathcal{C}$ represent shape codes, deformation codes, and conditional codes, respectively, with each i-th code denoted by the corresponding lowercase letter. The mesh is symbolized by $\mathcal{M} = \{\mathcal{V}, \mathcal{F}\}$, where $\mathcal{V}$ and $\mathcal{F}$ correspond to the vertices and faces set of the mesh. Furthermore, $\mathbf{X}$ refers to surface points from ground truth meshes for learning shape, and $\mathbf{X}^d$ refers to corresponding sampled points by FPS. We dedicate $\mathbf{X}_{\text{src}}, \mathbf{X}_{\text{tgt}}$ for representing the sampled surface points for learning deformation, and $\mathbf{X}^d_{\text{src}}, \mathbf{X}^d_{\text{tgt}}$ refer to sampled points by FPS. Moreover, $\mathcal{Q}$ represents the set of query points, where $\mathbf{q}$ refers to a point inside the point set, and $\mathbf{q}'$ is the output by feeding the query point into the deformation network.

*Equal Contribution.
†Corresponding author.
‡Work done during master's thesis.

2. Network Architectures

The inputs of our model are a sequence of T frames of sparse or partial noisy point clouds, represented by $\mathcal{P} = \{\mathbf{P}^t\}_{t=1}^{T}$, where $\mathbf{P}^t = \{\mathbf{p}_i \in \mathbb{R}^3\}_{i=1}^{L}$, L represents the number of points. The goal is to reconstruct continuous 3D meshes with high fidelity, denoted as $\{\mathcal{M}^t\}_{t=1}^{T} = \{\mathcal{V}^t, \mathcal{F}^t\}_{t=1}^{T}$, where $\mathcal{V}^t$ and $\mathcal{F}^t$ refer to the set of vertices faces of the reconstructed mesh at time frame t.

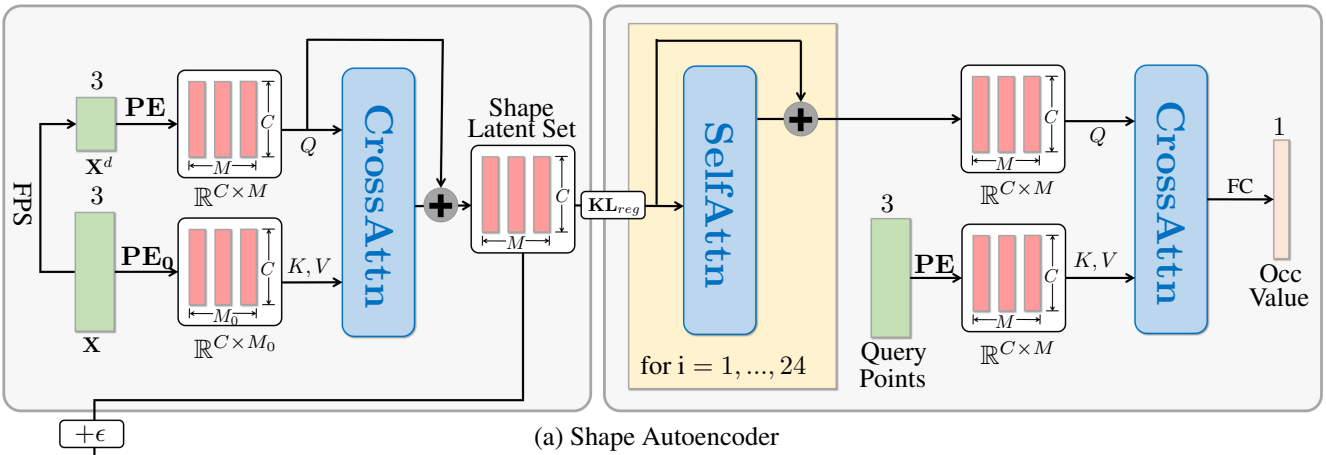
2.1. Shape Diffusion

In the shape diffusion part, we leverage the first frame of the sequence, $\mathbf{P}^1$, to reconstruct the object shape. As illustrated in Fig. 1, this process is divided into two distinct networks: (a) *Shape Autoencoder Network* and (b) *Shape Vector Set Diffusion Network*.

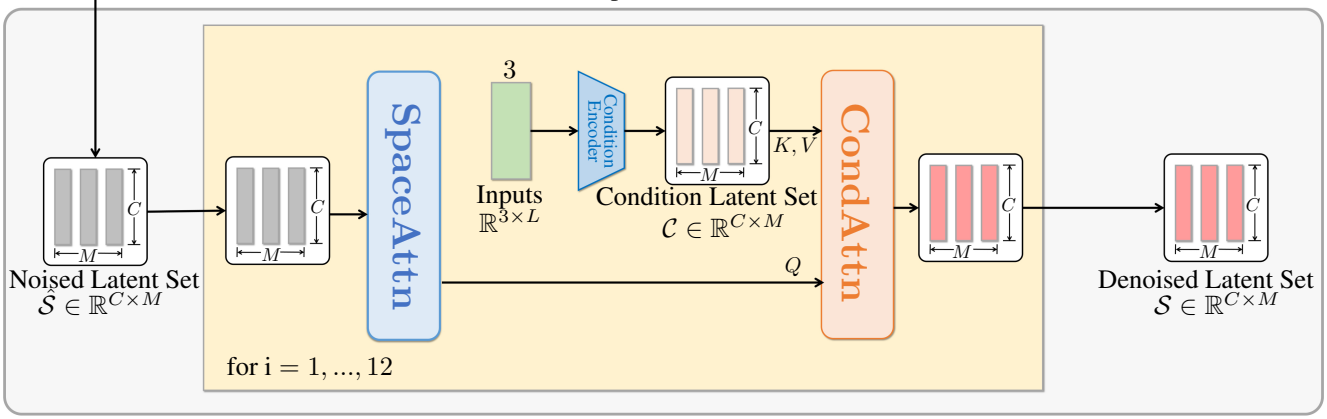
Shape Autoencoder To optimize computational efficiency, we adopt the furthest point sampling (FPS) technique. This method pinpoints crucial points within a point cloud, thereby thinning its density.:

$$\mathbf{X}^d = \text{FPS}(\mathbf{X}) \tag{1}$$

Subsequently, a cross-attention block, designed to compute attention weights across various points, is employed. This block fuses the features of the subsampled points and generates a shape latent set, denoted as $\mathcal{S} = \{\mathbf{s}_i \in \mathbb{R}^C\}_{i=1}^{M}$. Here, M represents the overall count of codes and C denotes their dimensionality. In this process, the positional embeddings derived from the points after FPS sampling are utilized as the query, whereas those obtained before FPS sampling serve as the key and value in the attention mechanism. Also, consistent with the latent diffusion framework proposed by [8], our model incorporates KL-regularization within the latent space. This regularization strategy plays a crucial role in modulating feature diversity, ensuring the preservation of high-level features. The query points are encoded and passed to the cross-attention block with the generated shape code. The resulting fused code is then mapped



(a) Shape Autoencoder



(b) The denoiser network of Shape Vector Set Diffusion

Figure 1. Network architecture of **Shape Diffusion**

to a dimension of 1 via a fully connected (FC) layer, providing the predicted occupancy value for the query points.

Shape Vector Set Diffusion Nosied shape codes $\hat{\mathcal{S}}$ are sent to the denoising neural network. The denoiser consists of two blocks: The space-attention block facilitates positional information exchange among M codes in different positions, while the condition-attention block injects information from sparse or partial points (conditional input). After repeating this process, we get the denoised latent set $\mathcal{S}$.

2.2. Synchronized Deformation Diffusion

As shown in Fig. 2, the deformation diffusion also contains two parts: *Deformation Autoencoder Network* and *Synchronized Deformation Vector Set Diffusion Network*.

Deformation Autoencoder In the deformation autoencoder, both surface and near-surface points of different frames are sampled according to the same face indexes, which ensures the correspondence within a sequence. These point cloud frames are pairwise paired together, each with a source point cloud $\mathbf{X}_{\text{src}}$ and a target point cloud $\mathbf{X}_{\text{tgt}}$.

Similarly to shape diffusion, to ensure the correspondence between the source point cloud and the target point cloud, we use the same FPS for downsampling. We have:

$$\mathbf{X}^d_{\text{src}} = \text{FPS}(\mathbf{X}_{\text{src}}) \tag{2}$$

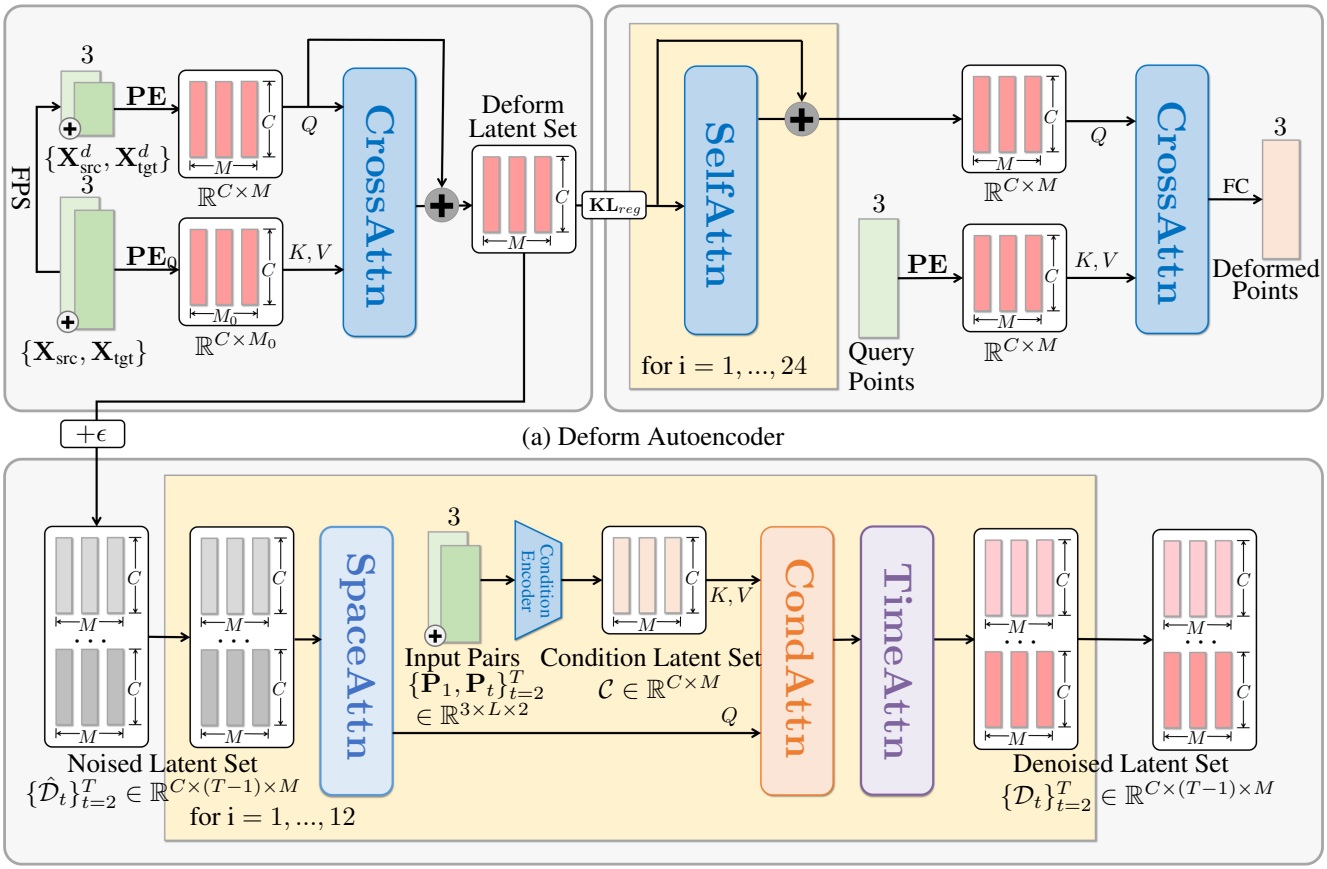
$$\mathbf{X}^d_{\text{tgt}} = \text{FPS}(\mathbf{X}_{\text{tgt}}) \tag{3}$$

As shown in Fig. 2 (a), after obtaining the key points of the source and target point clouds. Positional embeddings of the FPS downsampled and original point clouds are concatenated along the last dimension to preserve spatial consistency, where PosEmb: : $\mathbb{R}^3 \rightarrow \mathbb{R}^M$ refers to positional embedding functions:

$$\mathbf{PE} = \text{Concate}\left(\text{PosEmb}(\mathbf{X}_{\text{src}}), \text{PosEmb}(\mathbf{X}_{\text{tgt}})\right) \tag{4}$$

$$\mathbf{PE}^d = \text{Concate}\left(\text{PosEmb}(\mathbf{X}^d_{\text{src}}), \text{PosEmb}(\mathbf{X}^d_{\text{tgt}})\right) \tag{5}$$

We employ the cross attention $\text{CrossAttn}(Q, KV)$ throughout our method, where Q denotes the query, and KV denotes the key-value pair. Similarly, we use KL-divergence to retain high-level features to facilitate the



(a) Deform Autoencoder

(b) The denoiser network of Synchronized Deformation Vector Sets Diffusion

Figure 2. Network architecture of **Synchronized Deformation Diffusion.**

learning of the diffusion model. Here we get the deformation latent set $\mathcal{D} = \{\mathbf{d}_i \in \mathbb{R}^C\}_{i=1}^M$:

$$\mathcal{D}(\mathbf{X}_{\text{src}}, \mathbf{X}_{\text{tgt}}) = \text{CrossAttn}(\mathbf{PE}, \mathbf{PE}^d) \tag{6}$$

At inference time, we take the surface points of the source mesh as query points. The interaction between the predicted deformation codes sampled from the *Shape Vector Set Diffusion Network*, and the positional embedding of query points through cross-attention yields the approximated latent features. Subsequently, a linear layer derives the predicted positions of the target point cloud.

Synchronized Deformation Vector Set Diffusion In this stage, the sparse noisy input point clouds $\{\mathbf{P}_{\text{src}}, \mathbf{P}_{\text{tgt}}\}$ pairs green blocks in Fig. 2 (b) are processed with a conditional encoder. The conditional encoder has the same structure with the deformation encoder, through which we get the conditional latent set $\mathcal{C}^t(\mathbf{P}^1, \mathbf{P}^t) = \{\mathbf{c}_i \in \mathbb{R}^C\}_{i=1}^M$. Simultaneously, the shape latent set from the deformation encoder is added with Gaussian noise and sent to the denoiser of the diffusion model. As illustrated in the figure, after the space self-attention block, the conditional latent set is injected into the cross-attention block. Following the cross-attention along the temporal domain, and after 18 repetitions, we get the denoised deformation codes.

3. Implementation Details

3.1. Dataset

Train/Val/Test Split The datasets used in our method, D-FAUST [2] and DT4D-A [4], encompass a diverse range of human and animal motions, respectively. D-FAUST includes human motions like "chicken wings", "shake shoulders", and "shake hips." In contrast, DT4D-A features animal animations such as "bear3EP death", "bunnyQ walk", and "deer2MB rotate." Following the approach in NSDP [10] and CaDeX [3], we organize our train and validation sets with data from seen identities performing seen motions. The test set is divided into two categories: unseen motions of seen identities and seen motions of unseen identities. Specifically, for DT4D-A [4], our training set comprises 835 sequences, the validation set includes 59 sequences, and the test set is divided into 89 sequences for unseen motions and 108 sequences for unseen identities. Similarly, in

Symbol	Meaning
T	# frames
$\mathcal{P} = \{\mathbf{P}^t\}_{t=0}^{T-1}$	Sparse point clouds as input for diffusion models
$\mathbf{P}^t = \{\mathbf{p}_i \in \mathbb{R}^3\}_{i=0}^{L}$	A frame of sparse point clouds
L	# points in sparse point clouds
$\mathcal{S} = \{\mathbf{s}_i \in \mathbb{R}^C\}_{i=0}^{M}$	One set of shape codes with M latent codes
s_i	i-th shape code
$\mathcal{D} = \{\mathbf{d}_i \in \mathbb{R}^C\}_{i=0}^{M}$	One set of deformation codes with M latent codes
d_i	i-th deformation code
$\mathcal{C} = \{\mathbf{c}_i \in \mathbb{R}^C\}_{i=0}^{M}$	One set of conditional codes with M latent codes
c_i	i-th deformation code
C	# latent code channels
$\mathcal{V}, \mathcal{F}$	Vertices and faces of mesh
$\mathbf{X}$	Points from meshes as input for shape autoencoder
$\mathbf{X}^d$	Points sampled by FPS from meshes as input for shape autoencoder
$\mathbf{X}_{\text{src}}, \mathbf{X}_{\text{tgt}}$	Points from source and target meshes as input for deformation autoencoder
$\mathbf{X}^d_{\text{src}}, \mathbf{X}^d_{\text{tgt}}$	Points sampled by FPS from source and target meshes as input for deformation autoencoder
$\mathcal{Q}$	Query points set
$\mathbf{q}$	Query point
$\mathbf{q}'$	Deformed query point

Table 1. **Notation table** includes the mathematical symbols we mentioned in the paper.

D-FAUST [2], the training set contains 104 sequences, the validation set has 5 sequences, and the test set includes 9 sequences for unseen motions and 11 sequences for unseen identities.

Data Processing Our data processing strategy is designed to facilitate the learning of shape encoding and deformation. We utilize two distinct datasets for this purpose.

For shape encoding, our process begins with the application of the Butterfly Subdivision method [11], an interpolating subdivision technique widely used in computer graphics for generating smooth surfaces from polygonal meshes. Following the methodology outlined in [6], we center and rescale all meshes to ensure that the bounding box of each mesh is centered at the origin $(0, 0, 0)$ and the longest edge is normalized to a length of 1. We then uniformly sample 200k points within this normalized cube and compute their occupancy values to determine whether they lie inside or outside the mesh, as detailed in [6]. To enhance our model's understanding of surface properties, we introduce Gaussian noise at two different levels to the mesh surface points. This process generates 200k near-surface points, whose occupancy values are also computed. These points, combined with an additional 200k points sampled directly from the mesh surface, constitute the input to our network, providing a comprehensive set of data points for both the uniform space and near-surface regions.

For deformation encoding, we depart from using the Butterfly Subdivision method [11], which was applied in the context of shape encoding. This choice is primarily driven by the need to maintain spatial correspondence between points. To achieve this, we directly sample 200k surface points from the mesh. Subsequently, we sample an equal number of near-surface points. These near-surface points are generated along the normal direction of each surface point, with a predefined distance that ensures closeness to the surface while preserving the detail of the mesh structure. Crucially, both sets of points are selected based on the same face indices of the mesh. This methodical selection guarantees that the spatial correspondence is not disrupted, allowing for a more accurate representation of deformation.

Within the context of the partial challenge configuration, a camera was positioned with a fixed viewing angle of 45 degrees directed towards the human or animal subject within the scene. This positioning was undertaken to capture a partial depth observation of the subject, as illustrated in Figure 3. Additionally, in the more restricted partial setting, only half of the human body was observed. This was accomplished by intentionally selecting vertex indices corresponding to the upper body of the SMPL model [5].

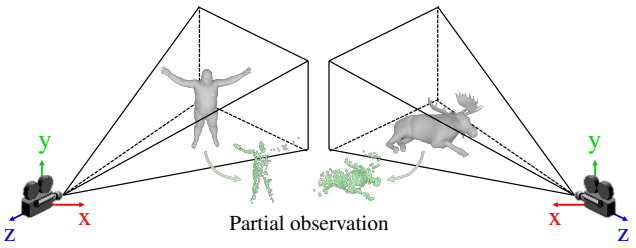


Figure 3. The camera setup for the generation of partial observation in D-FAUST [2] and DT4D-A [4] dataset.

D-FAUST	Unseen Motion			Unseen Individual		
	IoU↑	CD↓	Corr↓	IoU↑	CD↓	Corr↓
W/o. diffusion	79.1%	0.070	0.076	69.3%	0.089	0.098
Full	**90.7%**	**0.033**	**0.047**	**83.7%**	**0.045**	**0.064**

Table 2. Quantitative comparisons of ablation study on the diffusion model from **sparse and noisy** point clouds on D-FAUST [2].

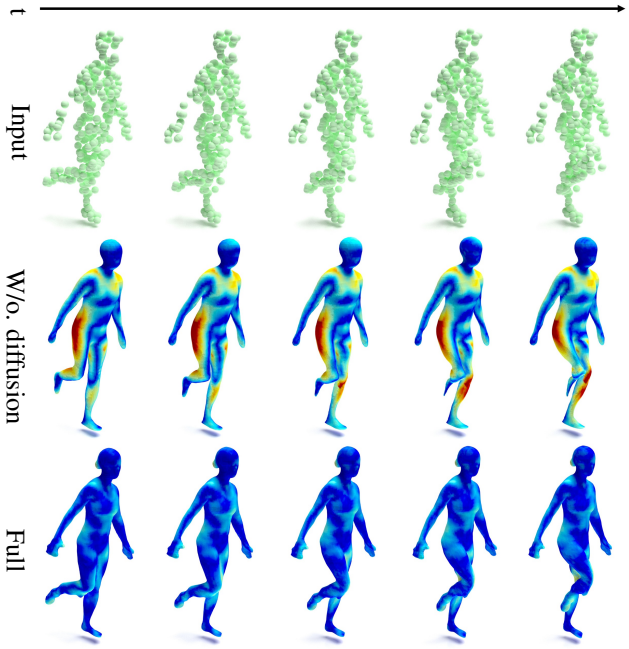


Figure 4. Qualitative comparisons of ablation study on the diffusion model from **sparse and noisy** point clouds on D-FAUST [2]. Our method with diffusion model exhibits lower errors.

4. Additional Results

4.1. Effectiveness of diffusion models

To further demonstrate the efficacy of the diffusion model, we expanded our ablation study to include 4D Shape Reconstruction from sparse and noisy point cloud sequences, utilizing the D-FAUST dataset [2]. This study provides a more comparative analysis against variants of one-step regression integrated with diffusion models, where the input comprises sequences of point clouds of size $L = 300$. Fig. 4 and Tab. 2 present both quantitative and qualitative comparisons. The results clearly indicate that incorporating the diffusion model significantly reduces reconstruction error and yields more precise motion outputs.

4.2. 4D Shape Reconstruction

In evaluating our model's performance in 4D shape reconstruction, we adopted a comprehensive set of metrics: Intersection over Union (IoU), chamfer distance, and correspondence distance. These metrics were chosen for their relevance in accurately quantifying shape reconstruction quality. IoU measures the overlap between predicted and ground truth shapes, Chamfer distance quantifies the average closest point distance, and correspondence distance evaluates the accuracy of point-wise correspondences. These metrics align with the standards set in recent studies, such as those by LPDC [9] and OFlow [7]. We present the average metrics across all 17 frames for both D-FAUST [2] and DT4D-A [4] datasets. The quantitative results are shown in Tab. 4, Tab. 5, Tab. 6, and Tab. 7. Our analysis reveals that our model demonstrates superior generalization capabilities in scenarios involving both unseen motions and unseen individuals. Specifically, it outperforms existing approaches, including OFlow [7], LPDC [9], and CaDeX [3], across all evaluated metrics for all time frames. This advancement underscores the efficacy of our approach in handling the complexities of 4D shape reconstruction. Furthermore, qualitative results in Fig. 7 and Fig. 8 show a selection of 8 frames, chosen to represent a diverse range of motions and shapes, from the total 17 to illustrate our model's performance. In each figure, the upper part displays the results for unseen motion, while the lower part corresponds to unseen individuals. We utilize a chamfer distance error map for visualization, where blue indicates lower error and red signifies higher error. The color-coded error map, computed based on the distance between predicted and ground truth points, provides an intuitive understanding of the model's accuracy in different scenarios. Our model not only has an overall smaller error on both datasets, but also captures motions more accurately.

BEHAVE	OFlow	LPDC	CaDeX	Ours
Chamfer↓	0.137	0.201	0.126	**0.062**

Table 3. Quantitative evaluation on real dataset BEHAVE [1]. The chamfer distance is computed from the reconstructed mesh and partial point cloud input.

4.3. 4D Shape Completion

In the more challenging task of 4D shape completion from partial point clouds, our model shows noticable improvements over existing state-of-the-art methods. The quantitative results, as illustrated in Fig. 9 and Fig. 10, demonstrate substantial performance enhancements, with a lower error rate in scenarios involving unseen motions and individuals. This underscores the robustness of our approach. Also, we present more results about the challenging half human setting. The result is demonstrated in the Fig. 6.

5. Real-world Data Test

In this section, we validate our model using data from the real-world BEHAVE dataset [1], employing four Kinect RGB-D cameras to capture RGB color and depth frames.

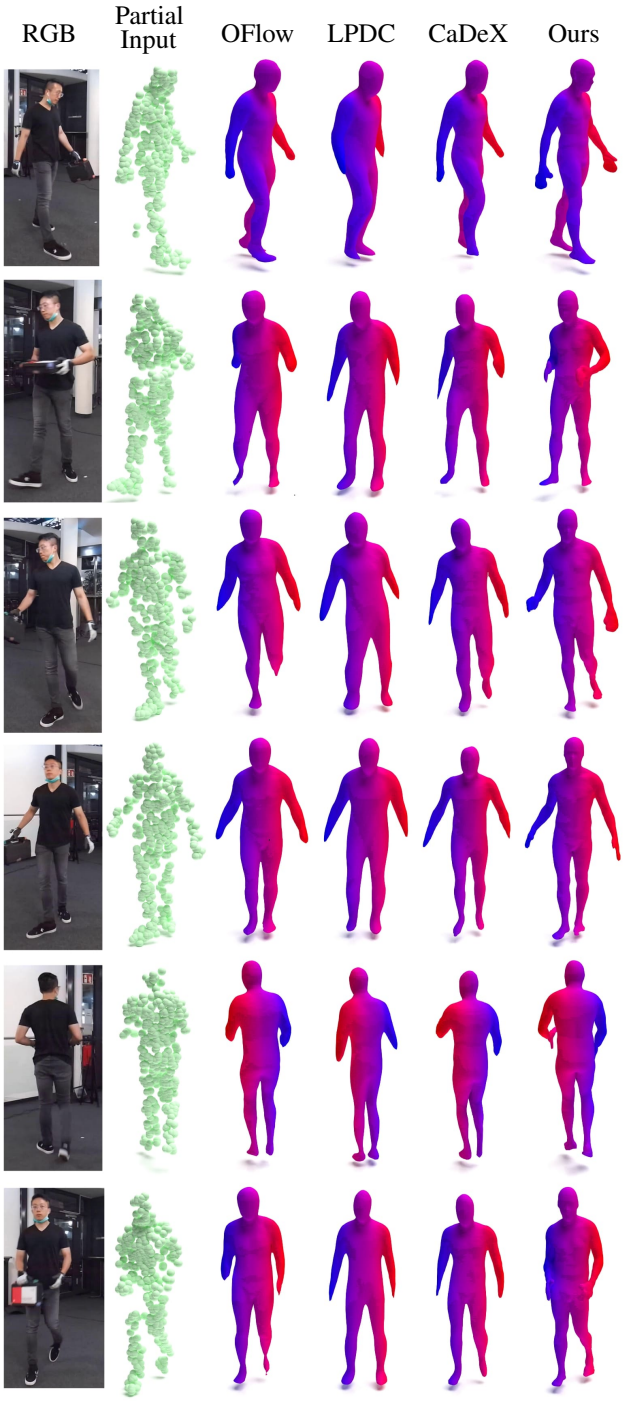


Figure 5. 4D Shape Completion on the BEHAVE dataset [1].

In our case, we utilize a single view from a fixed camera to align with our previous partial scan setting. Similarly, the depth map is back-projected into 3D point cloud, serving as a partial input for our model. In Fig. 5 and Tab. 3, we present a qualitative and quantitative evaluation of our reconstruction process with the corresponding input and reference RGB frame. The results demonstrate the robustness of our model in scenarios characterized by incomplete scans, such as instances where limbs, like the leg or arm, are obscured by structures, such as a grasped object in the hand. This resilience stems from the inherent capabilities of the diffusion model, empowering our model to infer potential structures even in the presence of significant occlusions.

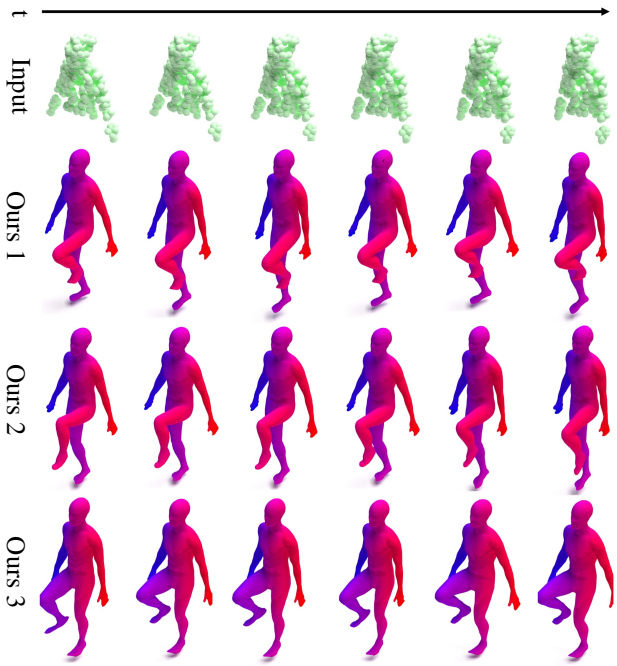


Figure 6. More results of 4D Shape Completion from highly partial point clouds (half human) on the D-FAUST [2] dataset.

Time step	IoU				Chamfer				Correspond.			
	OFlow	LPDC	CaDex	Ours	OFlow	LPDC	CaDex	Ours	OFlow	LPDC	CaDex	Ours
0	83.1%	85.6%	89.1%	**91.2%**	0.059	0.052	0.044	**0.031**	0.057	0.047	0.043	**0.031**
1	83.1%	85.5%	89.2%	**91.2%**	0.059	0.053	0.044	**0.031**	0.062	0.053	0.047	**0.034**
2	82.8%	85.4%	89.3%	**91.1%**	0.061	0.053	0.043	**0.031**	0.069	0.059	0.054	**0.036**
3	82.5%	85.3%	89.4%	**91.0%**	0.061	0.053	0.043	**0.032**	0.077	0.064	0.061	**0.039**
4	82.2%	85.0%	89.4%	**91.0%**	0.062	0.054	0.043	**0.032**	0.083	0.070	0.068	**0.040**
5	82.0%	85.1%	89.5%	**90.9%**	0.063	0.054	0.043	**0.032**	0.088	0.074	0.074	**0.043**
6	81.8%	85.1%	89.5%	**90.7%**	0.064	0.054	0.043	**0.033**	0.092	0.078	0.080	**0.045**
7	81.6%	85.1%	89.5%	**90.6%**	0.064	0.054	0.043	**0.032**	0.095	0.082	0.085	**0.047**
8	81.4%	85.0%	89.5%	**90.6%**	0.065	0.055	0.043	**0.033**	0.098	0.085	0.090	**0.048**
9	81.3%	85.0%	89.5%	**90.5%**	0.066	0.055	0.043	**0.034**	0.101	0.088	0.094	**0.051**
10	81.1%	84.6%	89.5%	**90.3%**	0.066	0.055	0.043	**0.034**	0.103	0.090	0.097	**0.052**
11	81.0%	84.6%	89.5%	**90.3%**	0.067	0.055	0.043	**0.034**	0.106	0.092	0.100	**0.054**
12	80.8%	84.6%	89.5%	**90.2%**	0.068	0.056	0.043	**0.034**	0.108	0.094	0.103	**0.055**
13	80.6%	84.4%	89.4%	**90.2%**	0.069	0.056	0.043	**0.034**	0.111	0.095	0.105	**0.056**
14	80.3%	84.3%	89.3%	**90.0%**	0.070	0.056	0.044	**0.035**	0.114	0.096	0.107	**0.057**
15	80.0%	84.3%	89.2%	**90.0%**	0.071	0.056	0.044	**0.035**	0.119	0.097	0.109	**0.059**
16	79.5%	84.3%	89.1%	**90.0%**	0.073	0.056	0.044	**0.035**	0.125	0.098	0.110	**0.059**
Mean	81.5%	84.9%	89.4%	**90.7%**	0.065	0.055	0.043	**0.033**	0.095	0.080	0.084	**0.047**

Table 4. **4D Shape Reconstruction for Unseen Motions (DFAUST).** We evaluate IoU, Chamfer distance, and correspondence distance for 17 timeframes for the 4D shape reconstruction from sparse point clouds on seen individuals but unseen motions of DFAUST [2] dataset.

Time step	IoU				Chamfer				Correspond.			
	OFlow	LPDC	CaDex	Ours	OFlow	LPDC	CaDex	Ours	OFlow	LPDC	CaDex	Ours
0	74.2%	76.8%	80.4%	**84.6%**	0.077	0.068	0.055	**0.042**	0.077	0.065	0.057	**0.044**
1	74.1%	76.8%	80.5%	**84.6%**	0.077	0.069	0.055	**0.042**	0.082	0.071	0.060	**0.046**
2	73.8%	76.7%	80.6%	**84.5%**	0.078	0.069	0.054	**0.043**	0.089	0.075	0.064	**0.048**
3	73.4%	76.5%	80.7%	**84.4%**	0.079	0.069	0.054	**0.043**	0.096	0.080	0.069	**0.051**
4	73.0%	76.5%	80.8%	**84.4%**	0.081	0.070	0.054	**0.043**	0.102	0.085	0.074	**0.053**
5	72.7%	76.6%	80.8%	**84.2%**	0.082	0.070	0.054	**0.044**	0.108	0.089	0.078	**0.056**
6	72.4%	76.3%	80.8%	**84.0%**	0.083	0.070	0.054	**0.043**	0.113	0.094	0.083	**0.059**
7	72.2%	76.2%	80.8%	**83.8%**	0.084	0.071	0.054	**0.045**	0.117	0.098	0.086	**0.061**
8	72.0%	76.0%	80.8%	**83.6%**	0.085	0.071	0.054	**0.046**	0.121	0.101	0.090	**0.065**
9	71.9%	76.1%	80.8%	**83.6%**	0.085	0.071	0.054	**0.046**	0.124	0.104	0.093	**0.067**
10	71.8%	76.1%	80.8%	**83.5%**	0.086	0.072	0.054	**0.046**	0.127	0.107	0.096	**0.069**
11	71.7%	75.9%	80.7%	**83.3%**	0.086	0.072	0.054	**0.047**	0.130	0.109	0.098	**0.072**
12	71.5%	75.9%	80.7%	**83.1%**	0.087	0.072	0.054	**0.048**	0.133	0.112	0.101	**0.076**
13	71.4%	75.8%	80.6%	**83.1%**	0.087	0.072	0.054	**0.048**	0.136	0.114	0.103	**0.077**
14	71.3%	75.7%	80.5%	**83.0%**	0.088	0.073	0.055	**0.049**	0.140	0.116	0.105	**0.079**
15	71.0%	75.6%	80.4%	**82.8%**	0.089	0.073	0.055	**0.049**	0.145	0.119	0.106	**0.081**
16	70.7%	75.7%	80.2%	**82.8%**	0.090	0.074	0.056	**0.049**	0.150	0.121	0.107	**0.082**
Mean	72.3%	76.2%	80.6%	**83.7%**	0.084	0.071	0.054	**0.045**	0.117	0.098	0.086	**0.064**

Table 5. **4D Shape Reconstruction for Unseen Individuals (DFAUST).** We evaluate I0U, Chamfer distance, and correspondence distance for 17 timeframes for the 4D shape reconstruction from sparse point cloud task on unseen individuals of DFAUST [2] dataset.

Time step	IoU				Chamfer				Correspond.			
	OFlow	LPDC	CaDex	**Ours**	OFlow	LPDC	CaDex	**Ours**	OFlow	LPDC	CaDex	**Ours**
0	74.8%	63.3%	79.6%	**89.6%**	0.191	0.302	0.063	**0.046**	0.163	0.252	0.078	**0.045**
1	74.4%	62.6%	79.8%	**89.6%**	0.193	0.308	0.062	**0.047**	0.182	0.285	0.082	**0.048**
2	73.7%	62.3%	80.0%	**89.4%**	0.197	0.310	0.061	**0.047**	0.208	0.311	0.091	**0.051**
3	73.1%	62.0%	80.2%	**89.3%**	0.202	0.313	0.060	**0.048**	0.233	0.341	0.100	**0.053**
4	72.7%	61.6%	80.3%	**89.2%**	0.206	0.316	0.060	**0.048**	0.254	0.372	0.110	**0.056**
5	72.3%	61.3%	80.2%	**89.1%**	0.209	0.320	0.059	**0.049**	0.271	0.402	0.118	**0.058**
6	72.1%	61.0%	80.5%	**88.9%**	0.211	0.323	0.059	**0.050**	0.285	0.431	0.126	**0.060**
7	71.9%	60.6%	80.6%	**88.9%**	0.213	0.327	0.059	**0.050**	0.298	0.459	0.133	**0.062**
8	71.7%	60.3%	80.6%	**88.7%**	0.215	0.331	0.058	**0.050**	0.308	0.485	0.139	**0.063**
9	71.5%	59.9%	80.7%	**88.6%**	0.216	0.335	0.058	**0.051**	0.318	0.510	0.145	**0.065**
10	71.3%	59.6%	80.7%	**88.6%**	0.218	0.339	0.059	**0.051**	0.327	0.533	0.150	**0.066**
11	71.1%	59.2%	80.6%	**88.6%**	0.219	0.343	0.059	**0.052**	0.335	0.555	0.154	**0.067**
12	70.9%	58.9%	80.5%	**88.5%**	0.221	0.347	0.059	**0.052**	0.343	0.576	0.158	**0.068**
13	70.7%	58.6%	80.4%	**88.5%**	0.223	0.351	0.060	**0.052**	0.352	0.598	0.162	**0.069**
14	70.3%	58.2%	80.2%	**88.4%**	0.226	0.355	0.060	**0.052**	0.364	0.619	0.166	**0.070**
15	69.7%	57.9%	80.0%	**88.4%**	0.231	0.360	0.061	**0.053**	0.380	0.641	0.169	**0.071**
16	68.9%	57.6%	79.6%	**88.3%**	0.239	0.365	0.063	**0.053**	0.402	0.662	0.173	**0.072**
Mean	71.8%	60.3%	80.3%	**88.9%**	0.214	0.332	0.060	**0.050**	0.295	0.472	0.133	**0.061**

Table 6. **4D Shape Reconstruction for Unseen Motions (DT4D-A)** We evaluate IoU, Chamfer distance, and correspondence distance for 17 timeframes for the 4D shape reconstruction from sparse point cloud task on seen individuals but unseen motions of DT4D-A [4] dataset.

Time step	IoU				Chamfer				Correspond.			
	OFlow	LPDC	CaDex	**Ours**	OFlow	LPDC	CaDex	**Ours**	OFlow	LPDC	CaDex	**Ours**
0	62.4%	53.5%	64.2%	**84.8%**	0.294	0.404	0.129	**0.056**	0.216	0.296	0.150	**0.057**
1	62.1%	52.8%	64.4%	**84.7%**	0.296	0.412	0.128	**0.056**	0.235	0.330	0.157	**0.060**
2	61.7%	52.6%	64.5%	**84.6%**	0.300	0.414	0.127	**0.056**	0.260	0.358	0.169	**0.062**
3	61.4%	52.4%	64.6%	**84.4%**	0.304	0.417	0.127	**0.057**	0.283	0.390	0.183	**0.065**
4	61.1%	52.2%	64.6%	**84.3%**	0.307	0.420	0.126	**0.057**	0.303	0.422	0.197	**0.068**
5	60.9%	51.9%	64.7%	**84.1%**	0.309	0.423	0.126	**0.057**	0.321	0.453	0.211	**0.070**
6	60.7%	51.7%	64.8%	**83.9%**	0.311	0.427	0.126	**0.058**	0.336	0.483	0.224	**0.072**
7	60.5%	51.4%	64.8%	**83.8%**	0.313	0.430	0.125	**0.058**	0.350	0.511	0.235	**0.074**
8	60.4%	51.2%	64.8%	**83.6%**	0.315	0.433	0.125	**0.059**	0.362	0.538	0.246	**0.076**
9	60.2%	51.0%	64.8%	**83.4%**	0.317	0.437	0.125	**0.059**	0.374	0.563	0.256	**0.077**
10	60.1%	50.7%	64.8%	**83.3%**	0.319	0.440	0.125	**0.059**	0.385	0.588	0.266	**0.078**
11	60.0%	50.5%	64.8%	**83.3%**	0.321	0.443	0.125	**0.059**	0.395	0.611	0.274	**0.080**
12	59.8%	50.3%	64.8%	**83.2%**	0.323	0.446	0.125	**0.060**	0.405	0.633	0.282	**0.081**
13	59.7%	50.1%	64.7%	**83.1%**	0.325	0.449	0.126	**0.060**	0.416	0.654	0.289	**0.082**
14	59.4%	49.9%	64.6%	**83.0%**	0.329	0.452	0.126	**0.060**	0.428	0.675	0.296	**0.083**
15	59.1%	49.7%	64.5%	**82.9%**	0.334	0.455	0.127	**0.060**	0.445	0.696	0.303	**0.084**
16	58.6%	49.5%	64.3%	**82.8%**	0.340	0.459	0.128	**0.061**	0.466	0.716	0.309	**0.085**
Mean	60.5%	51.3%	64.6%	**83.7%**	0.315	0.433	0.126	**0.058**	0.352	0.525	0.238	**0.074**

Table 7. **4D Shape Reconstruction for Unseen Individuals (DT4D-A)** We evaluate IoU, Chamfer distance, and correspondence distance for 17 timeframes for the 4D shape reconstruction from sparse point cloud task on unseen individuals of DT4D-A [4] dataset.

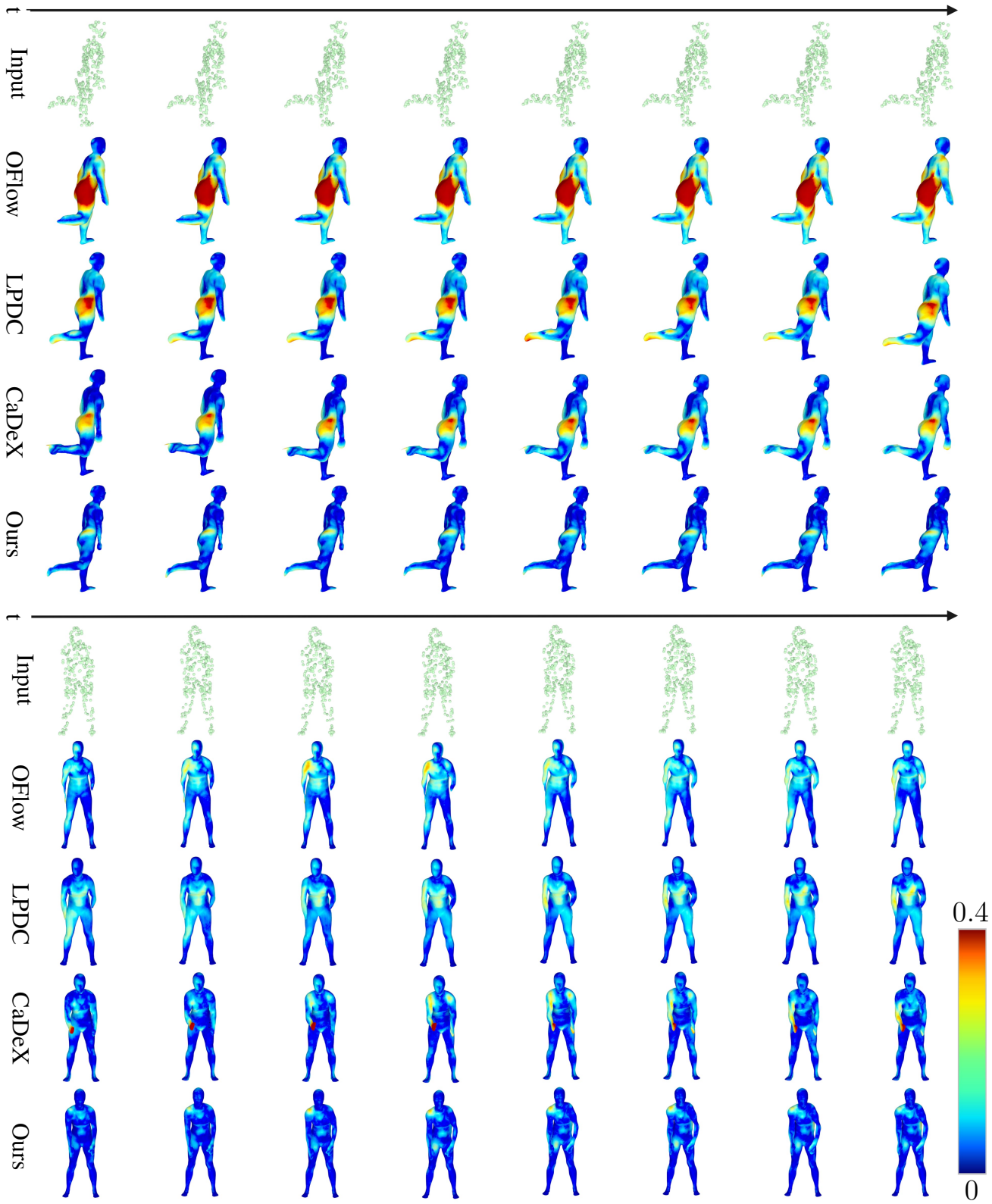


Figure 7. **4D Shape Reconstruction from sparse and noisy point clouds on the D-FAUST[2] dataset**. One for unseen motion (upper) and another for unseen individuals (lower).

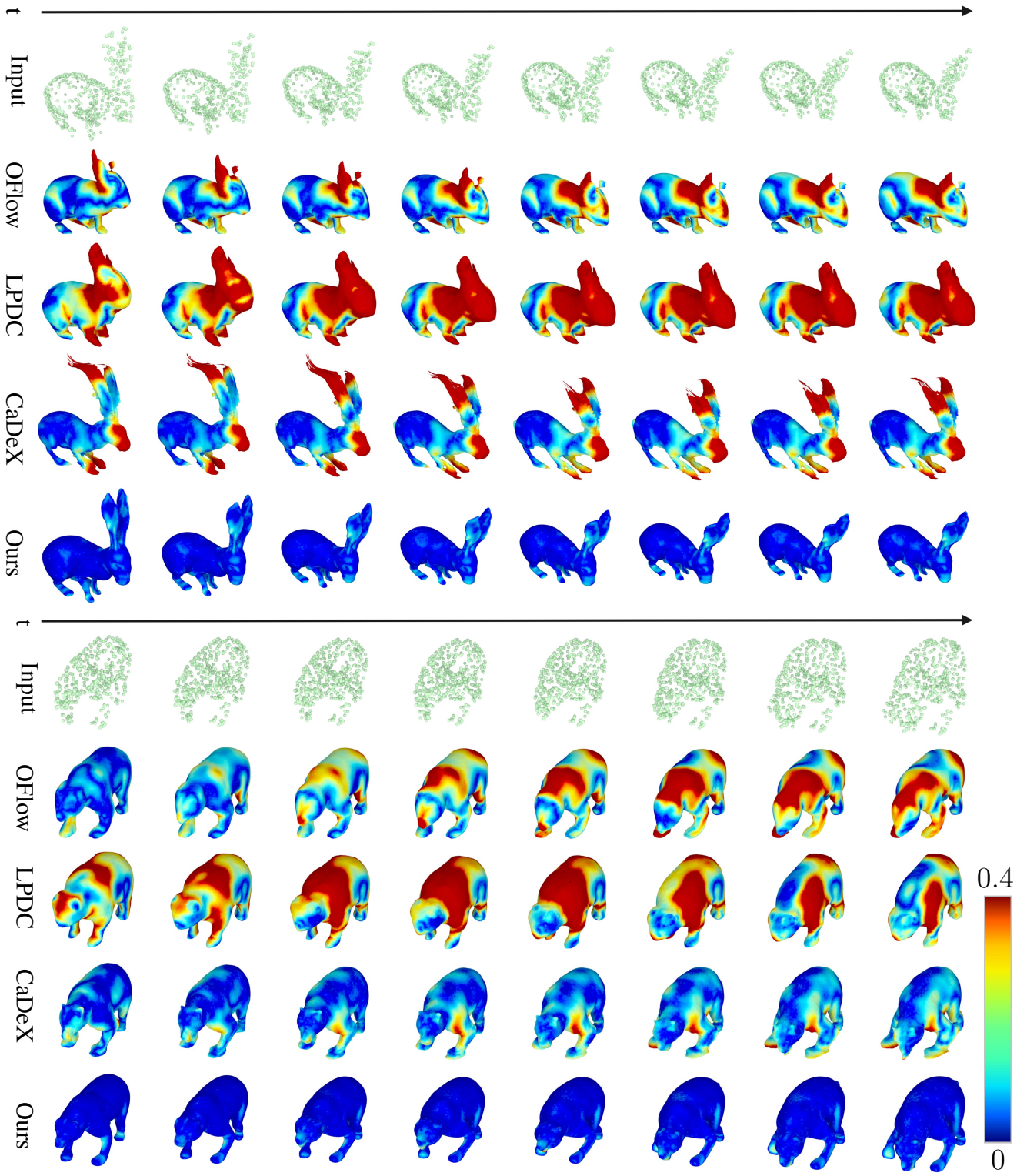


Figure 8. **4D Shape Reconstruction from sparse and noisy point clouds on the DT4D-A [4] dataset.** One for unseen motion (upper) and another for unseen individuals (lower).

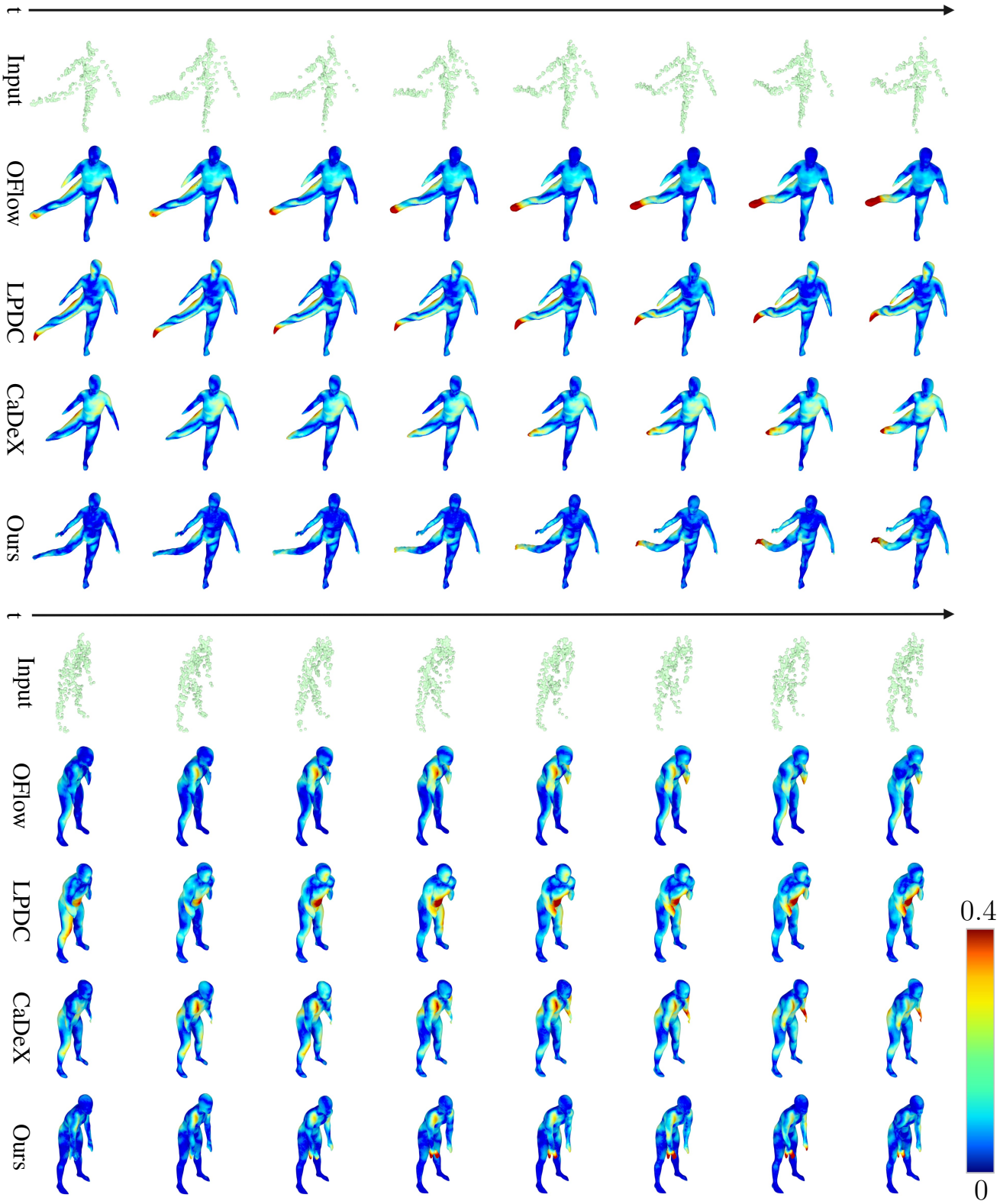


Figure 9. **4D Shape Completion from partial point clouds on the D-FAUST[2] dataset**. One for unseen motion (upper) and another for unseen individuals (lower).

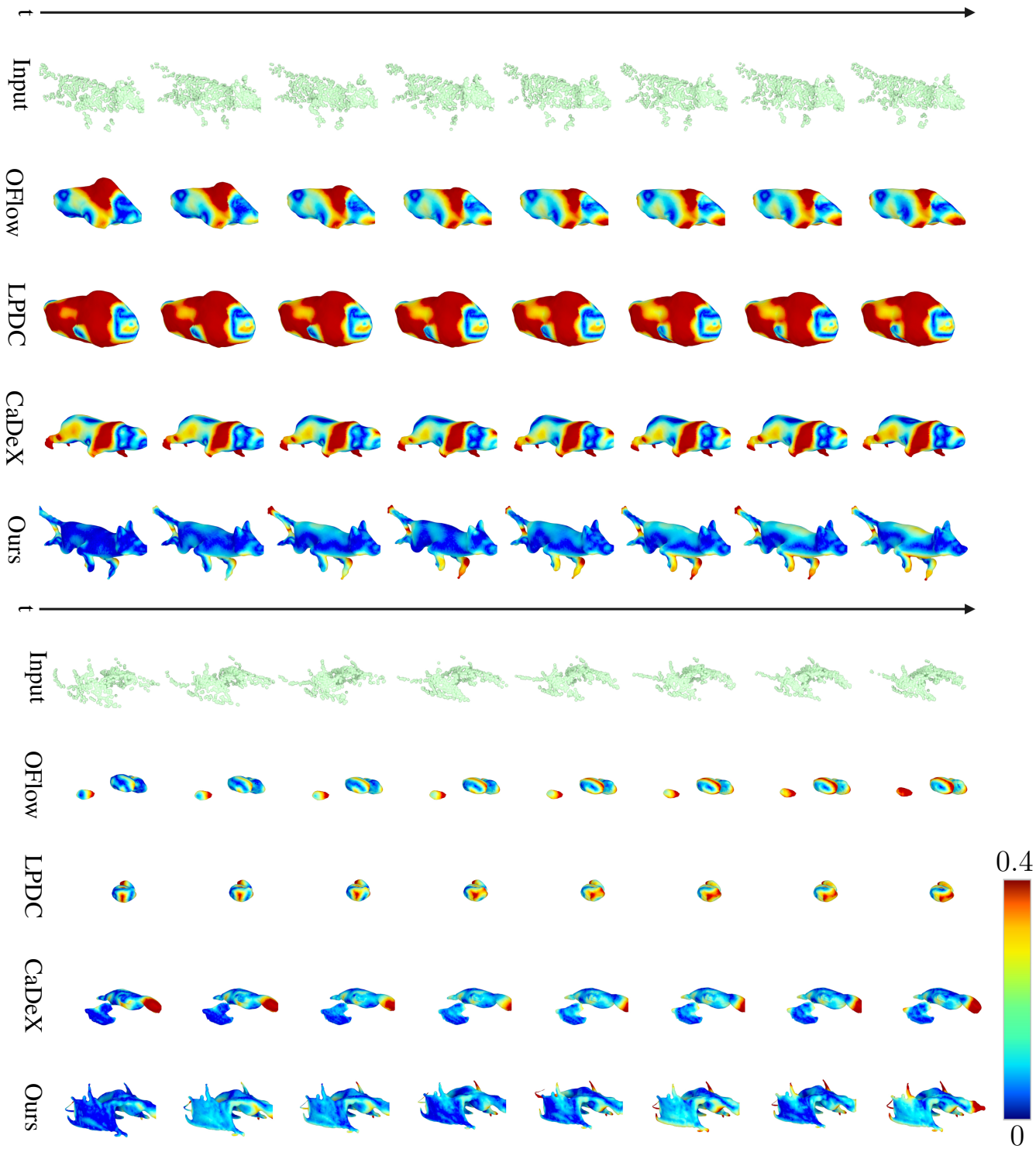


Figure 10. **4D Shape Completion from partial point clouds on the DT4D-A [4] dataset.** One for unseen motion (upper) and another for unseen individuals (lower).

References

[1] Bharat Lal Bhatnagar, Xianghui Xie, Ilya Petrov, Cristian Sminchisescu, Christian Theobalt, and Gerard Pons-Moll. Behave: Dataset and method for tracking human object interactions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2022. 5, 6

[2] Federica Bogo, Javier Romero, Gerard Pons-Moll, and Michael J. Black. Dynamic FAUST: Registering human bodies in motion. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3, 4, 5, 6, 7, 9, 11

[3] Jiahui Lei and Kostas Daniilidis. Cadex: Learning canonical deformation coordinate space for dynamic surface representation via neural homeomorphism. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 3, 5

[4] Yang Li, Hikari Takehara, Takafumi Taketomi, Bo Zheng, and Matthias Nießner. 4dcomplete: Non-rigid motion estimation beyond the observable surface. *IEEE International Conference on Computer Vision (ICCV)*, 2021. 3, 4, 5, 8, 10, 12

[5] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, 2015. 4

[6] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 4

[7] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Occupancy flow: 4d reconstruction by learning particle dynamics. In *International Conference on Computer Vision*, 2019. 5

[8] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 1

[9] Jiapeng Tang, Dan Xu, Kui Jia, and Lei Zhang. Learning parallel dense correspondence from spatio-temporal descriptors for efficient and robust 4d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6022–6031, 2021. 5

[10] Jiapeng Tang, Lev Markhasin, Bi Wang, Justus Thies, and Matthias Nießner. Neural shape deformation priors. *Advances in Neural Information Processing Systems*, 35: 17117–17132, 2022. 3

[11] Denis Zorin, Peter Schröder, and Wim Sweldens. Interpolating subdivision for meshes with arbitrary topology. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 189–192, 1996. 4