Predicting Loose-Fitting Garment Deformations Using Bone-Driven Motion Networks

XIAOYU PAN and JIAMING MAI, State Key Lab of CAD&CG, Zhejiang University; ZJU-Tencent Game and Intelligent Graphics Innovation Technology Joint Lab, China

XINWEI JIANG, DONGXUE TANG, and JINGXIANG LI, Tencent NExT Studios, China

TIANJIA SHAO and KUN ZHOU, State Key Lab of CAD&CG, Zhejiang University, China

XIAOGANG JIN*, State Key Lab of CAD&CG, Zhejiang University; ZJU-Tencent Game and Intelligent Graphics Innovation Technology Joint Lab, China

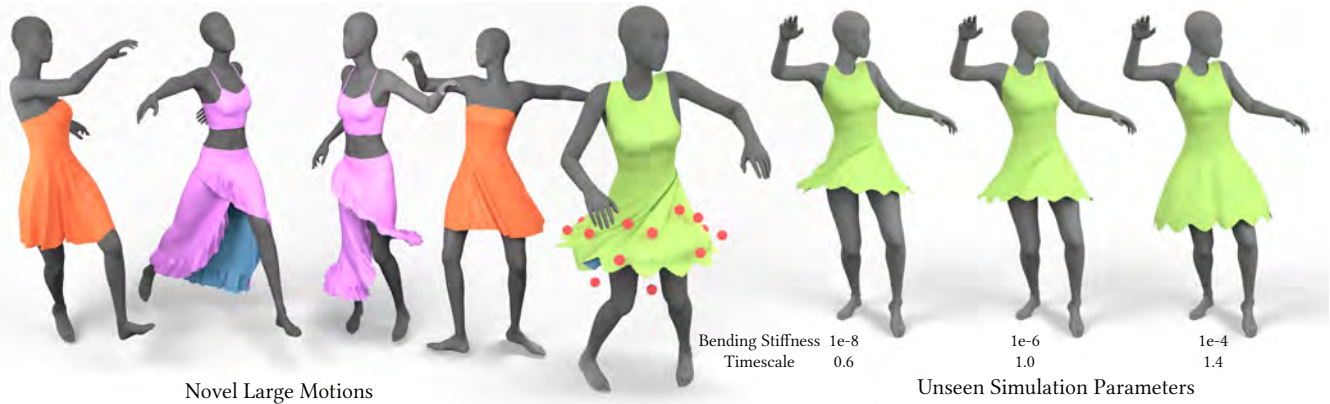DINESH MANOCHA, University of Maryland, USA

Fig. 1. Given the body motion sequence and simulation parameters, our approach predicts the deformations of loose-fitting garments effectively. We transfer the body motion sequence to generate motion of garment's bones (red balls shown in the middle figure on the lower part of the garment). We use these bone-driven motion networks to predict large-scale deformations caused by complex motions, such as flying, swirling and dropping on dresses (left). Our method can also be generalized to simulate dresses under unseen simulation parameters at an interactive rate (right).

We present a learning algorithm that uses bone-driven motion networks to predict the deformation of loose-fitting garment meshes at interactive rates. Given a garment, we generate a simulation database and extract virtual bones from simulated mesh sequences using skin decomposition. At runtime, we separately compute low- and high-frequency deformations in a sequential manner. The low-frequency deformations are predicted by transferring body motions to virtual bones' motions, and the high-frequency deformations are estimated leveraging the global information of virtual bones' motions and local information extracted from low-frequency meshes. In addition, our method can estimate garment deformations caused by variations of the simulation parameters (e.g., fabric's bending stiffness) using an RBF kernel ensembling trained networks for different sets of simulation parameters. Through extensive comparisons, we show that our method outperforms state-of-the-art methods in terms of prediction accuracy of mesh deformations by about 20% in RMSE and 10% in Hausdorff distance and STED. The code and data are available at *https://github.com/non-void/VirtualBones*.

CCS Concepts: • **Computing methodologies** → **Animation**; **Machine learning**.

Additional Key Words and Phrases: cloth animation, deep learning, skinning decomposition

*Corresponding author.

1 INTRODUCTION

Garments, including loose-fitting garments with complex deformations, are critical for dressing humans. For dancing characters, the dynamic motion of their dresses, such as flying, dropping, swirling, or swinging, not only help express emotion [Moody et al. 2010], but also present rich characteristic details. Many interactive applications, such as games and VR, need the capability to dress characters or digital avatars with such loose-fitting garments.

Xiaoyu Pan, Jiaming Mai, Xinwei Jiang, Dongxue Tang, Jingxiang Li,
Tianjia Shao, Kun Zhou, Xiaogang Jin, and Dinesh Manocha

Many data-driven or learning-based methods have been proposed for predicting cloth deformations. However, many of these methods [Bertiche et al. 2020, 2021a,b; Gundogdu et al. 2019; Jin et al. 2020; Lahner et al. 2018; Patel et al. 2020; Pumarola et al. 2019; Santesteban et al. 2019, 2021; Vidaurre et al. 2020] either focus on predicting garment shapes from static poses, or dynamics of tight-fitting garments, with deformations that closely follow the body. GarNet [Gundogdu et al. 2022, 2019] presents a method that retains the geometry features on garments by adopting curvature losses. Almost all deep-learning based methods demand a large amount of data, and PBNS [Bertiche et al. 2021a] introduces a neural simulator trained by unsupervised learning to alleviate this issue. Few methods [Wang et al. 2019a; Zhang et al. 2021] predict the deformations of loose-fitting garments, and even these methods may not be robust for large motions, or may not model high-frequency deformations [Zhang et al. 2021]. Moreover, current learning-based methods do not account for adjustments to simulation parameters (e.g., fabric's bending stiffness and simulator's timescale [SideFX 2021]) in an interactive manner. They either infer simulation parameters *implicitly* given artist edited mesh [Wang et al. 2019a], or adjust them by modifying weights of physical loss terms in the network [Bertiche et al. 2021a].

To simulate loose-fitting garments using deep learning, there are two main issues: complex deformations of loose-fitting garments and the heterogeneity between simulation parameters and body motions. In particular, these deformations may not follow the body shape closely, as is the case for tight-fitting garments. Many prior methods [Patel et al. 2020; Santesteban et al. 2019, 2021], which deform garments according to their associated bodies' skinning functions, may result in artifacts. Many image-based methods, such as geometry images [Pumarola et al. 2019], displacement maps [Jin et al. 2020], and normal maps [Lahner et al. 2018], are not suitable to represent such complex deformations, as they are mostly limited to tight-fitting clothes. Other learning methods use 3D mesh representations [Patel et al. 2020; Santesteban et al. 2019, 2021; Wang et al. 2019a], but are unable to handle complex deformations for loose-fitting garments robustly (see Fig. 8). Moreover, the body motions and simulation parameters play distinctly different roles for such garments. Therefore, it is hard for a single network to learn them jointly. We need better network architectures that can simultaneously learn the two inputs.

Main Results: We present a learning-based method to predict the 3D deformation of loose-fitting garments from body motion sequences and cloth simulation parameters automatically. A key aspect of our approach is extracting the virtual bones of the garment from a pre-computed simulation sequence and using them to drive the motion networks. These virtual bones are used as an intermediate representation and to divide the prediction task into two sub-tasks: low- and high-frequency deformation learning.

As a preprocess, we generate physically-based simulations of loose-fitting garments using Houdini Vellum [SideFX 2021], and decompose them into low-frequency and high-frequency deformations. We then create a set of virtual bones from the simulation sequence by using skinning decomposition [Le and Deng 2012]. At runtime, our model first learns to transfer the body motions to

the virtual bones' transformations to estimate the garment's low-frequency deformations. We treat the virtual bones' motions as global information and also have the local information extracted by graph neural networks [Pan et al. 2021; Wang et al. 2019b] from low-frequency mesh to generate high-frequency deformations. To handle heterogeneity between simulation parameters and body motions, we model deformations caused by simulation parameters and body poses differently. We train multiple networks that take body motions as inputs, and ensemble them using an RBF kernel to model simulation parameters. The novel aspects of our work include:

- The first deep-learning-based method for simulating the complex deformations of loose-fitting garments using virtual bones to efficiently represent low-frequency deformations and infer high-frequency deformations.
- A high-frequency deformation estimator that uses local information of the low-frequency mesh and global information of virtual bones' motions.
- A novel method to handle the heterogeneity between simulation parameters and body motions by modeling them in different types of networks.

We compare and highlight the benefits of our approach with state-of-the-art methods [Chen et al. 2021; Patel et al. 2020; Santesteban et al. 2019; Zhang et al. 2021]. Quantitatively, our method achieves an improvement of 20% in RMSE and 10% in Hausdorff Distance and STED compared to the strongest baseline. Our qualitative results show that our method can effectively estimate the garments' dynamics with complex deformations, and that it outperforms competitive methods in terms of generating plausible deformations.

2 RELATED WORK

Garment Animation and Deformation. Garment animation and simulation is a well-studied problem in computer graphics. Many approaches have been developed and they can be broadly divided into two categories: physically-based simulations (PBS) and data-driven models. A key issue is the balance between computational speed and accuracy. In general, PBS methods generate high-quality results but tend to have high computational costs. In contrast, data-driven models are less computationally costly but it is difficult for them to give any guarantee on accuracy. PBS methods tend to model the real-world physics based on material properties of garments and deform them according to laws of physics using time integration, collision detection, and response computation [Li et al. 2020; Müller 2008; Müller et al. 2015; Müller et al. 2007; Narain et al. 2012; Tang et al. 2016; Wang 2021; Wu et al. 2020].

Data-driven models learn from a set of ground-truth garments, which are obtained from PBS or digital scanning. These methods are complementary to PBS methods, and their main benefit is faster performance, as highlighted in many recent papers [Patel et al. 2020; Santesteban et al. 2019]. Most of recent mesh-based methods [Bertiche et al. 2020, 2021a,b; Gundogdu et al. 2019; Habermann et al. 2021; Jin et al. 2020; Lahner et al. 2018; Patel et al. 2020; Pumarola et al. 2019; Santesteban et al. 2019, 2021; Vidaurre et al. 2020; Yu et al. 2018] model tight-fitting clothes such as T-shirts and pants. [Santesteban et al. 2019] model garment deformations with dynamic wrinkles using a recurrent neural network. To avoid

smoothing out high-frequency details, TailorNet [Patel et al. 2020] disentangles garment deformations under static poses into low-frequency and high-frequency parts and models them separately. GarNet [Gundogdu et al. 2022, 2019] presents a method that retains the geometry features on garments by adopting curvature losses. Almost all deep-learning based methods demand a large amount of data, and PBNS [Bertiche et al. 2021a] introduces a neural simulator trained by unsupervised learning to alleviate this issue. Some techniques [Kim et al. 2013; Wang et al. 2019a; Zhang et al. 2021] have been proposed for loose-fitting garments with complex deformations that do not closely follow the body. [Kim et al. 2013] use data-driven motion graphs to exhaustively model dynamic deformations of loose-fitting garments. In contrast, our learning-based approach can be generalized to unseen motions and simulation parameters. [Wang et al. 2019a] learn an intrinsic garment space that allows shape authoring, which projects the mesh to the latent space and *implicitly* infers the latent representation of simulation parameters using the garment latent and the motion signature. However, their method may not model high-frequency deformations. [Zhang et al. 2021] adopt neural rendering to synthesize details onto images to retain these fine details. Our approach is complimentary to these methods and focuses on representing dynamic loose-fitting garments using virtual bones. Another line of works [Božič et al. 2021; Ma et al. 2021a,b; Saito et al. 2021; Tiwari et al. 2021] uses neural fields to represent clothed human. Their outputs are unified neural fields, where human and cloth are inseparable and only one set of garments is associated with the character. In contrast, our method explicitly and directly models garments' meshes. Moreover, we can dress each character with different sets of garments.

Rigging. Rigging is currently the predominant approach for animating characters, where users animate the mesh by controlling its underlying skeleton. Among these methods, linear blend skinning (LBS) [Lewis et al. 2000] is the most popular algorithm because of its simplicity and efficiency. However, the representation space of LBS is considered to be limited to objects with rigid transformations. To represent deformable objects with rigging, researchers have developed techniques to expand the expressive power of LBS by enriching the space of skinning weights or using more transformations [Kavan et al. 2009; Mohr and Gleicher 2003]. Another line of work tries to augment skeleton-based rigs with physics, using coarse finite element simulations [Capell et al. 2002; McAdams et al. 2011]. The former usage of rigging can be categorized into animation editing [Kavan et al. 2009; Lewis et al. 2000; Mohr and Gleicher 2003; Pan et al. 2021], and joint extraction [Le and Deng 2012]. We also use LBS in our approach.

3 OVERVIEW

Given the body motions $M_b^{\{1,...,t\}}$, i.e., the rotations of joints and translations of the body, and the set of simulation parameters θ, e.g., fabric's bending stiffness, and simulator's timescale [SideFX 2021], our method aims to predict vertex positions of a 3D garment mesh sequence $G^{\{1,...,t\}} \in \mathbb{R}^{t \times V \times 3}$. We assume that the garment mesh has a fixed topology. In this section, we first describe our use of virtual bones, how they drive the motion network and how we handle the variations in simulation parameters.

The virtual bones correspond to a set of bones $\{B\}$ that drive the garment deformations using rigid transformations, each of which controls a part of the garment during animation. Compared with 3D coordinates of vertices, the rigid transformations of virtual bones offer powerful and robust representations of complex deformations. In the animation, the deformations of the garment are driven by the virtual bones' motions using LBS: $G = LBS(M_v)$, where the virtual bones' motions M_v consist of the rotations and translations of each of them $M_v^{\{1,...,t\}} = [R_{v,j}|T_{v,j}]^{\{1,...,t\}}$. The virtual bones are extracted from the smoothed simulation sequence $\hat{G}_{LF}^{\{1,...,t\}}$ using skin decomposition [Le and Deng 2012; Liu et al. 2020], as is shown on the top row of Fig. 2. We highlight two examples of virtual bones and corresponding skin weights in Fig. 3. The virtual bones are different from the skeletons of characters rigged by artists [Kavan et al. 2009; Lewis et al. 2000]. First, artists tend to create hierarchical structures for bones to form skeletons when rigging characters, restricting the bones to rotating in relation to their parents. On the other hand, virtual bones are non-hierarchical and can be rotated and translated freely. Second, the artist-created skeletons have semantic meanings, while virtual bones are extracted from a simulation dataset and may not correspond to semantic representations. In its name, *virtual* refers to lacking semantic meaning, and *bone* means that they are rigid bones extracted by SSDR.

A motion network D is a network driven by virtual bones whose motions M_v are used to model the garment deformations caused by body motions M_b. Using the virtual bone as an intermediate representation, the motion network can effectively model the complex deformations on garments while retaining fine details. In our motion network, we use the low- and high-frequency modules D_{LF}, D_{HF} sequentially to estimate the corresponding low- and high-frequency garment deformations G_{LF}, G_{HF}, where the former refer to the overall garment shape and the latter represent the high-frequency details (fine wrinkles) on the garments. The low-frequency module $D_{LF}(M_b)$ transfers body motion sequence $M_b^{\{1,...,t\}}$ to virtual bones' motions $M_v^{\{1,...,t\}}$ using Gated Recurrent Units (GRU) [Gers et al. 2000]. The low-frequency deformations can be generated by the skinning function of virtual bones: $G_{LF} = LBS(M_v)$. The high-frequency module D_{HF} takes the virtual bones' motions estimated by the previous module and the local information on low-frequency mesh to predict high-frequency deformations $G_{HF} = D_{HF}(M_v, G_{LF})$. The former are processed using a GRU similar to the one in the low-frequency module, and the latter are extracted by a Graph Neural Network (GNN).

Our approach tries to estimate cloth deformations caused by the body motions and changes in the simulation parameters using separate learning methods. In particular, we model the deformations due to body motions using the motion networks; the variations in simulation parameters are modeled using an RBF kernel Ψ [Broomhead and Lowe 1988]. We use multiple motion networks $D_{1,...,K}$ as pivot motion networks, where each corresponds to a set of simulation parameters $\theta_1, ..., \theta_K$. The garment mesh corresponding to the input simulation parameters θ is a weighted summation of resultant meshes of pivot motion networks $G = \sum w_i G_i$, where the weighting coefficients depend on an RBF kernel that quantifies the difference

Xiaoyu Pan, Jiaming Mai, Xinwei Jiang, Dongxue Tang, Jingxiang Li,
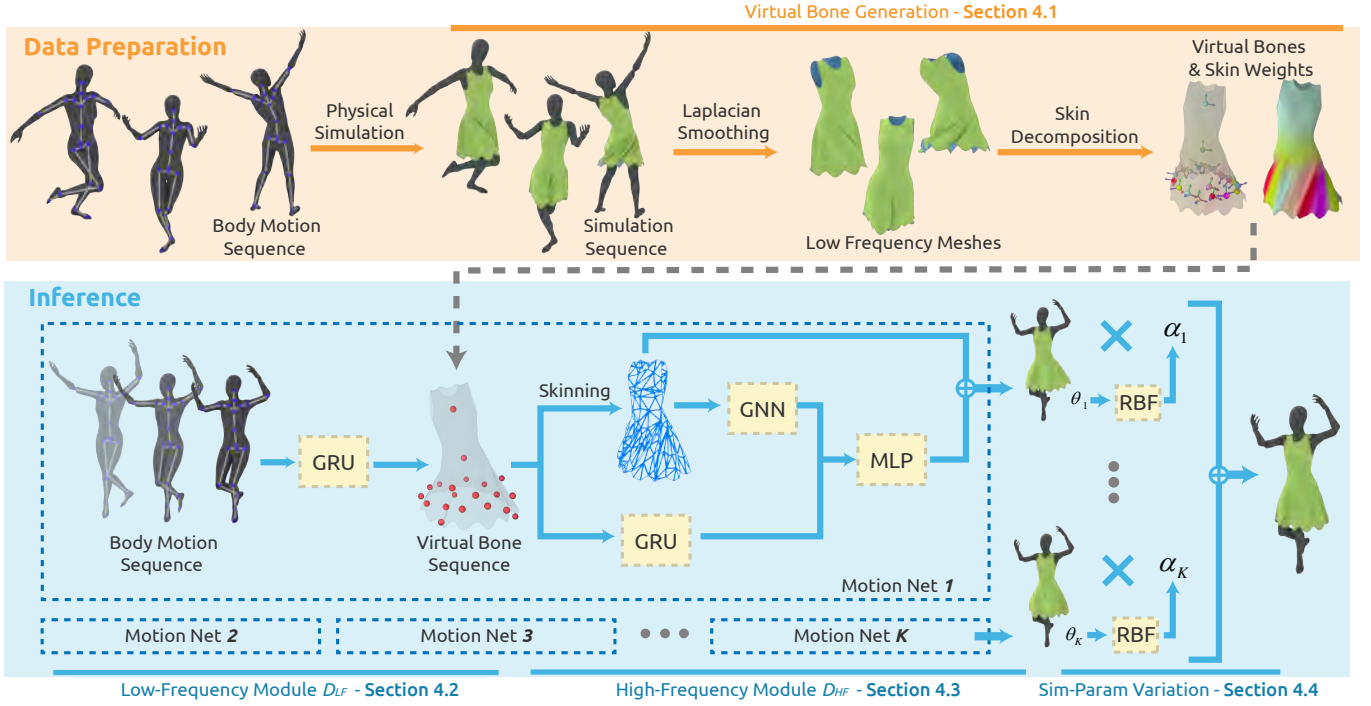Tianjia Shao, Kun Zhou, Xiaogang Jin, and Dinesh Manocha

Fig. 2. The top figure illustrates the generation of virtual bones and corresponding skin weights, which are obtained by performing skin decomposition on smoothed simulation sequences. The bottom figure demonstrates our inference process. The deformations caused by motions are modeled using K pivot motion networks, which correspond to different sets of simulation parameters and take body motions as inputs. A motion network is composed of a low-frequency module and a high-frequency module. The former transfers the body motions to virtual bones' motions to predict the low-frequency deformations, and the latter estimates high-frequency deformations leveraging local information of the mesh and global information of virtual bones' motions. In terms of variations of simulation parameters, we obtain the result by combining the results of pivot motion networks using weights inferred by an RBF kernel.
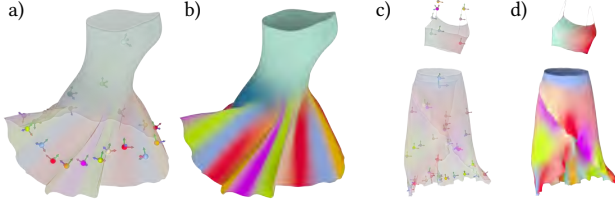

Fig. 3. The virtual bones and corresponding skin weights, where the former are represented by balls and the latter are indicated by the shades on the mesh. (a), (b) 20 virtual bones and skin weights of Dress 3. (c), (d) 40 virtual bones and skin weights of Dress 2.

between their simulation parameters and the input simulation parameters: $w_i = \Psi(\theta_i, \theta)$. These are highlighted in Fig. 2.

4 LEARNING ALGORITHM

4.1 Virtual Bone Generation

In this subsection we describe how to generate the virtual bones given the simulation sequence of garments $\hat{G}^{\{1,2,...,t\}}$. Similar to TailorNet [Patel et al. 2020], our method decomposes the garment meshes to the low-frequency and the high-frequency deformations.

Given a garment mesh $\hat{G}$, we perform Laplacian smoothing to obtain a smoothed mesh, which can be treated as the low-frequency deformation $\hat{G}_{LF}$, and take the residual as the high-frequency deformation $\hat{G}_{HF} = \hat{G} - \hat{G}_{LF}$.

We adopt the state-of-the-art skinning decomposition method Smooth Skinning Decomposition with Rigid Bones (SSDR) [Le and Deng 2012] to transfer the low-frequency mesh sequence into an LBS model with a sequence of virtual bones' motions. Formally, given the mesh sequence $\hat{G}_{LF}^{\{1,...,t\}}$, the SSDR algorithm computes a rest pose mesh P, a skin weights matrix $W_{skin} \in \mathbb{R}^{V \times |B|}$, and a sequence of virtual bones' transformation matrices $\hat{M}_v = [\hat{R}_v | \hat{T}_v] = \{[\hat{R}_{v,j}^t | \hat{T}_{v,j}^t]\}$. The position of the mesh's i-th vertex at the t-th frame v_i^t can be calculated as follows:

$$v_i^t = LBS(\hat{M}_v^t; P, W_{skin}) = \sum_{j=1}^{B} w_{skin,ij}(\hat{R}_{v,j}^t p_i + \hat{T}_{v,j}^t) + \epsilon, \quad (1)$$

where ϵ is the residual of the SSDR algorithm, p_i is the position of the i-th vertex of the rest pose mesh P, $w_{skin,ij}$ indicates the skin weight of v_i regarding to the j-th virtual bone, and $\hat{R}_v, \hat{T}_v$ are the virtual bones' ground truth rotations and translations, respectively.

4.2 Low-Frequency Module

In the first stage of inference, the low-frequency module D_{LF} transfers the body motion sequence to corresponding virtual bones' motions. In contrast to tight clothes, the history-dependent deformations play a central role in the loose-fitting garments' deformations. Therefore, we use recurrent neural networks [Lipton et al. 2015] in the low-frequency module, which sequentially processes body poses by memorizing the current states and updating them according to the new poses. We specifically leverage GRU [Gers et al. 2000] as the building block, which has proven to be successful in modeling dynamic garments [Santesteban et al. 2019]. Since the translation of the body strongly affects the dynamics of loose-fitting garments, for each frame, we concatenate the body translation with the rotation of body joints to construct the input of the low-frequency module $M_b^t = (T_b^t || R_{b_1}^t || ... || R_{b_K}^t)$, where $||$ represents the concatenation operator and K is the number of body joints. The outputs of the module are the rotations and translations of all virtual bones corresponding to this frame: $M_v^t = (T_{v,1}^t || ... || T_{v,|B|}^t) || (R_{v,1}^t || ... || R_{v,|B|}^t)$. The network is trained via the loss containing two terms:

$$
\begin{aligned}
\mathcal{L}_{LF}^t = &||\hat{G}_{LF}^t - LBS(M_v^t; P, W_{skin})||_2 + \\
&\lambda_{Lap}||\Delta(\hat{G}_{LF}^t) - \Delta(LBS(M_v^t; P, W_{skin}))||_2,
\end{aligned}
\tag{2}
$$

where the first term pushes the mesh generated by LBS to be close to the ground truth low-frequency mesh, and the second term encourages the Laplacians between the two meshes to be similar. λ_{Lap} is a predefined weight factor. P and W_{skin} are precomputed by the skin decomposition algorithm and fixed during network training.

4.3 High-Frequency Module

In this subsection, we introduce the high-frequency module D_{HF}, which estimates the high-frequency deformation G_{HF} of the garment. Our approach is based on the strategy used by GNNs [Pan et al. 2021; Wang et al. 2019b], which leverages global features and local features to enhance the size of the receptive field. This module treats the virtual bones' motions as global information, extracts the local information from the low-frequency mesh and leverages them to estimate the high-frequency deformations. We also note that we choose virtual bones' motions rather than body motions as the global information, as the former better represents the deformations of the garments.

To extract the local information of the vertices, we first generate the low-frequency mesh, which is the LBS-deformed mesh driven by virtual bones' motions $G_{LF} = LBS(M_v^t; P, W_{skin})$, and use a stack of graph convolutional layers to extract the information from it. The input and output for every vertex are its position $p_{v_i} \in \mathbb{R}^3$ and its corresponding local feature f_{local,v_i}^t, respectively. Specifically, we adopt EdgeConv operator [Wang et al. 2019b], which has been shown to be efficient for extracting the local features from point clouds [Wang et al. 2019b] and meshes [Pan et al. 2021; Xu et al. 2020]. In the global stream, the overall motion information is processed using a GRU structure similar to the low-frequency module, while changing the input to the motions of virtual bones M_v and the output to the global feature of every vertex f_{global,v_i}^t. The high-frequency deformation of each vertex is obtained by processing the concatenation of local and global features via a shared weight MLP.

The result of the motion network is a simple addition of low- and high-frequency deformations $G = G_{LF} + G_{HF}$.

For the loss function, besides position term, we employ a collision term to avoid body-garment collisions: $\mathcal{L}_{collision}^t = max(-n_{B,k}^t(v_i^t - v_{B,k}^t), 0)$, where $v_{B,k}^t$, $n_{B,k}^t$ are position and normal of the k-th body vertex, which is closest to the i-th estimated garment vertex.

4.4 RBF-Based Simulation Parameter Variation

We describe our method combining different motion networks corresponding to different sets of simulation parameters. To ease the task of modeling garment deformations under different poses and simulation parameters, our method disentangles the deformations caused by the two sources, and models them using different networks.

Specifically, we use a convex combination of the pivot motion networks with different simulation parameters $D_{\theta_{1,...,K}}(M_b)$ to generalize to garment deformations under unseen simulation parameters θ. To build correspondence between pivot networks and simulation parameters, we train each of them using animation sequences simulated with corresponding simulation parameters. The pivot motion networks are chosen from the training set. The process for choosing them is explained below. The calculation of mixture weights is based on Radial Basis Function (RBF) [Broomhead and Lowe 1988], where motion networks with similar simulation parameters have higher weights and others lower. The overall computation corresponds to:

$$
G = \sum_{i=0}^{K} \overline{\Psi}(\theta_i, \theta) D_{\theta_i}(M_b),
\tag{3}
$$

$$
\Psi(\theta_i, \theta) = \exp(-\frac{||g(\theta_i) - g(\theta)||_2^2}{2\sigma^2}),
\tag{4}
$$

where g is an MLP that projects the simulation parameters to the latent space, Ψ is an RBF kernel with bandwidth σ evaluating mixture weights based on the distance between simulation parameters' latents, and $\overline{\Psi}$ is the normalized mixture weights.

To choose the K pivot motion networks that best cover the space of different simulation parameters, we first select the networks having with least one highest or lowest simulation parameter among all networks as pivot motion networks to fit each of the other networks. Next, we iteratively fit non-pivot networks using the pivot motion networks and add the network with the highest RMSE to pivot motion networks until we get K pivot motion networks.

5 RESULTS AND EXPERIMENTS

We use a garment animation dataset consisting of three different types of garments driven by a digital avatar. To animate the character, we collect the dance motions of MikuMikuDance (MMD) videos [Wikipedia 2021] from the internet, which consist of complex body motions. To prepare a dataset with different simulation parameters, we choose three simulation parameters that greatly affect the simulation results of garments: bending stiffness, mass density, and time scale. We sample 10 sets of simulation parameters and split 8/2 sets as training/testing sets. For each set, we generate 40000 frames of simulation data using the Houdini Vellum simulation system [SideFX 2021]. Our networks are implemented using PyTorch [Paszke et al. 2019] and PyTorch Geometric [Fey and Lenssen 2019]. In the low-frequency module D_{LF}, the network is a

Xiaoyu Pan, Jiaming Mai, Xinwei Jiang, Dongxue Tang, Jingxiang Li,
Tianjia Shao, Kun Zhou, Xiaogang Jin, and Dinesh Manocha

single GRU layer followed by a linear layer. In the high-frequency module D_{HF}, we use a similar structure in the GRU and stack three EdgeConv layers for GNN. In our setting, we use 80 virtual bones and 8 pivot motion models. More details about our dataset, network architectures, hyper-parameters and other implementation details are presented in the supplementary material.

5.1 Results and Evaluation

We present the result of our network for unseen motions in Fig. 4. Our method supports complex deformations of garments under large quick motions, e.g., legs wide apart or swirling swiftly. The results are close to the ground truth in the overall shape but may lack some fine details or wrinkles.

We present the performance of our network under different unseen simulation parameters in Fig. 5. Our network is capable of handling different simulation parameters: from left to right, the bending stiffness of the garment lessens, which leads to deeper creases; from top to bottom, the timescale of the simulator increases, causing the garments to be closer to their rest states. Tuning simulation parameters in physical simulations is tedious and consumes a lot of time when generating the results even with proper parameters (1.5 fps for garments used in our benchmarks). Other data-driven methods, which require regeneration of the dataset and training (about 12 hours), are more expensive. Our method supports interactively adjusting simulation parameters and generating new deformations.

An important parameter in our pipeline is the number of garments' virtual bones. We evaluate the effects of this parameter by training low- and high-frequency modules with different numbers of virtual bones (20/40/80/150/250/500) for each garment. The quantitative results are shown in Fig. 6. In terms of the low-frequency module, our method achieves lower RMSE using 40/80 virtual bones for different garments. This may be attributed to the expressiveness of LBS and network capacity. When the number of bones is low, the LBS cannot fully capture the mesh, as the reconstruction error of SSDR for Dress1 is $1e-2$ with 20 virtual bones. When the number is high, the network capacity is limited. As for the high-frequency module, the network achieves lower errors with a higher number of virtual bones, because they carry more information about the mesh deformations. In the rest of our benchmarks, we use 80 virtual bones for each garment for lower overall RMSE.

Fig. 4. Given unseen body motions, our method can estimate garment meshes close to ground truth meshes for different types of garments.
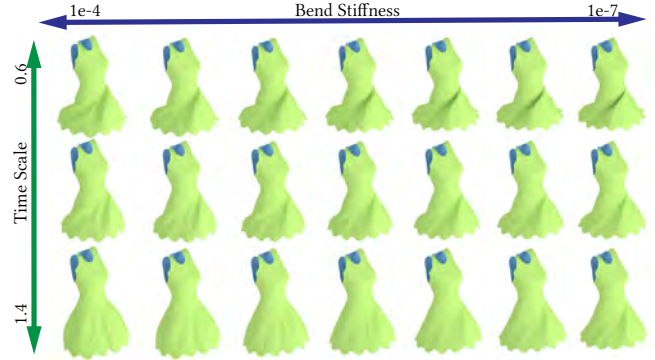

Fig. 5. Estimation of the same frame under different simulation parameters (bending stiffness and timescale). Our method effectively predicts garment meshes corresponding to different simulation parameters in real time.
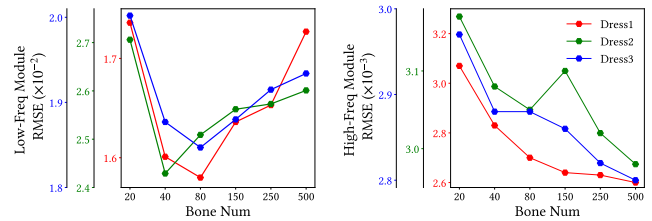

Fig. 6. Comparison of low- and high-frequency modules' performances with different numbers of virtual bones. The low-frequency module estimates meshes with the lowest RMSE with 40/80 virtual bones, and the high-frequency module achieves the lowest RMSE with 500 virtual bones.

Table 1. We compare the characteristics of our method with prior data-driven methods. Some of these methods are limited to tight-fitting garments or static poses and may not explicitly support changing the simulation parameters. Methods considering the dynamics of garments estimate the their deformations using the body motions of previous frames.

Method	Loose-fitting	Dynamic	Sim-Param Variations
TailorNet [Patel et al. 2020] GarNet [Gundogdu et al. 2019] [Jin et al. 2020] [Bertiche et al. 2021b] [Vidaurre et al. 2020] [Corona et al. 2021] [Wu et al. 2021]	✗	✗	✗
Deepwrinkles [Lahner et al. 2018] [Santesteban et al. 2019, 2021]	✗	✔	✗
Intrinsic [Wang et al. 2019a] DNG [Zhang et al. 2021] [Kim et al. 2013]	✔	✔	✗
PBNS [Bertiche et al. 2021a]	✗	✗	✔
N-Cloth [Li et al. 2021]	✔	✗	✗
Ours	✔	✔	✔

5.2 Comparisons with Prior Methods

In Table 1, we list the characteristics of different learning-based methods for modeling garments. Prior methods model tight garments [Bertiche et al. 2020, 2021a,b; Gundogdu et al. 2019; Jin et al. 2020; Lahner et al. 2018; Patel et al. 2020; Pumarola et al. 2019; Santesteban et al. 2019, 2021; Vidaurre et al. 2020] by rigging the garments using body motions, which causes ripping and non-smoothness on loose-fitting garments under large body motions. Moreover, the static methods [Bertiche et al. 2020, 2021a,b; Gundogdu et al. 2019; Jin et al. 2020; Patel et al. 2020; Pumarola et al. 2019; Vidaurre et al. 2020] estimate the garment using only the body pose of the current frame and may not capture all deformations. Other methods control the simulation parameters [Bertiche et al. 2021a] using the weight of loss terms in the network, which are hard to tune and may have no direct correspondence with the simulator. Our method explicitly models simulation parameters corresponding to the simulator using an RBF kernel.

We compare our method with others and perform ablation studies on three components: the low- and high-frequency modules and the overall performance of the motion network. Since prior methods do not focus on variations in simulation parameters, we perform comparisons using the dataset simulated with one set of simulation parameters (bending stiffness 1e-7, density 0.04, and simulator's timescale 1.0). All methods are trained with the same motion set and evaluated using the same unseen motions. We use three metrics to quantitatively evaluate the estimated meshes, namely RMSE (Root Mean Squared Error), Hausdorff distance and STED (Spatio-Temporal Edge Difference) [Vasa and Skala 2011]. The first two calculate distance between estimated and ground truth meshes, while STED measures difference between their dynamics. We demonstrate improvements on each module in Table 2. We have also compared our approach with other methods targeting overall deformations.

We also tested our method on public datasets containing tight garments. We provide the quantitative results in Table 3. The quantitative results show that our method generates comparable results with TailorNet [Patel et al. 2020] and [Santesteban et al. 2019].

We evaluate the choice of splitting the estimation of low- and high-frequency deformations by quantitatively comparing the performance of our network with other networks that have structure similar to our low-frequency module. We present the quantitative results in the left column of Table 2., which shows that separating the prediction slightly enhances the performance. This is due to the fact that our high-frequency module leverages the prediction results of the low-frequency module, as shown in the ablation study of the high-frequency module.

For the low-frequency module, we compare our method with TailorNet [Patel et al. 2020], Dynamic Neural Garments (DNG) [Zhang et al. 2021], and [Santesteban et al. 2019]. For DNG, we only use their method of generating coarse meshes. In the middle of Table 2, we provide a quantitative comparison. We observe that our method outperforms the baseline methods for all metrics, where RMSE is about 20% lower and Hausdorff distance and STED are about 10% lower than the best of them. We present two detailed comparisons in Fig. 8, where our method tends to generate a 3D mesh with the highest quality that is closest to the ground truth. Methods using static

poses such as TailorNet [Patel et al. 2020] fail to predict the loose part of the dress, and our improvement comes from both virtual bones and modelling dynamics. [Santesteban et al. 2019] generate meshes with high distortions on parts between thighs when the avatar's legs are wide open, as the garment is rigged to closely follow the body. Moreover, DNG [Zhang et al. 2021] may generate results with noticeable artifacts on loose parts.
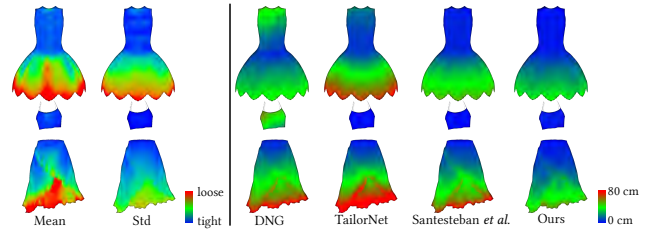

Fig. 7. We highlight loose parts on the garment and per vertex mean errors. On the left, we render the vertices' mean and std distances from the nearest body vertices during animation, where the red color indicates loose parts of the garment. On the right, we compare the per vertex mean error of our method with others, where ours results in lower errors on loose parts.
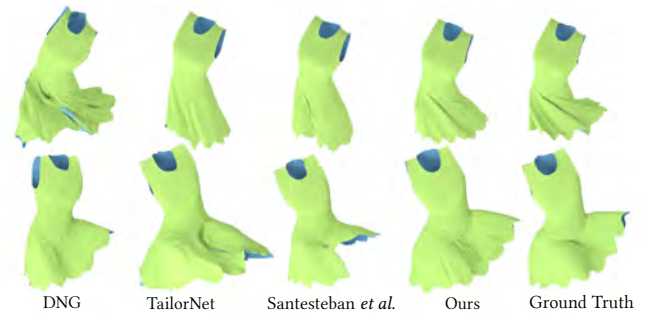

Fig. 8. Qualitative comparisons of the low-frequency module with different methods. Top / bottom row: Dress 1 driven by avatars swiftly spinning and with legs wide apart. Our method tends to generate the 3D deforming mesh with the highest quality that is closest to the ground truth.

For the high-frequency module, we compare our method with [Patel et al. 2020] and [Chen et al. 2021] and perform two ablation studies: the first uses the body motions as inputs, and the second does not use the local information extracted from the low-frequency mesh. We provide a quantitative comparison in the right column of Table 2, where our method outperforms the baseline methods. We present a detailed example in Fig. 9. Since TailorNet only takes static body motions as inputs, it generates non-smooth meshes when the body moves quickly. Our method generates smaller errors than [Chen et al. 2021], since our high-frequency module combines the local information of the low-frequency mesh and virtual bones' motions as input. The two ablation studies validate the choices of using virtual bones' motions and combining local information.

For variations of simulation parameters, we compare our method with two other methods. The first uses the inverse simulation parameter distances as the summation weights instead of RBF, and

Xiaoyu Pan, Jiaming Mai, Xinwei Jiang, Dongxue Tang, Jingxiang Li,
Tianjia Shao, Kun Zhou, Xiaogang Jin, and Dinesh Manocha

Table 2. Comparison with other methods and ablations for our motion model.

| | | Overall | | Low-Frequency Module | | | | High-Frequency Module | | | | |
		Ours (w/o splitting Low-High Freq)	Ours	TailorNet	DNG	[Santesteban et al. 2019]	Ours	TailorNet	[Chen et al. 2021]	Ours (using body motions)	Ours (w/o local stream)	Ours
Dress1	RMSE ↓	17.62	**17.38**	32.94	22.78	20.58	**15.76**	6.20	3.60	3.07	2.70	**2.64**
	Hausdorff ↓	70.94	**69.07**	91.79	75.23	75.62	**65.52**	16.28	16.02	15.89	13.31	**13.21**
	STED ↓	0.0810	**0.0786**	0.0977	0.0955	0.0871	**0.0729**	0.0346	0.0155	0.0151	0.0139	**0.0137**
Dress2	RMSE ↓	27.35	**27.10**	40.48	37.17	30.04	**24.29**	5.37	4.91	3.46	3.11	**3.05**
	Hausdorff ↓	100.65	**98.93**	128.56	116.63	104.15	**92.31**	20.01	22.30	19.94	18.21	**18.01**
	STED ↓	0.0782	**0.0732**	0.0905	0.1046	0.0759	**0.0691**	0.0300	0.0168	0.0151	0.0144	**0.0142**
Dress3	RMSE ↓	22.01	**21.93**	35.70	23.55	23.59	**18.47**	5.16	3.97	3.15	2.90	**2.86**
	Hausdorff ↓	62.31	**62.30**	89.88	71.78	75.40	**58.76**	15.94	16.47	15.49	**13.49**	13.51
	STED ↓	0.0841	**0.0807**	0.1045	0.0967	0.0850	**0.0779**	0.0383	0.0179	0.0167	0.01555	**0.01553**

Table 3. Comparison on the overall performance of the motion network on the dataset of [Santesteban et al. 2019] and TailorNet [Patel et al. 2020]. T-shirt and skirt are from [Santesteban et al. 2019] and pants are from TailorNet [Patel et al. 2020].

		TailorNet	[Santesteban et al. 2019]	Ours
T-shirt	RMSE ↓	**9.90**	10.25	10.52
	Hausdorff ↓	**27.02**	29.56	31.51
	STED ↓	**0.0418**	0.0449	0.0452
Skirt	RMSE ↓	22.95	20.96	**19.91**
	Hausdorff ↓	**76.80**	87.01	83.39
	STED ↓	0.0757	0.0745	**0.0722**
Pants	RMSE ↓	**4.84**	4.91	5.08
	Hausdorff ↓	**14.46**	14.87	18.75
	STED ↓	**0.0127**	0.0129	0.0166

Table 4. Ablation results on methods handling sim parameter variations.

		Linear	Large Net	Ours
Dress1	RMSE ↓	24.82	20.74	**16.69**
	Hausdorff ↓	82.94	78.69	**69.32**
	STED ↓	0.1147	0.1087	**0.0894**
Dress2	RMSE ↓	32.78	30.25	**26.98**
	Hausdorff ↓	113.92	108.37	**99.76**
	STED ↓	0.0910	0.0859	**0.0801**
Dress3	RMSE ↓	22.45	21.40	**19.82**
	Hausdorff ↓	72.43	69.38	**62.58**
	STED ↓	0.0921	0.0923	**0.0823**

Fig. 9. Qualitative comparison of the high-frequency module, the results and corresponding error maps for dress 2 driven by swift body motions. Our method estimates garment meshes with rich details and the lowest RMSE.

We have evaluated it on complex benchmarks and highlighted improved performance over prior learning-based methods. Compared with methods only using 3D coordinates, our virtual bone-driven approach can better predict complex deformations that are frequently observed in loose-fitting garments. Moreover, the virtual bones' motions used in our method can better guide the estimation of high-frequency details. Even for tight garments, our method generates results comparable with prior methods. Our method can also handle the variations of simulation parameters using an RBF-based model.

Our approach has some limitations. Our method uses LBS as skinning method for simplicity. However, its linear property may result in artifacts on areas influenced by multiple bones. Incorporating other skinning methods into our pipeline may be a good direction. The complex deformations can also result in self-collisions in the garment, which need to be prevented [Tan et al. 2021].

the second combines the simulation parameters with motions in the two modules of the motion network. The quantitative results are shown in Table 4, and they highlight the benefits of using an RBF kernel. We also note that the values in Tables 2. and 4. are not comparable, as they use different test sets.

6 CONCLUSION, LIMITATIONS, AND FUTURE WORK

We have presented the first learning-based method to learn the complex deformations of loose-fitting garments via bone deformations.

REFERENCES

Hugo Bertiche, Meysam Madadi, and Sergio Escalera. 2020. CLOTH3D: Clothed 3D Humans. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 344–359.

Hugo Bertiche, Meysam Madadi, and Sergio Escalera. 2021a. PBNS: Physically Based Neural Simulation for Unsupervised Garment Pose Space Deformation. *ACM Transactions on Graphics (TOG)* 40, 6, Article 198 (2021), 14 pages.

Hugo Bertiche, Meysam Madadi, Emilio Tylson, and Sergio Escalera. 2021b. DeePSD: Automatic Deep Skinning and Pose Space Deformation for 3D Garment Animation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 5471–5480.

Aljaž Božič, Pablo Palafox, Michael Zollhöfer, Justus Thies, Angela Dai, and Matthias Nießner. 2021. Neural Deformation Graphs for Globally-Consistent Non-Rigid Reconstruction. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1450–1459.

David Broomhead and David Lowe. 1988. *Radial Basis Functions, Multi-Variable Functional Interpolation and Adaptive Networks*. Technical Report. Royal Signals and Radar Establishment Malvern (United Kingdom).

Steve Capell, Seth Green, Brian Curless, Tom Duchamp, and Zoran Popović. 2002. Interactive Skeleton-Driven Dynamic Deformations. *ACM Transactions on Graphics (TOG)* 21, 3 (2002), 586–593.

Lan Chen, Lin Gao, Jie Yang, Shibiao Xu, Juntao Ye, Xiaopeng Zhang, and Yu-Kun Lai. 2021. Deep Deformation Detail Synthesis for Thin Shell Models. *CoRR* abs/2102.11541 (2021).

Enric Corona, Albert Pumarola, Guillem Alenya, Gerard Pons-Moll, and Francesc Moreno-Noguer. 2021. SMPLicit: Topology-Aware Generative Model for Clothed People. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 11875–11885.

Matthias Fey and Jan E. Lenssen. 2019. Fast Graph Representation Learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*.

Felix A. Gers, Jürgen Schmidhuber, and Fred Cummins. 2000. Learning to Forget: Continual Prediction with LSTM. *Neural Computation* 12, 10 (2000), 2451–2471.

Erhan Gundogdu, Victor Constantin, Shaifali Parashar, Amrollah Seifoddini, Minh Dang, Mathieu Salzmann, and Pascal Fua. 2022. GarNet++: Improving Fast and Accurate Static 3D Cloth Draping by Curvature Loss. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 44, 1 (2022), 181–195.

Erhan Gundogdu, Victor Constantin, Amrollah Seifoddini, Minh Dang, Mathieu Salzmann, and Pascal Fua. 2019. GarNet: A Two-Stream Network for Fast and Accurate 3D Cloth Draping. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 8738–8747.

Marc Habermann, Lingjie Liu, Weipeng Xu, Michael Zollhoefer, Gerard Pons-Moll, and Christian Theobalt. 2021. Real-Time Deep Dynamic Characters. *ACM Transactions on Graphics (TOG)* 40, 4, Article 94 (jul 2021), 16 pages.

Ning Jin, Yilin Zhu, Zhenglin Geng, and Ronald Fedkiw. 2020. A Pixel-Based Framework for Data-Driven Clothing. *Computer Graphics Forum (CGF)* 39, 8 (2020), 135–144.

Ladislav Kavan, Steven Collins, and Carol O'Sullivan. 2009. Automatic Linearization of Nonlinear Skinning. In *Proceedings of the 2009 Symposium on Interactive 3D Graphics and Games (I3D)*. 49–56.

Doyub Kim, Woojong Koh, Rahul Narain, Kayvon Fatahalian, Adrien Treuille, and James F. O'Brien. 2013. Near-Exhaustive Precomputation of Secondary Cloth Effects. *ACM Transactions on Graphics (TOG)* 32, 4, Article 87 (jul 2013), 8 pages.

Zorah Lahner, Daniel Cremers, and Tony Tung. 2018. DeepWrinkles: Accurate and Realistic Clothing Modeling. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 667–684.

Binh Le and Zhigang Deng. 2012. Smooth Skinning Decomposition with Rigid Bones. *ACM Transactions on Graphics (TOG)* 31, 6 (2012).

J. P. Lewis, Matt Cordner, and Nickson Fong. 2000. Pose Space Deformation: A Unified Approach to Shape Interpolation and Skeleton-Driven Deformation. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*. 165–172.

Cheng Li, Min Tang, Ruofeng Tong, Ming Cai, Jieyi Zhao, and Dinesh Manocha. 2020. P-Cloth: Interactive Complex Cloth Simulation on Multi-GPU Systems using Dynamic Matrix Assembly and Pipelined Implicit Integrators. *ACM Transactions on Graphics (TOG)* 39, 6 (2020), 1–15.

Yudi Li, Min Tang, Yun Yang, Zi Huang, Ruofeng Tong, Shuangcai Yang, Yao Li, and Dinesh Manocha. 2021. N-Cloth: Predicting 3D Cloth Deformation with Mesh-Based Networks. *CoRR* abs/2112.06397 (2021).

Zachary C. Lipton, John Berkowitz, and Charles Elkan. 2015. A Critical Review of Recurrent Neural Networks for Sequence Learning. *CoRR* abs/1506.00019 (2015).

Songrun Liu, Jianchao Tan, Zhigang Deng, and Yotam Gingold. 2020. Hyperspectral Inverse Skinning. *Computer Graphics Forum (CGF)* 39, 6 (2020), 49–65.

Qianli Ma, Shunsuke Saito, Jinlong Yang, Siyu Tang, and Michael J. Black. 2021a. SCALE: Modeling Clothed Humans with a Surface Codec of Articulated Local Elements. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 16082–16093.

Qianli Ma, Jinlong Yang, Siyu Tang, and Michael J. Black. 2021b. The Power of Points for Modeling Humans in Clothing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 10974–10984.

Aleka McAdams, Yongning Zhu, Andrew Selle, Mark Empey, Rasmus Tamstorf, Joseph Teran, and Eftychios Sifakis. 2011. Efficient Elasticity for Character Skinning with Contact and Collisions. *ACM Transactions on Graphics (TOG)* 30, 4, Article 37 (2011), 12 pages.

Alex Mohr and Michael Gleicher. 2003. Building Efficient, Accurate Character Skins from Examples. *ACM Transactions on Graphics (TOG)* 22, 3 (2003), 562–568.

Wendy Moody, Peter Kinderman, and Pammi Sinha. 2010. An Exploratory Study: Relationships between Trying on Clothing, Mood, Emotion, Personality and Clothing Preference. *Journal of Fashion Marketing and Management: An International Journal* (2010).

Matthias Müller. 2008. Hierarchical Position Based Dynamics. In *Workshop in Virtual Reality Interactions and Physical Simulation (VRIPHYS)*.

Matthias Müller, Nuttapong Chentanez, Tae-Yong Kim, and Miles Macklin. 2015. Air Meshes for Robust Collision Handling. *ACM Transactions on Graphics (TOG)* 34, 4, Article 133 (jul 2015), 9 pages.

Matthias Müller, Bruno Heidelberger, Marcus Hennix, and John Ratcliff. 2007. Position Based Dynamics. *Journal of Visual Communication and Image Representation* 18, 2 (2007), 109–118.

Rahul Narain, Armin Samii, and James F. O'Brien. 2012. Adaptive Anisotropic Remeshing for Cloth Simulation. *ACM Transactions on Graphics (TOG)* 31, 6, Article 152 (2012), 10 pages.

Xiaoyu Pan, Jiancong Huang, Jiaming Mai, He Wang, Honglin Li, Tongkui Su, Wenjun Wang, and Xiaogang Jin. 2021. HeterSkinNet: A Heterogeneous Network for Skin Weights Prediction. *Proceedings of the ACM in Computer Graphics and Interactive Techniques (PACMCGIT)* 4, 1, Article 10 (2021), 19 pages.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 8024–8035. http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

Chaitanya Patel, Zhouyingcheng Liao, and Gerard Pons-Moll. 2020. 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 (CVPR)*. 7363–7373.

Albert Pumarola, Jordi Sanchez-Riera, Gary Choi, Alberto Sanfeliu, and Francesc Moreno-Noguer. 2019. 3DPeople: Modeling the Geometry of Dressed Humans. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2242–2251.

Shunsuke Saito, Jinlong Yang, Qianli Ma, and Michael J Black. 2021. SCANimate: Weakly Supervised Learning of Skinned Clothed Avatar Networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2886–2897.

Igor Santesteban, Miguel A. Otaduy, and Dan Casas. 2019. Learning-Based Animation of Clothing for Virtual Try-On. *Computer Graphics Forum (CGF)* 38, 2 (2019), 355–366.

Igor Santesteban, Nils Thuerey, Miguel A Otaduy, and Dan Casas. 2021. Self-Supervised Collision Handling via Generative 3D Garment Models for Virtual Try-On. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 11763–11773.

SideFX. 2021. *Houdini Vellum*.

Qingyang Tan, Zherong Pan, and Dinesh Manocha. 2021. LCollision: Fast Generation of Collision-Free Human Poses using Learned Non-Penetration Constraints. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 3913–3921.

Min Tang, Huamin Wang, Le Tang, Ruofeng Tong, and Dinesh Manocha. 2016. CAMA: Contact-Aware Matrix Assembly with Unified Collision Handling for GPU-based Cloth Simulation. *Computer Graphics Forum (CGF)* 35, 2 (2016), 511–521.

Garvita Tiwari, Nikolaos Sarafianos, Tony Tung, and Gerard Pons-Moll. 2021. Neural-GIF: Neural Generalized Implicit Functions for Animating People in Clothing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 11708–11718.

Libor Vasa and Vaclav Skala. 2011. A Perception Correlated Comparison Method for Dynamic Meshes. *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 17, 2 (2011), 220–230.

Raquel Vidaurre, Igor Santesteban, Elena Garces, and Dan Casas. 2020. Fully Convolutional Graph Neural Networks for Parametric Virtual Try-On. *Computer Graphics Forum (CGF)* 39, 8 (2020), 145–156.

Huamin Wang. 2021. GPU-Based Simulation of Cloth Wrinkles at Submillimeter Levels. *ACM Transactions on Graphics (TOG)* 40, 4, Article 169 (2021), 14 pages.

Tuanfeng Y. Wang, Tianjia Shao, Kai Fu, and Niloy J. Mitra. 2019a. Learning an Intrinsic Garment Space for Interactive Authoring of Garment Animation. *ACM Transactions on Graphics (TOG)* 38, 6, Article 220 (2019), 12 pages.

Xiaoyu Pan, Jiaming Mai, Xinwei Jiang, Dongxue Tang, Jingxiang Li,
Tianjia Shao, Kun Zhou, Xiaogang Jin, and Dinesh Manocha

Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. 2019b. Dynamic Graph CNN for Learning on Point Clouds. *ACM Transactions on Graphics (TOG)* 38, 5, Article 146 (2019), 12 pages.

Wikipedia. 2021. MikuMikuDance, Wikipedia. https://en.wikipedia.org/wiki/MikuMikuDance [Online; accessed 2021-12-13].

Longhua Wu, Botao Wu, Yin Yang, and Huamin Wang. 2020. A Safe and Fast Repulsion Method for GPU-Based Cloth Self Collisions. *ACM Transactions on Graphics (TOG)* 40, 1, Article 5 (dec 2020), 18 pages.

Nannan Wu, Qianwen Chao, Yanzhen Chen, Weiwei Xu, Chen Liu, Dinesh Manocha, Wenxin Sun, Yi Han, Xinran Yao, and Xiaogang Jin. 2021. AgentDress: Realtime Clothing Synthesis for Virtual Agents using Plausible Deformations. *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 27, 11 (2021), 4107–4118.

Zhan Xu, Yang Zhou, Evangelos Kalogerakis, Chris Landreth, and Karan Singh. 2020. RigNet: Neural Rigging for Articulated Characters. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 58.

Tao Yu, Zerong Zheng, Kaiwen Guo, Jianhui Zhao, Qionghai Dai, Hao Li, Gerard Pons-Moll, and Yebin Liu. 2018. DoubleFusion: Real-Time Capture of Human Performances with Inner Body Shapes from a Single Depth Sensor. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 7287–7296.

Meng Zhang, Tuanfeng Y. Wang, Duygu Ceylan, and Niloy J. Mitra. 2021. Dynamic Neural Garments. *ACM Transactions on Graphics (TOG)* 40, 6, Article 235 (2021), 15 pages.