# AutoAvatar: Autoregressive Neural Fields for Dynamic Avatar Modeling

Ziqian Bai[1,2*]    Timur Bagautdinov[2]    Javier Romero[2]    Michael Zollhöfer[2]
Ping Tan[1]    Shunsuke Saito[2]

[1]Simon Fraser University    [2]Reality Labs Research

**Abstract.** Neural fields such as implicit surfaces have recently enabled avatar modeling from raw scans without explicit temporal correspondences. In this work, we exploit autoregressive modeling to further extend this notion to capture dynamic effects, such as soft-tissue deformations. Although autoregressive models are naturally capable of handling dynamics, it is non-trivial to apply them to implicit representations, as explicit state decoding is infeasible due to prohibitive memory requirements. In this work, for the first time, we enable autoregressive modeling of implicit avatars. To reduce the memory bottleneck and efficiently model dynamic implicit surfaces, we introduce the notion of articulated observer points, which relate implicit states to the explicit surface of a parametric human body model. We demonstrate that encoding implicit surfaces as a set of height fields defined on articulated observer points leads to significantly better generalization compared to a latent representation. The experiments show that our approach outperforms the state of the art, achieving plausible dynamic deformations even for unseen motions. https://zqbai-jeremy.github.io/autoavatar.

## 1 Introduction

Animatable 3D human body models are key enablers for various applications ranging from virtual try-on to social telepresence [4]. While modeling of human avatars from 3D scans without surface registration is gaining more and more attention in recent years [43,24,48,8,26], complex temporal dynamics are often completely ignored and the resulting deformations are often treated exclusively as a function of the pose parameters. However, the body shape is not uniquely determined by the current pose of the human, but also depends on the history of shape deformations due to secondary motion effects. The goal of our work is to realistically model these history-dependent dynamic effects for human bodies without requiring precise surface registration.

To this end, we propose AutoAvatar, a novel autoregressive model for dynamically deforming human bodies. AutoAvatar models body geometry implicitly - using a signed distance field (SDF) - and is able to directly learn from raw scans without requiring temporal correspondences for supervision. In addition, akin to physics-based simulation, AutoAvatar infers the complete shape of an avatar given history of shape and motion. The
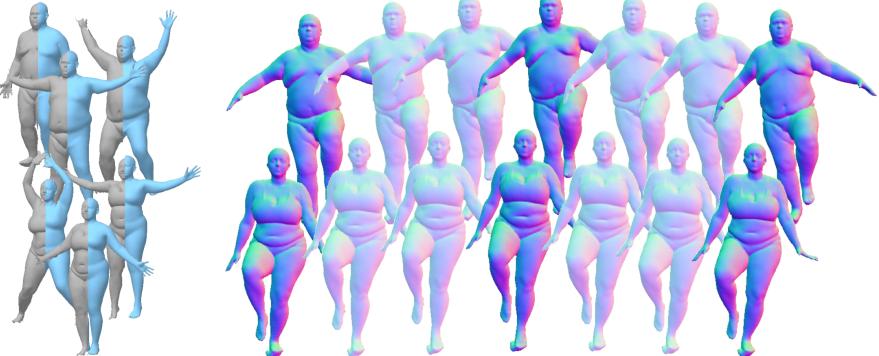
---

Fig. 1: **AutoAvatar.** Given raw 4D scans with self-intersections, holes, and noise (grey meshes) and fitted SMPL models (blue meshes), AutoAvatar automatically learns highly detailed animatable body models with plausible secondary motion dynamics without requiring a personalized template or surface registration (right).

aforementioned properties lead to a generalizable method that models complex dynamic effects including inertia and elastic deformations without requiring a personalized template or precise temporal correspondences across training frames.

To model temporal dependencies in the data, prior work has typically resorted to autoregressive models [39,22,28,45]. While the autoregressive framework naturally allows for incorporation of temporal information, combining it with neural implicit surface representations [29,36,9] for modeling human bodies is non-trivial. Unlike explicit shape representations, such neural representations implicitly encode the shape in the parameters of the neural network and latent codes. Thus, in practice, producing the actual shape requires expensive neural network evaluation at each voxel of a dense spatial grid [36]. This aspect is particularly problematic for autoregressive modeling, since most of the successful autoregressive models rely on rollout training [28,19] to ensure stability of both training and inference. Unfortunately, rollout training requires multiple evaluations of the model for each time step, and thus becomes prohibitively expensive both in terms of memory and compute as the resolution of the spatial grid grows. Another approach would be learning an autoregressive model using latent embeddings that encode dynamic shape information [15]. However, it is infeasible to observe the entire span of possible surface deformations from limited real-world scans, which makes the model prone to overfitting and leads to worse generalization at test time.

By addressing these limitations, we, for the first time, enable autoregressive training of a full-body geometry model represented by a neural implicit surface. To tackle the scalability issues of rollout training for implicit representations, we introduce the novel notion of articulated observer points. Intuitively, articulated observer points are temporally coherent locations on the human body surface which store the dynamically changing state of the implicit function. In practice, we parameterize the observer points using the underlying body model [22], and then represent the state of the implicit surface as signed heights with respect to the vertices of the pose-dependent geometry produced by the articulated model (see Fig. 3a). The number of query points is significantly lower than the number of voxels in a high-resolution grid, which allows for a significant reduction

in terms of memory and compute requirements, making rollout training tractable for implicit surfaces. In addition, we demonstrate that explicitly encoding shapes as signed height fields is less prone to overfitting compared to latent embeddings, a common way to represent autoregressive states [19,52].

Our main contributions are the following:

- The first autoregressive approach for modeling history-dependent implicit surfaces of human bodies,
- Articulated observer points to enable autogressive training of neural fields, and
- Extensive experiments showing that our approach outperforms existing methods both on shape interpolation and extrapolation tasks.

## 2    Related Work

*Parametric Human Models*  Since the anatomical structure of humans is shared across identities, various methods have been proposed to parameterize shape and pose of human bodies from large-scale 3D scan data [3,13,22,33,54,1]. SCAPE [3,13] learns statistical human model models using triangle deformations. The pioneering work by Allen et al. [2] used a vertex-based representation enhanced with pose-dependent deformations, but the model was complex and trained with insufficient data, resulting in overfitting. SMPL [22] improved the generalizability of [2] by training on more data and removing the shape dependency in the pose-dependent deformations. More recent works show that sparsity in the pose correctives reduces spurious correlations [33], and that non-linear deformation bases parameterized by neural networks achieve better modeling accuracy [54]. While most works focus on modeling static human bodies under different poses, Dyna [39] and DMPL (Dynamic SMPL) [22] enable parametric modeling of dynamic deformations by learning a linear autoregressive model. Kim et al. [16] combine a volumetric parametric model, VSMPL, with an external layer driven by the finite element method to enable soft tissue dynamics. SoftSMPL [45] learns a more powerful recurrent neural network to achieve better generalization to unseen subjects. Xiang et al. [52] model dynamically moving clothing from a history of poses. Importantly, the foundation of the aforementioned works is accurate surface registration of a template body mesh [6,7], which remains non-trivial. Habermann et al. [12] also model dynamic deformations from a history of poses. While they relax the need of registration by leveraging image-based supervision, a personalized template is still required as a preprocessing step.

Recently, neural networks promise to enable the modeling of animatable bodies without requiring surface registration or a personalized template [10,24,43,48,8]. These methods leverage structured point clouds [24,26,56] or 3D neural fields [53] to learn animatable avatars. Approaches based on neural fields parameterize human bodies as compositional articulated occupancy networks [10] or implicit surface in canonical space with linear blend skinning [30,43,8,51] and deformation fields [48,35]. Since implicit surfaces do not require surface correspondences for training, avatars can be learned from raw scans. Similarly, neural radiance fields [31] have been applied to body modeling to build animatable avatars from multi-view images [37,20]. However, these approaches represent avatars as a function of only pose parameters, and thus are unable to model
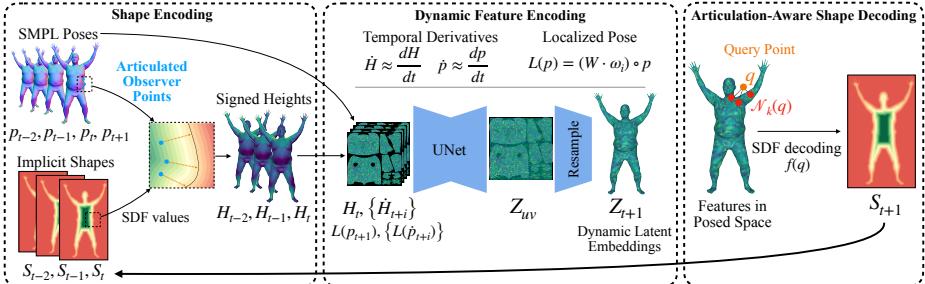
**Fig. 2: Overview.** AutoAvatar learns a pose-driven animatable human body model with plausible dynamics including secondary motions. Notice that our approach takes the history of implicit shapes in an autoregressive manner for learning dynamics.

dynamics. While our approach is also based on 3D neural fields to eliminate the need for surface registration, our approach learns not only pose-dependent deformations but also history-dependent dynamics by enabling autoregressive training of neural implicit surfaces.

*Learning Dynamics*  Traditionally, physics-based simulation [46] is used to model dynamics of objects. While material parameters of physics simulation can be estimated from real data [5,50,47,55], accurately simulating dynamic behavior of objects remains an open question. In addition, authenticity of physics-based simulation is bounded by the underlying model, and complex anisotropic materials such as the human body are still challenging to model accurately. For this reason, several works attempt to substitute a deterministic physics-based simulation with a learnable module parameterized by neural networks [14,57,44,38]. Such approaches have been applied to cloth simulation [14,38], fluid [44], and elastic bodies [57]. Subspace Neural Physics [14] learns a recurrent neural network from offline simulation to predict the simulation state in a subspace. Deep Emulator [57] first learns an autoregressive model to predict deformations using a simple primitive (sphere), and applies the learned function to more complex characters. While we share the same spirit with the aforementioned works by learning dynamic deformations in an autoregressive manner, our approach fundamentally differs from them. The aforementioned approaches all assume that physical quantities such as vertex positions are observable with perfect correspondence in time, and thus results are only demonstrated on synthetic data. In contrast, we learn dynamic deformations from real-world observation while requiring only coarse temporal guidance by the fitted SMPL models. This property is essential to model faithful dynamics of real humans.

## 3   Method

Our approach is an autoregressive model, which takes as inputs human poses and a shape history and produces the implicit surface for a future frame. Fig. 2 shows the overview of our approach. Given a sequence of $T$ implicitly encoded shapes $\{S_{t-T+1}, ..., S_t\}$ and $T+1$ poses $\{p_{t-T+1}, ..., p_{t+1}\}$ with $t$ being the current time frame, our model predicts the implicit surface $S_{t+1}$ of the future frame $t+1$. The output shape $S_{t+1}$ is
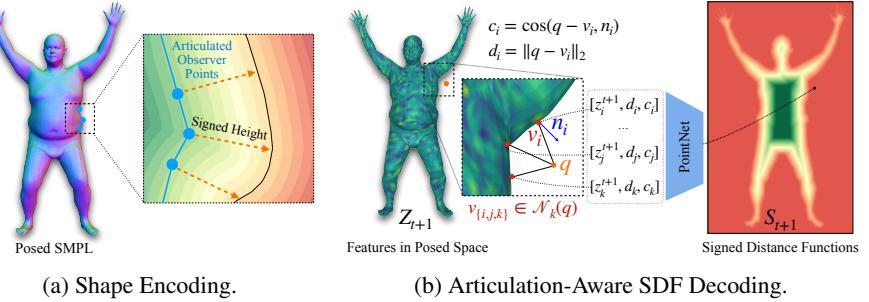
Fig. 3: **Shape Encoding/Decoding.** Our novel shape encoding via articulated observer points and articulated-aware SDF decoding lead to faithful modeling of dynamics.

then passed as an input to the next frame prediction in an autoregressive manner. Our model is supervised directly with raw body scans, and requires a training dataset of 4D scans (sequences of 3D scans) along with fitted SMPL body models [22]. Unfortunately, explicitly representing shapes $S_t$ as levelsets of implicit surface is prohibitively expensive for end-to-end training. To this end, we introduce the concept of *articulated observer points* - vertex locations on the underlying articulated model - which are used as a local reference for defining the full body geometry. The underlying implicit surface is encoded as a height field with respect to the articulated observer points (Sec. 3.1). Given a history of height fields and pose parameters, we convert those to dynamic latent feature maps in UV space (Sec. 3.2). Finally, we map the resulting features to SDFs by associating continuous 3D space with the learned features on the SMPL vertices, which are directly supervised by point clouds with surface normals (Sec. 3.3).

## 3.1 Shape Encoding via Articulated Observer Points

The core of our approach is an autoregressive model that operates on implicit neural surfaces, allowing us to incorporate temporal shape information necessary for modeling challenging dynamics. The key challenge that arises when training such autoregressive models is finding a way to encode the shape - parameterized implicitly as a neural field - into a representation that can be efficiently computed and fed back into the model. The most straightforward way is to extract an explicit geometry representation by evaluating the neural field on a dense spatial grid and running marching cubes [23]. However, in practice this approach is infeasible due to prohibitive memory and computational costs, in particular due to the cubic scaling with respect to the grid dimensions. Instead, we propose to encode the state of the implicit surface into a set of observer points.

Encoding geometry into discrete point sets has been shown to be efficient and effective for learning shape representations from point clouds [40]. Prokudin et al. [40] relies on a fixed set of randomly sampled observer points in global world coordinates, which is not suitable for modeling dynamic humans due to the articulated nature of human motion. Namely, a model relying on observer points with a fixed 3D location needs to account for extremely large shape variations including rigid transformations, making the learning task difficult. Moreover, associating randomly sampled 3D points with a parametric human body is non-trivial. To address these limitations, we further extend the

notion of observer points to an articulated template represented by the SMPL model [22], which provides several advantages for modeling dynamic articulated geometries. In particular, soft-tissue dynamic deformations appear only around the minimally clothed body, and we can rely on this notion as an explicit prior to effectively allocate observer points only to the relevant regions. In addition, the SMPL model provides a mapping of 3D vertices to a common UV parameterization, allowing us to effectively process shape information using 2D CNNs in a temporally consistent manner.

More specifically, to encode the neural implicit surface into the articulated observer points, we compute "signed heights" $H = \{h_i\}_{i=1}^{M} \in \mathbb{R}^M$ from $M$ vertices on a fitted SMPL model. For each vertex, the signed height $h_i$ is the signed distance from the vertex to the zero-crossing of the implicit surface along the vertex normal (see Fig. 3a). We use the iterative secant method as in [32] to compute the zero-crossings. Note that there can be multiple valid signed heights per vertex since the line along the normal can hit the zero-crossing multiple times. Based on the observation that the SMPL vertices are usually close to the actual surface with their normals roughly facing into the same direction, we use the minimum signed height within a predefined range $[h_{min}, h_{max}]$ (in our experiments, we use $h_{min} = -2cm, h_{max} = 8cm$). If no zero-crossing is found inside this range, we set the signed height to $h_{min}$. Note that the computed heights are signed because the fitted SMPL can go beyond the actual surface due to its limited expressiveness and inaccuracy in the fitting stage.

### 3.2   Dynamic Feature Encoding

The essence of AutoAvatar is an animatable autoregressive model. In other words, a reconstructed avatar is driven by pose parameters, while secondary dynamics is automatically synthesized from the history of shapes and poses. To enable this, we learn a mapping that encodes the history of shape and pose information to latent embeddings containing the shape information of the future frame. More specifically, denoting the current time frame as $t$, we take as input $T + 1$ poses $\{p_{t-T+1}, ..., p_{t+1}\}$ and $T$ signed heights vectors $\{H_{t-T+1}, ..., H_t\}$, and produce dynamic features $Z_{t+1} \in \mathbb{R}^{M \times C}$. Given these inputs, we also compute the temporal derivatives of poses $\{\dot{p}_{t+i}\}_{i=-T+2}^{1}$ and signed heights $\{\dot{H}_{t+i}\}_{i=-T+2}^{0}$ as follows:

$$\dot{p}_k = p_k p_{k-1}^{-1}$$
$$\dot{H}_k = H_k - H_{k-1}. \tag{1}$$

Note that in practice $p_*$ are represented as quaternions, and $p_k p_{k-1}^{-1}$ is computed by first converting multipliers to rotation matrices, multiplying those, and then converting the product back to quaternions. To emphasize small values in $\dot{H}$, we apply the following transformation $g(x) = \text{sign}(x) \cdot \ln(\alpha|x| + 1) \cdot \beta$, where $\alpha = 1000$ and $\beta = 0.25$. Following prior works [43,4], we also localize pose parameters to reduce long range spurious correlations as follows:

$$L(p) = (W \cdot \omega_i) \circ p, \tag{2}$$

where $\circ$ denotes the element-wise product, $i$ is the vertex index, $W \in \mathbb{R}^{J \times J}$ is an association matrix of $J$ joints, and $\omega_i \in \mathbb{R}^{J \times 1}$ is the skinning weights of the $i$-th vertex.

We set $W_{n,m} = 1$ if the $n$-th joint is within the 1-ring neighborhood of the $m$-th joint (otherwise $W_{n,m} = 0$). Note that the derivative of the root transformation is included in $\{L(\dot{\boldsymbol{p}}_{t+i})\}$ without localization. Finally, we map $\boldsymbol{H}_t$, $\{\dot{\boldsymbol{H}}_{t+i}\}$, $L(\boldsymbol{p}_{t+1})$, and $\{L(\dot{\boldsymbol{p}}_{t+i})\}$ to UV space using barycentric interpolation. The concatenated features are fed into a UNet [42] to generate a feature map $\boldsymbol{Z}_{uv}$. We then resample $\boldsymbol{Z}_{uv}$ on the UV coordinates corresponding to SMPL vertices to obtain the per-vertex dynamic latent embeddings $\boldsymbol{Z}$. We empirically found that incorporating temporal derivatives further improves the realism of dynamics (see Supp. Mat. video for comparison).

### 3.3 Articulation-Aware Shape Decoding

Given the dynamic feature $\boldsymbol{Z}_{t+1} = \{z_1^{t+1}, ..., z_M^{t+1}\}$ and a query point $\boldsymbol{q}$, we decode signed distance fields $f(\boldsymbol{q})$ to obtain the surface geometry of the dynamic avatar. Several methods model the implicit surface in canonical space by jointly learning a warping function from the posed space to the canonical space [43,8]. However, we observe that the canonicalization step is very sensitive to small fitting error in the SMPL model, and further amplifies the error in the canonical space, making it difficult to learn dynamics (see discussion in Sec. 4). Therefore, we directly model the implicit surface in a posed space while being robust to pose changes. Inspired by Neural Actor [20], we associate a queried 3D point with a human body model and pose-agnostic spatial information. Specifically, Neural Actor uses height from the closest surface point on the SMPL model to the query location together with a feature vector sampled on the same surface point. However, we find that their approach based on the single closest point leads to artifacts around body joints (e.g., armpits) for unseen poses. To better distinguish regions with multiple body parts, we instead use $k$-nearest neighbor vertices. Fig. 3b shows the illustration of our SDF decoding approach. Given a query point $\boldsymbol{q}$, we first compute the k-nearest SMPL vertices $\{\boldsymbol{v}_i\}_{i \in \mathcal{N}_k(\boldsymbol{q})}$, where $\mathcal{N}_k(\boldsymbol{q})$ is a set of indices of k-nearest neighbor vertices. To encode pose-agnostic spatial information, we use rotation-invariant features. Specifically, we compute the distance $d_i = \|\boldsymbol{q} - \boldsymbol{v}_i\|_2$ and cosine value $c_i = \cos(\boldsymbol{x}_i, \boldsymbol{n}_i)$, where $\boldsymbol{x}_i$ is the vertex-to-query vector $\boldsymbol{x}_i = \boldsymbol{q} - \boldsymbol{v}_i$, and $\boldsymbol{n}_i$ is the surface normal on $\boldsymbol{v}_i$. We feed the concatenated vector $[z_i^{t+1}, d_i, c_i]$ into a PointNet-like [41] architecture to compute the final SDFs with the max pooling replaced by a weighted average pooling based on jointly predicted weights for better continuity.

As in [43], we employ implicit geometric regularization (IGR) [11] to train our model directly from raw scans without requiring watertight meshes. Note that in contrast, other methods [30,8,48] require watertight meshes to compute ground-truth occupancy or signed distance values for training. Our final objective function $L$ is the following:

$$L = L_s + L_n + \lambda_{igr} L_{igr} + \lambda_o L_o, \tag{3}$$

where $\lambda_{igr} = 1.0, \lambda_o = 0.1$. $L_s$ promotes SDFs which vanish on the ground truth surface, while $L_n$ encourages that its normal align with the ones from data: $L_s = \sum_{\boldsymbol{q} \in \boldsymbol{Q}_s} |f(\boldsymbol{q})|$, $L_n = \sum_{\boldsymbol{q} \in \boldsymbol{Q}_s} \|\nabla_{\boldsymbol{q}} f(\boldsymbol{q}) - \boldsymbol{n}(\boldsymbol{q})\|_2$, where $\boldsymbol{Q}_s$ is the surface of the input raw scans. $L_{igr}$ is the Eikonal regularization term [11] that encourages the function $f$ to satisfy the Eikonal equation: $L_{igr} = \mathbb{E}_{\boldsymbol{q}} (\|\nabla_{\boldsymbol{q}} f(\boldsymbol{q})\|_2 - 1)^2$, and $L_o$ prevents off-surface SDF values from being too close to the zero-crossings as follows: $L_o = \mathbb{E}_{\boldsymbol{q}} (\exp(-\gamma \cdot |f(\boldsymbol{q})|))$, where $\gamma = 50$.

### 3.4   Implementation Details

**Network Architectures.** In our experiments, we use a UV map of resolution $256 \times 256$, $T = 3$, and $k = 20$. To reduce the imbalance of SMPL vertex density for k-NN computation, we use 3928 points subsampled by poisson-disk sampling on the SMPL mesh. Before being fed into the UNet, $L(\boldsymbol{p}_{k+1})$ and $\{L(\dot{\boldsymbol{p}}_{t+i})\}$ are compressed to 32 channels using $1 \times 1$ convolutions. The UNet uses convolution and transposed convolution layers with untied biases, a kernel size of 3, no normalization, and LeakyReLU with a slope of 0.2 as the non-linear activation, except for the last layer which uses TanH. The SDF decoder is implemented as an MLP, which takes as input 64-dim features from the UNet, positional encoded $d_i$ and $c_i$ up to 4-th order Fourier features. The number of intermediate neurons in the first part of the MLP is $(128, 128, 129)$, where the output is split into a 128-dim feature vector and a 1-dim scalar, which is converted into non-negative weights by softmax across the k-NN samples. After weighted average pooling, the aggregated feature is fed into another MLP with a neuron size of $(128, 128, 1)$ to predict the SDF values. The MLPs use Softplus with $\beta = 100$ and a threshold of 20 as non-linear activation except for the last layer which does not apply any activation.

**Training.** Our training consists of two stages. First, we train our model using ground-truth signed heights without rollout for 90000 iterations. Then, we finetune the model using a rollout of 2 frames for another 7500 iterations to reduce error accumulation for both training and inference. We use the Adam optimizer with a learning rate of $1.0 \times 10^{-4}$ $(1.0 \times 10^{-5})$ at the first (second) stage. To compute $L_n$, we sample $10000(1000)$ points on the scan surface. Similarly, for $L_{igr}$, we sample $10000(1000)$ points around the scan surface by adding Gaussian noise with standard deviation of 10cm to uniformly sampled surface points, and sample $2000(500)$ points within the bounding box around the raw scans. The points uniformly sampled inside the bounding box are also used to compute $L_o$. Both stages are trained with a batch size of 1.

**Inference.** At the beginning of the animations, we assume ground-truth raw scans are available for the previous $T$ frames for initialization. If no ground truth initial shape is available, we initialize the first $T$ frames with our baseline model conditioned only on pose parameters. Note that the scan data is extremely noisy around the hand and foot areas, and the SMPL fitting of the head region is especially inaccurate. Therefore, we fix the dynamic features on the face, hands, and feet to the ones of the first frame.

## 4   Experimental Results

### 4.1   Datasets and Metrics

**Datasets**. We use the DFaust dataset [7] for both training and quantitative evaluation, and AIST++ [49,18] for qualitative evaluation on unseen motions. For the DFaust dataset, we choose 2 subjects (50002 and 50004), who exhibit the most soft-tissue deformations. The interpolation test evaluates the fidelity of dynamics under the same type of motions as in training but at different time instance, and the extrapolation test evaluates performance on unseen motion. For 50002, we use the 2nd half of `chicken_wings` and `running_on_spot` for the interpolation test, `one_leg_jump` for the extrapolation test, and the rest for training. For 50004, we use the 2nd half of

`chicken_wings` and `running_on_spot` for interpolation, `one_leg_loose` for extrapolation, and the rest for training. The fitted SMPL parameters in DFaust are provided by the AMASS [27] dataset that uses sparse points on the registered data as approximated mocap marker locations and computes the parameters using MoSh [21]. Note that more accurate pose can be obtained by using all the registration vertices (see Appendix A), but this is not required by our method to recover soft-tissue deformation.

**Metrics**. For evaluation, we extract the 0-level set surface at each time step using Marching Cubes [23] with a resolution of $256^3$. We also use simplified scans with around 10000 vertices and outlier points (distance to the nearest SMPL vertex larger than 10cm) have been removed. We evaluate the accuracy of the predicted surface in terms of its position and dynamics accuracy. The surface position accuracy is measured by averaging the distance from each simplified scan vertex to the closest prediction surface point. Evaluating the dynamics accuracy of the implicit surface efficiently is more challenging. We approximate the local occupied volume as a scalar per registration vertex representing the ratio of surrounding points contained in the interior of the (ground-truth or inferred) surface. We use 10 points uniformly sampled inside a 5cm cube centered at the vertex. The head, hands and feet vertices are ignored due to their high noise levels. The temporal difference of this scalar across adjacent frames can be interpreted as a dynamic measure of the local volume evolution. We report the mean square difference between this dynamic descriptor as computed with the ground truth simplified scan and the inferred implicit surface. Since a small phase shift in dynamics may lead to large cumulative error, reporting only the averaged errors from the entire frames can be misleading. Therefore, we report errors along the progression of rollout predictions. For each evaluation sequence, we start prediction every 20 frames (i.e., 20th frame, 40th frame, ...), and use the ground truth pose and shape history only for the first frame, followed by the autoregressive predictions for the error computation. In Tab. 1, we report the averaged errors for both metrics after 1, 2, 4, 8, 16, and 30 rollouts. The errors for small number of rollouts evaluate the accuracy of future shape prediction given the ground-truth shape history, whereas the errors with longer rollouts evaluate the accumulated errors by autoregressively taking as input the predictions of previous frames. We discuss the limitation of error metrics with longer rollouts in Sec. 4.2.

## 4.2   Evaluation

In this section, we provide comprehensive analysis to validate our design choices and highlight the limitations of alternative approaches and SoTA methods based on both implicit and explicit shape representations. Note that all approaches use the same training set, and are trained with the same number of iterations as our method for fair comparison.

**Effectiveness of Autoregressive Modeling.** While autoregressive modeling is a widely used technique for learning dynamics [39,57,38], several recent methods still employ only the history of poses for modeling dynamic avatars [52,12]. Thus, to evaluate the effectiveness of autoregressive modeling we compare AutoAvatar with pose-dependent alternatives that use neural implicit surfaces. More specifically, we design the following 3 non-autoregressive baselines:

Table 1: **Quantitative Comparison with Baseline Methods.** Our method produces the most accurate predictions of the future frames given the ground-truth shape history among all baseline methods (see rollout 1-4). For longer rollouts, more dynamic predictions lead to higher error than less dynamic results due to high sensitivity to initial conditions in dynamic systems [34] (see discussion in Sec. 4.2).

(a) Mean Scan-to-Prediction Distance (mm) ↓ on DFaust.

| | | Rollout (# of frames) | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 4 | 8 | 16 | 30 |
| *Interpolation Set* | | | | | | | |
| Non-Autoregressive | SNARF [8] | 7.428 | 7.372 | 7.337 | 7.476 | 7.530 | 7.656 |
| | Pose | 4.218 | 4.202 | 4.075 | 4.240 | 4.409 | 4.426 |
| | PoseTCN | 4.068 | 4.118 | 4.086 | 4.228 | 4.405 | 4.411 |
| | Pose + dPose | 3.852 | 3.841 | 3.764 | 3.972 | 4.164 | 4.156 |
| Autoregressive | G-embed | 2.932 | 3.006 | 3.131 | 3.462 | 3.756 | 3.793 |
| | L-embed | 1.784 | 2.138 | 2.863 | 4.250 | 5.448 | 5.916 |
| | Ours | 1.569 | 1.914 | 2.587 | 3.627 | 4.736 | 5.255 |
| *Extrapolation Set* | | | | | | | |
| Non-Autoregressive | SNARF [8] | 7.264 | 7.287 | 7.321 | 7.387 | 7.308 | 7.251 |
| | Pose | 4.303 | 4.306 | 4.308 | 4.299 | 4.385 | 4.398 |
| | PoseTCN | 4.090 | 4.091 | 4.105 | 4.119 | 4.233 | 4.257 |
| | Pose + dPose | 3.984 | 3.991 | 4.017 | 4.063 | 4.162 | 4.190 |
| Autoregressive | G-embed | 2.884 | 2.926 | 3.043 | 3.258 | 3.577 | 3.787 |
| | L-embed | 1.329 | 1.539 | 2.079 | 3.326 | 4.578 | 5.192 |
| | Ours | 1.150 | 1.361 | 1.834 | 2.689 | 3.789 | 4.526 |

(b) Mean Squared Error of Volume Change ↓ on DFaust.

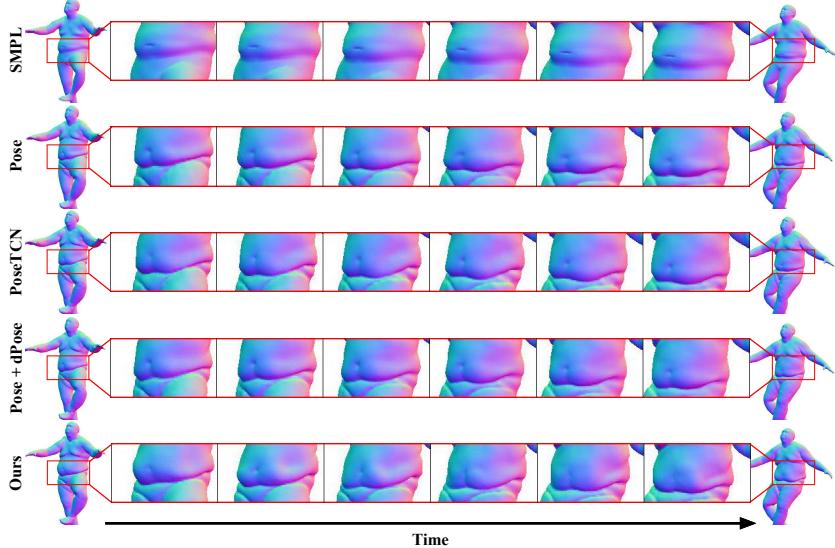| | | Rollout (# of frames) | | | | |
|---|---|---|---|---|---|---|
| | | 2 | 4 | 8 | 16 | 30 |
| *Interpolation Set* | | | | | | |
| Non-Autoregressive | SNARF [8] | 0.01582 | 0.01552 | 0.01610 | 0.01658 | 0.01682 |
| | Pose | 0.01355 | 0.01305 | 0.01341 | 0.01367 | 0.01387 |
| | PoseTCN | 0.01364 | 0.01323 | 0.01350 | 0.01399 | 0.01416 |
| | Pose + dPose | 0.01288 | 0.01247 | 0.01273 | 0.01311 | 0.01321 |
| Autoregressive | G-embed | 0.01179 | 0.01168 | 0.01199 | 0.01248 | 0.01265 |
| | L-embed | 0.01003 | 0.01180 | 0.01466 | 0.01716 | 0.01844 |
| | Ours | 0.00902 | 0.01053 | 0.01258 | 0.01456 | 0.01565 |
| *Extrapolation Set* | | | | | | |
| Non-Autoregressive | SNARF [8] | 0.01178 | 0.01194 | 0.01251 | 0.01228 | 0.01206 |
| | Pose | 0.01027 | 0.01039 | 0.01074 | 0.01052 | 0.01039 |
| | PoseTCN | 0.01020 | 0.01038 | 0.01064 | 0.01040 | 0.01029 |
| | Pose + dPose | 0.00992 | 0.01014 | 0.01048 | 0.01029 | 0.01013 |
| Autoregressive | G-embed | 0.00936 | 0.00959 | 0.00995 | 0.00996 | 0.00998 |
| | L-embed | 0.00648 | 0.00821 | 0.01100 | 0.01308 | 0.01402 |
| | Ours | 0.00567 | 0.00715 | 0.00915 | 0.01039 | 0.01107 |

Fig. 4: **Qualitative Comparison with Non-Autoregressive Baselines.** In contrast to the rigid results in non-autoregressive baselines, our approach produces high quality non-rigid dynamics.

1. Pose: We only feed pose parameters of the next frame $L(\boldsymbol{p}_{t+1})$ in our architecture. Prior avatar modeling methods based on neural fields employ this pose-only parameterization [43,48,8].
2. PoseTCN: Temporal convolutional networks (TCN) [17] support the incorporation of a long-range history for learning tasks, and have been used in several avatar modeling methods [52,12]. Thus, we use a TCN that takes as input the sequence of poses with the length of 16. We first compute localized pose parameters, as in our method, for each frame and apply the TCN to obtain 64-dim features for each SMPL vertex. The features are then fed into the UNet and SDF decoders identical to our method.
3. Pose+dPose: Our approach without autoregressive components ($\boldsymbol{H}_t$, $\{\dot{\boldsymbol{H}}_{t+i}\}$).

Tab. 1 shows that our approach outperforms the baseline methods for the first 8 frames for interpolation, and first 16 frames for extrapolation. In particular, there is a significantly large margin for the first 4-8 frames, indicating that our method achieves the most accurate prediction of the future frames given the ground-truth shape history. We also observe that the non-autoregressive methods tend to collapse to predicting the "mean" shape under each pose without faithful dynamics for unseen motions (see Fig. 4 and Supp. Mat. video). Since the accumulation of small errors in each frame may lead to large deviations from the ground-truth due to high sensitivity to initial conditions in dynamic systems [34], for longer rollouts mean predictions without any dynamics can produce lower errors than more dynamic predictions. In fact, although our method leads to slightly higher errors on longer rollouts, Fig. 4 clearly shows that our approach produces the most visually plausible dynamics on the AIST++ sequences. Importantly, we do not observe any instability or explosions in our autoregressive model for longer rollouts, as can be seen from the error behavior shown in Fig. 7. We also highly encourage
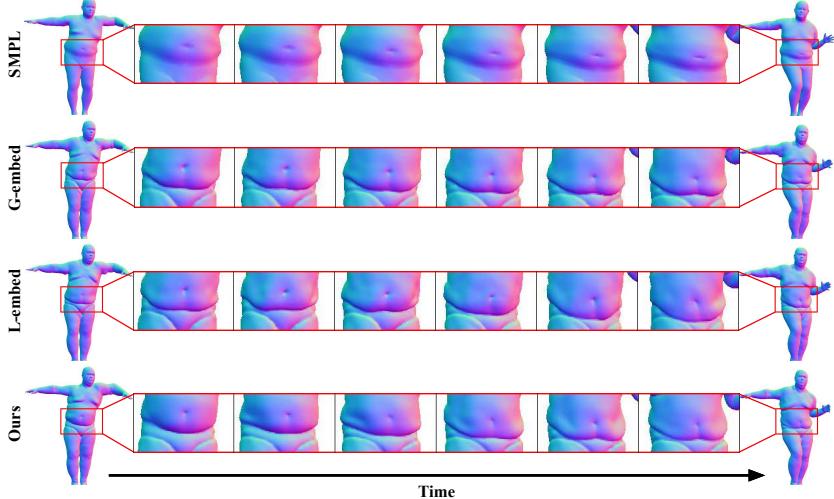
Fig. 5: **Qualitative Comparison with Latent-space Autoregression.** While the latent space-based autoregression approaches suffer either from overfitting to pose parameters (G-embed) or instability (L-embed, see Supp. Mat. video), our approach based on a physically meaningful quantity (signed height) achieves the most stable and expressive synthesis of dynamics.

readers to see Supp. Mat. video for qualitative comparison in animation. In summary, our results confirm that autoregressive modeling plays a critical role for generalization to unseen motions and improving the realism of dynamics.

**Explicit Shape Encoding vs. Latent Encoding.** Efficiently encoding the geometry of implicit surfaces is non-trivial. While our proposed approach encodes the geometry via signed heights on articulated observer points, prior approaches have demonstrated shape encoding based on a learned latent space [36,4]. Therefore, we also investigate different encoding methods for autoregressive modeling with the following 2 baselines:

1. G-embed: Inspired by DeepSDF [36], we first learn per-frame global embeddings $l_g \in \mathbb{R}^{512}$ with the UNet and SDF decoder by replacing $\boldsymbol{H}_t$, $\{\dot{\boldsymbol{H}}_{t+i}\}$, $\{L(\dot{\boldsymbol{p}}_{t+i})\}$ with repeated global embeddings. Then, we train a small MLP with three 512-dim hidden layers using Softplus except for the last layer, taking as input $\boldsymbol{p}_{t+1}$, $\{\dot{\boldsymbol{p}}_{t+i}\}$, and 3 embeddings of previous frames to predict the global embedding at time $t + 1$.

2. L-embed: For modeling mesh-based body avatars, localized embeddings are shown to be effective [4]. Inspired by this, we also train a model with localized embeddings $l_l \in \mathbb{R}^{16 \times 64 \times 64}$. We first learn per-frame local embeddings $l_l$ together with the UNet and SDF decoder by replacing $\boldsymbol{H}_t$, $\{\dot{\boldsymbol{H}}_{t+i}\}$, $\{L(\dot{\boldsymbol{p}}_{t+i})\}$ with bilinearly upsampled $l_l$. Then we train another UNet that takes as input $L(\boldsymbol{p}_{t+1})$, $\{L(\dot{\boldsymbol{p}}_{t+i})\}$, and 3 embeddings of previous frames to predict the localized embeddings at time $t + 1$.

Note that for evaluation, we optimize per-frame embeddings for test sequences using Eq. (3) such that the baseline methods can use the best possible history of embeddings for autoregression. Tab. 1 shows that our method outperforms L-embed in all cases because L-embed becomes unstable for the test sequences. For G-embed, we observe the same trend as for the non-autoregressive baselines: our approach achieves significantly
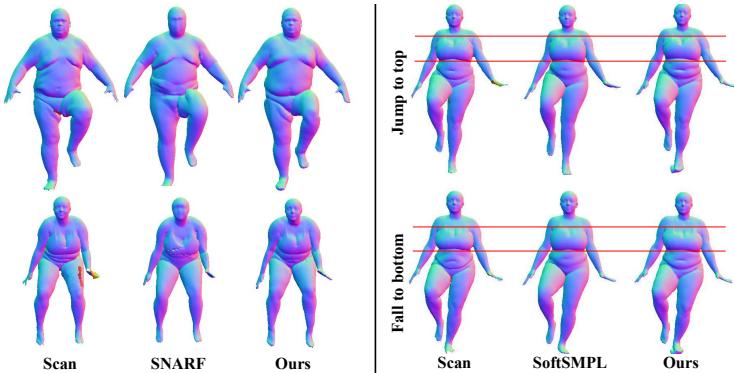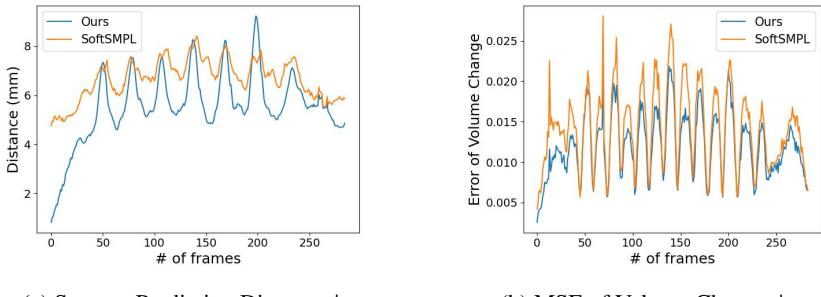
Fig. 6: **Qualitative Comparison with SoTA Methods.** Our approach produces significantly more faithful shapes and dynamics than the state-of-the-art implicit avatar modeling method [8], and shows comparable dynamics with prior art dependent on registrations with fixed topology [45].



(a) Scan-to-Prediction Distance ↓                    (b) MSE of Volume Change ↓

Fig. 7: **Comparison with SoftSMPL [45].** We plot the errors on the sequence of `one_leg_loose` for subject 50004. Surprisingly, our registration-free approach mostly outperforms this baseline that has to rely on registered data with fixed topology.

more accurate predictions of the future frames given the ground-truth trajectories (see the errors for 1-4 rollouts), and G-embed tends to predict "mean" shapes without plausible dynamics. The qualitative comparison in Fig. 5 confirms that our approach produces more plausible dynamics. Please refer to Supp. Mat. video for detailed visual comparison in animation. We summarize that physically meaningful shape encodings (e.g., signed heights) enable more stable learning of dynamics via autoregression than methods relying on latent space.

**Comparison to SoTA Methods..** We compare our approach with state-of-the-art methods for both implicit surface representations and mesh-based representations. As a method using neural implicit surfaces, we choose SNARF [8], which jointly learns a pose-conditioned implicit surface in a canonical T-pose and a forward skinning network for reposing. Similar to ours, SNARF does not require temporal correspondences other than the fitted SMPL models. We use the training code released by the authors using the same training data and the fitted SMPL parameters as in our method. Note that in the DFaust experiment in [8], SNARF is trained using only the fitted SMPL models to DFaust as ground-truth geometry, which do not contain any dynamic deformations.

Tab. 1 shows that our approach significantly outperforms SNARF for any number of rollouts. Interestingly, SNARF can produce dynamic effects for training data by severely overfitting to the pose parameters, but this does not generalize to unseen poses as the learned dynamics is the results of spurious correlations. As mentioned in Sec. 3.3, we also observe that the performance of SNARF heavily relies on the accuracy of the SMPL fitting for canonicalization, and any small alignment errors in the underlying SMPL registration deteriorates their test-time performance (see Fig. 6). Therefore, this experiment demonstrates not only the importance of autoregressive dynamic avatar modeling, but also the efficacy of our articulation-aware shape decoding approach given the quality of available SMPL fitting for real-world scans.

We also compare against SoftSMPL [45], a state-of-the-art mesh-based method that learns dynamically deforming human bodies from registered meshes. The authors of SoftSMPL kindly provide their predictions on the sequence of `one_leg_loose` for subject 50004, which is excluded from training for both our method and SoftSMPL for fair comparison. To our surprise, Fig. 7 show that our results are slightly better on both metrics for the majority of frames, although we tackle a significantly harder problem because our approach learns dynamic bodies directly from raw scans, whereas SoftSMPL learns from the carefully registered data. We speculate that the lower error may be mainly attributed to the higher resolution of our geometry using implicit surfaces in contrast to their predictions on the coarse SMPL topology (see Fig. 6). Nevertheless, this result is highly encouraging as our approach achieves comparable performance on dynamics modeling without having to rely on surface registration.

## 5   Conclusion

We have introduced AutoAvatar, an autoregressive approach for modeling high-fidelity dynamic deformations of human bodies directly from raw 4D scans using neural implicit surfaces. The reconstructed avatars can be driven by pose parameters, and automatically incorporate secondary dynamic effects that depend on the history of shapes. Our experiments indicate that modeling dynamic avatars without relying on accurate registrations is made possible by choosing an efficient representation for our autoregressive model.

**Limitations and Future Work.** While our method has shown to be effective in modeling the elastic deformations of real humans, we observe that it remains challenging, yet promising, to model clothing deformations that involve high-frequency wrinkles (see Appendix C for details). Our evaluation also suggests that ground-truth comparison with longer rollouts may not reliably reflect the plausibility of dynamics. Quantitative metrics that handle the high sensitivity to initial conditions in dynamics could be further investigated. Currently, AutoAvatar models subject-specific dynamic human bodies, but generalizing it to multiple identities, as demonstrated in registration-based shape modeling [39,22,45], is an interesting direction for future work. The most exciting venue for future work is to extend the notion of dynamics to image-based avatars [37,20]. In contrast to implicit surfaces, neural radiance fields [31] do not have an explicit "surface" as they model geometry using density fields. While this remains an open question, we believe that our contributions in this work such as efficiently modeling the state of shapes via articulated observer points might be useful to unlock this application.

# References

1. Alldieck, T., Xu, H., Sminchisescu, C.: imghum: Implicit generative models of 3d human shape and articulated pose. In: Proc. of International Conference on Computer Vision (ICCV). pp. 5461–5470 (2021) 3

2. Allen, B., Curless, B., Popović, Z., Hertzmann, A.: Learning a correlated model of identity and pose-dependent body shape variation for real-time synthesis. In: Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation. pp. 147–156 (2006) 3

3. Anguelov, D., Srinivasan, P., Koller, D., Thrun, S., Rodgers, J., Davis, J.: SCAPE: shape completion and animation of people. ACM Trans. on Graphics (TOG) **24**(3), 408–416 (2005) 3

4. Bagautdinov, T., Wu, C., Simon, T., Prada, F., Shiratori, T., Wei, S.E., Xu, W., Sheikh, Y., Saragih, J.: Driving-signal aware full-body avatars. ACM Trans. on Graphics (TOG) **40**(4), 1–17 (2021) 1, 6, 12

5. Bhat, K.S., Twigg, C.D., Hodgins, J.K., Khosla, P., Popovic, Z., Seitz, S.M.: Estimating cloth simulation parameters from video (2003) 4

6. Bogo, F., Romero, J., Loper, M., Black, M.J.: Faust: Dataset and evaluation for 3d mesh registration. In: Proc. of Computer Vision and Pattern Recognition (CVPR). pp. 3794–3801 (2014) 3

7. Bogo, F., Romero, J., Pons-Moll, G., Black, M.J.: Dynamic faust: Registering human bodies in motion. In: Proc. of Computer Vision and Pattern Recognition (CVPR). pp. 6233–6242 (2017) 3, 8, 19

8. Chen, X., Zheng, Y., Black, M.J., Hilliges, O., Geiger, A.: Snarf: Differentiable forward skinning for animating non-rigid neural implicit shapes. In: Proc. of International Conference on Computer Vision (ICCV) (2021) 1, 3, 7, 10, 11, 13, 19, 20

9. Chen, Z., Zhang, H.: Learning implicit fields for generative shape modeling. In: Proc. of Computer Vision and Pattern Recognition (CVPR). pp. 5939–5948. Computer Vision Foundation / IEEE (2019) 2

10. Deng, B., Lewis, J.P., Jeruzalski, T., Pons-Moll, G., Hinton, G.E., Norouzi, M., Tagliasacchi, A.: NASA neural articulated shape approximation. In: Proc. of European Conference on Computer Vision (ECCV). Lecture Notes in Computer Science, vol. 12352, pp. 612–628. Springer (2020) 3

11. Gropp, A., Yariv, L., Haim, N., Atzmon, M., Lipman, Y.: Implicit geometric regularization for learning shapes. In: Proceedings of the 37th International Conference on Machine Learning (ICML). Proceedings of Machine Learning Research, vol. 119, pp. 3789–3799. PMLR (2020) 7

12. Habermann, M., Liu, L., Xu, W., Zollhoefer, M., Pons-Moll, G., Theobalt, C.: Real-time deep dynamic characters. ACM Trans. on Graphics (TOG) **40**(4), 1–16 (2021) 3, 9, 11

13. Hasler, N., Stoll, C., Sunkel, M., Rosenhahn, B., Seidel, H.: A statistical model of human pose and body shape. Computer Graphics Forum **28**(2), 337–346 (2009) 3

14. Holden, D., Duong, B.C., Datta, S., Nowrouzezahrai, D.: Subspace neural physics: Fast data-driven interactive simulation. In: Proceedings of the 18th annual ACM SIGGRAPH/Eurographics Symposium on Computer Animation. pp. 1–12 (2019) 4

15. Kanazawa, A., Zhang, J.Y., Felsen, P., Malik, J.: Learning 3d human dynamics from video. In: Proc. of Computer Vision and Pattern Recognition (CVPR) (2019) 2

16. Kim, M., Pons-Moll, G., Pujades, S., Bang, S., Kim, J., Black, M.J., Lee, S.: Data-driven physics for human soft tissue animation. ACM Trans. on Graphics (TOG) **36**(4), 54:1–54:12 (2017) 3

17. Lea, C., Vidal, R., Reiter, A., Hager, G.D.: Temporal convolutional networks: A unified approach to action segmentation. In: Proc. of European Conference on Computer Vision (ECCV). pp. 47–54. Springer (2016) 11

18. Li, R., Yang, S., Ross, D.A., Kanazawa, A.: Ai choreographer: Music conditioned 3d dance generation with aist++. In: Proc. of International Conference on Computer Vision (ICCV). pp. 13401–13412 (2021) 8

19. Ling, H.Y., Zinno, F., Cheng, G., Van De Panne, M.: Character controllers using motion vaes. ACM Trans. on Graphics (TOG) **39**(4), 40–1 (2020) 2, 3

20. Liu, L., Habermann, M., Rudnev, V., Sarkar, K., Gu, J., Theobalt, C.: Neural actor: Neural free-view synthesis of human actors with pose control. ACM Trans. on Graphics (TOG) **40**(6), 1–16 (2021) 3, 7, 14, 19

21. Loper, M., Mahmood, N., Black, M.J.: Mosh: Motion and shape capture from sparse markers. ACM Trans. on Graphics (TOG) **33**(6), 1–13 (2014) 9, 19

22. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: SMPL: a skinned multi-person linear model. ACM Trans. on Graphics (TOG) **34**(6), 248:1–248:16 (2015) 2, 3, 5, 6, 14

23. Lorensen, W.E., Cline, H.E.: Marching cubes: A high resolution 3d surface construction algorithm. ACM siggraph computer graphics **21**(4), 163–169 (1987) 5, 9

24. Ma, Q., Saito, S., Yang, J., Tang, S., Black, M.J.: SCALE: Modeling clothed humans with a surface codec of articulated local elements. In: Proc. of Computer Vision and Pattern Recognition (CVPR) (Jun 2021) 1, 3

25. Ma, Q., Yang, J., Ranjan, A., Pujades, S., Pons-Moll, G., Tang, S., Black, M.J.: Learning to dress 3d people in generative clothing. In: Proc. of Computer Vision and Pattern Recognition (CVPR). pp. 6469–6478 (2020) 19

26. Ma, Q., Yang, J., Tang, S., Black, M.J.: The power of points for modeling humans in clothing. In: Proc. of International Conference on Computer Vision (ICCV) (Oct 2021) 1, 3

27. Mahmood, N., Ghorbani, N., Troje, N.F., Pons-Moll, G., Black, M.J.: Amass: Archive of motion capture as surface shapes. In: Proc. of International Conference on Computer Vision (ICCV). pp. 5442–5451 (2019) 9, 19, 20

28. Martinez, J., Black, M.J., Romero, J.: On human motion prediction using recurrent neural networks. In: Proc. of Computer Vision and Pattern Recognition (CVPR). pp. 2891–2900 (2017) 2

29. Mescheder, L.M., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy networks: Learning 3D reconstruction in function space. In: Proc. of Computer Vision and Pattern Recognition (CVPR). pp. 4460–4470. Computer Vision Foundation / IEEE (2019) 2

30. Mihajlovic, M., Zhang, Y., Black, M.J., Tang, S.: LEAP: Learning articulated occupancy of people. In: Proc. of Computer Vision and Pattern Recognition (CVPR) (Jun 2021) 3, 7

31. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: NeRF: Representing scenes as neural radiance fields for view synthesis. In: Proc. of European Conference on Computer Vision (ECCV). Lecture Notes in Computer Science, vol. 12346, pp. 405–421. Springer (2020) 3, 14

32. Niemeyer, M., Mescheder, L., Oechsle, M., Geiger, A.: Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In: Proc. of Computer Vision and Pattern Recognition (CVPR) (2020) 6

33. Osman, A.A.A., Bolkart, T., Black, M.J.: STAR: sparse trained articulated human body regressor. In: Proc. of European Conference on Computer Vision (ECCV). Lecture Notes in Computer Science, vol. 12351, pp. 598–613. Springer (2020) 3

34. Ott, E., Grebogi, C., Yorke, J.A.: Controlling chaos. Physical review letters **64**(11), 1196 (1990) 10, 11

35. Palafox, P., Božič, A., Thies, J., Nießner, M., Dai, A.: Npms: Neural parametric models for 3d deformable shapes. In: Proc. of International Conference on Computer Vision (ICCV). pp. 12695–12705 (2021) 3

36. Park, J.J., Florence, P., Straub, J., Newcombe, R.A., Lovegrove, S.: DeepSDF: Learning continuous signed distance functions for shape representation. In: Proc. of Computer Vision and Pattern Recognition (CVPR). pp. 165–174. Computer Vision Foundation / IEEE (2019) 2, 12

37. Peng, S., Zhang, Y., Xu, Y., Wang, Q., Shuai, Q., Bao, H., Zhou, X.: Neural body: Implicit neural representations with structured latent codes for novel view synthesis of dynamic humans. In: Proc. of Computer Vision and Pattern Recognition (CVPR). pp. 9054–9063 (2021) 3, 14

38. Pfaff, T., Fortunato, M., Sanchez-Gonzalez, A., Battaglia, P.: Learning mesh-based simulation with graph networks. In: International Conference on Learning Representations (2021) 4, 9

39. Pons-Moll, G., Romero, J., Mahmood, N., Black, M.J.: Dyna: A model of dynamic human shape in motion. ACM Trans. on Graphics (TOG) **34**(4), 1–14 (2015) 2, 3, 9, 14

40. Prokudin, S., Lassner, C., Romero, J.: Efficient learning on point clouds with basis point sets. In: Proc. of International Conference on Computer Vision (ICCV). pp. 4332–4341 (2019) 5

41. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proc. of Computer Vision and Pattern Recognition (CVPR). pp. 652–660 (2017) 7

42. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241. Springer (2015) 7

43. Saito, S., Yang, J., Ma, Q., Black, M.J.: SCANimate: Weakly supervised learning of skinned clothed avatar networks. In: Proc. of Computer Vision and Pattern Recognition (CVPR) (Jun 2021) 1, 3, 6, 7, 11

44. Sanchez-Gonzalez, A., Godwin, J., Pfaff, T., Ying, R., Leskovec, J., Battaglia, P.: Learning to simulate complex physics with graph networks. In: International Conference on Machine Learning. pp. 8459–8468. PMLR (2020) 4

45. Santesteban, I., Garces, E., Otaduy, M.A., Casas, D.: Softsmpl: Data-driven modeling of nonlinear soft-tissue dynamics for parametric humans. In: Computer Graphics Forum. vol. 39, pp. 65–75. Wiley Online Library (2020) 2, 3, 13, 14

46. Sifakis, E., Barbic, J.: Fem simulation of 3d deformable solids: a practitioner's guide to theory, discretization and model reduction. In: Acm siggraph 2012 courses, pp. 1–50 (2012) 4

47. Srinivasan, S.G., Wang, Q., Rojas, J., Klár, G., Kavan, L., Sifakis, E.: Learning active quasistatic physics-based models from data. ACM Trans. on Graphics (TOG) **40**(4), 1–14 (2021) 4

48. Tiwari, G., Sarafianos, N., Tung, T., Pons-Moll, G.: Neural-gif: Neural generalized implicit functions for animating people in clothing. In: Proc. of International Conference on Computer Vision (ICCV). pp. 11708–11718 (2021) 1, 3, 7, 11

49. Tsuchida, S., Fukayama, S., Hamasaki, M., Goto, M.: Aist dance video database: Multi-genre, multi-dancer, and multi-camera database for dance information processing. In: ISMIR. vol. 1, p. 6 (2019) 8

50. Wang, H., O'Brien, J.F., Ramamoorthi, R.: Data-driven elastic models for cloth: modeling and measurement. ACM Trans. on Graphics (TOG) **30**(4), 1–12 (2011) 4

51. Wang, S., Mihajlovic, M., Ma, Q., Geiger, A., Tang, S.: Metaavatar: Learning animatable clothed human models from few depth images. Proc. of Advances in Neural Information Processing Systems (NeurIPS) **34** (2021) 3

52. Xiang, D., Prada, F., Bagautdinov, T., Xu, W., Dong, Y., Wen, H., Hodgins, J., Wu, C.: Modeling clothing as a separate layer for an animatable human avatar. ACM Trans. on Graphics (TOG) **40**(6), 1–15 (2021) 3, 9, 11

53. Xie, Y., Takikawa, T., Saito, S., Litany, O., Yan, S., Khan, N., Tombari, F., Tompkin, J., Sitzmann, V., Sridhar, S.: Neural fields in visual computing and beyond. arXiv preprint arXiv:2111.11426 (2021) 3

54. Xu, H., Bazavan, E.G., Zanfir, A., Freeman, W.T., Sukthankar, R., Sminchisescu, C.: GHUM & GHUML: generative 3D human shape and articulated pose models. In: Proc. of Computer Vision and Pattern Recognition (CVPR). pp. 6183–6192. IEEE (2020) 3

55. Yang, S., Liang, J., Lin, M.C.: Learning-based cloth material recovery from video. In: Proc. of International Conference on Computer Vision (ICCV). pp. 4393–4403. IEEE Computer Society (2017) 4

56. Zakharkin, I., Mazur, K., Grigorev, A., Lempitsky, V.: Point-based modeling of human clothing. In: Proc. of International Conference on Computer Vision (ICCV). pp. 14718–14727 (2021) 3

57. Zheng, M., Zhou, Y., Ceylan, D., Barbic, J.: A deep emulator for secondary motion of 3d characters. In: Proc. of Computer Vision and Pattern Recognition (CVPR). pp. 5932–5940 (2021) 4, 9

# Appendix

## A  Analysis on Input Pose Accuracy

We investigate how the accuracy of the input SMPL fitting influences the results on subject 50002 of DFaust [7]. As discussed in Sec. 4.1, the fitted SMPL parameters in DFaust are provided by the AMASS [27] dataset that uses sparse points on the registered data as approximated motion capture marker locations and computes the parameters using MoSh [21]. We observe that the provided pose parameters sometimes exhibit small misalignment with respect to the input scans. While the fitting quality in the AMASS dataset is sufficient for our approach, we also evaluate the performance on more accurate pose parameters by using all the vertices on the registered meshes. More specifically, we first compute a better template by unposing the registered meshes in the first frame of each sequence using the LBS skinning weights of the SMPL template, and averaging over all the sequences. Using this new template, we optimize pose parameters for each frame with an L2-loss on all the registered vertices. Note that in this experiment, we use the original template with the refined pose parameters instead of the refined template in order not to unfairly favor our method over SNARF [8].

In Tab. 1, we report the mean absolute error of scan-to-prediction distance (mm) and the mean squared error of volume change for our method and SNARF. Tab. 1 shows that SNARF has a large error reduction with refined poses, indicating that SNARF is highly sensitive to the accuracy of the SMPL fit. We also observe that after pose refinement, SNARF overfits more to training poses (e.g., interpolation) as SNARF cannot model history-dependent dynamic deformations. In contrast, our method is more robust to the fitting errors, and significantly outperforms SNARF in most settings except for 16-30 rollouts in the interpolation set. Note that the results with longer rollouts favor "mean" predictions over more dynamic predictions, and do not inform us of the plausibility of the synthesized dynamics (see the discussion in Sec. 4.2).

## B  k-NN vs. Closest Surface Projection

As discussed in Sec. 3.3, our SDF decoding approach uses k-nearest neighbors (k-NN) of the SMPL vertices instead of closest surface projection [20]. Fig. H illustrates the limitation of this alternative approach proposed in Neural Actor [20]. As shown in Fig. H, we observe that associating a query location with a single closest point on the surface leads to poor generalization to unseen poses around regions with multiple body parts in close proximity (e.g. around armpits). In contrast, our approach, which associates query points with multiple k-NN vertices, produces more plausible surface geometry even for unseen poses.

## C  Limitation: Clothing Deformations

We also apply our method on the CAPE [25] dataset that contains 4D scans of clothed humans. We select the subject 03375_longlong, which exhibits the most visible

Table B: **Quantitative Evaluation on Input Pose Accuracy on Subject 50002.** We show the results of our approach and SNARF [8] using the poses provided by the AMASS [27] dataset and the ones after refinement using all vertices in the registered meshes. While SNARF is greatly influenced by the accuracy of pose parameters, the slight improvement in our method illustrates its robustness to SMPL fitting errors. In addition, our approach significantly outperforms SNARF even after pose refinement in most settings except for the 16-30 rollouts in the interpolation set.

(a) Mean Scan-to-Prediction Distance (mm) ↓

| | | Rollout (# of frames) | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 4 | 8 | 16 | 30 |
| *Interpolation Set* | | | | | | | |
| AMASS [27] | SNARF [8] | 7.898 | 7.715 | 7.588 | 7.840 | 7.898 | 8.238 |
| | Ours | 1.731 | 2.127 | 2.953 | 4.325 | 5.606 | 6.455 |
| Refined Poses | SNARF [8] | 3.982 | 4.001 | 3.964 | 4.068 | 4.029 | 4.158 |
| | Ours | 1.417 | 1.703 | 2.259 | 3.241 | 4.044 | 4.601 |
| *Extrapolation Set* | | | | | | | |
| AMASS [27] | SNARF [8] | 8.083 | 8.126 | 8.160 | 8.246 | 8.050 | 8.025 |
| | Ours | 1.259 | 1.479 | 1.984 | 2.883 | 4.023 | 4.867 |
| Refined Poses | SNARF [8] | 4.624 | 4.632 | 4.672 | 4.749 | 4.548 | 4.447 |
| | Ours | 1.149 | 1.329 | 1.745 | 2.486 | 3.313 | 3.855 |

(b) Mean Squared Error of Volume Change ↓

| | | Rollout (# of frames) | | | | |
|---|---|---|---|---|---|---|
| | | 2 | 4 | 8 | 16 | 30 |
| *Interpolation Set* | | | | | | |
| AMASS [27] | SNARF [8] | 0.01623 | 0.01590 | 0.01688 | 0.01703 | 0.01829 |
| | Ours | 0.00990 | 0.01135 | 0.01417 | 0.01597 | 0.01815 |
| Refined Poses | SNARF [8] | 0.01401 | 0.01349 | 0.01430 | 0.01426 | 0.01524 |
| | Ours | 0.00849 | 0.01002 | 0.01248 | 0.01389 | 0.01558 |
| *Extrapolation Set* | | | | | | |
| AMASS [27] | SNARF [8] | 0.01228 | 0.01244 | 0.01333 | 0.01292 | 0.01264 |
| | Ours | 0.00602 | 0.00756 | 0.00977 | 0.01082 | 0.01140 |
| Refined Poses | SNARF [8] | 0.01094 | 0.01092 | 0.01148 | 0.01099 | 0.01080 |
| | Ours | 0.00559 | 0.00691 | 0.00871 | 0.00953 | 0.01000 |

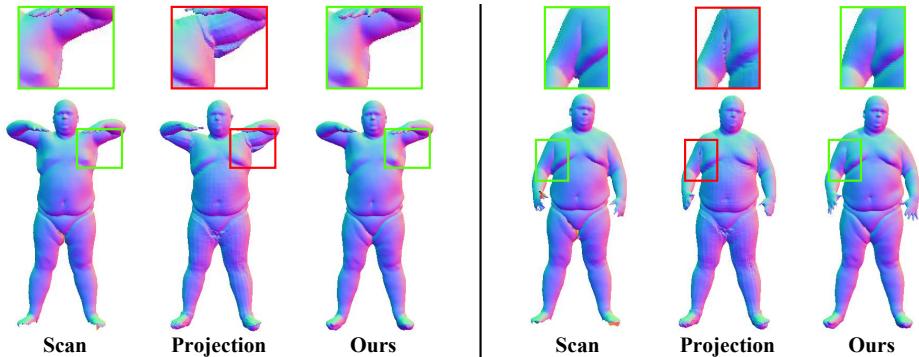| Scan | Projection | Ours | | Scan | Projection | Ours |

Fig. H: **k-NN vs. Closest Surface Projection.** While the closest surface projection suffers from artifacts around armpits, our SDF decoding based on k-NN produces more plausible surface geometry for unseen poses.

dynamic deformations for clothing. We exclude 6 sequences (`athletics`, `frisbee`, `volleyball`, `box_trial1`, `swim_trial1`, `twist_tilt_trial1`) from training, and use them for testing. We employ as input the template and SMPL poses provided by the CAPE dataset for training our model. Note that we approximate raw scans by sampling point clouds with surface normals computed on the registered meshes as the CAPE dataset only provides registered meshes for `03375_longlong`.

Please refer to the supplementary video for qualitative results. While our approach produces plausible short-term clothing deformations, it remains challenging to model dynamically deforming clothing with longer rollouts. Compared to soft-tissue deformations, dynamics on clothed humans involve high-frequency deformations and topology change, making the learning of clothing dynamics more difficult. We leave this for future work.