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

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

[1]Max Planck Institute for Informatics, Saarland Informatics Campus, Germany
[2]University of Würzburg, Germany
[3]Computer Graphics Lab, TU Braunschweig, Germany

In the following, we provide details for our main paper [1] regarding the hyperparameters used in experiments and illustrate the continuous feature extraction used during shape decoding (Sec. 1). Further, we explain the waterproofing method we have used to create our Humans dataset (Sec. 2). Finally, we show further results for the *Single-View Human Reconstruction* experiment (Sec. 3). Here we additionally demonstrate the performance of our method for dynamic sequence reconstruction on a frame-by-frame basis.

1. Implementation Details

Hyperparameters. Next, the hyperparameters for the main paper experiments are given. Please refer to Section 3 of the main paper for a detailed description of the symbols. The number of all point samples with their ground truth occupancy is $S = 100.000$ in all experiments. During training, sub-samples of size $R = 50.000$ are used. For optimizing the loss $\mathcal{L}_{\mathcal{B}}(\mathbf{w})$ in Equation 5, we use the Adam optimizer with parameters $lr = 1e-4$, $betas = (0.9, 0.999)$, $eps = 1e-8$, $weight\_decay = 0$ in all experiments. The first feature grid $\mathbf{F}_1$ has a resolution of $N = 32$ for super-resolution from 32^3 voxels, $N = 128$ for super-resolution from 128^3 voxels, $N = 128$ for point cloud reconstruction from Shapenet, and $N = 256$ for single-view reconstruction. The decoder $f()$ consists of 3 fully connected layers for the human voxel experiments and of 4 fully connected layers for all other experiments. For sampling ground truth points in the vicinity of the surface, we used $\sigma_1 = 0.01$, $\sigma_2 = 0.15$ for ShapeNet reconstruction from 3000 points and $\sigma_1 = 0.01$, $\sigma_2 = 0.1$ for the other ShapeNet experiments, $\sigma_1 = 0.015$, $\sigma_2 = 0.15$ for human super-resolution and $\sigma_1 = 0.015$, $\sigma_2 = 0.2$ for single-view reconstruction. For feature extraction along the XYZ axes, we used a distance of $d = 0.035$ for ShapeNet 32^3 voxels and $d = 0.072$ in all other experiments. We used $n = 4$ feature grids for 32^3 Humans and ShapeNet voxels, $n = 5$ for 128^3 Humans voxels, $n = 6$ for ShapeNet with 128^3 and 3000 points and $n = 7$ for single-view reconstruction. Code is available at https://virtualhumans.mpi-inf.mpg.de/ifnets/.

Continuous Feature Extraction. Next we further illustrate Eq. 3 of the main paper. To this end, we briefly recap the shape decoding of IF-Nets (cf. Section 3.2 main paper): Given an input shape $\mathbf{X}$ the encoder g produces the shape encoding $g(\mathbf{X}) = \mathbf{F}_1, .., \mathbf{F}_n$. The decoding is done in a point wise fashion, that is, given a query point $\mathbf{p} \in \mathbb{R}^3$ and the encoding, the task of the decoder is to classify the point as inside or outside of the shape. To this end, the decoder f is fed with local and global features extracted from the encoding $\mathbf{F}_1, .., \mathbf{F}_n$ at point $\mathbf{p}$. In order to encode information of the local neighborhood into the point encoding, even at early grids with small receptive fields (e.g. $\mathbf{F}_1$), we extract features at a query point's location $\mathbf{p}$ itself and *additionally* at surrounding points in a distance d along the Cartesian axes:

$$\{\mathbf{p} + a \cdot \mathbf{e}_i \cdot d \in \mathbb{R}^3 | a \in \{1, 0, -1\}, i \in \{1, 2, 3\}\},$$

where $d \in \mathbb{R}$ is the distance to the center point $\mathbf{p}$ and $\mathbf{e}_i \in \mathbb{R}^3$ is the $i-th$ Cartesian axis unit vector. Figure 1 illustrates this sampling strategy. The additional points from the local neighborhood are depicted as ●.

2. Humans Dataset Waterproofing

To compute ground truth occupancies for our Humans dataset, we first need to waterproof all scans. Hereby, it is crucial to not loose desired detail. To this end, we have developed a new waterproofing algorithm. Instead of explicitly manipulating the meshes, we determine the occupancy value of a 3D point in implicit space. We call our approach *implicit waterproofing*.

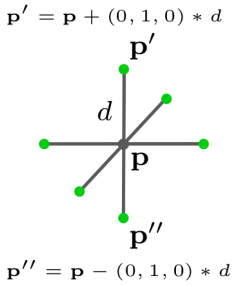

Figure 1. Illustration of locations for feature extraction. In order to add information on the local neighborhood into the point encoding, we extract features not only at a query point's location $\mathbf{p}$ itself, but *also* at surrounding points (●) in a distance d along the Cartesian axes.

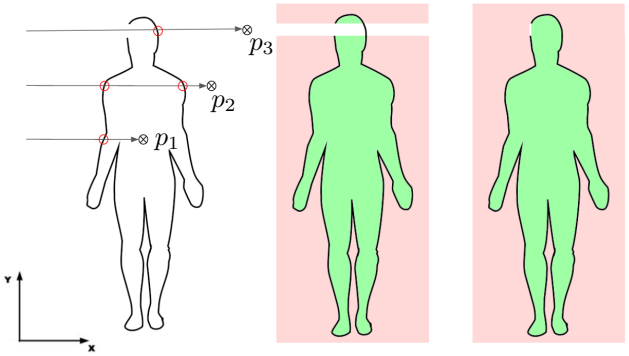

Figure 2. Visualization of the implicit waterproofing algorithm. See text for in-depth explanation.

We assume, like in the real world, every object has a certain thickness. Please note that this assumption is met for all scans of the Humans dataset but sometimes does not hold for artificially generated meshes representing thin objects by only one surface. To compute if a point $p \in \mathbb{R}^3$ is inside or outside a mesh, we compute a ray though p parallel to one of the Cartesian axes (e.g. X-axis, see Figure 2 left). For a point inside the mesh (e.g. p_1), the ray must intersect the surface in an odd number of times before passing through the point and in an even number of times in total, e.g. the ray enters the mesh, passes the point, and leaves the mesh. For a point outside the mesh (e.g. p_2), the ray must intersect the surface in an even number of times before passing through the point, e.g. the ray enters and leaves the mesh before passing the point. Thus, for meshes without holes such collision detection suffices to compute occupancies. Now consider a ray passing though a hole in the surface before or after passing its corresponding point (e.g. p_3): A ray through a hole must intersect the surface in an odd number of times in total. We can now classify every point to be inside the mesh (green), outside (red) or unknown (white), cf. Figure 2 (middle). In order to classify the remaining unknown points, we rotate the object by some degrees and repeat classification for the unclassified regions (Figure 2 right). This process is iterated multiple times until convergence. Remaining unknown points are assumed to be outside. On the Humans dataset three 90 degree rotations around the XYZ-axes have been sufficient to obtain good results.

3. More Results

Single-View Human Reconstruction. In Figure 3, we show more qualitative results for reconstructed full-body clothed humans from partial point clouds. All point clouds contain data only from one view-point and no information about the back-side of the subjects - the typical output of a depth camera.

Single-View Video Reconstruction. In this additional experiment on *Single-View Video Reconstruction*, we reconstruct a motion sequence of a subject from the BUFF [2] dataset. The given input is a sequence of single-view point clouds of a human in motion with fully occluded back, i.e. point clouds only depicting the front of the person.

Precisely, the input is a sequence of 297 frames acquired at 60 frames per second, i.e. circa 5 seconds long. To generate single-view point clouds from the 4D scans in the BUFF dataset, we synthesize depth images per frame with only 250×250px resolution, producing around 5000 points per frame on the visible side of the subject. Back-projection of the depth-pixels into 3D space generates the single-view point clouds. For this task, we add no additional temporal constrains to IF-Nets. Instead, we directly apply the trained network for Single-View Reconstruction from the previous experiment on a frame-by-frame basis.

To successfully fulfil this task, IF-Nets have again to fulfil the requirements of the *static* single-view reconstruction, namely: reconstruct unknown articulations, retain fine details, and meaningfully complete highly incomplete data at the same time. For this task of *dynamic* single-view reconstruction, IF-Nets are additionally tested to show their generalization capabilities: a) generalization to dynamic data (motions) and b) generalization to a new source of data unseen during training. Dynamic data is particularly challenging, as its reconstruction has to be temporally and spatially smooth.

Figures 4, 5 and 6 show 9 frames of the motion sequence. The input point cloud is shown in the first and second columns with a front view and a side view respectively. The third and forth columns analogously show the IF-Net reconstructions. Please refer to the supplemental video for the whole reconstructed sequence.

Again, IF-Nets reliably retain details present in the input e.g. cloth wrinkles, hair style, or facial details. Further,

IF-Nets plausibly complete the back of the subject. Both properties demonstrate good generalization to the unseen data source. Despite not being trained on sequential data, viewing the resulting sequence as a video shows surprising temporally smooth reconstructions. This demonstrates that IF-Nets allow to coherently reconstruct single-view video while featuring high diversity in terms of clothing details and variety of poses.

References

[1] Julian Chibane, Thiemo Alldieck, and Gerard Pons-Moll. Implicit functions in feature space for 3d shape reconstruction and completion. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2020. 1

[2] Chao Zhang, Sergi Pujades, Michael Black, and Gerard Pons-Moll. Detailed, accurate, human shape estimation from clothed 3D scan sequences. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

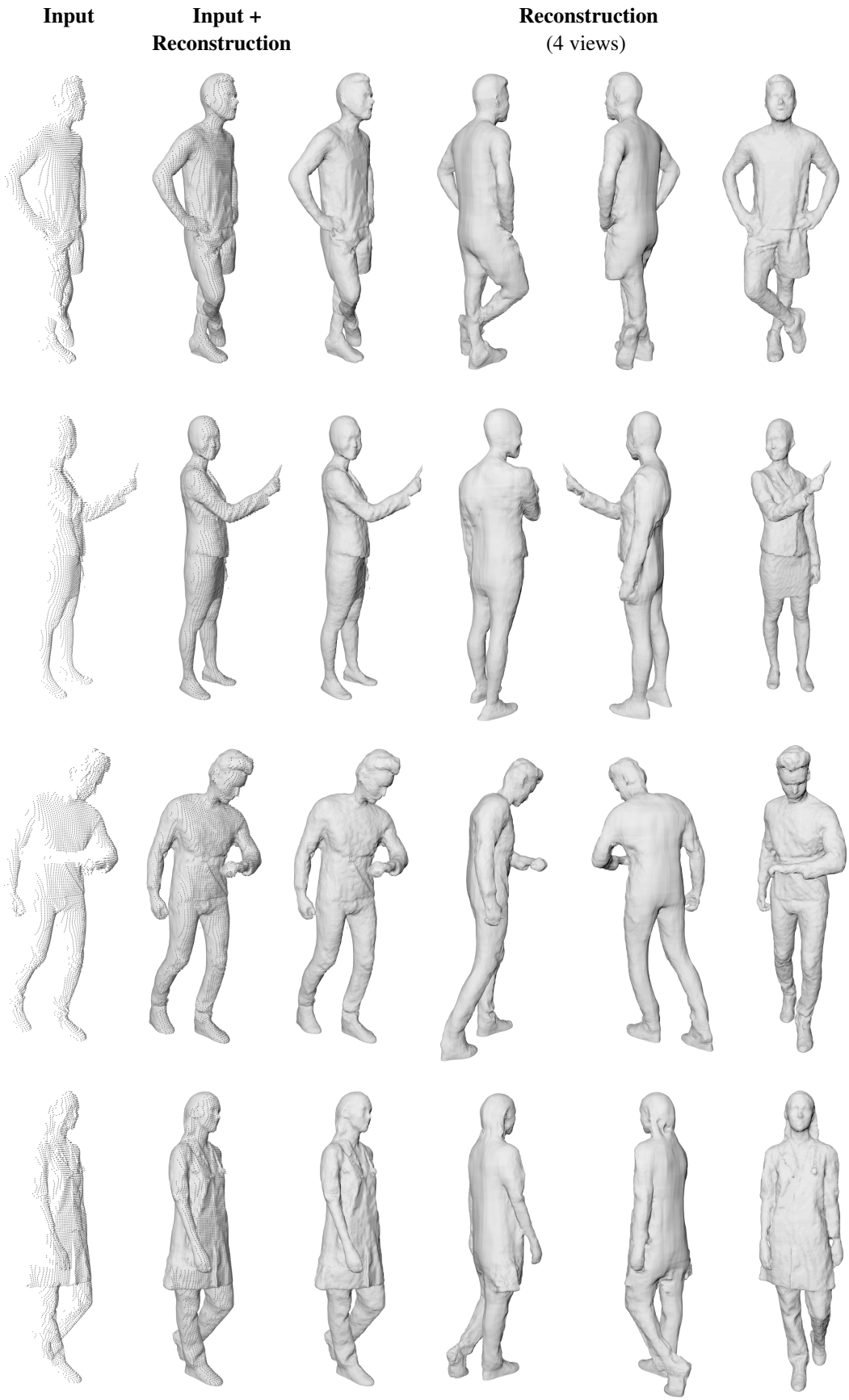

Figure 3. More results for *Single-View Human Reconstruction*.

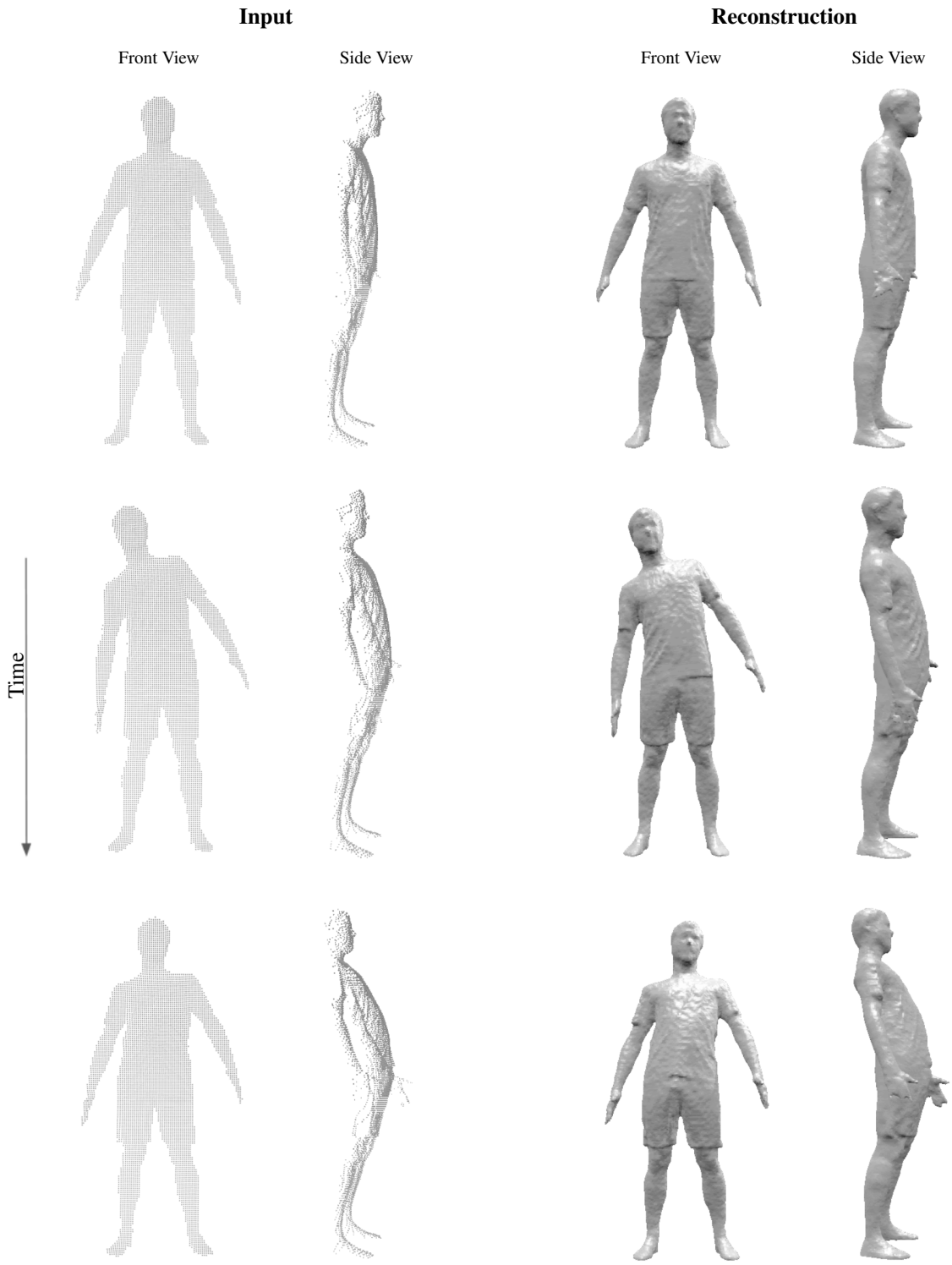

Figure 4. Single-view point clouds and reconstructions of subsequent frames (top to bottom). The given input is a sequence of single-view point clouds of a human in motion with fully occluded back from the BUFF dataset (left). Reconstructions (right) have been computed frame-by-frame. The IF-Net has been trained only on static single-views from the Humans dataset. The results demonstrate that the IF-Net generalizes to a new data source and to unseen poses during training and produces temporal coherence.

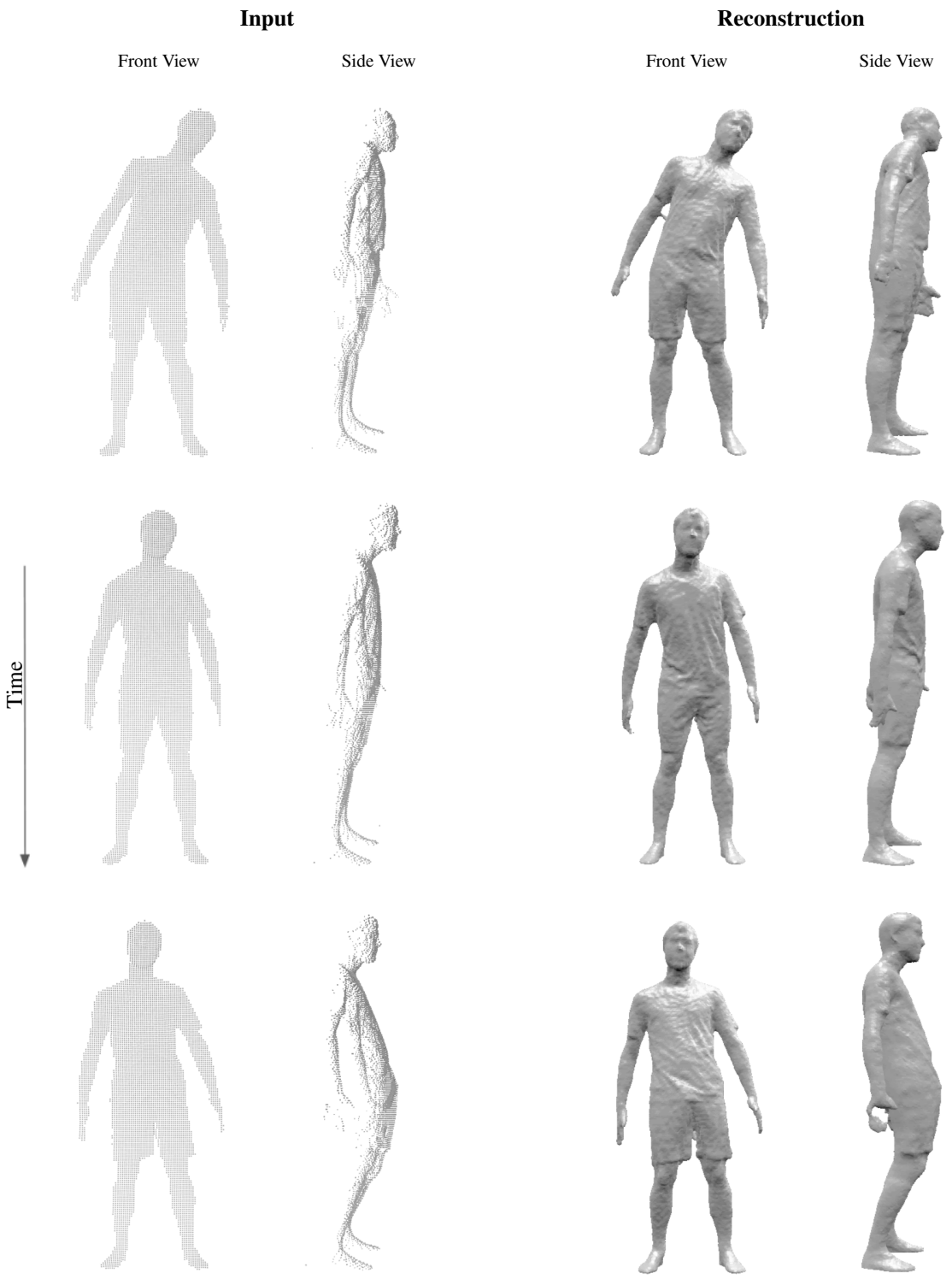

Figure 5. Continuation of Figure 4. Single-view point clouds and reconstructions of subsequent frames.

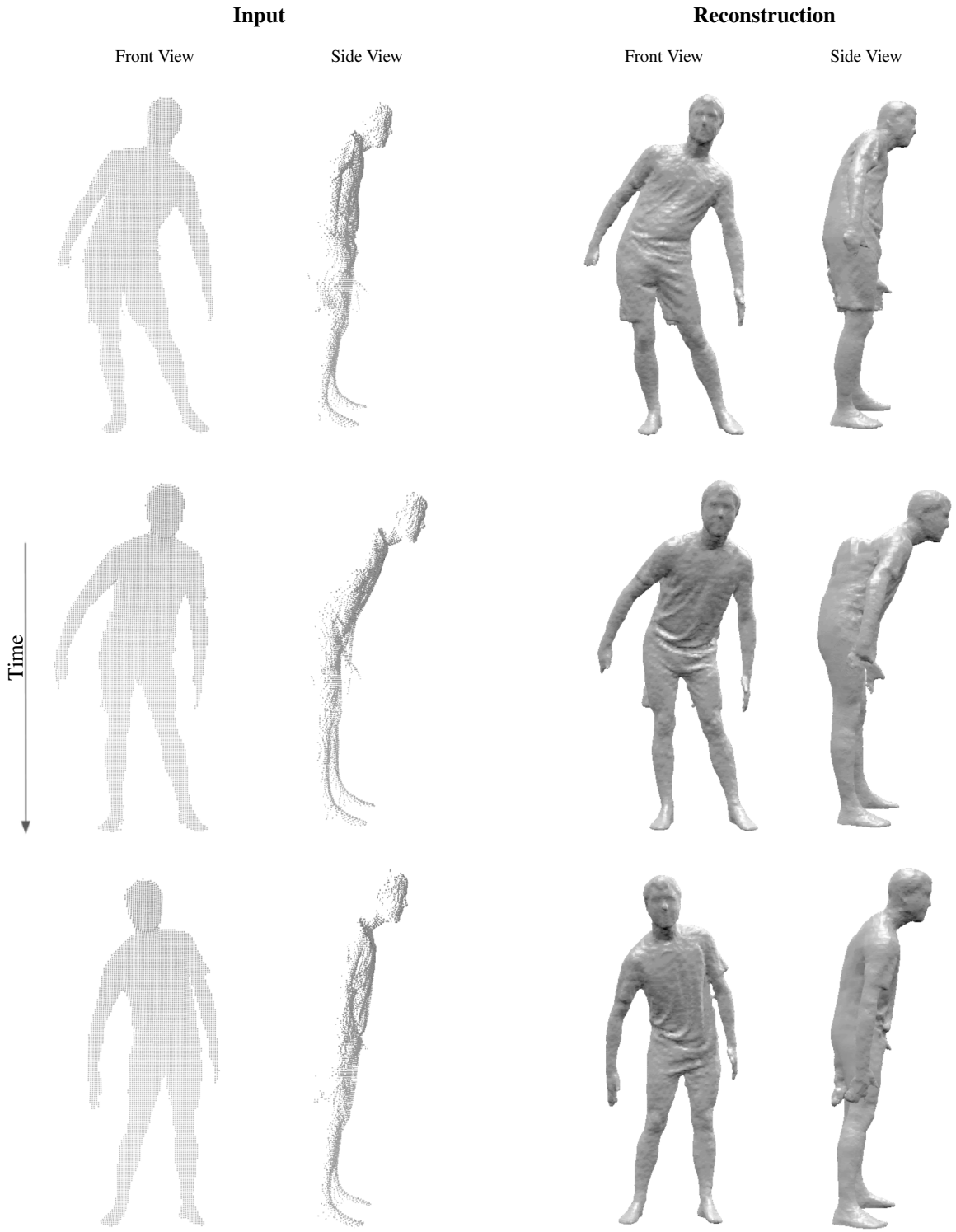

Figure 6. Continuation of Figure 4 and 5. Single-view point clouds and reconstructions of subsequent frames.