Spatio-Temporal Human Shape Completion With Implicit Function Networks

Boyao Zhou, Jean-Sébastien Franco, Federica Bogo, Edmond Boyer

Spatio-Temporal Human Shape Completion With Implicit Function Networks

Boyao Zhou[*†] Jean-Sébastien Franco[*] Federica Bogo[‡] Edmond Boyer [*]
{boyao.zhou, jean-sebastien.franco, edmond.boyer}@inria.fr febogo@microsoft.com

Abstract

We address the problem of inferring a human shape from partial observations, such as depth images, in temporal sequences. Deep Neural Networks (DNN) have been shown successful to estimate detailed shapes on a frame-by-frame basis but consider yet little or no temporal information over frame sequences for detailed shape estimation. Recently, networks that implicitly encode shape occupancy using MLP layers have shown very promising results for such single-frame shape inference, with the advantage of reducing the dimensionality of the problem and providing continuously encoded results. In this work we propose to generalize implicit encoding to spatio-temporal shape inference with spatio-temporal implicit function networks or STIF-Nets, where temporal redundancy and continuity is expected to improve the shape and motion quality. To validate these added benefits, we collect and train with motion data from CAPE for dressed humans, and DFAUST for body shapes with no clothing. We show our model's ability to estimate shapes for a set of input frames, and interpolate between them. Our results show that our method outperforms existing state of the art methods, in particular the single-frame methods for detailed shape estimation.

1. Introduction

In this paper, we examine the problem of 3D human shape estimation from incomplete 3D observations, *e.g.* depth images, under motion obtained from a single camera. This under-constrained problem requires additional information which can be provided by a learned model but also by leveraging observations over time when considering temporal sequences. We investigate how to benefit from both through data driven spatio-temporal modeling.

Building human shape models from incomplete 3D observations over time is a challenging task with many applications in augmented and virtual reality or telepresence applications, among others. Part of the difficulty lies in the choice of the shape representation, which can be global and encode high level features of human shape such as pose, or more local features to identify details on the surface of observed subjects. In fact many shape models used to complete partial shape data combine both aspects to leverage both of their advantages, *e.g.* by constraining local shape refinement with a global pose and body model of humans underlying to the shape, *e.g.* [24, 32, 45, 3]. But how to balance those aspects is often manually hardwired in the existing methods, especially with classic pre-learning inference and reconstruction models.

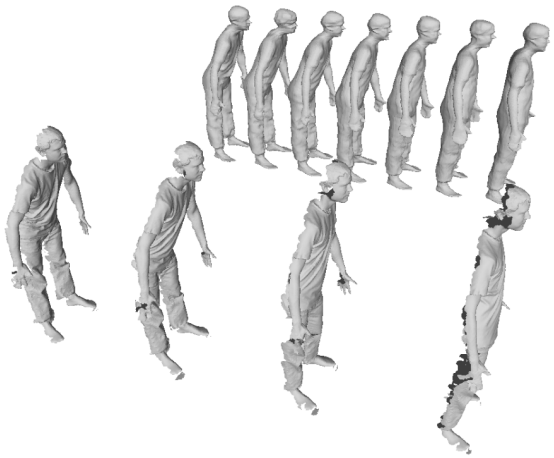

Figure 1. Given incomplete temporal 3D observations as input, here 4 samples, our STIF-Nets reconstruct their complete 3D shape and provide unseen interpolated frames.

The advent of Deep neural Networks (DNN) has brought a whole new set of possibilities to enhance inference and tackle single-view 3D reconstruction problems with data-driven priors, but has simultaneously made the representation problem even more open, because allowing DNNs to account and optimally operate on 3D information has proven to be non-trivial. This is even more prominent when one tackles 4D space-time applications due to the added dimensionality, and the even more massive amount of training data needed to train models in this context. In fact this has been such a barrier that the literature in data driven 4D

[*]Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP (Institute of Engineering Univ. Grenoble Alpes)
[†]Microsoft Research-Inria Joint Center
[‡]Microsoft Zürich, Switzerland

shape modeling is quite scarce at the time of this writing.

With this in mind, a particular category of implicit methods for 3D shape representation is rapidly gaining attention [28, 10]. By encoding the shape implicitly using an indicator function parameterized by MLPs to express the occupation at a given point in space, these methods have succeeded in reducing the dimensionality of the network needed for 3D shape inference problems and allowed a continuous representation of 3D shapes to be embedded in the inference problem. Expectedly, these desirable characteristics have translated to 3D shape inference results of very promising quality. To our knowledge, they have yet to be extended to spatio-temporal evolution of shapes, for which the reduction of dimension could provide a key benefit.

Our intent is to bridge this gap, by providing a new implicit spatio-temporal model by which better shape inference and completion can be achieved, given temporal depth sequences. In so doing, we target improved shape and motion quality by going beyond static shape priors to learn spatio-temporal shape-motion priors. To this goal, our model uses a U-Net encoder to produce an image-dimensioned feature map, similarly to [28, 31, 10, 37], but instead of only encoding a per-pixel implicit depth indicator function, our features parameterize a per-pixel implicit space-*time* depth evolution indicator function. To balance global and local temporal aspects in our model, we use the bidirectional GRU [12] to connect latent global features encoded by the U-Net from previous and subsequent frames, an architecture we coin U-GRU Encoder.

Using two databases of human motion in and without clothing, we assess the qualitative gain of this architecture on analysing monocular video sequences for 3D dynamic human shape estimation, comparing with per-frame estimation methods over the set of input frames, and also examining temporal densification, which can be performed by sampling shape estimates at intermediate time stamps of the implicit spatio-temporal function. Our experiments not only show high quality results for the interpolation task, comparable to results obtained from per-frame methods had they been provided with intermediate inputs; but also show improvement of quality of the 3D models retrieved with actually observed frames, with respect to per-frame methods.

2. Related Work

In this section we focus on previous works that address shape completion in the spatial and temporal domains.

2.1. Spatial Shape Completion

In order to complete a shape given a partial observation at a given time, methods have been proposed that differ with respect to the shape representations they consider. These representations can be either discrete with *e.g.* point based representations or continuous as with neural implicit functions that encode occupancy.

With explicit point based representations, strategies were explored that use prior assumptions, such as parametric or template shape models, to guide shape completion. For instance, [33, 7, 21] reconstruct the human body by inferring the parameters of the SMPL model [24], a popular parametric model for undressed humans. In another work, LBS-AE [23], Linear Blending Skinning parameters are learned from point cloud in a self-supervised way. Relaxing somewhat the constraints on the shape model, other approaches use a template, for example a mesh, to model human shapes. In this line of work approaches such as 3D-CODED [17, 15, 18, 47] deform a template using global features extracted from partial observations with PointNet [34]. BPS [33] learns the 3D point cloud descriptor and is able to reconstruct the SMPL-topology mesh from it. While strong prior models clearly help the completion task they also limit its applicability to reduced shape spaces, *e.g.* undressed human bodies.

Implicit representations have also been largely exploited to model occupancy in 3D. In the discrete case of voxels, the grid regularity allows the extension of CNN-based methods to 3D and the ability to infer human shapes with data-driven strategies as in [41, 46]. Voxel based representations suffer anyway from complexity issues and recent strategies have taken a continuous approach with implicit representations, *e.g.* [28, 42, 37, 10, 38, 13]. In this case, occupancy is modeled at any 3D location and learned with ground-truth examples. OccNet [28], a seminal work in this category, can be used to complete shapes but is missing local input features, which are crucial to preserve human shape details. SAL [4], IGR [16] and SALD [5] learn the implicit representation of human, but rely on optimizing the latent code to fit the uniformly sampled dense point cloud. NASA [14] encodes the articulated human conditioning only on pose with implicit function, which is identity-dependent. LEAP [29] and SCANimate [39] learn the human implicit representation by using the Linear Blending Skinning. However, LEAP [29] requires key joint information which is identity- and pose-dependent. In general, optimization-based post-processing stage in [4, 16, 5, 14, 39] could not be easily applied with partial view input. Recently, IF-Net [10], which stacks 3D convolutions to extract features at different scales for query points, can preserve human body details. It is in practice compute-hungry to train, as a consequence of the voxel representation required to perform 3D convolutions. While we use a similar strategy, we lift the problem to the spatio-temporal domain and reduce the complexity by basing our network in the 2D pixel domain as for [37] but with depth-time implicit queries. We also note in this category of work [6] which combines IF-Net and SMPL to register shapes, with still the aforementioned limitations.

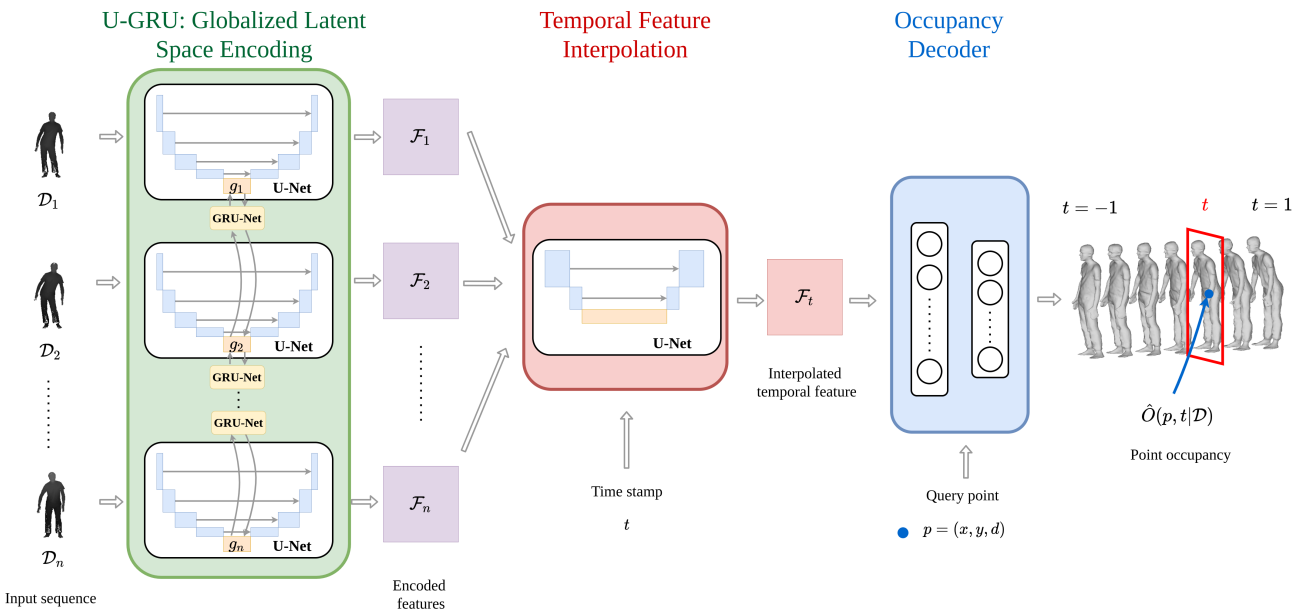

Figure 2. STIF-Nets Overview. Our architecture is articulated among three phases: (1) simultaneous encoding-decoding of input sequence frames to a set of features using U-Nets whose latent spaces are interconnected thanks to GRU networks (§3.1); (2) Temporal interpolation of the feature space with U-Net encoder (§3.2); (3) Occupancy decoding along the viewing line (§3.3).

2.2. Spatio-Temporal Shape Completion

Besides model based approaches that perform temporal shape predictions using parametric representations based on rigged skeletons *e.g.* [27, 48, 11, 9] or more elaborate models such as SMPL *e.g.* [40, 2, 3], few works consider spatio-temporal completion per-se. [1] can complete shape sequences both spatially and temporally by using decompositions over spatio-temporal basis and by assuming temporally consistent shape models, *i.e.* with fixed topology, for that purpose. The approach targets more sparse missing data in temporal trajectories than large completions of shapes with inconsistent topologies as we do. Considering point trajectories, flow based methods have been proposed that model spatio-temporal shape evolution and can therefore perform predictions, *e.g.* [31, 44, 20, 43]. DynamicFusion [30] reconstructs 3D information by using the fusion of multiview depth image, which is not in our case of single view completion. In this category, OccFlow [31] is close to our objective with an approach that predicts spatial-temporal occupancies. However, considering point trajectories implies temporal correspondences which are often difficult to obtain and also sparse missing data instead of significant completions.

3. Method

Our goal in this paper, given a monocular sequence of input depth frames representing incomplete shapes, is to infer a set of complete and temporally densified or predicted shapes. By incomplete inputs, we mean that frames are typically presumed to be obtained from time of flight cameras or front scans with depth sensing technologies, with back and occluded data missing.

Let $\mathcal{D} = \{\mathcal{D}_i\}_{i\in\{1,\cdots,\mathrm{n}\}}$ be the discrete time sequence of input depth frames of resolution $\mathrm{res} \times \mathrm{res}$. As we seek benefit from the lean and continuous parameterization an implicit representation offers, following several similar papers [28, 10] we model the problem with a DNN representing the occupancy regression function of a query point $p = (x, y, d)$ with continuous pixel coordinates $(x, y) \in [1, \mathrm{res}]^2$, and continuous depth $d \in \mathbb{R}$, and given a time stamp in the continuous interval $t \in [-1, 1]$ representing the initial frame interval $[1, \mathrm{n}]$. This function produces predictions of the occupancy $\hat{\mathcal{O}}(p, t|\mathcal{D}) \in [-1, 1]$ of the query point given the input depth images $\mathcal{D}$.

Note that, contrary to other similar inspirational works which focus on 3D volumetric inference [10], we choose a similar 2D discrete support grid similar to [37] instead, for targeted memory and computational efficiency improvements necessary to deal with the additional temporal dimension. We also explicitly focus our method on the output surface at the zero crossing of the occupancy function $\hat{\mathcal{O}}(p, t|\mathcal{D})$, similar to TSDF-based methods [19], which can be efficiently extracted using a marching cubes [25, 22] algorithm, and evaluate points in the vicinity of the surface, as opposed to volume-centric approaches [10] which tend to infer volumetric occupancy functions in $[0, 1]$ for regular volumetric grids.

In order to make the problem tractable and decompose the function according to its main factors, we build our network architecture along three phases, illustrated in Fig. 2. In the first (§3.1), we decode a set of 2D feature maps $\mathcal{F} = \{\mathcal{F}_i\}_{i\in\{1,\cdots,\mathrm{n}\}}$ from each 2D input depth image $\mathcal{D}_i$ of identical resolution $\mathrm{res} \times \mathrm{res}$ but introduce a global correlation to allow the network to learn global motion features linking them. We input the queried continuous time variable t in the second phase (§3.2), and jointly use it with the full set of feature maps to decode a t-specific interpolated 2D feature map $\mathcal{F}_t$ also matching the input resolution $\mathrm{res} \times \mathrm{res}$. This feature map is then used jointly with the query point p to decode the final occupancy result $\hat{\mathcal{O}}(p, t|\mathcal{D}) \in [-1, 1]$, as described in §3.3.

3.1. Globalized Latent Space Encoding

In this phase, we want to allow the network to extract relevant feature maps $\mathcal{F}_i$ for each input $\mathcal{D}_i$ that preserve some detail, while simultaneously allowing the method to be aware of global aspects such as the underlying subject motion. To this goal we opt for a 2D U-Net encoder-decoder structure [36] per-input frame, which projects its inputs on a low dimensional latent space and lifts it back to an output matching the input size, using four symmetric downsampling convolution and upsampling deconvolution layers. U-Net also has the property to balance global aspects of the frame with local ones, using skip connections between matched convolution and deconvolution layers, that allow to preserve local and high frequency details for creation of the feature map, while still allowing for efficient training. Accounting for the expected symmetry between the features extracted for the various input frames, we propose to train the n U-Net instances with shared weights.

We however still need to account for shared temporal aspects and inter-frame motion. To this end, we link each U-Net's latent space vector with those of both temporally adjoining frames using a bidirectional Gated Recurrent Unit, or GRU, to learn interframe residuals of the latent space. The intent is to force interframe phenomena to be treated as a global, transpixel phenomena. The choice of GRU is motivated by its use in Natural Language Processing, where it was shown to perform similarly to LSTMs with fewer parameters and easier training [12]. We show the combination of U-Net and GRU, which we coin U-GRU, to significantly improve training results (Tab. 3). Thus phase 1 of our network can be seen as a global feature decoder solution from the input frame set to the set of intermediate feature maps, which are all individually made aware of interframe cues:

$$\mathcal{F} = \text{U-GRU}(\mathcal{D}). \tag{1}$$

3.2. Temporal Feature Interpolation

With the feature map set still global to the entire input sequence, we propose in a second phase to extract an interpolated feature map $\mathcal{F}_t$ which is specialized for the queried time t. We concatenate all feature maps together and add t weighed by a constant normalization factor $c_t \times t$ as an additional constant input channel $\mathcal{T}$ to every pixel of the map, and feed this aggregate to a simpler U-Net [36] with a pixel-wise 1×1 convolution operator, two levels of downsampling convolutions and upsampling operations, to decode $\mathcal{F}_t$ from $\mathcal{F}$:

$$\mathcal{F}_t = \text{U-NET}(\mathcal{F}, \mathcal{T}). \tag{2}$$

With this architecture choice, the network can learn its temporal interpolation function while automatically adjusting between both global and local per-pixel components of the interpolation. So the U-Net here is able to reduce the aliasing effect of the interpolated feature. We believe that the Temporal Feature Interpolation could remedy the missing information, *e.g.* hole, noise or occlusion, from one single frame by considering temporal information from previous and next frames(Fig. 3(e)(g)). Note again that the network can be trained with any continuous $t \in [-1, 1]$ where -1 stands for the first given frame and 1 represents the last frame.

3.3. Occupancy Decoder

This third and last phase focuses on spatial decoding of the occupancy $\hat{\mathcal{O}}(p, t|\mathcal{D})$ of a given query point $p = (x, y, d)$. We bilinearly interpolate a feature $\mathcal{F}_{x,y,t}$ specific to the real-valued (x, y) from feature map $\mathcal{F}_t$ for query point p. Then we associate the depth query value d weighed by normalizing constant $c_d \times d$ with $\mathcal{F}_{x,y,t}$ as the input for an MLP regressor with the following characteristics. The aggregate feature $\{\mathcal{F}_{x,y,t}, d\}$ of query point p at time t is sent to two linear layers. The first one is activated using the widely used RELU and second one using TANH which conveniently produces occupancy values $\hat{\mathcal{O}}(p, t|\mathcal{D})$ in the target interval $[-1, 1]$:

$$\hat{\mathcal{O}}(p, t|\mathcal{D}) \triangleq \text{MLP}(\text{U-NET}(\text{U-GRU}(\mathcal{D}), \mathcal{T}), p). \tag{3}$$

3.4. Training

The proposed network can be trained for various tasks, *i.e.* shape completion of input frames, temporal interpolation or densification of frames. We propose a uniform supervised training procedure for all of these cases. For this we consider that, for a given batch of ground truth training sequences B, we are given occupancy samples $\mathcal{O}_{p,j}$ with a randomized point set $p \in P_j$ and their matching inputs $\mathcal{D} = \{\mathcal{D}_j\}_{j\in1,\cdots,\mathrm{m}}$, from a set of ground truth frames with time stamps $\{t_j\}_{j\in1,\cdots,\mathrm{m}}$. Typically this set will include time stamps that match the input frames and some additional training examples regularly interspaced between input frames. The training can then be realized by minimizing

a mean square loss over the set of network parameters θ:

$$\theta^* = \arg\min_\theta \sum_B \sum_{t \in T} \sum_{p \in P_j} \|\hat{\mathcal{O}}(p,t|\mathcal{D}) - \mathcal{O}_{p,j}\|^2. \quad (4)$$

To account for the continuous nature of the occupancy function $\hat{\mathcal{O}}(p,t|\mathcal{D})$, we create more temporal samples than given in training, by drawing t from a randomized, denser set T within the training interval, and use j of the time stamp closest to t in the above training procedure.

Point sampling strategy. The choice of the training point set P_j is an important one. Naive strategies would be to use uniformly randomized or regularly spaced samples over the whole sequence's bounding box to present the training with positive and negative examples. This is however quite inefficient as it wastes most of the advantage of modeling the occupancy as an implicit function, the main point being to decorrelate the training complexity from dense 3D space sampling that would occur with regular grid CNNs. [28, 10] use a Gaussian sampling strategy at the vicinity of the surface and train with a classification loss. We propose a simpler yet experimentally efficient sampling which leverages our surface level set parameterization, by providing samples from 3 distinct surfaces in the vicinity of the true surface: one corresponding to the true surface location with training label 0, the expanded and the shrunk surfaces with positive and negative displacement along the surface normal, with respective label sets in $o \in \{-0.5, +0.5\}$. The shrinking and expansion factor along the normal we choose is the label o multiplied by a constant scale factor l. n_s samples are used for every surfaces, and we also sample n_s points from inner part of shrunk surface with $o = +1$ and $3 \times n_s$ points from outer part of expanded surface with $o = -1$ as we empirically observe the need for more negative samples with the expansion.

3.5. Implementation Details and Inference

We implement our STIF-Nets in PyTorch and train it from scratch. In practice, we set the frame number $\mathrm{n} = 4$, the sampling number $\mathrm{n}_s = 300$, the expanded/shrunken length $l = 0.02$ which is a ratio w.r.t. the bounding box scale, two coefficients $c_t = \mathrm{res}, c_d = \max(\mathrm{res}, 256)$. Due to the limitation of GPU memory for sequential data, we set the batch size to 1 and we drop the Batch Normalization in the original U-Net implementation. We use Adam optimizer with the learning rate of 0.0001. During inference, we set the resolution of 3D occupancy grid to 256^3 for all occupancy-based methods, except the one reported as depth/grid resolution $512/512$ in Tab. 4, which is different from res for the depth image. Marching cubes [25, 22] is applied to extract the zero-level set of the computed occupancy grid as a surface mesh.

4. Experimental Evaluation

In order to evaluate STIF-Nets we conducted quantitative and qualitative comparisons on the shape completion task given depth image sequences, this for both input frames and new interpolated, focusing on body shapes in motion. In the following we provide numerical results as well as ablation studies that shed light on how the main components of STIF-Nets impact the performance, which will be more expansively presented in the supplementary material. We first detail the data and metrics used in our evaluation.

4.1. Data and metric

We collected human motion data from a clothed human dataset CAPE [26] which is based on 4D capture ClothCap [32],with two clothing styles dressed on each character, and an undressed human dataset DFAUST [8]. Both datasets contain real scans that were captured at 60fps and fit with the SMPL model [24], which provides our ground truth surface models. From the scans we created front view depth images where the depth of a pixel is determined by the front-facing scanned 3D point that is closest to the pixel viewing-line. Note that we preserve the hole and noise of real scans in our processed depth image in order to test the robustness of our method, see Fig. 1, 2 and 7.

Training and Test Sets: The training set includes 8 characters, 2 male and 2 female from each dataset. For each character, we selected 3 or 4 motion sequences for a total of 28 sequences. Within each motion sequence, we extracted 6 sub-sequences composed of 4 frames. These sub-sequences are of 2 types: 4-interval sequences with interframe intervals of 4 and approximately 200ms durations; and 10-interval sequences with interframe intervals of 10 and approximately 500ms durations. In addition, 3 frames, taken randomly within each sequence were added to the 4 frames with the objective to more robustly train interpolation. Both 4 and 10-interval sequences were used in the training for a total of 336 input sequences. Note that we train our STIF-Nets only once across dressed and undressed characters and short-term and long-term sequences. The test set includes 9 characters, 4 from CAPE and 5 from DFAUST, who perform 54 input sequences. 2 characters in the test set were completely unseen during training. The seen shape characters perform the different motion styles from the training set.

Metric: We evaluated the completions using 2 metrics: The volumetric intersection over union (IoU) and a surface based Chamfer-L1 distance. Note that numerical values were computed with the meshes obtained by the Marching cubes algorithm applied on the occupancies predicted by STIF-Nets. In practice, we noticed that the IoU metric, which is volumetric, hardly differentiates the approaches whereas the Chamfer distance, a surface metric, provides more insights though being more sensitive to noise.

Figure 3. Qualitative results with front-view completions. From left to right, (a) partial scan, reconstruction of (b) 3D-CODED [17], (c) OccNet [28], (d) IF-Net [10], (e) our static, (f) our naive dynamic and (g) our STIF-Nets. See more results in supplementary material.

4.2. Frame Completion

Using the data and the metrics mentioned in the previous section we conducted comparisons of STIF-Nets with representative state of the art methods: one point based method, 3D-CODED [17] and two recent implicit function based methods, OccNet [28] and IF-Net [10]. We retrain the 3D-CODED, OccNet and IF-Net on our dataset and the numeric results with 4 frames intervals are shown in Tab. 1.

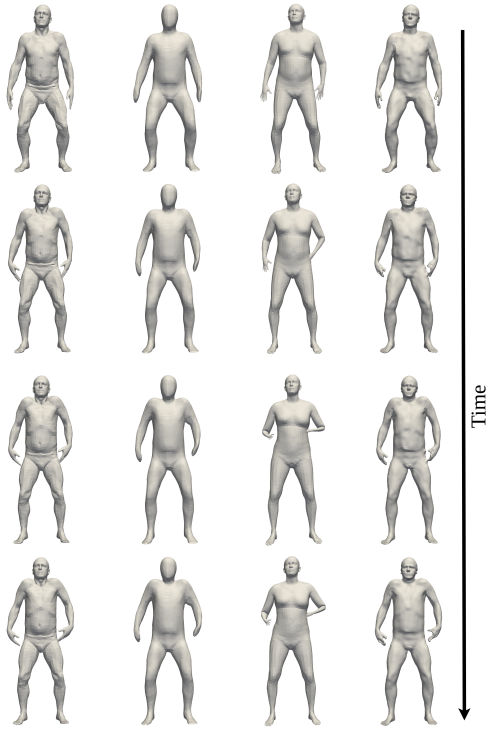

Figure 4. Sequence reconstruction. From left to right, (a) ground truth, reconstruction of (b) OccFlow [31], (c) BPS [33] and (d) our STIF-Nets.

Data	CAPE		DFAUST	
Method	IoU ↑	Chamfer ↓	IoU ↑	Chamfer ↓
3D-CODED [17]	0.455	0.591	0.578	0.347
OccNet [28]	0.488	0.476	0.604	0.340
IF-Net [10]	0.787	0.155	0.822	0.134
[10](×2)	0.804	0.143	0.840	0.127
OccFlow [31]	-	-	0.740	0.231
BPS [33]	-	-	0.761	0.197
Our STIF	**0.822**	**0.123**	**0.858**	**0.111**

Table 1. Spacial completion with IoU and Chamfer-L1 distances ($\times 10^{-1}$) for 4 interframe intervals.

Note that our models provide clearly better results with both IoU and Chamfer distance. IF-Net processes the partial scan data into voxel occupancy. To fairly compare with IF-Net, we set the same total resolution of voxels as our processed depth image, and also compare with IF-net inputting the voxels of 2 times resolution. Note that our method outperforms both. In addition, we evaluate OccFlow [31] and BPS [33] on DFAUST. OccFlow is pretrained on DFAUST dataset and BPS is pretrained on CAESAR [35] which is a very large undressed human dataset, as supplied by authors.

Static completion comparisons are presented in Fig. 3. They show that both 3D-CODED [17] and OccNet [28] have difficulties preserving shape details, as a result of the missing local features in these methods. We also prepare

Data	CAPE		DFAUST	
Method	IoU ↑	Chamfer ↓	IoU ↑	Chamfer ↓
Our Static(depth)	0.651	0.512	0.712	0.380
(neighbour)	0.753	0.158	0.777	0.154
(latent)	0.788	0.157	0.812	0.154
3D-CODED [17]	0.456	0.592	0.578	0.349
OccNet [28]	0.488	0.475	0.604	0.337
IF-Net [10]	0.791	0.158	0.826	0.143
[10](×2)	**0.806**	0.150	0.841	0.143
Our STIF	**0.806**	**0.139**	**0.842**	**0.133**

Table 2. Temporal interpolation with IoU and Chamfer-L1 distances ($\times 10^{-1}$) for 4 interframe intervals.

our static model which replaces the U-GRU encoder by the U-Net and remove the second Feature Interpolation phase. The qualitative results in Fig. 3 show that all static approaches including ours and IF-Net [10] present artefacts, often resulting from holes and noise in the raw input scans. On the other hand, dynamic approaches appear more robust, with our STIF-Nets outperforming the naive temporal baseline. The Feature Interpolation phase is able to reduce the aliasing effect of bilinear sampling of feature map in the Occupancy Decoder, which preserves the high frequency feature on the face, belly and even cloth.

The best comparison with OccFlow and BPS would require full scan data instead partial scans, which is significantly less challenging than our scenario, where the performance of these two methods degrades. In Fig. 4, OccFlow is not able to preserve the high frequency feature on the human body and the BPS descriptor is sensitive to the noise on the real scan data.

We also experiment the influence of interval frames on the completion result. Note that we do not train again our STIF-Nets for 2, 6 or 8 interframe intervals. In Fig. 5, as can be expected the dynamic model efficiency reduces with increasing inter frame intervals, but this gap is not large between short-term sequence and long-term sequence.

Our method outperforms the best method in the compared baseline, IF-Net, both in speed and performance. In our tests, the IF-Net average per-frame computation time is 4.791 s/frame, whereas our static baseline and STIF-Nets run in 0.343 and 0.349 s/frame on a GeForce RTX 2080Ti, using the same total input and occupancy grid resolution. Our STIF-Nets pays only 6ms per frame penalty for using the temporal information.

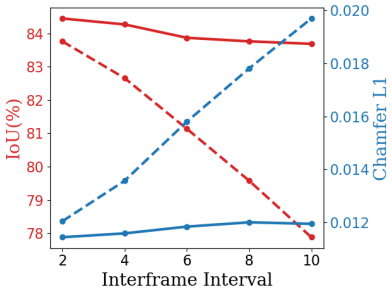

Figure 5. Evaluation of our STIF-Nets with different interframe intervals. The solid line —— stands for completion task and the dashed one - - - stands for interpolation task.

4.3. Frame Interpolation

We also experimented the ability to interpolate between the input frames from partial data. For the interpolation task, the evaluation is performed at the 3 middle frames within the 4 intervals of each sequence. From the static case, a naive interpolation baseline can be achieved using different strategies: nearest neighbor frame, depth image interpolation or latent representation interpolation. Numerical comparisons in Tab. 2 show that the latter performs the best and we therefore only report results with latent space interpolation in Tab. 2 and 3 for other static methods. Tab. 2 also demonstrates that STIF-Nets outperforms static interpolation with both metrics. Fig. 6 illustrates frame interpolation. The spatio-temporal modeling is able to preserve the volume and the high frequency features even on the Interpolation task which is difficult for static models. In Fig. 5, the interframe interval notably influences the interpolation performance, which degrades gracefully given that we did not increase the supervision of interpolation task during the training of long interval.

4.4. Ablation Studies

Tab. 3 reports on two crucial elements in our method: sampling and dynamic modeling. It shows that our proposed sampling strategy benefits with respect to the Gaussian sampling from surface used in the IF-Net [10]. Again STIF-Nets quite significantly outperforms the static approach quantitatively. To illustrate the efficiency of our dynamic modeling, we prepare a naive dynamic baseline which drops the GRU in Encoder, considers the 4 frames as 4 channels to extract the feature map and uses a Temporal Feature Interpolation with lower dimensionality. For both spatial completion and temporal interpolation tasks, our STIF-Nets better preserve the reconstruction volume (Fig. 3). The Feature Interpolation phase is able to predict the feature map at any queried time stamp in order to fill the gap between spatial completion and temporal interpolation.

We also experiment the STIF-Nets with different resolution of input depth image. Tab. 4 reports that increasing the input depth image resolution could benefit the reconstruction accuracy and the 3D occupancy grid resolution, used for the Marching cubes [25, 22] during inference, plays as well an important role. Fig. 7 shows that more details on the face and belly are extracted by STIF-Nets with high resolution depth image than the low resolution one.

	Input Frame Completion				Interpolation			
Data	CAPE(dressed)		DFAUST(undressed)		CAPE		DFAUST	
Method	IoU ↑	Chamfer ↓	IoU ↑	Chamfer ↓	IoU ↑	Chamfer ↓	IoU ↑	Chamfer ↓
Our Static + basic sampling	0.788	0.143	0.829	0.129	0.784	0.178	0.825	0.176
Our Static	0.804	0.128	0.845	0.113	0.788	0.157	0.812	0.154
Our Native Dynamic	0.713	0.183	0.737	0.180	0.733	0.176	0.760	0.175
Our STIF	**0.822**	**0.123**	**0.858**	**0.111**	**0.806**	**0.139**	**0.842**	**0.133**

Table 3. Quantitative comparisons with IoU and Chamfer-L1 distances ($\times 10^{-1}$) for 4 interframe intervals on both completion and interpolation tasks.

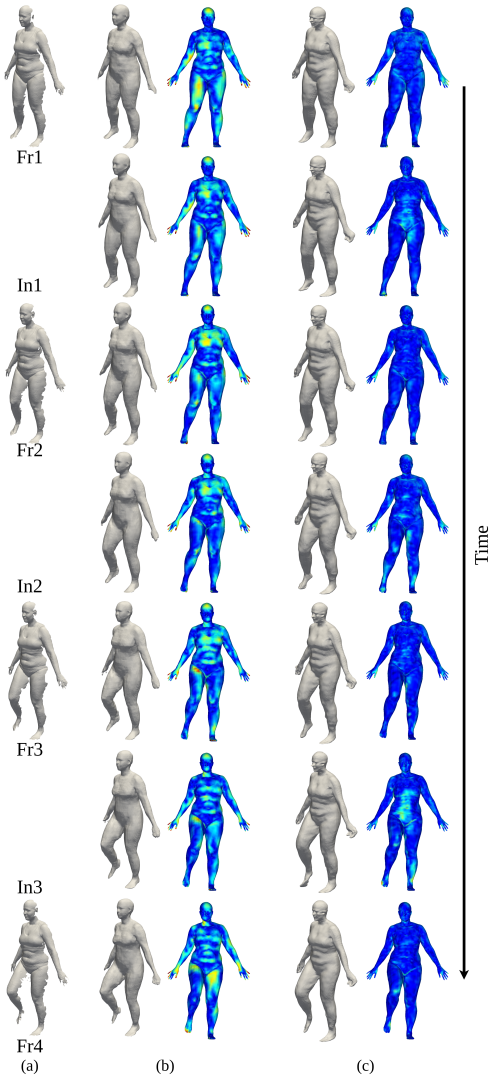

Figure 6. Qualitative results for a 4 frame input sequence with frame completion (Fr1, Fr2, Fr3, Fr4) and interpolation (In1, In2, In3). From left to right, (a) Partial scan, reconstruction/heatmap of (b) IF-Net [10], and of (c) our STIF-Nets. For the heatmap, we compute the Chamfer-L1 distance from reconstruction to the ground truth and we set 0.03 as the maximum error.

depth(^{2})/grid(^{3}) resolution	IoU ↑	Chamfer ↓
128/256	0.810	0.143
256/256	0.843	0.116
512/256	0.846	0.113
512/512	0.858	0.104

Table 4. Impact of the depth image and occupancy grid resolution for shape completion with IoU and Chamfer-L1 distances ($\times 10^{-1}$).

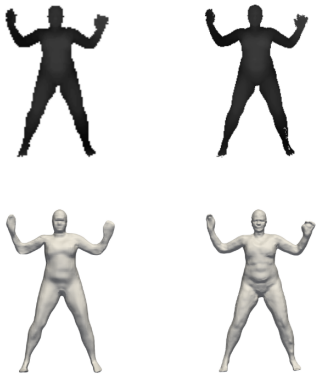

Figure 7. Qualitative results with (a) 128^2-resolution depth image and (b) 256^2-resolution depth image for STIF-Nets.

5. Conclusion

We have presented STIF-Nets, a deep network architecture to model shapes from incomplete observations. STIF-Nets builds on neural implicit function representations, which has proved efficient for shape modeling. The key contribution with respect to existing works is to lift these representations to the spatio-temporal domain, hence leveraging information over time and enabling shape completions over both the spatial and temporal domains. Experiments demonstrate that STIF-Nets contributes with improved robustness, shape quality and generalization abilities with respect to purely spatial strategies. We believe that STIF-Nets can trigger new research in 4D shape modeling.

References

[1] Ijaz Akhter, Tomas Simon, Sohaib Khan, Iain Matthews, and Yaser Sheikh. Bilinear spatiotemporal basis models. *ACM Transactions on Graphics (TOG)*, 31(2):17, 2012. 3

[2] Thiemo Alldieck, Marcus Magnor, Weipeng Xu, Christian Theobalt, and Gerard Pons-Moll. Detailed human avatars from monocular video. In *International Conference on 3D Vision*, pages 98–109, Sep 2018. 3

[3] Thiemo Alldieck, Marcus Magnor, Weipeng Xu, Christian Theobalt, and Gerard Pons-Moll. Video based reconstruction of 3d people models. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 1, 3

[4] Matan Atzmon and Yaron Lipman. Sal: Sign agnostic learning of shapes from raw data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2020. 2

[5] Matan Atzmon and Yaron Lipman. Sald: Sign agnostic learning with derivatives. In *International Conference on Learning Representations*, 2021. 2

[6] Bharat Lal Bhatnagar, Cristian Sminchisescu, Christian Theobalt, and Gerard Pons-Moll. Combining implicit function learning and parametric models for 3d human reconstruction. In *Proceedings of the European Conference on Computer Vision*. Springer, August 2020. 2

[7] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J. Black. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In *Computer Vision – ECCV 2016*, Lecture Notes in Computer Science, pages 561–578. Springer International Publishing, Oct. 2016. 2

[8] Federica Bogo, Javier Romero, Gerard Pons-Moll, and Michael J Black. Dynamic faust: Registering human bodies in motion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6233–6242, 2017. 5

[9] Zhe Cao, Hang Gao, Karttikeya Mangalam, Qizhi Cai, Minh Vo, and Jitendra Malik. Long-term human motion prediction with scene context. In *Proceedings of the European Conference on Computer Vision*, 2020. 3

[10] Julian Chibane, Thiemo Alldieck, and Gerard Pons-Moll. Implicit functions in feature space for 3d shape reconstruction and completion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, jun 2020. 2, 3, 5, 6, 7, 8

[11] Hsu-kuang Chiu, Ehsan Adeli, Borui Wang, De-An Huang, and Juan Carlos Niebles. Action-agnostic human pose forecasting. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, pages 1423–1432. IEEE, 2019. 3

[12] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014. 2, 4

[13] Boyang Deng, JP Lewis, Timothy Jeruzalski, Gerard Pons-Moll, Geoffrey Hinton, Mohammad Norouzi, and Andrea Tagliasacchi. Neural articulated shape approximation. In *Proceedings of the European Conference on Computer Vision*. Springer, August 2020. 2

[14] Boyang Deng, John P Lewis, Timothy Jeruzalski, Gerard Pons-Moll, Geoffrey Hinton, Mohammad Norouzi, and Andrea Tagliasacchi. Nasa neural articulated shape approximation. In *Proceedings of the European Conference on Computer Vision*, pages 612–628. Springer, 2020. 2

[15] Theo Deprelle, Thibault Groueix, Matthew Fisher, Vladimir Kim, Bryan Russell, and Mathieu Aubry. Learning elementary structures for 3d shape generation and matching. In *Advances in Neural Information Processing Systems*, pages 7435–7445, 2019. 2

[16] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. In *Proceedings of Machine Learning and Systems 2020*, pages 3569–3579. 2020. 2

[17] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan Russell, and Mathieu Aubry. 3d-coded : 3d correspondences by deep deformation. In *Proceedings of the European Conference on Computer Vision*, pages 235–251, 2018. 2, 6, 7

[18] Oshri Halimi, Ido Imanuel, Or Litany, Giovanni Trappolini, Emanuele Rodolà, Leonidas Guibas, and Ron Kimmel. The whole is greater than the sum of its nonrigid parts. In *Proceedings of the European Conference on Computer Vision*, 2020. 2

[19] Matthias Innmann, Michael Zollhöfer, Matthias Nießner, Christian Theobalt, and Marc Stamminger. Volumedeform: Real-time volumetric non-rigid reconstruction. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Proceedings of the European Conference on Computer Vision*, pages 362–379, 2016. 3

[20] Chiyu Jiang, Jingwei Huang, Andrea Tagliasacchi, and Leonidas Guibas. Shapeflow: Learnable deformations among 3d shapes. In *Advances in Neural Information Processing Systems*, 2020. 3

[21] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7122–7131, 2018. 2

[22] Thomas Lewiner, Hélio Lopes, Antônio Wilson Vieira, and Geovan Tavares. Efficient implementation of marching cubes' cases with topological guarantees. *Journal of graphics tools*, 8(2):1–15, 2003. 3, 5, 7

[23] Chun-Liang Li, Tomas Simon, Jason Saragih, Barnabás Póczos, and Yaser Sheikh. Lbs autoencoder: Self-supervised fitting of articulated meshes to point clouds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11967–11976, 2019. 2

[24] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Transaction on Graphics (TOG)*, 34(6):248:1–248:16, Oct. 2015. 1, 2, 5

[25] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, 21(4):163–169, 1987. 3, 5, 7

[26] Qianli Ma, Jinlong Yang, Anurag Ranjan, Sergi Pujades, Gerard Pons-Moll, Siyu Tang, and Michael J. Black. Learn-

ing to dress 3d people in generative clothing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2020. 5

[27] Julieta Martinez, Michael J. Black, and Javier Romero. On human motion prediction using recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 3

[28] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 2, 3, 5, 6, 7

[29] Marko Mihajlovic, Yan Zhang, Michael J. Black, and Siyu Tang. LEAP: Learning articulated occupancy of people. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2021. 2

[30] Richard A Newcombe, Dieter Fox, and Steven M Seitz. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 343–352, 2015. 3

[31] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Occupancy flow: 4d reconstruction by learning particle dynamics. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, Oct. 2019. 2, 3, 6

[32] Gerard Pons-Moll, Sergi Pujades, Sonny Hu, and Michael J Black. Clothcap: Seamless 4d clothing capture and retargeting. *ACM Transactions on Graphics (TOG)*, 36(4):1–15, 2017. 1, 5

[33] Sergey Prokudin, Christoph Lassner, and Javier Romero. Efficient learning on point clouds with basis point sets. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4332–4341, Oct. 2019. 2, 6

[34] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 652–660, 2017. 2

[35] Kathleen M Robinette, Sherri Blackwell, Hein Daanen, Mark Boehmer, and Scott Fleming. Civilian american and european surface anthropometry resource (caesar), final report. volume 1. summary. Technical report, Sytronics Inc Dayton Oh, 2002. 6

[36] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 4

[37] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2304–2314, 2019. 2, 3

[38] Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2020. 2

[39] Shunsuke Saito, Jinlong Yang, Qianli Ma, and Michael J. Black. SCANimate: Weakly supervised learning of skinned clothed avatar networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2021. 2

[40] Hsiao-Yu Tung, Hsiao-Wei Tung, Ersin Yumer, and Katerina Fragkiadaki. Self-supervised learning of motion capture. In *Advances in Neural Information Processing Systems*, pages 5236–5246, 2017. 3

[41] Gul Varol, Duygu Ceylan, Bryan Russell, Jimei Yang, Ersin Yumer, Ivan Laptev, and Cordelia Schmid. Bodynet: Volumetric inference of 3d human body shapes. In *Proceedings of the European Conference on Computer Vision*, September 2018. 2

[42] Qiangeng Xu, Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. Disn: Deep implicit surface network for high-quality single-view 3d reconstruction. In *Advances in Neural Information Processing Systems 32*, pages 492–502. 2019. 2

[43] Guandao Yang, Xun Huang, Zekun Hao, Ming-Yu Liu, Serge Belongie, and Bharath Hariharan. Pointflow: 3d point cloud generation with continuous normalizing flows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, October 2019. 3

[44] Shuaihang Yuan, Xiang Li, Anthony Tzes, and Yi Fang. 3dmotion-net: Learning continuous flow function for 3d motion prediction. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2020. 3

[45] Chao Zhang, Sergi Pujades, Michael J Black, and Gerard Pons-Moll. Detailed, accurate, human shape estimation from clothed 3d scan sequences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4191–4200, 2017. 1

[46] Zerong Zheng, Tao Yu, Yixuan Wei, Qionghai Dai, and Yebin Liu. Deephuman: 3d human reconstruction from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, October 2019. 2

[47] Boyao Zhou, Jean-Sebastien Franco, Federica Bogo, Bugra Tekin, and Edmond Boyer. Reconstructing human body mesh from point clouds by adversarial gp network. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, November 2020. 2

[48] Yi Zhou, Zimo Li, Shuangjiu Xiao, Chong He, Zeng Huang, and Hao Li. Auto-conditioned recurrent networks for extended complex human motion synthesis. In *International Conference on Learning Representations*, 2018. 3