Pyramidal Signed Distance Learning for Spatio-Temporal Human Shape Completion

Boyao Zhou[1], Jean-Sébastien Franco[1], Martin deLaGorce[2], and Edmond Boyer[1]

[1] Inria-Univ. Grenoble Alpes-CNRS-Grenoble INP-LJK, France
{boyao.zhou, jean-sebastien.franco, edmond.boyer}@inria.fr
[2] Microsoft. Cambridge, United-Kingdom
{martin.delagorce}@microsoft.com

Abstract. We address the problem of completing partial human shape observations as obtained with a depth camera. Existing methods that solve this problem can provide robustness, with for instance model-based strategies that rely on parametric human models, or precision, with learning approaches that can capture local geometric patterns using implicit neural representations. We investigate how to combine both properties with a novel pyramidal spatio-temporal learning model. This model exploits neural signed distance fields in a coarse-to-fine manner, this in order to benefit from the ability of implicit neural representations to preserve local geometry details while enforcing more global spatial consistency for the estimated shapes through features at coarser levels. In addition, our model also leverages temporal redundancy with spatio-temporal features that integrate information over neighboring frames. Experiments on standard datasets show that both the coarse-to-fine and temporal aggregation strategies contribute to outperform the state-of-the-art methods on human shape completion.

1 Introduction

Completing shape models is a problem that arises in many applications where perception devices provide only partial shape observations. This is the case with single depth cameras which only perceive the front facing geometric information. Completing the 3D geometry in such a situation while retaining the observed geometric details is the problem we consider in this work. We particularly focus on human shapes and take benefit of the camera ability to provide series of depth images over time.

The task is challenging since different and potentially conflicting issues must be addressed. Strong priors on human shapes and their clothing are required as large shape parts are usually occluded in depth images. However, such priors can in practice conflict with the capacity to preserve geometric details present in the observations. Another issue lies in the ability to leverage information over time with local geometric patterns that can be either temporally consistent or time varying, as with folds on human clothing.

Existing methods fall into two main categories with respect to their global or local approach to the human shape completion problem. On the one hand, model-based methods build on parametric human models which can be fitted to incomplete observations, for instance SMPL [25] Dyna [39] or NPMs [34]. Approaches in this category can leverage strong priors on human shapes and hence often yield robust solutions to the shape completion problem. However, parametric models usually impose that the estimated shapes lie in a low dimensional space, which limits the ability to generalize over human shapes, in particular when considering clothing. Another consequence of the low dimensional shape space is that local geometric details tend to be filtered out when encoding shapes into that space.

On the other hand, neural network based approaches, *e.g.* NDF [13], IF-Net [12] or STIF [51], use implicit representations to model the observed surfaces. Such representations give access to a larger set of shapes, although this set is still bounded by the coverage in the training data. They also present better abilities to preserve geometric details as they are, by construction in these approaches, local representations. Besides, in contrast to model based methods, the local aspect of the representation impacts the robustness with estimated shapes that can often be spatially inconsistent.

In order to retain the benefit of local representations while providing better robustness and spatial coherence we propose a novel hierarchical coarse-to-fine strategy that builds on implicit neural representations [35,30]. This strategy applies to the spatial domain as well as to the temporal domain. We investigate the temporal dimension since redundancy over time can contribute to shape completion, as demonstrated in [51]. This dimension naturally integrates into the pyramidal spatio-temporal model we present. Our approach considers as input consecutive depth images of humans in a temporal sequence and estimates in turn a sequence of 3D distance functions that maps any point in the space time cube covered by the input frames to its distance to the observed human shape. This distance mapping function is learned over temporally coherent 3D mesh sequences and through a MLP decoder that is fed with multi-scale spatio-temporal features, as encoded from the input depth images. Our experiments demonstrate the spatial and temporal benefit of the hierarchical strategy we introduce with both local and global shape properties that are substantially better preserved with respect to the state of the art.

The main contributions of this paper are, first, to show the benefit of a pyramidal architecture that aggregates a residual pyramid of features in the context of implicit surface regression, and second, to propose a tailored training strategy that exploits this residual architecture by using losses that are distributed at each scale of the feature map both spatially and temporally. Code is released at https://gitlab.inria.fr/bzhou/PyramidSpatioTempoHumanShapeComplete.git.

2 Related Work

The set of methods proposed in the literature to estimate full human body surfaces from a sequence of depth images can be broadly divided in 4 main categories: model-based methods, regression-based methods, dynamic fusion based methods and hybrid methods. We detail here these different methods and discuss their strengths and limitations.

Model-based methods build on learned parametric human models that are fitted to incomplete observations. The parametric model generally represents a 3D undressed human with global shape and pose parameters. The surface can be represented explicitly using a triangulated surface similarly to SMPL [4,25,36,46] or Dyna [39], or implicitly using either a parameterized occupancy or a signed distance function [34,5,18,6]. Model based methods that decouple the body shape parameters from the pose like [25,34] can be used to efficiently exploit temporal coherence in a sequence by estimating a single shape parameter vector for the whole sequence and different pose for each frame. The low dimensionality of the parameterized space inherently limits the ability of these models to represent details such as clothing or hair present in the data. To model the details beyond the parameterized space [38,29,2,3,8] add a cloth displacement map to represent 3D dressed humans, which significantly improves the accuracy of the reconstructed surface but often still impose some constraint on the topology of the surface and thus do not allow to fully explain complex observed data.

Regression-based methods directly predict the shapes from the incomplete data. This allows to predict less constrained surfaces and keep more of the details from the input data than model-based methods. The representation of the predicted surface can be explicit [19,40,52] or implicit which allows for changes in the topology [11,35]. This approach has been used for non-articulated objects [24,30,47,33,37,13] and articulated human body [42,43,12,20,16,51,40]. The implicit surface can be represented using a (truncated) signed distance function, an unsigned distance function [13] or an occupancy function as in IF-Net [12]. Learning truncated signed distance functions tends to lead to more precise surfaces than learning occupancy, as the signed distance provides a richer supervision signal for points that are sampled near but not on the surface. One main challenge with these approaches is to efficiently exploit the temporal coherence to extract surface detail from previous frames.

Dynamic fusion approaches [31,17] are another classical set of approaches that are neither model-based nor regression-based and which consist in tracking point cloud correspondences, aligning point cloud in a non-rigid manner and fusing the aligned point clouds into a single surface representation. This allows to exploit temporal coherence to complete the surface in regions that are observed in other frames in the sequence. Omitting strong priors on the human body surface, either in the form of a model or a trained regressor (although human prior can be used to help the tracking), these methods are able to retain specific details of the observed surface but are also, consequently, not able to predict the surface in regions that are not observed in any frame. These methods are also subject to drift and error accumulation when used on challenging long sequences with fast motion and topological changes.

Hybrid methods that mix various of these approaches have been proposed to combine their advantages. IP-Net [7] generates the implicit surface with [12] and register the surface with SMPL+D [1,21] and thus inherits the limited ability of model based methods to represent geometric details. Function4D [50] combines a classical dynamic fusion method with a post processing step to repair holes in the surface using a learned implicit surface regression. While these methods improve results w.r.t each of the combined methods, they still tend to retain part of their drawbacks.

Exploiting temporal coherence is a main challenge for regression based methods, as mentioned above. Using a pure feed-forward approach where a set of frames is fed into the regressor becomes quickly unmanageable as the size of the temporal window increases. Oflow [32] exploits the temporal coherence by estimating a single surface for the sequence initial frame which is deformed over all frames using a dense correspondence field conditioned on the whole sequence inputs. This strategy enforces by construction temporal consistency, but it also prevents temporal information to benefit to the shape model. STIF [51] addresses the problem of temporal integration using a recurrent GRU [14,15] layer to aggregate information. Such layer can however only be used at the low-resolution features and hence surface details do not propagate from one frame to the next.

In this paper we suggest a pyramidal approach that can exploit both spatial and temporal dimensions. Taking inspiration from pyramidal strategies applied successfully to other prominent vision problems such as object detection [23], multi-view stereo [48], 3D reconstruction [45] and optical flow [44] we devise a method that aggregates spatio-temporal information in a coarse to fine manner, propagating features from low to high resolution through up-sampling, concatenation with higher resolution features and the addition of residuals.

3 Network Architecture

We wish to compute the completion of a sequence of shapes observed from noisy depth maps. These maps, noted $\mathcal{D} = \{\mathcal{D}_t\}_{t\in\{1,...,N\}}$, provide a truncated front point cloud of the human shape in motion. Our output, by contrast, is a corresponding sequence of *complete* shapes encoded as a set of per-frame Truncated Signed Distance functions, using neural implicit functions $\mathrm{SDF}_t(p)$ with $t \in \{1, \dots, N\}$, which can be each queried for arbitrary 3D points $p = (x, y, z)$. The surface can then be extracted using standard algorithms [26,22].

In the following, we explain how the $\mathrm{SDF}_t(p)$ functions can be coded with a network inspired from point cloud SDF networks [13,35], but which also jointly leverages the analysis of small temporal frame subgroups of size K. Without loss of generality we expose the framework for a given subgroup of frames $\{1, \dots, K\}$ and drop the more general index N. We also propose a more general pyramidal coarse-to-fine architecture to balance global information and local detail in the inference, which was previously only demonstrated with explicit TSDF en-

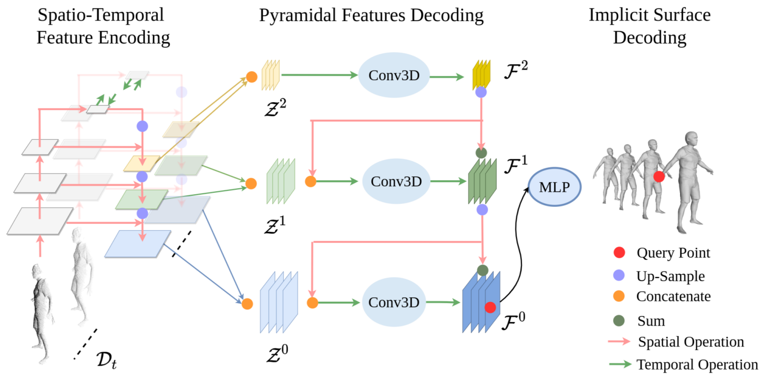

Fig. 1. Network Architecture. Data is processed in 3 phases: depth images are processed with Spatio-Temporal Feature encoding backbone in §3.1 into three-level-feature maps. These feature maps are latter aggregated with 3D convolution considering the temporal dimension in the coarse-to-fine manner in §3.2. The aggregated feature map is used to predict the signed distance field with MLP in §3.3.

codings for a different incremental reconstruction problem [45]. Our network is articulated around three main phases represented in Fig. 1, a feature pyramid extraction phase (§3.1), a pyramidal feature decoding phase (§3.2), and an implicit surface decoding phase (§3.3), detailed as follows.

3.1 Spatio-Temporal Feature Encoding

To build a hierarchical inference structure, a time-tested pattern (*e.g.* [48,44]) is to build a feature hierarchy or feature pyramid [23] to extract various feature scale levels. The recent literature [51] shows that a measurable improvement to this strategy, in the context of temporal input images sequences, is to link the coarsest layer in each input image's pyramid with a bidirectional recurrent GRU component [14,15], instead of building independent per-frame pyramids. This allows the model to learn common global characteristics and their mutual updates at a modest computational cost. We modify the backbone encoder U-GRU [51] in order to hierarchically yield a set of per-frame feature maps $\mathcal{Z}_t^0$, $\mathcal{Z}_t^1$, $\mathcal{Z}_t^2$ at three levels of respectively high, mid and low resolutions, for each input depth map $\mathcal{D}_t$ with $t \in \{1, \dots, K\}$. In the rest of the discussion we will only use temporal aggregates of each feature level $\mathcal{Z}^0 = \{\mathcal{Z}_t^0\}, \mathcal{Z}^1 = \{\mathcal{Z}_t^1\}$, and $\mathcal{Z}^2 = \{\mathcal{Z}_t^2\}$ with $t \in \{1, \dots, K\}$, such that we can denote their extraction:

$$\{\mathcal{Z}^0, \mathcal{Z}^1, \mathcal{Z}^2\} = \text{Encode}(\mathcal{D}_1, \dots, \mathcal{D}_K), \tag{1}$$

The feature map here contains 3 dimensions: x, y and t in the three different levels of the pyramid. However, the coarsening is only done in the spatial domain at this stage so that all scale feature maps contain the same number of frames K. More details on these feature map's dimensions are provided in Section 3.4.

3.2 Pyramidal Feature Decoding

In the next stage, our goal is to allow the network to progressively decode and refine spatio-temporal features of the sequence that will be used for TSDF decoding. To this goal, we first process the coarsest feature aggregate $\mathcal{Z}^2$ with a full 3D (2D + t) convolution to extract a sequence feature $\mathcal{F}^2$:

$$\mathcal{F}^2 = \text{Conv3D}(\mathcal{Z}^2). \tag{2}$$

We then subsequently process the finer feature levels $l \in \{1, 0\}$, first by up-sampling the previous-level sequence feature map in the spatial domain using bilinear interpolation, noted $\mathcal{U}^l$, then concatenating it with the aggregate feature $\mathcal{Z}_l$ as input to 3D convolutions, as shown in Fig. 1. The output of the 3D convolutions is the residual correction of the previous-level aggregated feature, echoing successful coarse-to-fine architectures [48,44]:

$$\mathcal{U}^l = \text{Up-Sample}(\mathcal{F}^{l+1}), \tag{3}$$

$$\mathcal{F}^l = \mathcal{U}^l + \text{Conv3D}(\{\mathcal{U}^l, \mathcal{Z}^l\}). \tag{4}$$

3.3 Implicit Surface Decoding

The final high-resolution feature $\mathcal{F}^0$ obtained as output of the previous process serves as a latent vector map, from which we sample a latent vector at a given query point's image coordinates. The selected latent vector is then decoded using an MLP, and provides the TSDF value for that point. More formally, given a query point (x, y, z) and a time frame $t \in \{1, \ldots, K\}$ this signed distance function is computed by first sampling bi-linearly at the corresponding 2D location (x, y) the slice $\mathcal{F}_t^0$ of the high resolution 3D feature map that corresponds to frame t of the temporal window. We can note this sampler f_t^0:

$$f_t^0(x, y) = B(\mathcal{F}_t^0; x, y), \tag{5}$$

After obtaining the queried feature vector in the high resolution scale, we concatenate it with depth feature z. The signed distance function can then be computed by

$$\text{SDF}_t^0(x, y, z) = \text{MLP}(f_t^0(x, y), z), \tag{6}$$

where an MLP is used to decode the TSDF value for a given pixel x, y, and frame t of the high-resolution feature map, and an implicitly accounted z. Operational details and constants are discussed in the next section and supplementary.

3.4 Implementation Details

We adapt U-Net [41]-like encoder [51] as backbone, for the purpose of hierarchical learning, combined with GRU [14,15] for pyramidal feature extraction in §3.1. Each 3D convolution block contains 3 convolutions layers with 3×3×3 kernels in §3.2. We use zero padding spatially and temporally. In §3.3, the depth feature z

is multiplied by 128 in order to get a similar activation ranges as the values in the feature vector $f_t^0(x,y)$. The final output layer of the MLP is activated with *tanh* in order to bound the signed distance in the range $[-1,1]$. The pre-computed SDF is scaled with the factor of 75 and we set the truncate value as 1 in the normalized space. In practice, this multiplication improves also the numerical stability, otherwise the loss would be too small during the training. While it would be possible to compute the inference results over an input sequence in a sliding window fashion, for computational trade-off with quality we compute the results on consecutive groups of $K=4$ frames. A training batch is composed of 4×512^2 depth images and 4×2800 query points per scale for the SDF evaluation. It takes 2.86 seconds with GPU memory footprint about 7.86GB for a batch training.

4 Training Strategies

In the previous section, we discussed the network used and inference path for a given set of K input frames. The same path can be used for supervised training, but pyramidal architectures are usually trained with intermediate loss objectives that guide the coarser levels. This is easy to do with a supervised loss when the coarser objectives are just a down-sampled version of the expected result, *e.g.* for object detection, depth map or optical flow inference [48,44]. A main contribution of this paper is to show the benefit of such schemes with an implicit representation of the surface reconstructed from a residual pyramid of features. One of the key difficulties lies in characterizing the intermediate, lower resolution contribution to the final result of coarser layers of the implicit decoder. To this goal we present four training strategies, with several temporal and pyramidal combinations, and discuss their ablation in the experiments. A visual summary of these strategies is provided in Fig. 2.

4.1 Common Training Principles

We render depth maps of 3D raw scans from a training set to associate depth maps observations to ground truth computed SDF values for a given set of query points in the 3D observation space. Following general supervision principles of implicit networks [42,51], we provide training examples for three groups: on-surface points, points in the vicinity of the surface, and other examples uniformly sampled over the whole observed 3D volume. On-surface training examples are obtained simply by sampling k_s vertices from ground truth 3D model and associating them with SDF value 0. For surface vicinity points at any given time t, as shown in Fig. 2, for each of the k_s vertices, we randomly draw 4 training points with their ground truth SDF values among points on a 26-point 3D grid centered on a ground truth surface point, in which the closest projected grid edge length matches the observed feature map pixel size, as illustrated in the right most column in Fig.2. For the third, uniform point group, we randomly draw k_u sample training points covering the whole acquisition space, for which

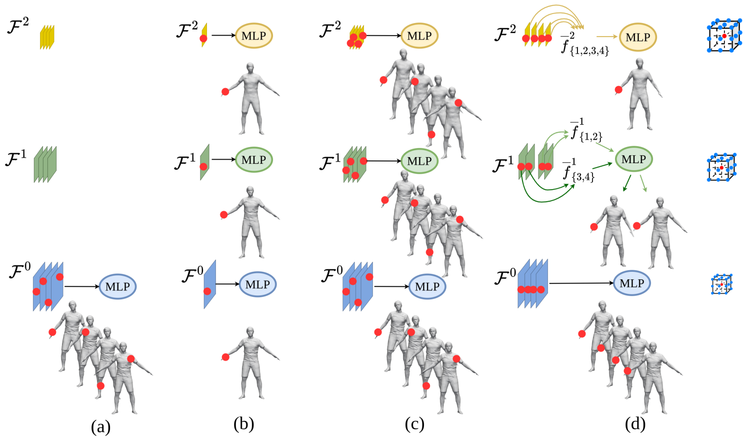

Fig. 2. Training strategy variants, from left to right: (a) temporal naive, (b) static pyramidal, (c) temporal pyramidal and (d) spatio-temporal pyramidal. More detailed explanations can be found in §4.3.

we provide the SDF to the ground truth surface. In our experiments, we keep $k_s = 400$ and $k_u = 800$ constant regardless of training scenario.

4.2 Pyramidal Training Framework

For explicit pyramidal training, we need to supply ground truth SDFs for all levels of the pyramid, including the coarser ones. To this goal, for a set of sampled ground truth surface points at time t, we extend the set of surface-vicinity training samples by randomly drawing them from three different 26-point-grid cubes centered on the ground truth surface point, instead of only one. Each of these grid cubes corresponds to a network pyramid level feature $\mathcal{F}_t^l$ with $l \in \{0, 1, 2\}$ and its closest projected grid edge length matches the finest feature map pixel size $\mathcal{F}_t^0$. We still draw 4 samples at each feature level l, among the 26 possibilities.

For each of the m^l query points $p_i = (x_i, y_i, z_i)$ in the training batch, noting its frame t_i and pyramid level l, we generalize equations 5 and 6 to provide an MLP decoding and training path of SDFs in the temporal window:

$$f_{t_i}^l(x_i, y_i) = B(\mathcal{F}_{t_i}^l; x_i, y_i) \tag{7}$$

$$\mathrm{SDF}_{t_i}^l(x_i, y_i, z_i) = \mathrm{MLP}(f_{t_i}^l(x_i, y_i), z_i) \tag{8}$$

Given the ground truth SDF value s_i^l computed from ground truth 3D models for every training point p_i for level l, the loss can then be defined as:

$$L = \sum_l \lambda^l \frac{1}{m^l} \sum_{i=1}^{m^l} \|\mathrm{SDF}_{t_i}^l(x_i, y_i, z_i) - s_i^l\|^2 \tag{9}$$

where λ is the weight to balance the final loss, in practice we set $\lambda^0 = 4$ for the finest level, $\lambda^1 = 1$ and $\lambda^2 = 0.1$ for the coarsest level.

4.3 Simpler Variants of the Pyramidal Approach

Based on the previous framework, various training variants can then be devised by ablating the 2D+t convolutional pyramid training or the supervision of all pyramid levels versus only limiting the loss to the finer level, as shown in Fig. 2.

- **Static pyramidal variant.** We use a simple training variant, by considering single input frames $K = 1$ and consequently replacing 3D (2D+t) convolutions by 2D convolutions, in effect keeping the aggregation from each single feature map $\mathcal{Z}^0$, $\mathcal{Z}^1$, and $\mathcal{Z}^2$ decoded from the input feature pyramid to the corresponding $\mathcal{F}^0$, $\mathcal{F}^1$, and $\mathcal{F}^2$. The pyramid is trained including training samples for all pyramid levels $\mathcal{F}^l$ in the loss L from (9). The static pyramidal baseline is illustrated in Fig. 2(b).
- The **temporal naive variant** keeps the (2D+t) convolutions for a temporal frame group $t \in \{1, \ldots, K\}$ but only supervises the finest pyramid level.
- The **temporal pyramidal variant** includes the loss terms for all 3 pyramid levels, and uses unmatched, untracked randomized sample training points in each frame of the processed temporal frame group. The temporal naive and temporal pyramidal variants are illustrated respectively in Fig. 2(a) and (c).

4.4 Full Spatio-Temporal Pyramidal Training

While the previous pyramidal variants account for spatial hierarchical aspects in training, no component in the previous training schemes accounts for hierarchical temporal aggregation or weak surface motion priors in the training losses. One would expect such terms to help the training balance global spatio-temporal aspects against local spatio-temporal details for the underlying surface in motion.

Here we propose a temporal coarse-to-fine spatio-temporal training supervision for $K = 4$ time frames, extending the previous temporal pyramidal variant. In this training strategy we add gradual pooling of the queried features from feature maps $\mathcal{F}^l_t$ to provide 4 supervision signals at the finer level with map $\mathcal{F}^0$, 2 supervision signals at the mid-level $\mathcal{F}^1$ from frame groups 1, 2 and 3, 4, and one signal global to the sequence for the coarsest level $\mathcal{F}^2$, averaging the feature over all 4 frames. At training time, for a point $p = (x, y, z)$ among the uniform group of training points, this can be done simply by retrieving the four features in the four temporal maps corresponding to point p in the coarse map $\mathcal{F}^2$ and the two averages by retrieving the two features from temporal frames 1, 2 and 3, 4 respectively. To perform this in a meaningful way for points on and at the vicinity of the surface however, the features pooled temporally should be aggregated from a coherent point trajectory in the temporal sequence. For this we leverage training datasets for which a temporal template fitting is provided (*e.g.* with SMPL [25]) to work with temporally registered vertices. Intuitively, this allows the proposed network to account for a weak prior on underlying point trajectory coherence when producing estimated SDF sequences at inference time.

To formalize this sampling strategy we can first denote v^t_i the 3D position of the i^{th} vertex in the temporal registered model in frame t. Then, similarly to what is done in Section 4.2, each of the query points p_i^1 generated in the vicinity to the surface in the first frame is obtained by choosing a point on a 26-point 3D grid centered around a vertex of index v_i^1 on the fitted ground truth model in the first frame. However instead of sampling the query point independently for each of the 4 frames in the temporal window, we reuse the same vertex indices and the same offset w.r.t to that vertex throughout the 4 frames to obtain query points $(p_i^t)_{t=1}^4$ along a trajectory that follows the body motion. i.e. $p_i^t = v_i^t + p_i^1 - v_i^1$. We can then formalize the averaging of the features extracted across several frames in a set S using the point trajectories $(p_i^t)_{t\in S}$ as:

$$\overline{f}^l_{\{S\}} = \frac{1}{|S|}\sum_{t\in S} f_t^l(x_i^t, y_i^t) \tag{10}$$

with $p_i^t = (x_i^t, y_i^t, z_i^t)$. As shown in Fig. 2(d), for each point sample we use features $\overline{f}^2_{\{1,2,3,4\}}$ to compute a low-level loss and the features $\overline{f}^1_{\{1,2\}}$ and $\overline{f}^1_{\{3,4\}}$ to compute two mid-level losses. For each of these pooled features, we use the mean of the point's ground truth SDFs as supervision signal after MLP decoding.

5 Experiments

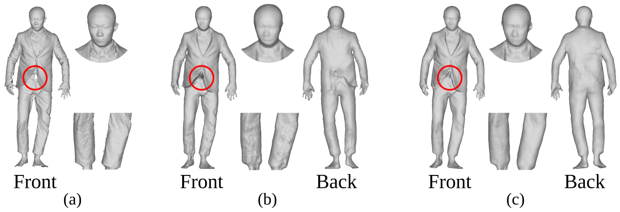

Fig. 3. Shape completion on CAPE. (a) front-view partial scan. Completion with: (b) our static pyramidal method, (c) our spatio-temporal method.

We provide quantitative and qualitative comparisons on real-world scan data. In particular, we discuss the results of the different training strategies introduced in §4. We also show numerical comparisons, on raw scan data, between our method and both learning-based method and model-based methods.

5.1 Dataset

Focusing on dynamic human shape completion, we collect data from the CAPE [27] dataset for dressed humans with different clothing styles for characters, and the DFAUST [9] dataset for undressed humans, as in competitive methods [34,51]. Both are captured at 60fps and provide temporally coherent mesh sequences alongside the raw scan data. We render depth images in resolution 512^2 from the raw scans in order to preserve the measurement noise. We also pre-compute pseudo ground-truth signed distances using the mesh models to avoid topological artifacts present in the raw data. Meshes in these datasets are actually obtained by fitting the SMPL [25] model to the raw scan data. However, thanks to the local reasoning, our network tries to capture the partwise space spanned by the local geometric patterns from the fitted models and is agnostic to the associated shapewise parametric space, here SMPL.

Data	CAPE		DFAUST	
Method	IoU ↑	Chamfer-L1 ↓	IoU ↑	Chamfer-L1 ↓
(i) naive temporal (a)	0.777	0.174	0.858	0.115
(ii) static pyramidal (b)	0.788	0.172	0.871	0.112
(iii) temporal pyramidal (c)	0.808	0.182	0.873	0.118
(iv) spatio-temporal pyramidal (d)	**0.839**	**0.161**	**0.898**	**0.103**
(v) spatio-temporal occupancy	0.800	0.168	0.872	0.109

Table 1. Pyramidal Approach Variants. Spatial completion with IoU and Chamfer-L1 distances ($\times 10^{-1}$) in the real 3D space.

5.2 Training Protocol

We follow the training protocol of [51]. We take 2 male and 2 female characters from each dataset, in total 8 characters. Each character performs 3 or 4 motion styles, in total there are 28 motion sequences. 6 sub-sequences with 4 frames are extracted in each motion sequence. We consider 4 consecutive frames as it experimentally proved to be a good trade-off between quality and computational cost. In addition, the 4-frame training sub-sequences are built with two durations: short and long. Note that we only train one model for the characters in the two datasets and short as well as long sub-sequences.

5.3 Ablation and Variants Comparison

We validate our method by testing the different variants of §4 on 152 frames from new released CAPE data and 100 frames of two unseen identities from DFAUST in Tab. 1. Our evaluation metrics are Intersection over Union(IoU) and Chamfer-L1 distance. These results show that the spatio-temporal pyramidal training strategy in §4.4 and illustrated in Fig. 2(d) yields the best results. In Tab. 1, from row(i) to row(iii), the pyramidal training is more effective than the naive training. In row(iv), we leverage the coarse-to-fine strategy not only spatially but also temporally and obtain the improvement from row(ii) to row(iv), especially for the more difficult case of clothed humans with the CAPE dataset. Comparing row(iv) and (v) highlights the effectiveness of our neural signed distance representation with respect to occupancy. In order to better illustrate the contribution of the temporal information, Fig. 3 shows how the spatio-temporal pyramidal model can correct artefacts still present with the static pyramidal.

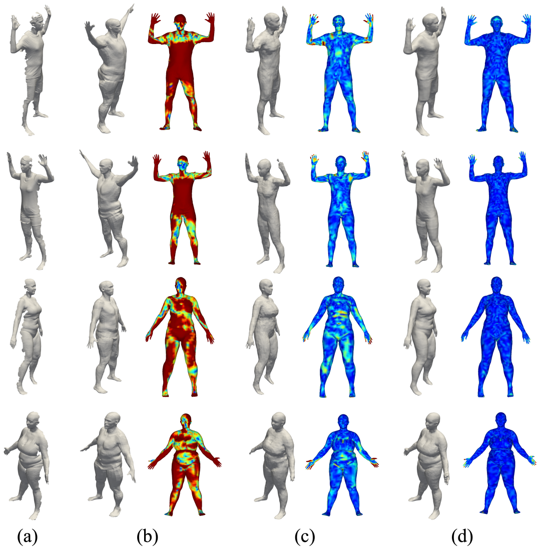

Fig. 4. Static Shape completion, top two identities are from CAPE and bottom two are from DFAUST. From left to right, (a) front-view partial scan, completion with (b) 3D-CODED [19], (c) IF-Net [12] and (d) our method. We set the maximum reconstruction to ground truth Chamfer-L1 distance as 2cm for heatmaps.

5.4 Learning-based Method Comparisons

Baselines We first compare our method with learning-based baselines in the following categories: 1. *Static methods* as 3D-CODED [19], ONet [30], BPS [40], IF-Net [12] and 2. *Dynamic methods* as OFlow [32] and STIF [51]. 3D-CODED and BPS are point-based methods which take point cloud as input and output the template-aligned mesh. ONet, IF-Net, OFlow and STIF-Net are implicit function learning methods as ours. IF-Net relies on the 3D convolution so it preprocesses the partial observation input into voxel grid. And STIF-Net inputs the depth image as ours. To fairly compare with IF-Net and STIF-Net, we use the

Data	CAPE		DFAUST	
Method	IoU ↑	Chamfer-L1 ↓	IoU ↑	Chamfer-L1 ↓
3D-CODED [19]	0.455	0.591	0.578	0.347
ONet [30]	0.488	0.476	0.604	0.340
IF-Net [12]	0.853	0.121	0.876	0.107
BPS [40]	-	-	0.761	0.197
OFlow [32]	-	-	0.740	0.231
STIF [51]	0.834	0.113	0.865	0.104
ours	**0.880**	**0.100**	**0.914**	**0.092**

Table 2. Comparison with learning-based methods with IoU and Chamfer-L1 distances ($\times 10^{-1}$) in the real 3D space on [51] benchmarck.

same resolution of input. The final mesh surface is extracted from 256^3 grid with marching cubes [22,26].

Results We follow the evaluation protocol of STIF-Net [51], evaluating on 9 characters, including 2 unseen ones w.r.t. training data, with Intersection over Union(IoU) and Chamfer-L1 distance. In Tab 2, we improve both IoU and Chamfer-L1 distance on both datasets. We would like to highlight the benefit from 2 aspects: the dynamic modeling and the pyramidal learning strategy. In Fig. 4, our method largely improves the qualitative results from the static methods, especially for detail preservation. Our pyramidal learning strategy still retains more detail than dynamic state-of-the-art [51], the wrinkle, face details and it could correct some artifacts, *e.g.* hand, leg in Fig. 5.

5.5 Model-based Method Comparisons

Baselines We compare with model-based methods OpenPose [10]+SMPL [25], IP-Net [7] and NPMs [34]. Such methods require a latent space optimization process. For more details, IP-Net extracts the surface from partial observation input with learning-based method [12], then optimizes the SMPL+D [1,21] model parameters to fit the surface. NPMs learns the neural parameter model from real-world data and synthetic data and optimize such parameters to fit the partial observation during inference. Note that IP-Net [7] and NPMs [34] train on 45000 frames from not only CAPE, but also from the synthetic data DeformingThings4D [49] and motion capture data AMASS [28], while we train on 1344 frames from the real scan data of CAPE and DFAUST only.

Results We follow the evaluation protocol of NPMs [34] on 4 identities in 2 cloth styles with IoU and Chamfer-L2 distance. In Tab. 3, our method improves the numeric result on IoU but not on the Chamfer-L2 loss. One may note that since our method is devoid of a parametric or latent control space, it has more freedom to reconstruct details, see Fig. 5. NPMs improves over a first inference phase by using a latent-space optimization which specifically minimizes for an L2 surface loss. Thus, NPMs is slightly better than ours on Chamfer-L2 metric but increases the inference time. While lacking such a stage, the results produced by our method are still observed to be generally competitive with the latter optimization methods.

Method	IoU ↑	Chamfer-L2 ↓
OpenPose+SMPL	0.68	0. 243
IP-Net [7]	0.82	0. 034
NPMs [34]	0.83	**0.022**
ours	**0.89**	0.029

Table 3. Comparison with model-based methods with IoU and Chamfer-L2 distances ($\times 10^{-3}$) for CAPE in the normalized space on [34] benchmark. Reuse the table of [34].

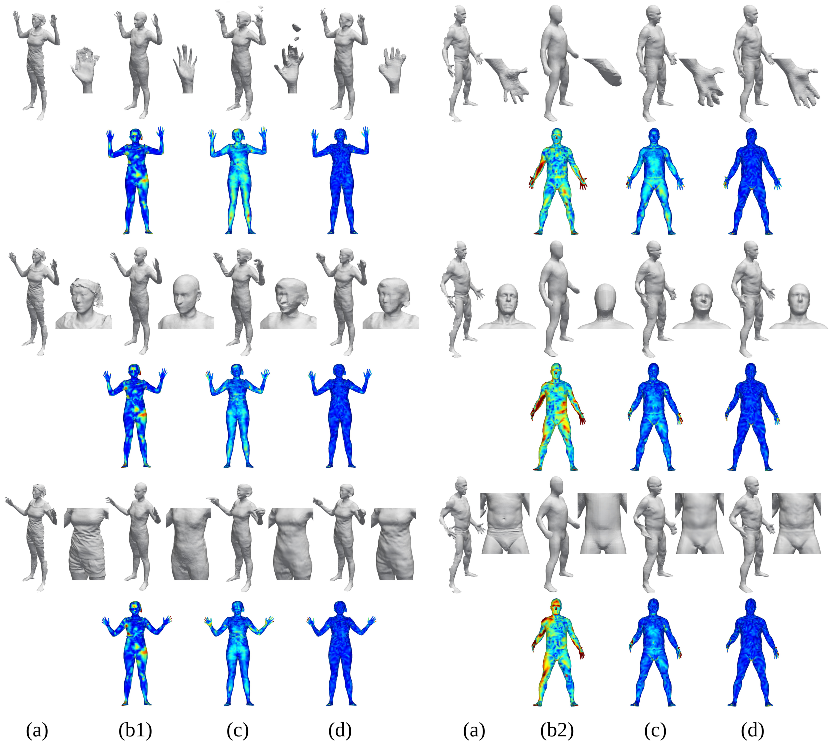

Fig. 5. Motion completion on CAPE(left) and DFAUST(right) datasets. From left to right, (a) front-view partial scan, completion with (b1) NPMs [34], (b2) OFlow [32], (c) STIF [51] and (d) our method. We set the maximum reconstruction to ground truth Chamfer-L1 distance as 2cm for heatmaps.

6 Conclusion

We propose in this paper a complete framework for coarse-to-fine treatment and training of implicit depth completion from partial depth maps. We demonstrate the substantial quality improvement that can be obtained by using an architecture that proposes a residual pyramid of features in the context of implicit surface regression and a corresponding tailored training strategy which distributes losses at each scale of the feature map both spatially and temporally. The proposed architecture benefits from the temporal consistency and allows to preserve the high-frequency details of the surface. This result suggests the applicability of such implicit pyramid schemes for other problems in the realm of 3D vision and reconstruction.

Acknowledgements This work has been partially supported by MIAI@Grenoble Alpes (ANR-19-P3IA-0003).

References

1. Alldieck, T., Magnor, M., Bhatnagar, B.L., Theobalt, C., Pons-Moll, G.: Learning to reconstruct people in clothing from a single rgb camera. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
2. Alldieck, T., Magnor, M., Xu, W., Theobalt, C., Pons-Moll, G.: Detailed human avatars from monocular video. In: International Conference on 3D Vision. pp. 98–109 (Sep 2018). `https://doi.org/10.1109/3{DV}.2018.00022`
3. Alldieck, T., Magnor, M., Xu, W., Theobalt, C., Pons-Moll, G.: Video based reconstruction of 3d people models. In: IEEE Conference on Computer Vision and Pattern Recognition (2018)
4. Anguelov, D., Srinivasan, P., Koller, D., Thrun, S., Rodgers, J., Davis, J.: Scape: shape completion and animation of people. ACM Transactions on Graphics (TOG) **24**(3), 408–416 (2005)
5. Atzmon, M., Lipman, Y.: Sal: Sign agnostic learning of shapes from raw data. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (June 2020)
6. Atzmon, M., Lipman, Y.: Sald: Sign agnostic learning with derivatives. In: International Conference on Learning Representations (2021), `https://openreview.net/forum?id=7EDgLu9reQD`
7. Bhatnagar, B.L., Sminchisescu, C., Theobalt, C., Pons-Moll, G.: Combining implicit function learning and parametric models for 3d human reconstruction. In: Proceedings of the European Conference on Computer Vision. Springer (August 2020)
8. Bhatnagar, B.L., Sminchisescu, C., Theobalt, C., Pons-Moll, G.: Loopreg: Self-supervised learning of implicit surface correspondences, pose and shape for 3d human mesh registration. In: Advances in Neural Information Processing Systems (NeurIPS) (December 2020)
9. Bogo, F., Romero, J., Pons-Moll, G., Black, M.J.: Dynamic faust: Registering human bodies in motion. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6233–6242 (2017)
10. Cao, Z., Hidalgo Martinez, G., Simon, T., Wei, S., Sheikh, Y.A.: Openpose: Realtime multi-person 2d pose estimation using part affinity fields. IEEE Transactions on Pattern Analysis and Machine Intelligence (2019)
11. Chen, Z., Zhang, H.: Learning implicit fields for generative shape modeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5939–5948 (2019)
12. Chibane, J., Alldieck, T., Pons-Moll, G.: 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)
13. Chibane, J., Mir, A., Pons-Moll, G.: Neural unsigned distance fields for implicit function learning. In: Advances in Neural Information Processing Systems (NeurIPS) (December 2020)
14. Cho, K., Van Merriënboer, B., Bahdanau, D., Bengio, Y.: On the properties of neural machine translation: Encoder-decoder approaches. arXiv preprint arXiv:1409.1259 (2014)
15. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555 (2014)

16. Deng, B., Lewis, J.P., Jeruzalski, T., Pons-Moll, G., Hinton, G., Norouzi, M., Tagliasacchi, A.: Nasa neural articulated shape approximation. In: Proceedings of the European Conference on Computer Vision. pp. 612–628. Springer (2020)
17. Dou, M., Khamis, S., Degtyarev, Y., Davidson, P., Fanello, S.R., Kowdle, A., Escolano, S.O., Rhemann, C., Kim, D., Taylor, J., Kohli, P., Tankovich, V., Izadi, S.: Fusion4d: Real-time performance capture of challenging scenes. ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH 2016 **35** (July 2016), `https://www.microsoft.com/en-us/research/publication/fusion4d-real-time-performance-capture-challenging-scenes-2/`
18. Gropp, A., Yariv, L., Haim, N., Atzmon, M., Lipman, Y.: Implicit geometric regularization for learning shapes. In: Proceedings of Machine Learning and Systems 2020, pp. 3569–3579 (2020)
19. Groueix, T., Fisher, M., Kim, V.G., Russell, B., Aubry, M.: 3d-coded : 3d correspondences by deep deformation. In: Proceedings of the European Conference on Computer Vision. pp. 235–251 (2018)
20. Huang, Z., Xu, Y., Lassner, C., Li, H., Tung, T.: Arch: Animatable reconstruction of clothed humans. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3093–3102 (2020)
21. Lazova, V., Insafutdinov, E., Pons-Moll, G.: 360-degree textures of people in clothing from a single image. In: International Conference on 3D Vision (3DV) (sep 2019)
22. Lewiner, T., Lopes, H., Vieira, A.W., Tavares, G.: Efficient implementation of marching cubes' cases with topological guarantees. Journal of graphics tools **8**(2), 1–15 (2003)
23. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 936–944 (2017)
24. Liu, S., Saito, S., Chen, W., Li, H.: Learning to infer implicit surfaces without 3d supervision (2019)
25. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: SMPL: A skinned multi-person linear model. ACM Transaction on Graphics (TOG) **34**(6), 248:1–248:16 (Oct 2015)
26. Lorensen, W.E., Cline, H.E.: Marching cubes: A high resolution 3d surface construction algorithm. ACM siggraph computer graphics **21**(4), 163–169 (1987)
27. 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: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (Jun 2020)
28. Mahmood, N., Ghorbani, N., Troje, N.F., Pons-Moll, G., Black, M.J.: AMASS: Archive of motion capture as surface shapes. In: International Conference on Computer Vision. pp. 5442–5451 (Oct 2019)
29. von Marcard, T., Henschel, R., Black, M., Rosenhahn, B., Pons-Moll, G.: Recovering accurate 3d human pose in the wild using imus and a moving camera. In: Proceedings of the European Conference on Computer Vision (sep 2018)
30. Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy networks: Learning 3d reconstruction in function space. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2019)
31. Newcombe, R.A., Fox, D., Seitz, S.M.: Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 343–352 (2015)

32. Niemeyer, M., Mescheder, L., Oechsle, M., Geiger, A.: Occupancy flow: 4d reconstruction by learning particle dynamics. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (Oct 2019)
33. Niemeyer, M., Mescheder, L., Oechsle, M., Geiger, A.: Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3504–3515 (2020)
34. Palafox, P., Božič, A., Thies, J., Nießner, M., Dai, A.: Npms: Neural parametric models for 3d deformable shapes. In: Proceedings of the International Conference on Computer Vision (2021)
35. Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: Deepsdf: Learning continuous signed distance functions for shape representation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 165–174 (2019)
36. Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A.A.A., Tzionas, D., Black, M.J.: Expressive body capture: 3d hands, face, and body from a single image. In: Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (2019)
37. Peng, S., Niemeyer, M., Mescheder, L., Pollefeys, M., Geiger, A.: Convolutional occupancy networks. In: European Conference on Computer Vision (ECCV) (2020)
38. Pons-Moll, G., Pujades, S., Hu, S., Black, M.J.: Clothcap: Seamless 4d clothing capture and retargeting. ACM Transactions on Graphics (TOG) **36**(4), 1–15 (2017)
39. Pons-Moll, G., Romero, J., Mahmood, N., Black, M.J.: Dyna: A model of dynamic human shape in motion. vol. 34, pp. 120:1–120:14 (Aug 2015)
40. Prokudin, S., Lassner, C., Romero, J.: Efficient learning on point clouds with basis point sets. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4332–4341 (Oct 2019)
41. 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)
42. Saito, S., Huang, Z., Natsume, R., Morishima, S., Kanazawa, A., Li, H.: Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2304–2314 (2019)
43. Saito, S., Simon, T., Saragih, J., Joo, H.: 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)
44. Sun, D., Yang, X., Liu, M.Y., Kautz, J.: PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume (2018)
45. Sun, J., Xie, Y., Chen, L., Zhou, X., Bao, H.: NeuralRecon: Real-time coherent 3D reconstruction from monocular video. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2021)
46. 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: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6184–6193 (2020)
47. Xu, Q., Wang, W., Ceylan, D., Mech, R., Neumann, U.: Disn: Deep implicit surface network for high-quality single-view 3d reconstruction. In: Advances in Neural Information Processing Systems 32, pp. 492–502 (2019)

48. Yang, J., Mao, W., Alvarez, J.M., Liu, M.: Cost volume pyramid based depth inference for multi-view stereo. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
49. Yang Li, Hikari Takehara, T.T.B.Z., Nießner, M.: 4dcomplete: Non-rigid motion estimation beyond the observable surface. (2021)
50. Yu, T., Zheng, Z., Guo, K., Liu, P., Dai, Q., Liu, Y.: Function4d: Real-time human volumetric capture from very sparse consumer rgbd sensors. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (June 2021)
51. Zhou, B., Franco, J.S., Bogo, F., Boyer, E.: Spatio-temporal human shape completion with implicit function networks. In: Proceedings of the International Conference on 3D Vision (2021)
52. Zhou, B., Franco, J.S., Bogo, F., Tekin, B., Boyer, E.: Reconstructing human body mesh from point clouds by adversarial gp network. In: Proceedings of the Asian Conference on Computer Vision (ACCV) (November 2020)