DOI: 10.1111/cgf.14903

Eurographics Symposium on Geometry Processing 2023
P. Memari and J. Solomon
(Guest Editors)

COMPUTER GRAPHICS *forum*
Volume 42 (2023), Number 5
e14903 pp. 1 - 13

Deep Deformation Detail Synthesis for Thin Shell Models

Lan Chen[1,2], Lin Gao[†1,3], Jie Yang[1,3], Shibiao Xu[†4], Juntao Ye[2], Xiaopeng Zhang[2] and Yu-Kun Lai[5]

[1]University of Chinese Academy of Science, China
[2]Institute of Automation, Chinese Academy of Sciences, China
[3]Institute of Computing Technology, Chinese Academy of Sciences, China
[4]Beijing University of Posts and Telecommunications, China
[5]Cardiff University, United Kingdom

Abstract

In physics-based cloth animation, rich folds and detailed wrinkles are achieved at the cost of expensive computational resources and huge labor tuning. Data-driven techniques make efforts to reduce the computation significantly by utilizing a preprocessed database. One type of methods relies on human poses to synthesize fitted garments, but these methods cannot be applied to general cloth animations. Another type of methods adds details to the coarse meshes obtained through simulation, which does not have such restrictions. However, existing works usually utilize coordinate-based representations which cannot cope with large-scale deformation, and requires dense vertex correspondences between coarse and fine meshes. Moreover, as such methods only add details, they require coarse meshes to be sufficiently close to fine meshes, which can be either impossible, or require unrealistic constraints to be applied when generating fine meshes. To address these challenges, we develop a temporally and spatially as-consistent-as-possible deformation representation (named TS-ACAP) and design a DeformTransformer network to learn the mapping from low-resolution meshes to ones with fine details. This TS-ACAP representation is designed to ensure both spatial and temporal consistency for sequential large-scale deformations from cloth animations. With this TS-ACAP representation, our DeformTransformer network first utilizes two mesh-based encoders to extract the coarse and fine features using shared convolutional kernels, respectively. To transduct the coarse features to the fine ones, we leverage the spatial and temporal Transformer network that consists of vertex-level and frame-level attention mechanisms to ensure detail enhancement and temporal coherence of the prediction. Experimental results show that our method is able to produce reliable and realistic animations in various datasets at high frame rates with superior detail synthesis abilities compared to existing methods.

CCS Concepts
• ***Computing methodologies*** → *Physical simulation; Artificial intelligence;*

1. Introduction

Creating dynamic general clothes or garments on animated characters has been a long-standing problem in computer graphics (CG). In the CG industry, physics-based simulations (PBS) are used to achieve realistic and detailed folding patterns for garment animations. However, it is time-consuming and requires expertise to synthesize fine geometric details since high-resolution meshes with tens of thousands or more vertices are often required. For example, 10 seconds are required for the physics-based simulation of a frame for detailed skirt animation shown in Fig. 1. Not surprisingly, garment animation remains a bottleneck in many applications. Recently, data-driven methods provide alternative solutions to fast and effective wrinkling behaviors for garments. Depending on body poses, some data-driven methods [WHRO10, FYK10, dASTH10, SOC19, WSFM19, PMJ*22] are capable of generating tight or loose-fitting cloth animations successfully.

Instead of using human poses as guidance, wrinkle augmentation on coarse simulations provides another alternative. It utilizes very efficient coarse simulations to recover high-level deformation and leverages learning-based methods to add realistic wrinkles. Previous methods [KGBS11, ZBO13, CZY21, CYJ*18] commonly require dense correspondences between coarse and fine meshes, so that local details can be added without affecting global deformation. Such methods also require coarse meshes to be sufficiently close to fine meshes, as they only add details to coarse meshes. To maintain the correspondences for training data and ensure closeness between coarse and fine meshes, weak-form constraints such as various test functions [KGBS11,ZBO13,CYJ*18] are applied to make fine meshes track the coarse meshes, but as a result, the obtained high-resolution meshes do not fully follow physical behavior, leading to animations that lack realism. An example is shown in

[†] Corresponding authors are Lin Gao (gaolin@ict.ac.cn) and Shibiao Xu (shibiaoxu@bupt.edu.cn).

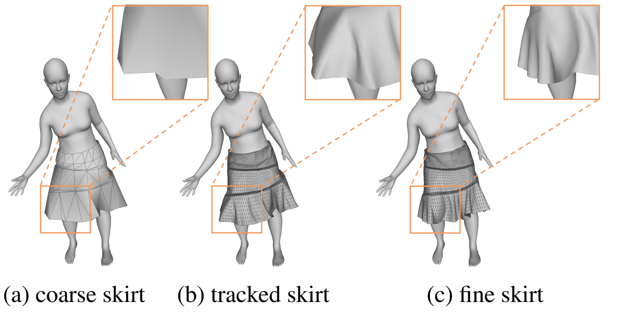


Figure 1: *One frame of the skirt in different representations. (a) coarse mesh (207 triangles), (b) tracked mesh (13,248 triangles) and (c) fine mesh (13,248 triangles). Both coarse and fine meshes are obtained by simulating the skirt using a physics-based method [NSO12]. The tracked mesh is obtained with a physics-based simulation involving additional constraints to track the coarse mesh. The tracked mesh exhibits stiff folds while the wrinkles in the fine simulated mesh are more realistic.*

Fig. 1 where the tracked skirt (b) loses a large amount of wrinkles which should appear when simulating on fine meshes (c).

Without requiring the constraints between coarse and fine meshes, we propose the DeformTransformer network to synthesize detailed thin shell animations from coarse ones, based on deformation transfer. This is inspired by the similarity observed between pairs of coarse and fine meshes generated by PBS. Although the positions of vertices from two meshes are not aligned, the overall deformation is similar, so it is possible to predict fine-scale deformation with coarse simulation results. Most previous works [KGBS11, ZBO13, CYJ*18] use explicit vertex coordinates to represent 3D meshes, which are sensitive to translations and rotations, so they require good alignments between low- and high-resolution meshes. In our work, we regard cloth animations as non-rigid deformation and propose a novel representation for mesh sequences, called TS-ACAP (Temporally and Spatially As-Consistent-As-Possible) representation. TS-ACAP is a local deformation representation, capable of representing and solving large-scale deformation problems, while maintaining the details of meshes. Compared to the original ACAP representation [GLY*19], TS-ACAP is fundamentally designed to ensure the temporal consistency of the extracted feature sequences, and meanwhile, it can maintain the original features of ACAP to cope with large-scale deformations.

With TS-ACAP representations for both coarse and fine meshes, we leverage a sequence transduction network to map the deformation from the coarse to fine level to ensure the temporal coherence of generated sequences. We propose DeformTransformer, a Transformer-based [VSP*17] deep model to deal with the sequential deformation mapping problem in a spatial-temporal inference manner. DeformTransformer consists of vertex-level and frame-level attention mechanisms for mesh sequence transduction optimization. More concretely, the spatial inference network takes a mesh in each frame individually and learns the self-attention information from the coarse TS-ACAP domain and maps it into the fine space. Additionally, the attention between multiple frames is learned in the temporal coherence network both in the encoder and decoder phases. Unlike existing works using recurrent neural networks (RNN) [SOC19], our network is based entirely on attention, without recursion modules, so that it can be trained significantly faster than architectures based on recurrent layers. With temporally consistent features and the DeformTransformer network, our method achieves stable general cloth synthesis with fine details in an efficient manner.

In summary, the main contributions of our work are as follows:

- We propose a novel framework for fast synthesis of cloth dynamics, by learning temporally consistent deformation from low-resolution meshes to high-resolution meshes with realistic dynamics.
- To achieve this, we propose a temporally and spatially as-consistent-as-possible deformation representation (TS-ACAP) to represent the cloth mesh sequences. It is able to deal with large-scale deformation, essential for mapping between coarse and fine meshes, while ensuring temporal coherence.
- Based on the TS-ACAP, we further design an effective neural network architecture (named DeformTransformer) with spatial and temporal Transformer components, which successfully enables the high-quality synthesis of dynamic wrinkles with rich details on thin shells and maintains temporal consistency on the generated high-resolution mesh sequences.

We qualitatively and quantitatively evaluate our method for various cloth types (T-shirts, pants, skirts, square, and disk tablecloth) with different motion sequences. In Sec. 2, we review the work most related to ours. We then give a detailed description of our method in Sec. 3. We present experimental results, including extensive comparisons with state-of-the-art methods and ablation study for key components evaluation in Sec. 4, and finally, we draw conclusions and discuss future work in Sec. 5.

2. Related work

2.1. Cloth Animation

Physics-based techniques for realistic cloth simulation have been widely studied in computer graphics, using methods such as implicit Euler integrator [BW98, HVS*09], iterative optimization [TPBF87, BMF03, GHDS03], collision detection and response [Pro97, VM95], etc. Although such techniques can generate realistic cloth dynamics, they are time-consuming for detailed cloth synthesis, and the robustness and efficiency of simulation systems are also of concern. To address these, on the one hand, graphics researchers [FTP16, WY16, WWW22] have devoted themselves to exploring a variety of optimization methods, such as preconditioned conjugate gradient, accelerated gradient descent and L-BFGS. For example, Wu *et al.* [WWW22] propose a GPU-based multilevel additive Schwarz preconditioner to simulate cloth with a high resolution, 50K to 500K vertices, in real-time. On the other hand, alternative methods have also been developed to generate the dynamic details of cloth animation via adaptive techniques [LYO*10, MC10, NSO12], data-driven approaches [dASTH10, GRH*12, WHRO10, KGBS11, ZBO13] and deep learning-based methods [CYJ*18, CZY21, GCS*19, LCT18, ZWCM20, BME20, GCP*20, SOC22, PMJ*22, TB23, ZWCM21, ZCM22], etc.

Data-driven methods offer faster cloth animations, but existing methods are limited to tighter garments [dASTH10, GRH*12, KKN*13, KV08]. Some approaches [KGBS11, ZBO13] learn a mapping from coarse to detailed garment shapes for free-flowing cloth simulation, but high-resolution cloth is required to track low-resolution cloth, thus cannot exhibit full high-resolution dynamics.

Recently, deep learning-based methods have been successfully applied for 3D animations of human faces [CWW*16, JZD*18], hair [ZWW*18, YSZZ19] and garments [LZT*19, WSFM19]. As for garment synthesis, some approaches [LCT18,SOC19,PLPM20] are proposed to utilize a two-stream strategy consisting of global garment fit and local wrinkle enhancement. Lähner *et al.* [LCT18] present DeepWrinkles, which recovers the global deformation from a 3D scan system and uses a conditional generative adversarial network to enhance a low-resolution normal map. Zhang *et al.* [ZWCM20] further generalize the augmentation method with normal maps to complex garment types as well as various motion sequences. These approaches add wrinkles on normal maps rather than geometry, and thus their effectiveness is restricted to adding fine-scale visual details, not large-scale dynamics. Based on the skinning representation, there is a tremendous amuount of research focusing on body- or skeleton-guided garment generation with neural networks, which aims to generalize to multiple body shapes [GCS*19, SOC19, GCP*20], loose-fitting garments [PMJ*22], semi-supervised or unsupervised generation [ZCM22, BME20, SOC22]. In addition, other works are devoted to generalizing neural networks to various cloth styles [PLPM20] or cloth materials [WSFM19]. Despite training with tight garments dressed on characters, some deep learning-based methods [CYJ*18, OLL18, ZWCM21] are demonstrated to work for cloth animation with higher degrees of freedom. Chen *et al.* [CYJ*18] represent coarse and fine meshes via geometry images and use a super-resolution network to learn the mapping. Oh *et al.* [OLL18] propose a multi-resolution cloth representation with fully connected networks to add details hierarchically. Since the free-flowing cloth dynamics are harder for networks to learn than tight garments, the results of these methods have not reached the realism of PBS. From another perspective, Zhang *et al.* [ZWCM21] propose to generate coarse garment proxies depending on joints, and then enhance realistic details in the garment image space. Still focusing on meshes, our method based on a novel deformation representation and network architecture has superior capabilities of learning the mapping between coarse and fine meshes, generating realistic cloth dynamics.

2.2. Representation for 3D Meshes

Unlike 2D images with a regular grid of pixels, 3D meshes have irregular connectivity, making learning difficult. Existing deep learning-based methods turn 3D meshes to various representations [XLZ*20], such as voxels, images, point clouds, meshes, etc. Volumetric representation has a regular structure but suffers from high space and time consumption. Thus Wang *et al.* [WLG*17] propose an octree-based convolutional neural network and encode the voxels sparsely. Image-based representations including depth images [EPF14, GGAM14] and multi-view images [SMKL15,LTT*19] are proposed to encode 3D models in a 2D domain. It is unavoidable that both volumetric and image-based representations lose some geometric details. Alternatively, geometry images are used in [SBR16,SUHR17,CYJ*18] for mesh classification or generation, which are obtained through cutting a 3D mesh to a topological disk, parameterizing it to a rectangular domain and regularly sampling the 3D coordinates in the 2D domain [GGH02]. However, this representation may suffer from parameterization distortion and seam line problems.

Instead of representing 3D meshes into other formats, recently there are methods [TGL*22, TGLX18, HHF*19, MBM*17, FLWM18, SACO22] applying neural networks directly to triangle meshes with various features. Researchers represent meshes as graphs and adopt graph convolutions for efficiency and convenience [WPC*20]. Mixture Model Network (MoNet) [MBM*17] adopts node pseudo-coordinates and a weight function to determine the relative position and weight between a node and its neighbor. There are many approaches under this framework, such as Geodesic CNN (GCNN) [MBBV15], Anisotropic CNN (ACNN) [BMRB16], Spline CNN [FLWM18], etc., by constructing nonparametric weight functions. Gao *et al.* [GLL*16] propose a deformation-based representation called rotation-invariant mesh difference (RIMD), which is translation and rotation invariant. However, it is expensive to reconstruct vertex coordinates from RIMD. A faster deformation representation based on an as-consistent-as-possible (ACAP) formulation is used to reconstruct meshes, but it does not guarantee temporal consistency when applied to a dynamic mesh sequence. We propose a temporally and spatially as-consistent-as-possible (TS-ACAP) representation, to ensure both spatial and temporal consistency of mesh deformation and can accelerate the computation of features.

2.3. Sequence Generation with DNNs (Deep Neural Networks)

Temporal information is crucial for stable and vivid sequence generation. Previously, recurrent neural networks (RNN) have been successfully applied in many sequence generation tasks [MKB*10, MKB*11]. However, it is difficult to train RNNs to capture long-term dependencies since RNNs suffer from the vanishing gradient problem [BSF94]. To deal with this problem, previous works proposed some variations of RNN, including long short-term memory (LSTM) [HS97] and gated recurrent unit (GRU) [CVMBB14]. These variations of RNN rely on the gating mechanisms to control the flow of information, thus performing well in the tasks that require capturing long-term dependencies, such as speech recognition [GMH13] and machine translation [BCB14, SVL14]. Recently, based on attention mechanisms, the Transformer network [VSP*17] has been verified to outperform many typical sequential models for long sequences. This structure is able to inject the global context information into each input. Based on Transformer, impressive results have been achieved in tasks with regard to audio, video and text, *e.g.* speech synthesis [LLL*19, OTSK20], action recognition [GCDZ19] and machine translation [VSP*17]. We utilize the Transformer network to learn the frame-level attention which improves the temporal stability of the generated animation sequences.

3. Approach

3.1. Overview

The overall architecture of our detail synthesis network is illustrated in Fig. 2. To synthesize realistic cloth animations, we propose

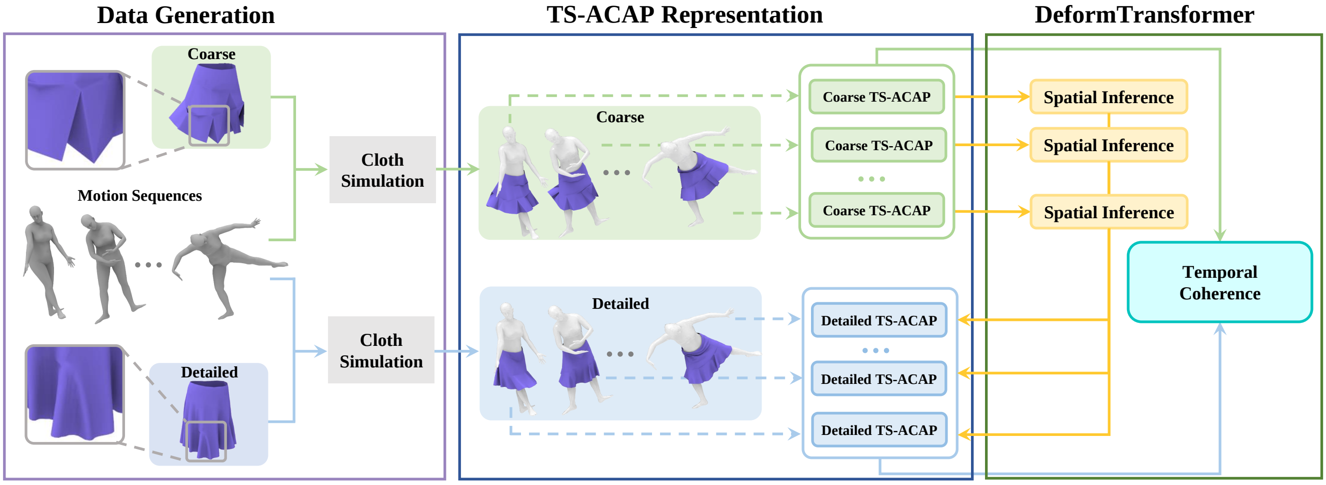


Figure 2: *The overall architecture of our detail synthesis network. At the data preparation stage, we generate low- and high-resolution thin shell animations via coarse and fine meshes and various motion sequences. Then we encode the coarse meshes and the detailed meshes to a deformation representation TS-ACAP, respectively. Our algorithm then learns to map the coarse features to fine features by designing a DeformTransformer network that consists of the spatial inference module and the temporal coherence module, and finally reconstructs the detailed animations.*

a method to simulate coarse meshes first and learn a temporally-coherent mapping to the fine meshes. To efficiently extract localized features with temporal consistency, we propose a new deformation representation, called TS-ACAP (temporally and spatially as-consistent-as-possible), which is able to cope with both large rotations and unstable sequences. Since the vertices of the fine models are typically more than ten thousand to simulate realistic wrinkles, it is hard to directly map the coarse features to the high-dimensional fine ones for the network. Therefore, convolutional encoder networks are applied to encode coarse and fine meshes in the TS-ACAP representation to their latent spaces, respectively. Unlike existing works using recurrent neural networks (RNNs) [SOC19], we use the Transformer [VSP*17], a sequence-to-sequence network architecture, based on attention mechanisms for our detail synthesis task, which is more efficient to learn and leads to superior results.

3.2. Deformation Representation

As discussed before, large-scale deformations are essential to represent thin shell mode dynamics such as cloth animations, because folding and wrinkle patterns during animation can often be complicated. Moreover, cloth animations are in the form of sequences, hence temporal coherence is very important for realism. Using 3D coordinates directly cannot cope with large-scale deformations well, and existing deformation representations are generally designed for static meshes, and directly applying them to cloth animation sequences on a frame-by-frame basis does not take temporal consistency into account. To cope with this problem, we propose a mesh deformation feature with spatial-temporal consistency, called TS-ACAP, to represent the coarse and fine deformed shapes, which exploits the localized information effectively and reconstructs meshes accurately.

Take coarse meshes $\mathcal{C}$ for instance and fine meshes $\mathcal{D}$ are processed in the same way. Assume that a sequence of coarse meshes contains n models with the same connectivity. A mesh with the same topology is chosen as the reference model, such as a garment mesh worn by a character in the T pose. We calculate the deformation gradient $\mathbf{T}_{t,i}$ to represent the local shape deformation firstly. Using polar decomposition, $\mathbf{T}_{t,i}$ can be decomposed into a rotation part and a scaling/shearing part $\mathbf{T}_{t,i} = \mathbf{R}_{t,i}\mathbf{S}_{t,i}$. The scaling/shearing transformation is uniquely defined, while the rotation $\mathbf{R}_{t,i}$ corresponds to infinite possible rotation angles (differed by multiples of 2π, along with possible opposite orientation of the rotation axis). Typical formulations often constrain the rotation angle to be within $[0,\pi]$ which is unsuitable for smooth large-scale animations.

In order to handle large-scale rotations, we first require the orientations of rotation axes and rotation angles of spatially adjacent vertices on the same mesh to be as consistent as possible. Especially for our sequence data, we further add constraints for adjacent frames to ensure the temporal consistency of the orientations of rotation axes and rotation angles on each vertex. We first consider consistent orientation for axes.

$$\begin{aligned}\arg\max_{o_{t,i}} \sum_{(i,j)\in\mathcal{E}} & o_{t,i}o_{t,j} \cdot s(\omega_{t,i}\cdot\omega_{t,j},\theta_{t,i},\theta_{t,j}) \\ & + \sum_{i\in\mathcal{V}} o_{t,i}\cdot s(\omega_{t,i}\cdot\omega_{t-1,i},\theta_{t,i},\theta_{t-1,i}) \\ & \text{s.t.}\quad o_{t,1}=1, o_{t,i}=\pm 1 (i\neq 1)\end{aligned} \tag{1}$$

where t is the index of the frame, $\mathcal{E}$ is the edge set, and $\mathcal{V}$ is the vertex set. Denote by $(\omega_{t,i},\theta_{t,i})$ one possible choice for the rotation axis and rotation angle that match $\mathbf{R}_{t,i}$. $o_{t,i}\in\{+1,-1\}$ specifies whether the rotation axis is flipped ($o_{t,i}=1$ if the rotation axis is unchanged, and -1 if its opposite is used instead). The first term promotes spatial consistency while the second term promotes temporal consistency. s is a function measuring orientation consistency, which is defined as follows:

$$s(\omega_{t,i},\omega_{t,j}) = \begin{cases} 0, & |\omega_{t,i}\cdot\omega_{t,j}|\le\varepsilon_1 \text{ or } \theta_{t,i}<\varepsilon_2 \text{ or } \theta_{t,j}<\varepsilon_2 \\ 1, & \text{Otherwise if } \omega_{t,i}\cdot\omega_{t,j}>\varepsilon_1 \\ -1, & \text{Otherwise if } \omega_{t,i}\cdot\omega_{t,j}<-\varepsilon_1 \end{cases}$$

$$\tag{2}$$

The first case here is to ignore settings where the rotation angle is near zero, as the rotation axis is not well defined in such cases. As for rotation angles, we optimize the following

$$\begin{aligned}\arg\min_{r_{t,i}} & \sum_{(i,j)\in\mathcal{E}} \|(r_{t,i}\cdot 2\pi + o_{t,i}\theta_{t,i}) - (r_{t,j}\cdot 2\pi + o_{t,j}\theta_{t,j})\|_2^2 \\ & + \sum_{i\in\mathcal{V}} \|(r_{t,i}\cdot 2\pi + o_{t,i}\theta_{t,i}) - (r_{t-1,i}\cdot 2\pi + o_{t,j}\theta_{t-1,i})\|_2^2 \\ \text{s.t.} \quad & r_{t,i}\in\mathbb{Z}, \quad r_{t,1}=0.\end{aligned} \tag{3}$$

where $r_{t,i}\in\mathbb{Z}$ specifies how many 2π rotations should be added to the rotation angle. The two terms here promote spatial and temporal consistencies of rotation angles, respectively. These optimizations can be solved using integer programming, and we use the mixed integer CoMISo [BZK09] which provides an efficient solver. See [GLY*19] for more details. A similar process is used to compute the TS-ACAP representation of the fine meshes.

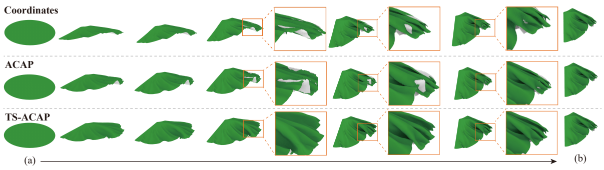


Figure 3: *Comparison of shape interpolation results with different representations: coordinates, ACAP and TS-ACAP. (a) and (b) are the source and target models respectively with large-scale deformation to be interpolated. The interpolated models with ACAP feature show plausible wrinkles in each frame while resulting in self-intersections and causing inconsistency in the temporal domain.*

Compared to the ACAP representation, our TS-ACAP representation considers temporal constraints to represent nonlinear deformation for optimization of axes and angles, which is more suitable for consecutive large-scale deformation sequences. We compare ACAP and TS-ACAP using a simple example of a simulated disk-shaped cloth animation sequence. Once we obtain deformation representations of the meshes in the sequence, we interpolate two meshes, the initial state mesh and a randomly selected frame, using linear interpolation of shape representations. In Fig. 3, we demonstrate the interpolation results with ACAP representation, which shows that it cannot handle such challenging cases with complex large-scale deformations. In contrast, with our temporally and spatially as-consistent-as-possible optimization, our TS-ACAP representation is able to produce consistent interpolation results.

3.3. DeformTransformer Networks

Unlike [TGLX18, WSFM19] which use fully connected layers for mesh encoder, we perform a convolution operator on vertices [DMI*15, TGL*22] where the output feature $\mathbf{f}$ at a vertex is obtained as a linear combination of input in its one-ring neighbors along with a bias (please see detailed formulation in the supplementary material). Let $\mathcal{F}_\mathcal{C}=\{\mathbf{f}_{\mathcal{C}_1},\ldots,\mathbf{f}_{\mathcal{C}_n}\}$ be the sequence of coarse mesh features, and $\mathcal{F}_\mathcal{D}=\{\mathbf{f}_{\mathcal{D}_1},\ldots,\mathbf{f}_{\mathcal{D}_n}\}$ be its counterpart, the sequence of detailed mesh features. To synthesize $\mathcal{F}_\mathcal{D}$ from $\mathcal{F}_\mathcal{C}$, the DeformTransformer framework is proposed to solve this sequence-to-sequence problem. As illustrated in Fig. 4, the DeformTransformer framework consists of two sub-networks: the *Spatial Inference* module, which learns feature mapping of individual frames using mesh transformer encoders and upsampling networks, and the *Temporal Coherence* module, which consists of two temporal transformer encoders and a temporal transformer decoder to generate temporally-coherent deformations.

Spatial Inference Module. Since input mesh features share the same vertex numbers and connected edges, we infer the high-resolution features while maintaining a fixed detailed mesh topology at the spatial level. We design a mesh transformer encoder to capture the deformations among all vertices. For coarse meshes with $V_\mathcal{C}$ vertices, the input features $\mathbf{f}_{\mathcal{C}_i}$ are $(V_\mathcal{C},K)$ dimensional, where K is the number of filter kernels in the last mesh convolution layer (where we set K as 9 in all our experiments). Then point-wise features are fed into the mesh transformer encoder composed of identical blocks each with two sub-modules, where one is the multi-head self-attention mechanism, and the other is the frame-wise fully connected feed-forward network. Here, an attention function [BCB14] learns a mapping from a query and a set of key-value pairs to an output. The layer output is computed as a weighted sum of the values and the weight on each value is computed by a compatibility function of the query with the corresponding key. Multi-head self-attention [VSP*17] uses different linear projections in parallel and the layer outputs are concatenated and once again projected, resulting in the final values. It is more beneficial than single attention with jointly learned information from different representation subspaces at different positions. We then employ a residual connection around the multi-head self-attention layer and the feed-forward layer, followed by layer normalization. After learning the vertex-wise relationships within the mesh feature maps, we use fully-connected mesh upsampling layers (where the layer number depends on the face scale factor, *e.g.* 3 layers for 64 times upsampling) and mesh convolutional layers relying on the adjacency of detailed mesh. Here we adopt upsampling and mesh convolution instead of a transformer decoder due to the huge GPU cost for the detailed meshes with more than ten thousand vertices.

Temporal Coherence Module. The temporal coherence module consists of several stacked encoder-decoder layers. To take the order of the sequence into consideration, triangle positional embeddings [VSP*17] are injected into frames of $\mathcal{F}_\mathcal{C}$ and $\mathcal{F}_\mathcal{D}$, respectively. The features of N frames $\{\mathbf{f}_{\mathcal{C}_1},\ldots\mathbf{f}_{\mathcal{C}_N}\}$ are concatenated and reshaped into $(N, V_\mathcal{C}\cdot K)$. The coarse temporal transformer encoder takes these sequential coarse features as input and encodes them to a temporally-dependent hidden space. The architecture of the encoder module is similar to the spatial mesh transformer encoder, but here we consider temporal-level element relationships. The multi-head attention is able to build the dependence between any frames, thus ensuring that each input can consider the global context of the whole sequence. Meanwhile, compared with other sequence models, this mechanism splits the attention into several subspaces so that it can model the frame relationships in multiple aspects. Besides, a masked temporal transformer encoder takes fine mesh sequence $\mathcal{F}_\mathcal{D}$ as input and encodes it similarly to the coarse encoder. Unlike the coarse encoder, detailed meshes are generated sequentially, and when predicting frame t, it should not attend to subse-

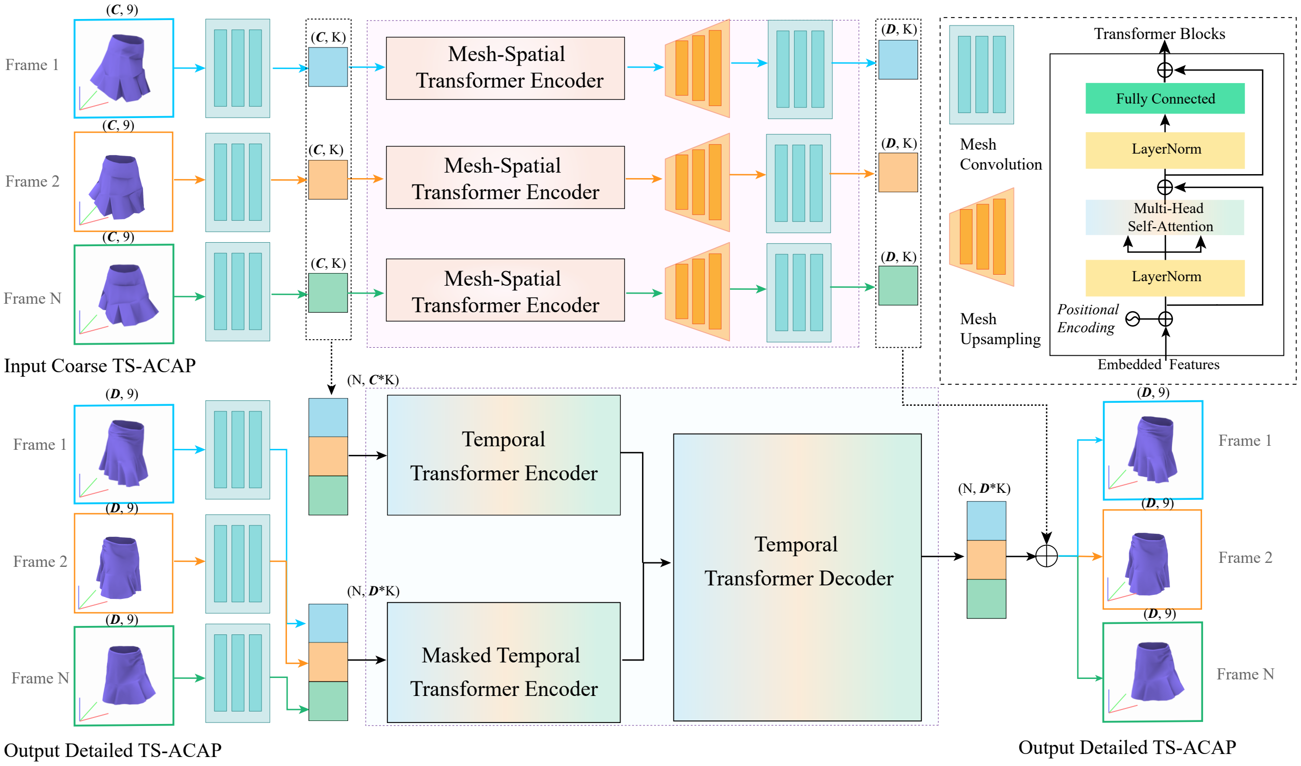


Figure 4: *The architecture of our DeformTransformer network. The coarse and fine mesh sequences are embedded into feature vectors using the TS-ACAP representation which is defined at each vertex as a 9-dimensional vector. Then two convolutional encoders map coarse and fine features to their latent spaces, respectively. The inference from sequential coarse to fine latent vectors is then modeled as spatial-temporal joint reasoning with our DeformTransformer network. The upper part shows the Spatial Inference module. The latent vectors in all vertices from one mesh are fed into a spatial mesh transformer encoder to learn feature embeddings in the spatial dimensions. Then the upsampling layers and mesh convolutional layers are applied to predict detailed features in each frame. The lower part illustrates a Temporal Coherence module consisting of two temporal transformer encoders embedding sequential features from coarse and fine spaces respectively, and a temporal transformer decoder for temporally-coherent deformation generation. The spatial and temporal predictions are fused, followed by a mesh convolution layer to get the detailed TS-ACAP features. Notice that in the training phase the input high-resolution TS-ACAP features are those from the ground truth, but during testing, these features are initialized to zeros, and once a new high-resolution frame is generated, its TS-ACAP feature is added. With predicted feature vectors, realistic and stable cloth animations are generated.*

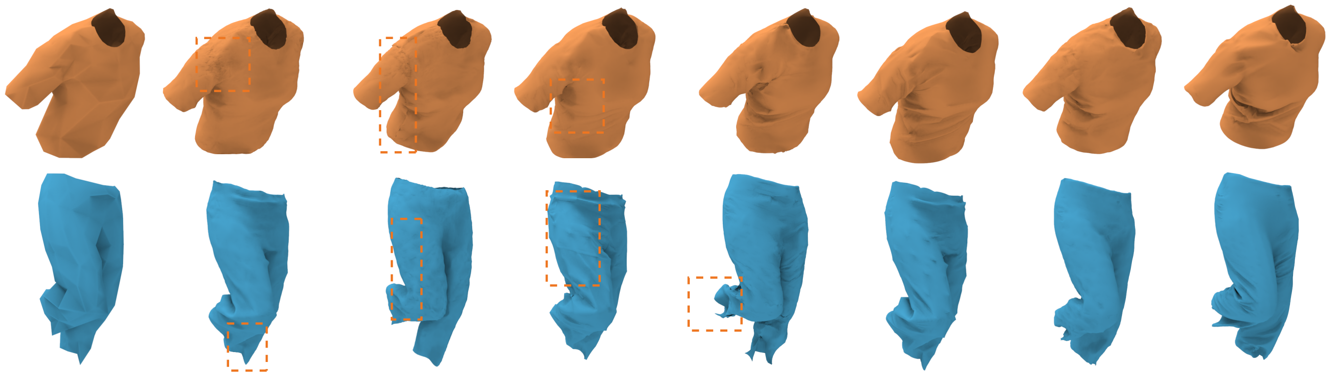
quent frames (with the position after frame t). To achieve this, we utilize a masking process for the multi-head self-attention module [VSP*17]. With the encoded coarse and fine latent vectors, the temporal transformer decoder network aims to reconstruct a sequence of fine mesh features. It performs multi-head attention over the output of the encoder, thus capturing the long-term dependencies between coarse mesh features $\mathcal{F}_\mathcal{C}$ and fine mesh features $\mathcal{F}_\mathcal{D}$.

We train the DeformTransformer network by minimizing the mean squared error between predicted features and the ground truth. Using the predicted TS-ACAP feature, we reconstruct the vertex coordinates of the target mesh applying the reconstruction algorithm for ACAP features (please refer to [GLY*19] for details). After that, a collision solving process is introduced to avoid body interaction (see details in the supplementary material).

4. Results

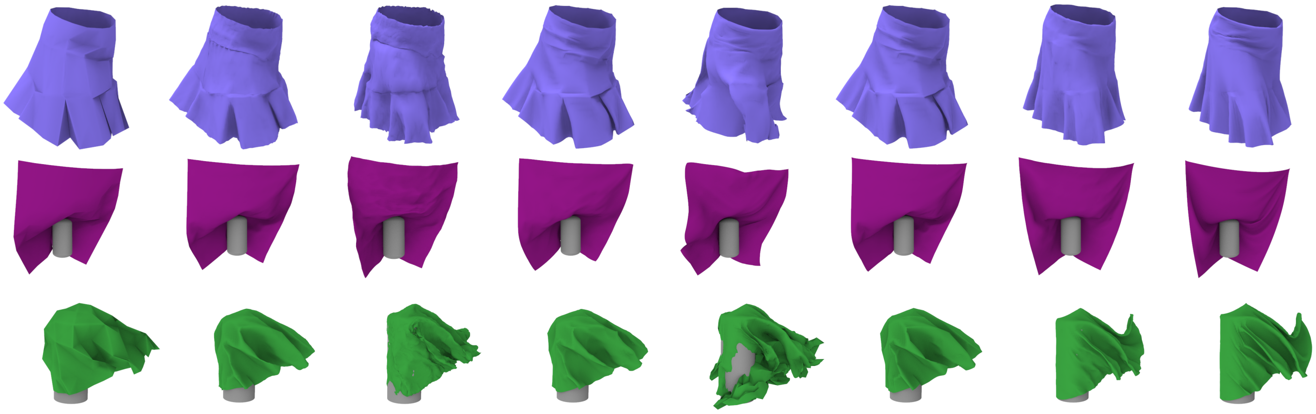
4.1. Runtime Performance

We implement our method on a computer with a 2.50GHz 4-Core Intel i5 CPU for coarse simulation and TS-ACAP extraction, and an NVIDIA GeForce® GTX 1080Ti GPU for fine TS-ACAP generation by the network and the coordinates reconstruction of the vertices. Table 1 shows the average per-frame execution time of our method for various cloth datasets (more detailed timing cost is in the supplementary material such as coarse simulation, TS-ACAP extraction, synthesis of high-resolution TS-ACAP and coordinates, and collision refinement). For reference, we also measure the time of the high-resolution physics-based simulation, with an open-source solver called ARCSim [NSO12]. The degree of optimization of the numerical algorithms is the second order. The SIMD optimization technique is AVX. This code runs in the same CPU with the coarse simulation. Our algorithm is 10 $\sim$ 35 times faster than this method. Note that there are faster physics-based simulation methods like [WWW22], and our method can also be highly accelerated for coarse simulation. The low computational cost of our method makes it suitable for interactive applications. Please refer to the supplementary material for detailed implementation and network architecture.



(a) Input (b) Chen *et al.* w/ track (c) Chen *et al.* w/o track (d) Zurdo *et al.* w/ track (e) Zurdo *et al.* w/o track (f) Tracked (g) Ours (h) GT

Figure 5: *Comparison of the reconstruction results for unseen data on the TSHIRT and PANTS datasets with tight garments. (a) coarse simulation, (b) results of [CZY21] trained on the tracked data, (c) results of [CZY21] trained on the full simulated data, (d) results of [ZBO13] trained on the tracked data, (e) results of [ZBO13] trained on the full simulated data, (f) results of tracked PBS, (g) our results, (h) ground truth generated by PBS. Our method produces detailed shapes of higher quality than Chen et al. and Zurdo et al., see the folds and wrinkles in the close-ups. Chen et al. results suffer from seam line problems. The results of Zurdo et al. exhibit clearly noticeable artifacts. It is highly recommended to zoom in for a detailed comparison.*



(a) Input (b) Chen *et al.* w/ track (c) Chen *et al.* w/o track (d) Zurdo *et al.* w/ track (e) Zurdo *et al.* w/o track (f) Tracked (g) Ours (h) GT

Figure 6: *Comparison of the reconstruction results for unseen data in the datasets of loose garments. (a) the coarse simulation, (b) the results of Chen et al. [CZY21], (c) the results of Zurdo et al. [ZBO13], (d) the results generated by physics-based tracking simulation [CZY21], (e) our results, (f) the ground truth generated by PBS.*

4.2. Fine Detail Synthesis Results and Comparisons

We demonstrate our method using various detail enhancement examples both quantitatively and qualitatively, including added wrinkles and rich dynamics. We compare our results with physics-based coarse simulations, our implementation of a deep learning-based method [CZY21], and a conventional machine learning-based method [ZBO13]. Since these two methods require dense correspondences between coarse and fine meshes, for fair comparison we also implement a tracking mechanism [CZY21] to produce paired tracked data, which is more consistent with coarse simulated results making detail refinement easier, at the cost of deviating from real ground truth. In the following comparisons, we show their results both trained with tracked data and without tracked data.

4.2.1. Qualitative Evaluation

We compare detail synthesis results on the TSHIRT and PANTS datasets with tight garments. In Fig. 5, we show (b∼e) the results of two compared methods [CZY21, ZBO13] trained with/without tracking scheme, (f) tracked PBS, (g) ours and (h) PBS as ground truth. All methods are able to reconstruct the garments completely with mid-scale wrinkles. However, Chen *et al.* [CZY21] suffer from seam line artifacts due to the geometry image representation. A geometry image is a parametric sampling of a shape made into a topological disk by cutting through seams. The boundaries of the disk

Table 1: *Statistics and timing (sec/frame) of the testing examples including five types of thin shell animations.*

Benchmark	#verts LR	#verts HR	ARCSim HR	ours	speedup
TSHIRT	246	14,190	8.72	0.867	**10**
PANTS	200	11,967	10.92	0.904	**12**
SKIRT	127	6,812	6.84	0.207	**33**
SHEET	81	4,225	2.48	0.157	**16**
DISK	148	7,729	4.93	0.139	**35**

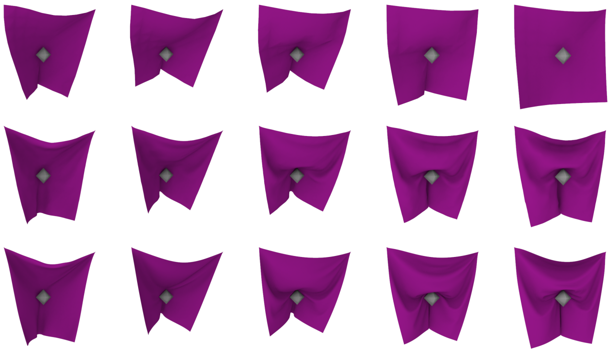
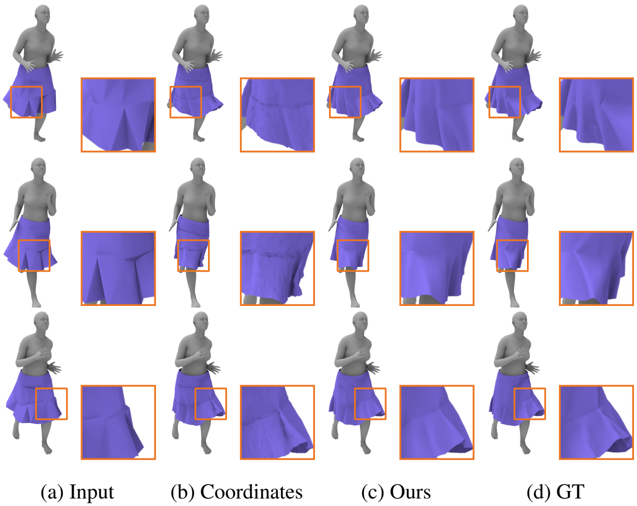


Figure 7: *Generalization evaluation for draping cloth with an octahedron unseen in the training data. From top to bottom, we show input coarse mesh, ours, and the ground truth generated by PBS.*

need to be fused to reconstruct the original topology. They introduce a padding scheme to align seam line vertices, but the predicted geometry images may not be entirely accurate, resulting in imprecise fused boundaries, *e.g.* clear seam lines on the shoulder and crooked boundaries on the left side of the waist for the examples in Figs. 5 (b) and (c). Zurdo *et al.* [ZBO13] utilize tracking algorithms for coarse and fine alignment, leading to constrained fine meshes with rigid artifacts and not exhibiting full physics-based simulation behavior. But without tracking constraints, their results may have artifacts where the meshes are not well aligned, *e.g.* the trouser legs. Different from these methods that reconstruct displacements or local coordinates, our method uses deformation-based features in both encoding and decoding phases which does not suffer from such restrictions and ensures physically-reliable results.



(a) Input (b) Coordinates (c) Ours (d) GT

Figure 8: *The evaluation of the TS-ACAP feature in our detail synthesis method.*

In addition, we show results of loose garments and free-flying cloth, with comparisons on the SKIRT, SHEET, and DISK datasets (shown in Fig. 6, more results are in the supplementary material). The results of the two compared methods exhibit obvious artifacts due to the significant misalignment between coarse and ground truth deformations (see (c) and (e)). The results with tracking (see (b) and (d)) smooth out sharp triangles, while only enhancing small wrinkles on coarse meshes maintaining global shapes. Our learned detail synthesis model provides better visual quality for shape generation and successfully reconstructs the swinging skirt (see the small wrinkles on the waist and the medium-level folds on the skirt hem) and overall drape of the disk (*i.e.* , how the tail of the disk flies like a fan in the wind). The transformer-based temporal module further ensures stable animation; please see the accompanying video.

Generalization. With an appropriately trained model with regularization, our DeformTransformer can be applied to test motions different from the training data. This capability is important for applications such as games or movies, since the variations in motions can be large and change over time. Trained on draping cloth sequences crashing with different obstacles, *e.g.* pole, sphere, torus, cube, icosahedron, the model is then applied to draping cloth crashing with an octahedron. As shown in Fig. 7, the middle-scale wrinkles can be captured and the cloth is properly deformed corresponding to the motions of the obstacle. However, the generalization ability of our method is still limited, as can be seen, the sharp wrinkles caused by the corner of octahedron are not captured, because similar scenarios did not appear in the training data. Adding more complicated examples in training data could address this problem.

4.2.2. Quantitative Evaluation

For quantitative comparison, we use three metrics: Root Mean Squared Error (RMSE), Hausdorff distance as well as spatio-temporal edge difference (STED) [VS11] designed for motion sequences with a focus on 'perceptual' error of models (shown in Table 2). Note that for the datasets from the top to bottom in the table, the Hausdorff distances between LR meshes and the ground truth are increasing. This tendency is in accordance with the deformation range from tighter garments to cloth with higher degrees of freedom. Since using positions cannot handle rotations well, the larger scale the models deform, the more artifacts Chen *et al.* [CZY21] and Zurdo *et al.* [ZBO13] would bring in the reconstructed models, leading to increased errors. The results indicate that our method has better reconstruction results quantitatively than the compared methods on the 5 datasets with all three metrics. Especially for the SKIRT, SHEET, and DISK datasets which contain loose cloth and hence larger and richer deformation, our method outperforms existing methods significantly since tracking between coarse and fine meshes is not required in our algorithm. We also show the distance evaluation between the ground truth and the results of compared methods using tracked data. Although their results have fewer ar-

Table 2: *Quantitative comparison with [CZY21] and [ZBO13] of reconstruction errors for unseen cloth animations in several datasets.*

		Methods					
Dataset	Metrics	LR	Chen *et al.* w/ track	Chen *et al.* w/o track	Zurdo *et al.* w/ track	Zurdo *et al.* w/o track	Ours
TSHIRT	RMSE $\times10^{-2}\downarrow$	-	1.508	0.76	1.52	1.04	**0.273**
	Hausdorff $\times10^{-2}\downarrow$	0.59	0.594	0.506	0.584	0.480	**0.254**
	STED $\downarrow$	-	0.238	0.277	0.217	0.281	**0.0684**
PANTS	RMSE $\times10^{-2}\downarrow$	-	1.38	1.82	1.39	1.89	**0.339**
	Hausdorff $\times10^{-2}\downarrow$	0.761	0.684	1.09	0.711	0.983	**0.293**
	STED $\downarrow$	-	0.121	0.176	0.0735	0.151	**0.0308**
SKIRT	RMSE $\times10^{-2}\downarrow$	-	3.35	2.72	3.35	2.19	**0.391**
	Hausdorff $\times10^{-2}\downarrow$	2.09	2.32	1.54	2.31	1.52	**0.352**
	STED $\downarrow$	-	0.132	0.227	0.0586	0.178	**0.0239**
SHEET	RMSE $\times10^{-2}\downarrow$	-	3.35	4.37	3.42	3.02	**0.543**
	Hausdorff $\times10^{-2}\downarrow$	2.61	2.83	2.60	2.89	2.34	**0.443**
	STED $\downarrow$	-	0.0407	0.155	0.0272	0.0672	**0.0259**
DISK	RMSE $\times10^{-2}\downarrow$	-	11.11	7.03	11.04	11.40	**2.19**
	Hausdorff $\times10^{-2}\downarrow$	3.12	3.49	2.27	3.48	2.23	**1.46**
	STED $\downarrow$	-	0.0867	0.244	0.0809	0.502	**0.0542**

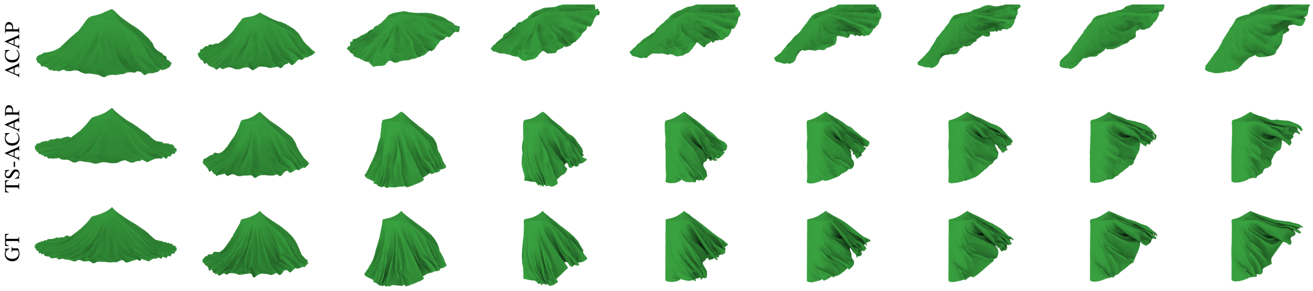


Figure 9: *Consecutive generated frames from a testing sequence in the DISK dataset with ACAP, TS-ACAP, and ground truth.*

Table 3: *Per-vertex error (RMSE $\times10^{-2}$) on predictions with different representations: 3D coordinates, ACAP and TS-ACAP.*

Dataset	TSHIRT	PANTS	SKIRT	SHEET	DISK
3D coordinates	1.01	1.93	0.941	0.86	18.5
ACAP	0.614	0.785	0.693	0.606	3.51
TS-ACAP	**0.273**	**0.339**	**0.391**	**0.543**	**2.19**

tifacts which largely reduces the STED value, their wrinkles and deformations are not similar to the full-model simulation thus still resulting in high RMSE and Hausdorff distances.

4.3. Ablation Study

We conduct an ablation study to evaluate the effectiveness of key components of our proposed method for several aspects: the capability of the TS-ACAP feature and the capability of the spatial temporal modules and the Transformer network. Besides, we show the impact of the resolution of coarse meshes on the detailed mesh synthesis. We evaluate our method qualitatively and quantitatively on different datasets.

Feature Representation Evaluation. The effectiveness of TS-ACAP is verified by comparing per-vertex position errors with 3D vertex coordinates and ACAP, with network layers and parameters adjusted accordingly to optimize performance alternatively. The details of numerical comparison are shown in Table 3. ACAP and TS-ACAP show quantitative improvements over 3D coordinates. In Fig. 8, we exhibit several compared examples of animated skirts using coordinates and TS-ACAP. The results using coordinates show a rough appearance, unnatural deformation and some artifacts, especially in the highlighted regions with details shown in the close-ups. TS-ACAP results are more similar to the ground truth than the ones with coordinates. ACAP has the problem of temporal inconsistency, thus the results are shaking or jumping frequently. Although the use of the Transformer network can somewhat mitigate this issue, such artifacts can appear even with the Transformer. Fig. 9 shows several consecutive frames from a testing sequence in the DISK dataset. TS-ACAP results show more consistent wrinkles than ACAP thanks to temporal constraints.

Spatial Temporal Module Evaluation. Since the key components of our network are the spatial and temporal modules, we evaluate the impact of each module (shown in Fig. 10). It is obvious that without the spatial inference module, the results only exhibit large-scale, smooth deformations without local details. As shown in the

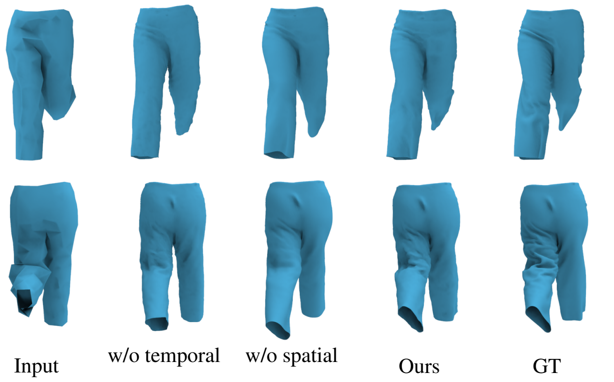


Figure 10: *Ablation study of network architecture."w/o spatial" and "w/o temporal" are our method without the spatial inference module, and without temporal coherence module, respectively.*

Table 4: *Comparison of RMSE between synthesized shapes and ground truth with different networks, i.e. an encoder-decoder (EncDec) dropping out sequential modules and multi-head attention mechanisms, EncDec + RNN, EncDec + LSTM and ours with the DeformTransformer network.*

Dataset	TSHIRT	PANTS	SKIRT	SHEET	DISK
EncDec	0.00909	0.01142	0.00831	0.00739	0.0427
EncDec + RNN	0.0435	0.0357	0.0558	0.0273	0.157
EncDec + LSTM	0.0351	0.0218	0.0451	0.0114	0.102
Ours	**0.00273**	**0.00339**	**0.00391**	**0.00543**	**0.0219**

fourth column, the spatial sub-network improves the quality of local details on the back of the knee of the synthesized pants. On the other hand, the results of our model without a temporal coherence module show inappropriate artifacts and a lack of temporal consistency. By adding the temporal module, the results confirm that the temporal transformer network does properly learn the temporal relationships between coarse and fine TS-ACAP features. The spatial inference module and temporal coherence module jointly improve the quality of high-resolution mesh synthesis.

Transformer Blocks Evaluation. We also evaluate the impact of the Transformer blocks in our pipeline. We compare our method to an encoder-decoder network (EncDec) dropping out sequential modules and multi-head attention mechanisms, EncDec with the recurrent neural network (RNN), and with the long short-term memory (LSTM) module. An example of T-shirts is given in Fig. 11, showing 5 frames in order. The results without any temporal modules show artifacts on the sleeves and neckline since these places have strenuous forces. The models using RNN and LSTM stabilize the sequence via eliminating dynamic and detailed deformation, but all the results keep wrinkles on the chest from the initial state, lacking rich dynamics. Besides, they are not able to generate stable and realistic garment animations that look similar to the ground truth, while our method with the Transformer network apparently improves the temporary stability, producing results close to the ground truth. We also quantitatively evaluate the performance of the Transformer network in our method via per-vertex error. The RMSE of our model is smaller than the others (shown in Table 4).

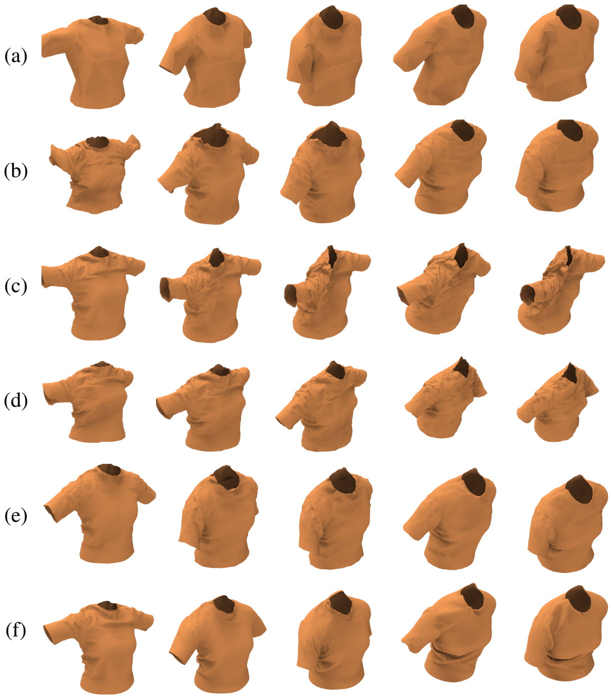


Figure 11: *The evaluation of the Transformer blocks for wrinkle synthesis. From top to bottom, we show (a) input coarse mesh, (b) the results with an encoder-decoder (EncDec) dropping out sequential modules and multi-head attention mechanisms, (c) the results with EncDec + RNN [CGCB14], (d) the results with EncDec + LSTM [HS97], (e) ours, and (f) the ground truth generated by PBS.*

5. Conclusion and Future Work

In this paper, we introduce a novel algorithm for synthesizing robust and realistic cloth animations via deep learning. To achieve this, we propose a geometric deformation representation named TS-ACAP which well embeds the details and ensures temporal consistency. Benefiting from the deformation-based feature, there is no explicit requirement of tracking between coarse and fine meshes in our algorithm. We also use the Transformer network based on attention mechanisms to map the coarse TS-ACAP to fine TS-ACAP, maintaining the stability of our generation. Quantitative and qualitative results reveal that our method can synthesize realistic-looking wrinkles in various datasets, such as draping tablecloth, tight or loose garments dressed on human bodies, etc.

Since our algorithm synthesizes details based on the coarse meshes, the time for coarse simulation is unavoidable. Especially for tight garments like T-shirts and pants, the collision solving phase is time-consuming. In the future, we intend to generate coarse sequences for tight cloth via skinning-based methods in order to reduce the computation for our pipeline. Model compression and acceleration can be achieved via distillation and quantization. Another limitation is that our current network is not able to deal with all kinds of garments with different topologies.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. U21A20515, 62171321, 62162044, 62102414 and 62061136007), the Beijing Municipal Natural Science Foundation for Distinguished Young Scholars (No. JQ21013), the Royal Society Newton Advanced Fellowship (No. NAF\R2\192151) and the Innovation Funding of ICT, CAS (No.E361090).

References

[BCB14] BAHDANAU D., CHO K., BENGIO Y.: Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473* (2014). 3, 5

[BME20] BERTICHE H., MADADI M., ESCALERA S.: PBNS: physically based neural simulator for unsupervised garment pose space deformation. *arXiv preprint arXiv:2012.11310* (2020). 2, 3

[BMF03] BRIDSON R., MARINO S., FEDKIW R.: Simulation of clothing with folds and wrinkles. In *Proc. Symp. Computer Animation* (2003), pp. 28–36. 2

[BMRB16] BOSCAINI D., MASCI J., RODOLÀ E., BRONSTEIN M.: Learning shape correspondence with anisotropic convolutional neural networks. *Advances in Neural Information Processing Systems 29* (2016). 3

[BSF94] BENGIO Y., SIMARD P., FRASCONI P.: Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks 5*, 2 (1994), 157–166. 3

[BW98] BARAFF D., WITKIN A.: Large steps in cloth simulation. In *SIGGRAPH* (1998), pp. 43–54. 2

[BZK09] BOMMES D., ZIMMER H., KOBBELT L.: Mixed-integer quadrangulation. *ACM Trans. Graph. 28*, 3 (July 2009). 5

[CGCB14] CHUNG J., GULCEHRE C., CHO K., BENGIO Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555* (2014). 10

[CVMBB14] 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). 3

[CWW*16] CAO C., WU H., WENG Y., SHAO T., ZHOU K.: Real-time facial animation with image-based dynamic avatars. *ACM Transactions on Graphics 35*, 4 (2016). 3

[CYJ*18] CHEN L., YE J., JIANG L., MA C., CHENG Z., ZHANG X.: Synthesizing cloth wrinkles by CNN-based geometry image superresolution. *Computer Animation and Virtual Worlds 29*, 3-4 (2018), e1810. 1, 2, 3

[CZY21] CHEN L., ZHANG X., YE J.: Multi-feature super-resolution network for cloth wrinkle synthesis. *J. Comput. Sci. Technol. 36* (2021), 478–493. 1, 2, 7, 8, 9

[dASTH10] DE AGUIAR E., SIGAL L., TREUILLE A., HODGINS J. K.: Stable spaces for real-time clothing. *ACM Trans. Graph. 29*, 3 (2010), 106:1–106:9. 1, 2, 3

[DMI*15] DUVENAUD D. K., MACLAURIN D., IPARRAGUIRRE J., BOMBARELL R., HIRZEL T., ASPURU-GUZIK A., ADAMS R. P.: Convolutional networks on graphs for learning molecular fingerprints. In *Advances in neural information processing systems* (2015), pp. 2224–2232. 5

[EPF14] EIGEN D., PUHRSCH C., FERGUS R.: Depth map prediction from a single image using a multi-scale deep network. In *Advances in Neural Information Processing Systems* (2014), pp. 2366–2374. 3

[FLWM18] FEY M., LENSSEN J. E., WEICHERT F., MÜLLER H.: SplineCNN: Fast geometric deep learning with continuous B-spline kernels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 869–877. 3

[FTP16] FRATARCANGELI M., TIBALDO V., PELLACINI F.: Vivace: A practical Gauss-Seidel method for stable soft body dynamics. *ACM Transactions on Graphics (TOG) 35*, 6 (2016), 1–9. 2

[FYK10] FENG W.-W., YU Y., KIM B.-U.: A deformation transformer for real-time cloth animation. *ACM Trans. Graph. 29*, 4 (2010), 108:1–108:10. 1

[GCDZ19] GIRDHAR R., CARREIRA J., DOERSCH C., ZISSERMAN A.: Video action transformer network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2019), pp. 244–253. 3

[GCP*20] GUNDOGDU E., CONSTANTIN V., PARASHAR S., SEIFODDINI A., DANG M., SALZMANN M., FUA P.: GarNet++: Improving fast and accurate static 3D cloth draping by curvature loss. *IEEE Transactions on Pattern Analysis and Machine Intelligence 44*, 1 (2020), 181–195. 2, 3

[GCS*19] GUNDOGDU E., CONSTANTIN V., SEIFODDINI A., DANG M., SALZMANN M., FUA P.: GarNet: A two-stream network for fast and accurate 3D cloth draping. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)* (2019), 8738–8747. 2, 3

[GGAM14] GUPTA S., GIRSHICK R., ARBELÁEZ P., MALIK J.: Learning rich features from RGB-D images for object detection and segmentation. In *European conference on computer vision* (2014), Springer, pp. 345–360. 3

[GGH02] GU X., GORTLER S. J., HOPPE H.: Geometry images. *ACM Trans. on Graph. 21*, 3 (2002), 355–361. 3

[GHDS03] GRINSPUN E., HIRANI A. N., DESBRUN M., SCHRÖDER P.: Discrete shells. In *Proc. Symp. Computer Animation* (2003), pp. 62–67. 2

[GLL*16] GAO L., LAI Y.-K., LIANG D., CHEN S.-Y., XIA S.: Efficient and flexible deformation representation for data-driven surface modeling. *ACM Transactions on Graphics (TOG) 35*, 5 (2016), 1–17. 3

[GLY*19] GAO L., LAI Y.-K., YANG J., LING-XIAO Z., XIA S., KOBBELT L.: Sparse data driven mesh deformation. *IEEE Transactions on Visualization and Computer Graphics* (2019). 2, 5, 6

[GMH13] GRAVES A., MOHAMED A.-R., HINTON G.: Speech recognition with deep recurrent neural networks. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing* (2013), IEEE, pp. 6645–6649. 3

[GRH*12] GUAN P., REISS L., HIRSHBERG D. A., WEISS A., BLACK M. J.: DRAPE: Dressing any person. *ACM Trans. Graph. 31*, 4 (2012), 35:1–35:9. 2, 3

[HHF*19] HANOCKA R., HERTZ A., FISH N., GIRYES R., FLEISHMAN S., COHEN-OR D.: MeshCNN: a network with an edge. *ACM Transactions on Graphics (TOG) 38*, 4 (2019), 1–12. 3

[HS97] HOCHREITER S., SCHMIDHUBER J.: Long short-term memory. *Neural Computation 9*, 8 (1997), 1735–1780. 3, 10

[HVS*09] HARMON D., VOUGA E., SMITH B., TAMSTORF R., GRINSPUN E.: Asynchronous Contact Mechanics. *ACM Trans. Graph. 28*, 3 (2009). 2

[JZD*18] JIANG L., ZHANG J., DENG B., LI H., LIU L.: 3D face reconstruction with geometry details from a single image. *IEEE Transactions on Image Processing 27*, 10 (2018), 4756–4770. 3

[KGBS11] KAVAN L., GERSZEWSKI D., BARGTEIL A. W., SLOAN P.-P.: Physics-inspired upsampling for cloth simulation in games. *ACM Trans. Graph. 30*, 4 (2011), 93:1–93:10. 1, 2, 3

[KKN*13] KIM D., KOH W., NARAIN R., FATAHALIAN K., TREUILLE A., O'BRIEN J. F.: Near-exhaustive precomputation of secondary cloth effects. *ACM Trans. Graph. 32*, 4 (2013), 87:1–87:8. 3

[KV08] KIM T.-Y., VENDROVSKY E.: Drivenshape: A data-driven approach for shape deformation. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2008), pp. 49–55. 3

[LCT18] Lähner Z., Cremers D., Tung T.: Deepwrinkles: Accurate and realistic clothing modeling. In *European Conference on Computer Vision* (2018), Springer, pp. 698–715. 2, 3

[LLL*19] Li N., Liu S., Liu Y., Zhao S., Liu M.: Neural speech synthesis with transformer network. In *Proceedings of the AAAI Conference on Artificial Intelligence* (2019), vol. 33, pp. 6706–6713. 3

[LTT*19] Li Y., Tsiminaki V., Timofte R., Pollefeys M., Gool L. V.: 3D appearance super-resolution with deep learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2019), pp. 9671–9680. 3

[LYO*10] Lee Y., Yoon S.-e., Oh S., Kim D., Choi S.: Multi-resolution cloth simulation. In *Computer Graphics Forum* (2010), vol. 29, Wiley Online Library, pp. 2225–2232. 2

[LZT*19] Liu L., Zheng Y., Tang D., Yuan Y., Fan C., Zhou K.: NeuroSkinning: Automatic skin binding for production characters with deep graph networks. *ACM Transactions on Graphics (TOG) 38*, 4 (2019), 1–12. 3

[MBBV15] Masci J., Boscaini D., Bronstein M., Vandergheynst P.: Geodesic convolutional neural networks on Riemannian manifolds. In *Proceedings of the IEEE International Conference on Computer Vision workshops* (2015), pp. 37–45. 3

[MBM*17] Monti F., Boscaini D., Masci J., Rodola E., Svoboda J., Bronstein M. M.: Geometric deep learning on graphs and manifolds using mixture model CNNs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 5115–5124. 3

[MC10] Müller M., Chentanez N.: Wrinkle meshes. In *Symposium on Computer Animation* (2010), Madrid, Spain, pp. 85–91. 2

[MKB*10] Mikolov T., Karafiát M., Burget L., Černockỳ J., Khudanpur S.: Recurrent neural network based language model. In *Eleventh Annual Conference of the International Speech Communication Association* (2010). 3

[MKB*11] Mikolov T., Kombrink S., Burget L., Černockỳ J., Khudanpur S.: Extensions of recurrent neural network language model. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2011), IEEE, pp. 5528–5531. 3

[NSO12] Narain R., Samii A., O'Brien J. F.: Adaptive anisotropic remeshing for cloth simulation. *ACM Trans. on Graph. 31*, 6 (2012), 147:1–10. 2, 6

[OLL18] Oh Y. J., Lee T. M., Lee I.-K.: Hierarchical cloth simulation using deep neural networks. In *Proceedings of Computer Graphics International 2018*. 2018, pp. 139–146. 3

[OTSK20] Okamoto T., Toda T., Shiga Y., Kawai H.: Transformer-based text-to-speech with weighted forced attention. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2020), IEEE, pp. 6729–6733. 3

[PLPM20] Patel C., Liao Z., Pons-Moll G.: TailorNet: Predicting clothing in 3D as a function of human pose, shape and garment style. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020), pp. 7365–7375. 3

[PMJ*22] Pan X., Mai J., Jiang X., Tang D., Li J., Shao T., Zhou K., Jin X., Manocha D.: Predicting loose-fitting garment deformations using bone-driven motion networks. In *ACM SIGGRAPH 2022 Conference Proceedings* (2022), pp. 1–10. 1, 2, 3

[Pro97] Provot X.: Collision and self-collision handling in cloth model dedicated to design garments. In *EG Workshop on Computer Animation and Simulation* (1997), pp. 177–189. 2

[SACO22] Sharp N., Attaiki S., Crane K., Ovsjanikov M.: DiffusionNet: Discretization agnostic learning on surfaces. *ACM Transactions on Graphics (TOG) 41*, 3 (2022), 1–16. 3

[SBR16] Sinha A., Bai J., Ramani K.: Deep learning 3D shape surfaces using geometry images. In *European Conference on Computer Vision (ECCV)* (2016), pp. 223–240. 3

[SMKL15] Su H., Maji S., Kalogerakis E., Learned-Miller E. G.: Multi-view convolutional neural networks for 3D shape recognition. In *IEEE ICCV* (2015). 3

[SOC19] Santesteban I., Otaduy M. A., Casas D.: Learning-based animation of clothing for virtual try-on. In *Computer Graphics Forum* (2019), vol. 38, Wiley Online Library, pp. 355–366. 1, 2, 3, 4

[SOC22] Santesteban I., Otaduy M. A., Casas D.: SNUG: Self-supervised neural dynamic garments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022), pp. 8140–8150. 2, 3

[SUHR17] Sinha A., Unmesh A., Huang Q., Ramani K.: SurfNet: Generating 3D shape surfaces using deep residual networks. In *CVPR* (2017), pp. 6040–6049. 3

[SVL14] Sutskever I., Vinyals O., Le Q. V.: Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems* (2014), pp. 3104–3112. 3

[TB23] Tiwari L., Bhowmick B.: GarSim: Particle based neural garment simulator. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (2023), pp. 4472–4481. 2

[TGL*22] Tan Q., Gao L., Lai Y., Yang J., Xia S.: Mesh-based autoencoders for localized deformation component analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence 44*, 10 (2022), 6297–6310. 3, 5

[TGLX18] Tan Q., Gao L., Lai Y.-K., Xia S.: Variational autoencoders for deforming 3D mesh models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2018). 3, 5

[TPBF87] Terzopoulos D., Platt J., Barr A., Fleischer K.: Elastically deformable models. In *SIGGRAPH* (1987), pp. 205–214. 2

[VM95] Volino P., Magnenat-Thalmann N.: Collision and self-collision detection: efficient and robust solutions for highly deformable surfaces. In *Computer Animation and Simulation* (1995), pp. 55–65. 2

[VS11] Váša L., Skala V.: A perception correlated comparison method for dynamic meshes. *IEEE Transactions on Visualization and Computer Graphics 17* (02 2011), 220–30. 8

[VSP*17] Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A. N., Kaiser Ł., Polosukhin I.: Attention is all you need. In *Advances in Neural Information Processing Systems* (2017), pp. 5998–6008. 2, 3, 4, 5, 6

[WHRO10] Wang H., Hecht F., Ramamoorthi R., O'Brien J.: Example-based wrinkle synthesis for clothing animation. *ACM Trans. Graph. 29*, 4 (2010), 107:1–107:8. 1, 2

[WLG*17] Wang P.-S., Liu Y., Guo Y.-X., Sun C.-Y., Tong X.: O-CNN: Octree-based convolutional neural networks for 3D shape analysis. *ACM Transactions on Graphics (TOG) 36*, 4 (2017), 1–11. 3

[WPC*20] Wu Z., Pan S., Chen F., Long G., Zhang C., Philip S. Y.: A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems 32*, 1 (2020), 4–24. 3

[WSFM19] Wang T. Y., Shao T., Fu K., Mitra N. J.: Learning an intrinsic garment space for interactive authoring of garment animation. *ACM Transactions on Graphics (TOG) 38*, 6 (2019), 220. 1, 3, 5

[WWW22] Wu B., Wang Z., Wang H.: A GPU-based multilevel additive Schwarz preconditioner for cloth and deformable body simulation. *ACM Transactions on Graphics (TOG) 41*, 4 (2022), 1–14. 2, 6

[WY16] Wang H., Yang Y.: Descent methods for elastic body simulation on the GPU. *ACM Transactions on Graphics (TOG) 35*, 6 (2016), 1–10. 2

[XLZ*20] Xiao Y.-P., Lai Y.-K., Zhang F.-L., Li C., Gao L.: A survey on deep geometry learning: From a representation perspective. *Computational Visual Media 6*, 2 (2020), 113–133. 3

[YSZZ19] Yang L., Shi Z., Zheng Y., Zhou K.: Dynamic hair modeling from monocular videos using deep neural networks. *ACM Transactions on Graphics (TOG) 38*, 6 (2019), 1–12. 3

[ZBO13] Zurdo J. S., Brito J. P., Otaduy M. A.: Animating wrinkles by example on non-skinned cloth. *IEEE Trans. Visual. Comput. Graph. 19*, 1 (2013), 149–158. 1, 2, 3, 7, 8, 9

[ZCM22] Zhang M., Ceylan D., Mitra N. J.: Motion guided deep dynamic 3D garments. *ACM Transactions on Graphics (TOG) 41*, 6 (2022), 1–12. 2, 3

[ZWCM20] Zhang M., Wang T., Ceylan D., Mitra N. J.: Deep detail enhancement for any garment. *arXiv e-prints* (2020), arXiv–2008. 2, 3

[ZWCM21] Zhang M., Wang T. Y., Ceylan D., Mitra N. J.: Dynamic neural garments. *ACM Transactions on Graphics (TOG) 40*, 6 (2021), 1–15. 2, 3

[ZWW*18] Zhang M., Wu P., Wu H., Weng Y., Zheng Y., Zhou K.: Modeling hair from an RGB-D camera. *ACM Transactions on Graphics (TOG) 37*, 6 (2018), 1–10. 3