

# Frequency-Divided Learning of Fine-Grained Clothing Behavior via Flexible Dynamic Graphs

Tianxing Li, Rui Shi, Takashi Kanai, Qing Zhu

**Abstract**—Despite significant advancements in neural simulation techniques for clothing animation, these methods struggle to capture the dynamic details of garments during movement. This limitation restricts their applicability in scenarios where high-quality garment deformation is essential. To address this challenge, we introduce a novel graph learning-based approach to enhance deformation realism through designed mechanisms for mesh information propagation and external optimization strategies during model training. First, we address the issue of over-smoothing common in conventional graph processing techniques by introducing a flexible message-passing method. This approach effectively manages node interactions within the mesh, thereby improving the expressiveness of the model. Furthermore, acknowledging that uniform model supervision typically neglects high-frequency details during optimization, we analyze the spectral properties of clothing meshes. Based on this analysis, we introduce a frequency-division constraint aligned with the characteristics of different frequency bands, which aids in precisely controlling the generation of details. Our model further integrates self-collision and other physics-aware losses, enabling the learning of generalized and fine-grained dynamic deformations. Extensive evaluations and comparisons demonstrate the effectiveness of our approach, showing notable improvements over existing state-of-the-art solutions.

**Index Terms**—Clothing deformation, frequency division, graph message-passing, physical-informed learning.

## I. INTRODUCTION

LEARNING 3D clothing behavior for digital characters is a crucial technique across various industries, including video games, films, e-commerce, and the metaverse. Efficient simulation of high-quality clothing has always been a key goal for researchers in computer graphics, as it significantly increase interactivity and immersion. Traditional skinning based methods (*e.g.*, linear blend skinning [1], dual quaternion skinning [2]) and pose space deformation [3] simplify garment deformation by interpolating vertex positions according to skeletal movements. While these approaches are straightforward and quick, they always struggle to produce realistic dynamics and details. In contrast, physics-based simulation [4]–[6] leverages fundamental physics principles to generate plausible results. However, the high computational demands

Manuscript received xx xx 2024; revised xx xx xx; accepted xx xx xx.

Tianxing Li and Qing Zhu are with the College of Computer Science, Beijing University of Technology, Beijing 100124, China (e-mail: litianxing@bjut.edu.cn; ccgsqz@bjut.edu.cn)

Rui Shi is with the School of Information Science and Technology, Beijing University of Technology, Beijing 100124, China (e-mail: ruishi@bjut.edu.cn)

Takashi Kanai is with the Department of General Systems Studies, the University of Tokyo, Tokyo 153-8902, Japan (e-mail: kanait@acm.org)

This paper has supplementary downloadable material available at <http://ieeexplore.ieee.org/>, provided by the author. The material includes more implementation details and experimental results.

of these simulations often restrict their use in real-time applications.

Owing to advancements in artificial intelligence technology, learning-based approaches [7], [8] are demonstrating potential in enhancing both efficiency and quality in different types of real-world applications [9], [10]. State-of-the-art methods covering static [11] to dynamic deformations [12]–[14] and ranging from supervised [15]–[17] to unsupervised strategies [18]–[20] have substantially progressed by establishing critical connections between influencing factors and target behaviors through carefully designed models, thereby facilitating automatic deformation approximation. At the same time, while some methods [21], [22] aim to provide unified frameworks for garment animation, they often overlook the inherent frequency-domain characteristics or fail to directly control them, which are crucial for estimating detailed deformations. Consequently, they struggle to consistently achieve high-fidelity garment representations, particularly in capturing fine-grained deformations and intricate details across diverse motion scenarios.

This situation raises a critical question: *How can models for approximating clothing deformation reliably produce fine-grained details?* The garment mesh can be represented as a graph, where nodes correspond to mesh vertices and edges define the connections between them. By treating the attributes (features) of the mesh as signals defined over the graph, their spectral properties can be analyzed through graph frequency decomposition. Similar to signals commonly observed in physical systems, low-frequency components typically exhibit significantly higher energy compared to their high-frequency parts. Therefore, when a generic neural network processes this signal, the high-frequency details can easily be overwhelmed by the low-frequency contents, resulting in a biased representation of different frequency bands. This issue is particularly pronounced in most graph neural networks (GNNs) [23]–[25], which tend to induce over-smoothing because of their inherent low-pass filtering effect during neighborhood aggregation. Studies in [26] and [22] have also highlighted this issue and proposed solutions. The former abandons the use of GNNs, opting instead for multi-perceptron (MLP) layers, which compromises the generalization capabilities of the model. Conversely, the latter employs spectral normalization to control the Lipschitz constant of the graph-based model, indirectly mitigating bias but not achieving direct control of the finer details.

In this work, we introduce a novel, learning-based, generalized framework designed to enhance the estimation of detailed clothing deformations in animation. The key to our

success lies in proposing strategies that address the often-overlooked high-frequency details, focusing on both the *intrinsic* properties of the neural network (*i.e.*, the way graph information is processed within networks) and the *external* constraints during training (*i.e.*, objectives imposed externally on the network’s output). Intrinsically, we benefit from the abilities of graph attention networks to deal with irregular mesh information, while also recognizing the issue of over-smoothing in message-passing due to the feature aggregation way. To bridge this gap, we develop a graph attention mechanism with the flexibility that dynamically adjusts feature homogenization or heterogenization during propagation, thereby enhancing the expressiveness of the model. Externally, we find incorporating a supervised optimization process can simplify the complexity of parameter tuning inherent in purely unsupervised learning, because supervised losses could provide clear, goal-oriented guidelines during training. However, uniform error metrics (*e.g.*, directly constraining vertex position errors) tend to under-penalize high-frequency imperfections since larger errors in low-frequency components overshadow them, prompting the network to prioritize these discrepancies. This scheme might yield numerically favorable results, but it does not guarantee the visual completeness of details. To address it, we propose a frequency division strategy for meshes that decomposes clothing deformations into distinct frequency components. By analyzing these components, we design the frequency-division losses that directly control and optimize each frequency band.

To sum up, our main contributions are threefold:

- **Flexible graph information propagation mechanism.** We introduce a general method for feature processing in arbitrary clothing meshes. Unlike traditional GNNs that function as low-pass filters, our approach overcomes feature smoothing limitations by enhancing feature variability during message-passing, thereby improving model performance.
- **Direct control over the garment detail generation.** From a frequency domain perspective, we introduce a frequency-division constraint. This enables active control over different deformation frequency components, ensuring that personalized details are effectively captured and not overlooked.
- **Generalized physics-aware deformation method.** In addition to applying frequency-division loss for close alignment with the ground truth, we incorporate physics-inspired unsupervised losses like self-collision to enhance model realism. Our approach is applicable to a diverse range of garment topologies and supports personalization needs.

We demonstrate the superiority of our proposal through comprehensive validation and comparison with the state-of-the-art methods. Furthermore, our analysis and solutions can offer useful insights for 3D generation tasks, extending their applicability beyond clothing simulation to broader contexts.

## II. RELATED WORK

**Physics-based simulation** models clothing behavior by applying fundamental laws of physics, including gravity, elas-

ticity, friction, based on the intrinsic properties of materials involved. To enhance the precision of these simulations, advanced techniques such as time integration [27], [28], collision handling [29], [30], mechanical modeling [31], [32], differentiable simulation [33], [34], and yarn-level simulation [35], [36] have been proposed. Although these methods yield impressive results, they also present significant computational challenges, limiting their application in real-time scenarios. Recently, research has focused on leveraging GPU parallelization [37], [38], refining coarse meshes with additional details [39], [40], and employing projective dynamics [41], [42] to enhance simulation efficiency. However, these advancements still require considerable expertise and time for parameter tuning. Consequently, despite the progress made, a number of technical requirements tend to make these methods unsuitable for large-scale clothing simulation where rapid and scalable solutions are necessary.

**Learning-based methods** provide efficient alternatives to physics-based simulations, addressing challenges in automation and customizability. Early works, such as pose space deformation [3], laid the groundwork for learning parametric deformations, which have since expanded to body shapes [11], poses [43], [44], fit [16], clothing design [26], [45], and sizes [46]. Proxy-based methods [47] improve efficiency by generating low-resolution proxy meshes with optimized skinning weights to drive high-resolution cloth animations, while super-resolution approaches [48] enhance wrinkle details and maintain spatial-temporal coherence with interleaved simulation-correction frameworks. Other studies refine garment simulation accuracy through realistic wrinkle modeling [49], pixel-based data-driven frameworks [50], and texture-based geometric recovery [51], alongside physics-aware learning for physical plausibility [52], [53]. Recent advances include self-supervised garment generation and draping [54], shape and deformation priors for recovery [55], and differentiable simulation for garment optimization [56].

Several studies leverage GNNs for improved generalization in modeling articulated characters by generating skinning weights and blend shapes [57]–[60]. To capture high-quality cloth animation details, methods drape garments over SMPL [61], with GarNet [62] introducing a two-stream PointNet-based architecture for modeling body-cloth interactions. Mesh-based frameworks [13], [15] and graph U-Net [11] focus on deformation approximation but face challenges with generalization to new meshes or poses. SwinGar [22] advances dynamic deformation using temporal mesh graphs but struggles with achieving detailed physical realism.

In another promising line, researchers have reframed solving equations of motion in physics-based simulations as an optimization problem, employing self-supervised methods to eliminate the need for ground truth data [18]–[20], [63]. While effective, these approaches struggle with generalizing to diverse clothing meshes. HOOD [21] introduces a hierarchical graph-based solution to address this but faces challenges with runtime efficiency due to complex feature dependencies. Despite advancements, key issues remain, including tuning physical parameters in unsupervised optimization and generating dynamic details across varied contexts.

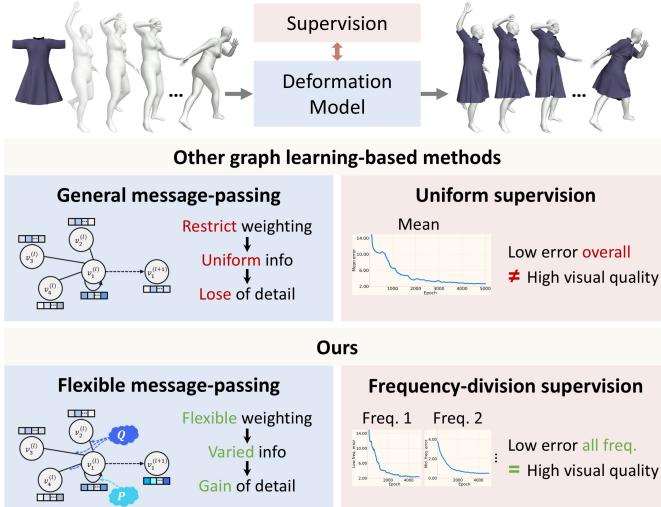


Fig. 1: Motivation for our study. The general message-passing and uniform supervision in existing graph learning-based methods limit the generation of high-quality details. Our approach proposes flexible message-passing and frequency-division supervision to overcome these limitations.

Motivated by existing works, this paper introduces a novel learning-based approach to approximate fine-grained clothing behavior (as depicted in Fig. 1). Unlike common methods [13], [15], [22], [64] that use GNNs, our approach introduces a flexible message-passing mechanism within the network and incorporates frequency-division supervision as external constraints to guide the training process, effectively mitigating the issue of detail loss. These will be further elaborated in the methodology section.

**Spectral analysis** is widely used in machine learning to capture high-frequency details and enhance performance across domains. Fourier features [65] help model high-frequency patterns in tasks like rendering and super-resolution [66], while frequency decomposition improves reconstruction by focusing on high-frequency residuals [67]. Periodic activations in implicit neural representations [68] advance 3D reconstruction by capturing fine geometric structures, and frequency-aware techniques sharpen depth estimation through high-frequency supervision [69]. In this paper, we propose a novel frequency-division constraint for the clothing domain, the first to apply frequency decomposition for independently supervising frequency components in garment deformation, enabling precise learning of fine details.

### III. METHODOLOGY

We consider a 3D garment mesh template with vertices  $\bar{T} \in \mathbb{R}^{N \times 3}$  and faces in its rest pose, where  $N$  is the vertex count. Additionally, we use a SMPL body in the rest pose, consisting of  $N_b$  vertices, which is parameterized by shape parameters  $\beta \in \mathbb{R}^{|\beta|}$  and pose parameters  $\theta \in \mathbb{R}^{|\theta|}$ . A motion sequence  $\phi^t = \{\theta^{(t-m)}, \dots, \theta^t\} \in \mathbb{R}^{|\theta| \times (m+1)}$ , spanning from time  $t - m$  to  $t$ , is also provided. Our goal is to develop a learning framework that links these elements to the resulting clothing behavior. The pipeline is schematically summarized

in Fig. 2. Initially, we process each source of information separately. Body attributes, including shape parameters  $\beta \in \mathbb{R}^{|\beta|}$  and vertex-to-joint distances  $Dis \in \mathbb{R}^{N_b \times S}$  (where  $S$  denotes the number of body joints), are encoded using MLP layers, while motion features  $\phi^t$  are processed by gated recurrent unit (GRU) layers. For the garment mesh, we model the complex geometric changes in its deformed state by regarding it as a graph structure composed of vertices and edges. In this graph, mesh vertices are treated as nodes, and the edges correspond to the original mesh connectivity between pairs of vertices. Specifically, the graph is defined as  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , where  $\mathcal{V} = \{v_1, \dots, v_N\}$  represents the feature set of all  $N$  vertices, and  $\mathcal{E}$  denotes the edges derived from the mesh, which remain unchanged during the deformation process. Each node is assigned features  $v_i$  such as vertex positions, normals, and distances to body joints, all of which are then processed using the designed FlexiGAT composed of graph attention layers (detailed in Sec. III-A).

Subsequently, the multiple encoded features are fused into a comprehensive graph representation that efficiently computes the dynamic skinning weights  $\mathcal{W}^t$  for the garments. The blend shapes  $B^t$ , crucial for capturing temporal changes, are produced through additional graph attention layers. Following the application of the skinning function, the dynamic garment mesh  $\tilde{\mathcal{M}}^t$  can be modelled as:

$$\tilde{\mathcal{M}}^t(\bar{T}, \beta, \phi^t) = W(T^t(\bar{T}, \beta, \phi^t), J, \mathcal{W}^t(\bar{T}, \beta, \phi^t)), \quad (1)$$

$$T^t(\bar{T}, \beta, \phi^t) = \bar{T} + B^t(\bar{T}, \beta, \phi^t), \quad (2)$$

where the blend shapes  $B^t$  are added to the garment mesh template vertices  $\bar{T}$  to form  $T^t$ , which is then deformed using the skinning function  $W(\cdot)$ , incorporating dynamic skinning weights  $\mathcal{W}^t$  and joint transformations from the joint location  $J$ . Building upon the generated  $\tilde{\mathcal{M}}^t$ , we aim to enrich the garment deformations with nonlinear details. To achieve this, we further construct the mesh as a graph, embedding vertex velocities, normals, and vertex-joints distances, and then process it through graph attention layers of FlexiGAT. This procedure outputs a refinement  $\Delta^t$ , allowing the final deformation of the garment to be expressed as:  $\mathcal{M}^t = \tilde{\mathcal{M}}^t + \Delta^t$ .

In terms of optimization, to facilitate effective learning, we analyze clothing behavior through a spectral perspective as discussed in Sec. III-B. Based on this analysis, we formulate a frequency-division loss function (Sec. III-C). This function is employed alongside self-collision and other physically-aware unsupervised losses to enhance model performance.

#### A. Flexible Graph Attention Mechanism

The transformation of clothing meshes in animation demonstrates a complex process where information dynamically flows through the edges between nodes within the mesh. To effectively model this, the mesh can be conceptualized as a graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  with  $N$  vertices, comprising a set of vertex features  $\mathcal{V}$  and edges  $\mathcal{E}$ . In updating this information, most GNNs follow the message-passing paradigm, where the representation of each node is computed by recursively aggregating and transforming the representations of its neighboring nodes. Unlike other GNN architectures that assign equal importance

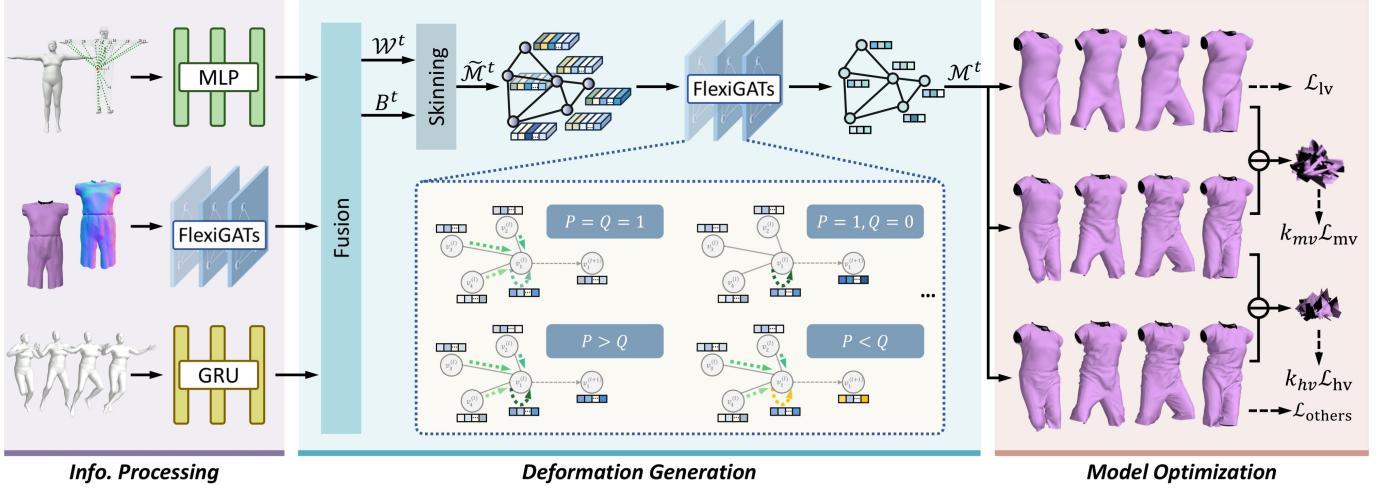


Fig. 2: Overview of our method. We start by extracting features from the body, garment, and motion, which are then integrated into a comprehensive graph. This graph is then processed to generate skinning weights and blend shapes, which are used to create the dynamic garment mesh  $\tilde{M}^t$ . Additional refinement process is also applied to enhance its details, leading to the final garment deformation  $M^t$ . Central to our approach is the message-passing mechanism of FlexiGAT and the model optimization strategy, allowing multi-fold feature aggregation (see different colored arrows in the center) and employing frequency-division losses for separate frequency band (*i.e.*, low, middle, and high) supervision as shown in the right part.

to all neighbors, the integration of attention mechanisms (*e.g.*, graph attention network, GAT [24]) has proven to enhance generalization and adaptability. Specifically, the raw attention coefficient  $e_{ij}^{(l)}$  at the  $l$ -th layer between node  $i$  and neighbor  $j$  is given by:

$$e_{ij}^{(l)} = \text{LeakyReLU} \left( a^{(l)\top} [W^{(l)} v_i^{(l)} \| W^{(l)} v_j^{(l)}] \right), \quad (3)$$

where  $v_i^{(l)}$  and  $v_j^{(l)}$  are the feature vectors of the nodes at layer  $l$ , residing in  $\mathbb{R}^d$ .  $W^{(l)} \in \mathbb{R}^{d' \times d}$  is a learned transformation matrix,  $a^{(l)} \in \mathbb{R}^{2d'}$  is a learned parameter, and  $\|$  represents concatenation. To make the coefficients comparable across different nodes, they are then normalized across all neighbors  $j \in \mathcal{N}_i$ :

$$\alpha_{ij}^{(l)} = \text{softmax}_j(e_{ij}) = \frac{\exp(e_{ij})}{\sum_{j' \in \mathcal{N}_i} \exp(e_{ij})}. \quad (4)$$

The matrix  $A^{(l)}$  contains the entry  $\alpha_{ij}^{(l)}$  in the  $i$ -th row and  $j$ -th column, representing the propagation operator in the message-passing. Then, for a general GAT, the processing of a layer can be expressed as:

$$V^{(l+1)} = \sigma \left( A^{(l)} V^{(l)} K^{(l)} \right), \quad (5)$$

where  $\sigma(\cdot)$  is a nonlinearity function,  $K^{(l)}$  can be regarded as a  $1 \times 1$  convolution. Here,  $V^{(l)} = [v_1^{(l)}, \dots, v_N^{(l)}]$  denotes the graph feature matrix at the  $l$ -th layer, and the output  $V^{(l+1)}$  is the updated graph feature matrix at the  $(l+1)$ -th layer. However, since the propagation aggregator is non-negative, a limitation arises: as information is propagated through multiple layers, node features become increasingly indistinguishable, leading to the frequently criticized issue of over-smoothing. To tackle

this, we propose a straightforward yet impactful modification to the standard propagation operation:

$$\begin{aligned} V^{(l+1)} &= \sigma \left( \left( P^{(l)} I + Q^{(l)} (A^{(l)} - I) \right) V^{(l)} K^{(l)} \right) \\ &= \sigma \left( \left( (P^{(l)} - Q^{(l)}) I + Q^{(l)} A^{(l)} \right) V^{(l)} K^{(l)} \right), \end{aligned} \quad (6)$$

where  $P^{(l)}$  and  $Q^{(l)}$  are the scalars learned at layer  $l$  that adjust the influence of self and neighbors. This flexibility allows the graph attention layer to function in various modes, which we have termed FlexiGAT. For instance, when  $P^{(l)} = Q^{(l)} = 1$ , Eq. (6) reverts to the standard GAT as in Eq. (5). When  $P^{(l)} = 1, Q^{(l)} = 0$ , the update focuses exclusively on self-features, isolating the node from its neighbors. If  $P^{(l)} > Q^{(l)}$ , the update process balances the node's own features with those of its neighbors, functioning as a blending mechanism. Conversely, when  $P^{(l)} < Q^{(l)}$ , the resulting operator has negative diagonal entries and positive off-diagonal entries, creating a sharpening effect that enhances feature distinctions.

Our FlexiGAT extends beyond the capabilities of the original GAT, which, despite its multi-attention mechanism that theoretically creates multiple propagation operators, is limited by its non-negative propagation weights. This inherent characteristic leads to over-smoothing, a significant challenge where node features become increasingly homogeneous across the network as more layers are applied. In contrast, by integrating flexibility into the propagation operators and allowing for both negative and positive weights, our method supports diverse scenarios and effectively counters the over-smoothing problem, maintaining sharper distinctions and more dynamic feature variations in message-passing than the single aggregation function of the original GAT.

### B. Spectral Analysis for Clothing Meshes

In addition to the challenges posed by efficient message-passing, the issue of over-smoothing in clothing deformations is also influenced by spectral bias. This bias refers to the tendency of neural networks to preferentially learn lower-complexity components during training, often neglecting intricate geometric details. In the context of garment neural simulation, this leads to missing essential wrinkles, leading to unrealistic results. To address this issue, we begin to explore the feasibility of dividing the mesh into distinct frequency bands, which would allow for a more focused attention on higher-frequency components. To do so, we first perform a spectral analysis of the deformed clothing mesh.

Given a mesh object with  $N$  vertices, the spectrum of the mesh graph is characterized by the eigenvalues and eigenvectors of the Laplacian matrix. The Laplacian matrix is defined as:  $L = D - C$ , where  $D = \text{diag}\{d_1, \dots, d_N\}$  is a diagonal degree matrix, with each element  $d_i$  representing the number of edges connected to vertex  $i$ . Here,  $C \in \mathbb{R}^{N \times N}$  is the adjacency matrix, which records whether there is an edge between pairs of nodes, with entries  $C_{ij} = 1$  if there is an edge, and  $C_{ij} = 0$  otherwise. Note that for a garment mesh, the Laplacian matrix is computed only once based on the rest pose topology and remains fixed during deformation. To standardize the influence of varying vertex degrees on graph analysis, the normalized Laplacian matrix  $L_{\text{norm}} = D^{-1/2}LD^{-1/2}$  is used. This matrix is then decomposed spectrally as follows:

$$L_{\text{norm}} = U\Lambda U^T, \quad (7)$$

where  $U = [u_1, \dots, u_N]$  is an orthonormal matrix with eigenvectors, and  $\Lambda = \text{diag}\{\lambda_1, \dots, \lambda_N\}$  contains the corresponding eigenvalues in non-decreasing order. Next, we can extract different components of mesh using the corresponding eigenvector elements. For the low-frequency band, we have:

$$X^l = U^l U^{l\top} X, \quad (8)$$

where  $U^l$  is the matrix containing the selected eigenvectors corresponding to the low-frequency band.  $X$  is the vertices of the garment mesh.  $X^l$  denotes the low frequency vertex positions. For the middle- and high- frequency band, we define  $X^m$  and  $X^h$  as:

$$X^m = U^m U^{m\top} X - X^l, \quad (9)$$

$$X^h = X - U^m U^{m\top} X, \quad (10)$$

where  $U^m$  correspond to the middle-frequency elements of the eigenvector  $U$ . Overall,  $X$  changes with the deformation of the mesh, causing the frequency components  $X^l$ ,  $X^m$ , and  $X^h$  to vary accordingly. The selection of  $U^l$  and  $U^m$  is discussed in detail in Sec. IV-D. As shown on the right side of Fig. 2, from top to bottom, the low-frequency part is associated with the eigenvectors corresponding to the smallest eigenvalues of the Laplacian matrix, capturing the global, smooth variations of the mesh. The middle-frequency part is crucial as it contains most of the detailed folds and wrinkles in clothing, which are essential for realistic simulations. In contrast, the high-frequency part, while capturing the finest details, often includes details that are imperceptible to the

naked eye and can introduce unwanted noise. Therefore, our simulation focuses more on the middle-frequency part to mitigate spectral bias and ensure that significant geometrical details are accurately represented.

### C. Model Optimization

To effectively learn the clothing behavior, we introduce a blend optimization scheme for the model. This strategy integrates both supervised and unsupervised losses to address the unawareness of physical information typically seen in purely supervised models and the lack of control often associated with purely unsupervised models.

Regarding the supervised loss, the conventional approach typically measures errors by directly comparing predictions with the ground truth. In this scenario, all frequency components are supervised together, potentially biasing the model learning. Despite diligent efforts to minimize error metrics, the model may primarily enhance performance on low-frequency components, as these are more prominent than high-frequency components. As a result, middle- and high- frequency details are often overlooked.

In response to these observations, and based on the spectral analysis of the garment mesh discussed in the previous subsection, we propose separately controlling the frequencies. Specifically, for vertex position constraints, we employ the  $L^2$  norm. As an example, we consider the low-frequency band:  $x_i^l$  represents the position of vertex  $i$  within this band, and  $x_i^{l,\text{GT}}$  denotes the corresponding ground truth data. For the middle- ( $x_i^m, x_i^{m,\text{GT}}$ ) and high-frequency parts ( $x_i^h, x_i^{h,\text{GT}}$ ), these values are no longer the vertex positions themselves but instead represent relative displacements derived from the frequency decomposition, as defined in Eqs. (9) and (10). The vertex loss for different frequency bands is defined as follows:

$$\mathcal{L}_{\text{lv}} = \frac{k_v}{N} \sum_{i=1}^N \|x_i^l - x_i^{l,\text{GT}}\|_2, \quad (11)$$

$$\mathcal{L}_{\text{mv}} = \frac{k_v}{N} \sum_{i=1}^N \|x_i^m - x_i^{m,\text{GT}}\|_2, \quad (12)$$

$$\mathcal{L}_{\text{hv}} = \frac{k_v}{N} \sum_{i=1}^N \|x_i^h - x_i^{h,\text{GT}}\|_2. \quad (13)$$

To achieve the frequency-division control, we employ two weight parameters to adjust the influence of the middle- and high-frequency components respectively:

$$\mathcal{L}_{\text{vert}} = \mathcal{L}_{\text{lv}} + k_{mv} \mathcal{L}_{\text{mv}} + k_{hv} \mathcal{L}_{\text{hv}}, \quad (14)$$

where  $k_{mv}$  and  $k_{hv}$  are the balancing weights for the middle and high-frequency bands. Experiment in Sec. IV-D will indicate the effect of different weight settings on the results.

Regarding the self-collision, our observations indicate that significant visual impacts typically occur at positions where large folds are formed, such as in a pleated dress during movement. To address these issues, we apply a repulsion loss to vertices that exceed a predefined proximity threshold. This approach is inspired by simulation studies of multi-layered clothing [70]–[73]. The repulsion allows the vertices to

maintain a safe distance from each other as folds are forming, thus effectively preventing self-collisions. The loss function is defined as follows:

$$\mathcal{L}_{sc} = \frac{k_{sc}}{\sum_{i,j} K_{i,j}} \sum_{i,j} K_{i,j} e^{-\lambda d_{ij}}, \quad (15)$$

where  $\lambda$  is used to control the exponential decay rate. The setting for this parameter needs to be determined based on the ranges of other losses. For our case,  $\lambda$  is set to 8.  $d_{ij}$  is the Euclidean distance between vertices  $i$  and  $j$ , scaled by a factor of 100 to ensure the unit of measurement remains consistent in centimeters. This loss function fundamentally introduces repulsive forces between vertices using the relation matrix  $K$ .  $K_{ij} = 1$  only if both of the following conditions are satisfied: (i) the geodesic distance between vertices  $i$  and  $j$  in the rest pose exceeds a threshold, and (ii) the Euclidean distance between these two vertices is less than another threshold. These conditions ensure that the repulsive force is applied only when vertices that are initially geodesically distant become undesirably close in Euclidean distance during deformation. The thresholds are adjustable based on fabric properties and are set here to 10 cm for geodesic distance and 2 cm for Euclidean distance, respectively. If either condition is not met, and no repulsive force is applied.

To more accurately model physically realistic behaviors, we incorporate additional loss terms inspired by prior studies: edge lengths [22], normal vector directions [62], garment-body collisions [20], gravity [19], stretch [20], [27], [63], and shear force [27]. Each term is controlled by specific weights:  $k_e$ ,  $k_n$ ,  $k_c$ ,  $k_g$ ,  $k_s$ , and  $k_h$ . For detailed definitions of these terms, please refer to the supplemental material.

#### IV. EXPERIMENTS

##### A. Implementation Details

**Dataset.** We collect a diverse array of garments from the CLOTH3D dataset [44], human body models from SMPL [61], and dynamic motions from the CMU Mocap dataset [74]. These elements are integrated to simulate garment-body interactions using silk-like fabrics in Blender. Our training dataset comprises 55 garment styles and nine distinct body shapes, encompassing approximately 50,000 poses. The validation dataset includes five garments and three body shapes, totaling around 3,000 poses. Meanwhile, the test dataset contains 15 garments paired with randomly generated body shapes, accounting for approximately 8,000 poses. We ensure there is no overlap among these datasets to maintain their independence. The supplemental material provides a comprehensive overview of the full range of garment data utilized in our study.

**Architecture.** As depicted in Fig. 2, our framework comprises two primary stages: information processing and deformation generation. In the first stage, body features are input into MLP layers with the hidden feature size of [128, 256, 128] and tanh activation. The garment mesh graph is processed using a three-layer FlexiGAT with feature sizes of [64, 64, 128], four heads, and tanh activation. Motion features are handled by a three-layer GRU, each with a feature size of 128. In the second stage, all processed features are initially fused

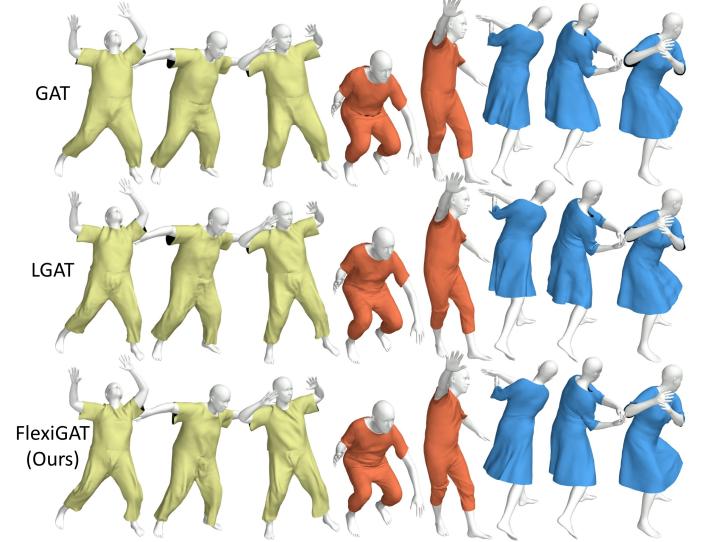


Fig. 3: Qualitative results of applying different graph processing technologies.

together by multiplication to form a comprehensive graph. Softmax is then applied to this graph to generate dynamic skinning weights  $\mathcal{W}^t$ . For generating the blend shape  $B^t$ , which is inherently complex, the graph is further processed through three additional layers of FlexiGATs with configurations of [64, 32, 3] and four heads. This detail is omitted in Fig. 2 for simplicity. Then, linear blend skinning is utilized to generate the dynamic garment mesh  $\tilde{M}^t$ , simultaneously forming a garment graph that includes node velocity and other features. This graph serves as the input for the detail correction generator, which uses FlexiGAT layers with the hidden feature size of [64, 64, 32, 32, 3] and four heads, to produce the final deformation refinement  $\Delta^t$ .

**Model training.** For the initialization of linear layers, we employ the geometric initialization method as stated in [75]. Conversely, the graph processing components are initialized using the Kaiming initialization method [76]. For optimization, we use the Adamax optimizer [77], with an initial learning rate of 3e-3, and apply cosine annealing to decay the learning rate.

Our training strategy leverages a progressive transition from supervised to unsupervised learning, capitalizing on the strengths of both approaches. Initially, a randomly initialized model is trained using supervised vertex and edge losses, with coefficients  $k_v = 100$  and  $k_e = 15$  respectively. The parameters  $k_{mv}$  and  $k_{hv}$  for weighting the middle and high-frequencies are 2 and 1.5 respectively. Once the vertex error falls below 5 cm, the normal loss with  $k_n = 50$  is activated to prevent over-smoothing and preserve wrinkle details.

We define an overfitting condition as less than 2.5% reduction in vertex error over 100 epochs. Upon reaching this condition, the information processing parameters are frozen, supervised loss weights are reduced by a factor of 0.1, and the garment-body collision loss is introduced with  $k_c = 1.5$ . Further, decreasing the loss value below 0.5 triggers the activation of the self-collision loss with  $k_{sc} = 15$ .

As the model stabilizes, supervised loss weights are halved

TABLE I: Quantitative deformation results. We measure five error metrics: the average vertex distance  $E_{\text{verts}}$  (cm), the average angular deviation of vertex normals  $E_{\text{vnorm}}$  ( $^{\circ}$ ), that of face normal  $E_{\text{fnorm}}$  ( $^{\circ}$ ), the relative edge length error  $E_{\text{edge}}$  (%), and discrete Gaussian curvature error  $E_{\text{curv}}$  between predictions and ground truth data. The term “FreqDiv” refers to the proposed frequency-division constraint.

	$E_{\text{verts}}$	$E_{\text{vnorm}}$	$E_{\text{fnorm}}$	$E_{\text{edge}}$	$E_{\text{curv}}$
GAT+FreqDiv	3.06	19.33	21.49	11.21	0.041
LGAT+FreqDiv	2.49	16.16	18.12	10.25	0.037
FlexiGAT+w/o FreqDiv	2.53	17.64	19.87	10.31	0.038
FlexiGAT+FreqDiv (Ours)	<b>2.28</b>	<b>13.62</b>	<b>14.96</b>	<b>8.94</b>	<b>0.029</b>

again, and the gravity and cloth model losses are incorporated with constants  $k_g = 0.1$ ,  $k_s = 5$ , and  $k_h = 2$ . This progressive integration of unsupervised losses, after an initial supervised phase, efficiently enhances convergence. Direct application of unsupervised losses typically renders the learning process highly unstable, often failing to converge when dealing with the complexities of multiple garments and diverse motions.

### B. Evaluation of Graph Processing Technologies

In this section, we compare our proposed graph processing mechanism, FlexiGAT, with previous techniques: the original GAT and the GAT with Lipschitz normalization (LGAT) as implemented in SwinGar [22] to mitigate spectral bias. In this experiment, the network structure and training strategy remain constant, with only the graph layer type being altered.

Fig. 3 shows the qualitative results obtained using these three graph processing methods across various garments and poses. Visually, the deformation result of the original GAT appears the coarsest, as it tends to smooth out the detailed wrinkles of the garment that should otherwise follow movement dynamics. This smoothing issue is a common problem with GAT and most GNNs, where the use of a non-negative propagation operator (*i.e.*,  $A^{(l)}$  in Eq. (5)) during message-passing acts like a low-pass filter. This results in poor differentiation of node features and significant performance degradation when multiple layers are stacked.

When incorporating Lipschitz normalization into GAT, it controls the spectral properties of the weight matrices, stabilizing learning and preventing the over-smoothing typical in standard GAT implementations. By bounding the Lipschitz constant, LGAT better preserves detailed features such as folds and wrinkles, which are essential for accurate garment deformation. Despite its potential, it does not completely solve the problem of feature homogenization over multiple layers. The normalization process may still fail to capture minor but significant variations in node features, especially in deep network architectures, resulting in visual effects that occasionally suffer from a loss of detail or exhibit incoherent bulging. This highlights the need for further refinement in balancing feature smoothing and detail preservation in GATs, particularly for complicated graph meshes.

Our FlexiGAT provides a dynamic adjustment in the influence shared between self and neighbor node features by

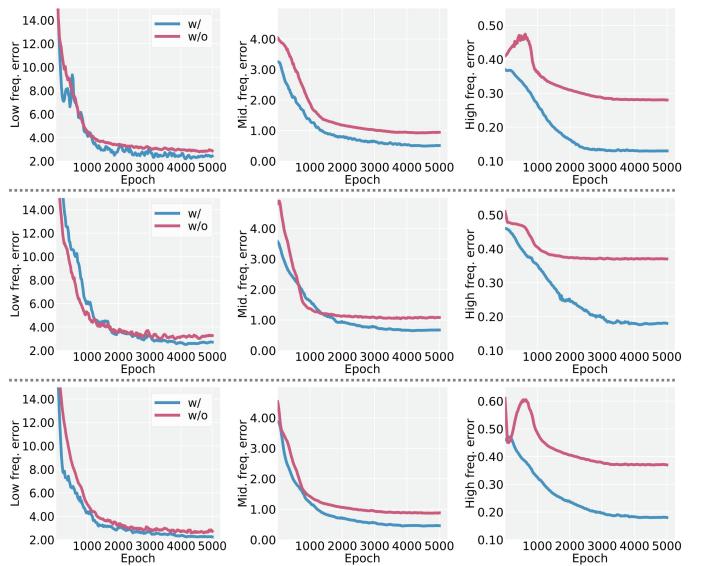


Fig. 4: Error of different frequency band w/ and w/o frequency division. The top, middle, and bottom rows correspond to the three garments displayed in Fig. 5.

tuning parameters  $P^{(l)}$  and  $Q^{(l)}$  in Eq. (6). This flexibility allows for a better feature expressiveness, thereby enhancing the performance in garment deformation approximation task. Moreover, the quantitative evaluation is presented at the first two rows and the last row of Tab. I. These results demonstrate that, with all other conditions kept constant and only the graph processing method changed, our approach achieves significant improvements over the original GAT across as well as GAT with Lipschitz normalization across various error metrics. These consistent results across various geometric and structural metrics validate FlexiGAT’s ability to effectively capture complex relationships and produce accurate, reliable predictions closely aligned with the ground truth.

### C. Evaluation on Frequency Division

In this section, we perform an ablation study to assess the impact of frequency division processing on deformation results. By applying the frequency-division loss, we gain active control over spectral information, which enhances the accuracy of middle-to-high frequency deformations essential for realistically producing garment wrinkles. To measure performance, we maintain model weights at various stages and use these weights to obtain deformation errors across different frequency bands. As shown in Fig. 4, the first row corresponds to the pink garment on the left in Fig. 5, the second row to the blue garment in the middle, and the third row to the coral orange garment on the right. Note that while the loss metrics are computed separately for different garments, the model weights used are consistent across all evaluations within the same epoch. Our method handle various types of garments within a unified model framework. Upon reaching training convergence, we observe low-frequency errors ranging between 2 and 4 cm. We initially think that w/o frequency division would give the lowest low-frequency error. However, contrary to

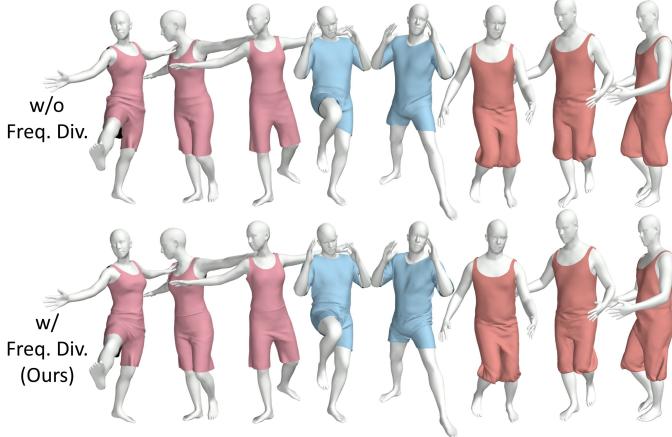


Fig. 5: Qualitative results of w/o and w/ frequency division. Applying frequency division loss to emphasize middle- and high-frequency information significantly enhances the visual quality of wrinkle generation.

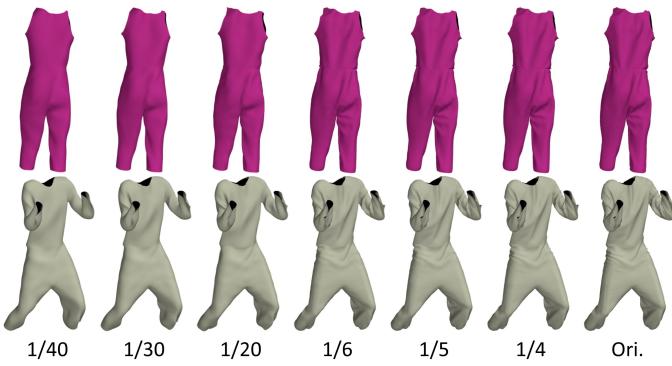


Fig. 6: Deformation results corresponding to different frequency bands. Using a lower vertex ratio, such as 1/40, corresponds to a lower frequency band and results in smoother clothing. Conversely, when the proportion reaches 1/4, nearly all deformation information is accurately reproduced.

our initial hypothesis, the model employing the frequency-division loss demonstrates superior overall performance. The error curves reveal larger fluctuations in low-frequency error with frequency-division loss compared to those without it. We speculate that enhancing middle-to-high frequencies indirectly influences the entire learning process, causing more fluctuations in low-frequency convergence, which ultimately leads to the improved model.

For the middle-frequency band, the introduction of frequency-division loss significantly enhances the results, which is essential for generating realistic wrinkles. In the high-frequency band, models trained without the designed loss typically exhibit an initial error increase followed by a decrease. In contrast, our proposed solution leads to a more consistent and gradual reduction in errors. Overall, the error curves across various training stages demonstrate the effectiveness of our frequency-division processing approach.

We also calculate the mean errors, shown in the last two rows of Tab. I. The inclusion of the frequency-division loss improves performance across all metrics compared to the

TABLE II: Quantitative deformation results across different frequency-division loss settings.

	$E_{verts}$	$E_{vnorm}$	$E_{fnorm}$	$E_{edge}$	$E_{curv}$
(1.5, 1.5)	2.31	14.23	15.98	9.28	<b>0.029</b>
(1.5, 2)	2.51	17.02	18.71	10.21	0.032
(1.5, 2.5)	2.57	17.21	19.19	10.44	0.034
(2, 2)	2.39	15.29	17.27	9.55	0.030
(2, 2.5)	2.54	17.68	19.23	10.13	0.036
(2.5, 1.5)	2.34	14.54	16.82	9.31	0.031
(2.5, 2)	2.42	16.67	19.04	9.37	0.031
(2.5, 2.5)	2.47	16.84	18.95	9.56	0.033
(2, 1.5) Ours	<b>2.28</b>	<b>13.62</b>	<b>14.96</b>	<b>8.94</b>	<b>0.029</b>

case without it. These results indicate that the direct control of different frequency bands can effectively emphasize high-frequency details like wrinkles while maintaining overall accuracy, highlighting its critical role in enhancing fine-grained surface deformation. Qualitative results, shown in Fig. 5, illustrate an intuitive contrast: without frequency-division loss, the results appear smoother and many wrinkles are lost. Conversely, our method retains finer and more realistic details.

#### D. Evaluation on Different Frequency Settings

In this section, we explore the effects of various frequency settings on clothing simulation. Initially, we select specific frequency bands by determining the number of eigenvectors based on the vertex count. This selection process constructs the matrix  $U$ , enabling the clothing deformations across different frequency bands as shown in Fig. 6. Notably, when using only 1/40 of the vertex count, the garment appears very smooth, nearly losing all detailed information. As we increase the ratio to 1/20 or 1/30 of the vertex count, finer details and wrinkles begin to emerge. Consequently, we establish 1/30 of the vertex count as the threshold separating low and middle-frequency bands. Further analysis involves examining ratios of 1/6, 1/5, and 1/4 of the vertex count to differentiate between middle and high frequencies. At 1/5 of the vertex count, a substantial portion of middle-frequency wrinkles is effectively captured. Thus, we choose 1/5 as the threshold between middle- and high-frequency bands. These thresholds consistently perform well across garments with varying vertex counts in the dataset.

Fig. 6 also reveals that while the low-frequency band is relatively narrow, it contains the majority of the “energy,” leading neural networks to predominantly focus on these frequencies. However, for achieving refined deformation effects, it is essential to augment the representation in middle frequencies to counteract spectral bias in GNNs.

Our frequency-division loss can optimize vertex position information across various frequency bands. We consistently maintain the weight of the low-frequency loss in Eq. (14) to one, serving as a reference. For middle and high frequencies, we assign different weights, *i.e.*,  $k_{mv}$  and  $k_{hv}$ , to adjust their influence. The errors of the resulting models are shown in Tab. II. Through a systematic grid search, the optimal weight combination is identified as (2, 1.5). The experimental results reveal that the key to achieving the desired effects lies in maintaining the weight of the middle frequency while moderately increasing the weight assigned to the high frequency.

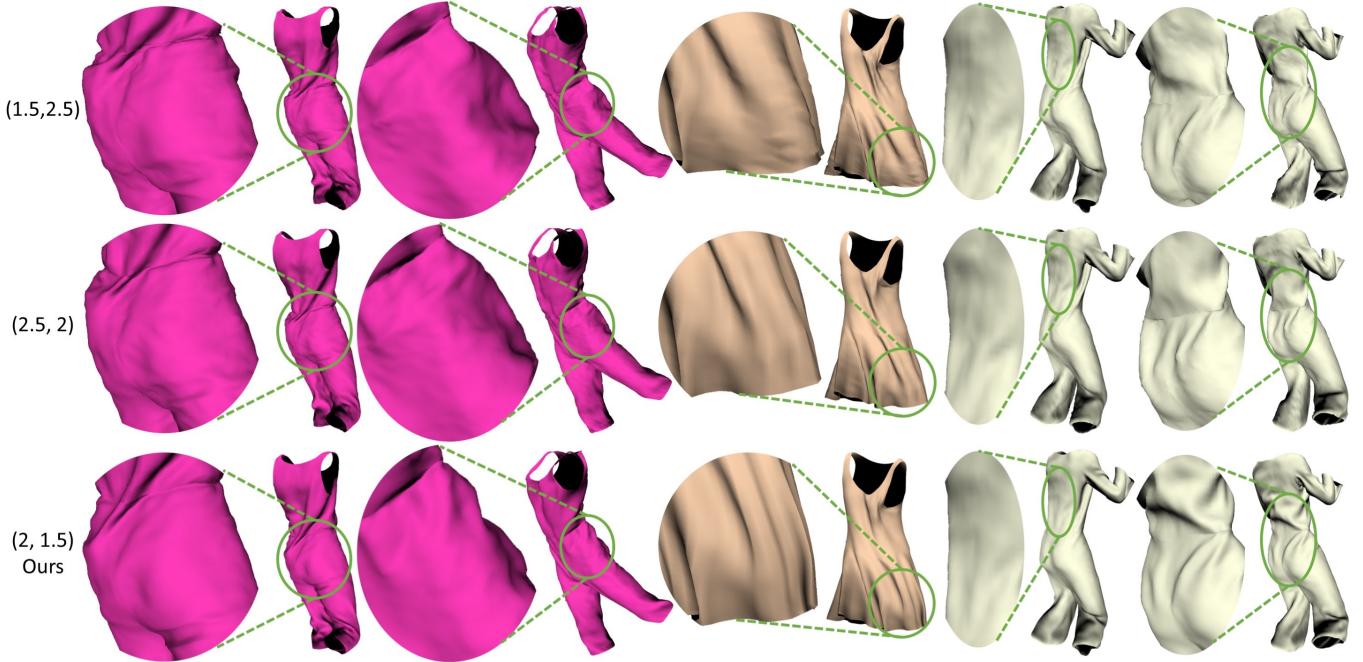


Fig. 7: Qualitative results of different frequency weights. While boosting high frequencies produces more wrinkles, it also introduces high-frequency noise, such as jittering and bulging in zoomed-in regions.

Fig. 7 presents the deformation results, comparing the least effective parameter pair (1.5, 2.5) with a moderately effective one (2.5, 2), and our selected settings. It is evident that overemphasizing the high frequencies leads to significant jitter and noticeable bulging in the deformations, which can be interpreted as high-frequency noise, markedly impairing the visual quality. Even the deformation with moderate settings also shows some bulging. Our chosen settings, however, demonstrate visually optimal deformation results.

### E. Evaluation on Self-Collision Loss

In this section, we evaluate the effectiveness of our self-collision loss  $\mathcal{L}_{sc}$  in Eq. (15) by excluding it from the total losses. We utilize the Möller-Trumbore intersection algorithm [78] to determine the self-collision ratio by identifying intersections between edges and faces and counting only intersecting faces; multiple intersections with the same face are counted once. To enhance computational efficiency, we implement a k-d tree to pre-filter vertices exceeding a Euclidean distance of 1.5 cm. Reasonable variations in this threshold do not substantially alter our conclusions. On average, the introduction of  $\mathcal{L}_{sc}$  significantly decreases the self-collision rate from 6.4% to 2.7%. As demonstrated in Fig. 8, we present line plots showing the rates (%) of self-collision across various garments and motions, accompanied by representative qualitative comparisons. As observed, self-collision frequently occur in areas where joints are clenched, such as armpits, waists, and crotches, as well as near the loose folds, including the bottoms of dress hem and trouser legs. In the absence of  $\mathcal{L}_{sc}$ , these specific areas tend to exhibit artifacts, adversely affecting the visual quality from certain viewpoints. Our proposed constraint effectively mitigates these issues by exerting a

TABLE III: Comparison with state-of-the-art methods in garment generalization, cloth model utilization, frequency control, and inference speed (fps).

	Clothing Generalization	Physics Awareness	Direct Freq. Control	Inference Speed
SwinGar	✓	✗	✗	307.8
GAPS	✗	✓	✗	370.4
HOOD	✓	✓	✗	13.4
Ours	✓	✓	✓	312.6

repulsive effect on non-neighboring vertices that come within proximity during movement. The proposed approach not only improves quantitative metrics by 57.8% but also enhances the physical plausibility in the neural clothing simulation.

### F. Comparison to State-of-the-Art Methods

To further evaluate the capabilities of our method, we conduct a comparative analysis with three state-of-the-art learning-based clothing deformation methods, including SwinGar [22], GAPS [63], and HOOD [21].

**Property comparison** is shown in Tab. III. A check mark signifies the presence of a property, while a cross denotes its absence. SwinGar, HOOD, and our model all share the ability to generalize to different types of garments with a single unified model. This is a convenient feature for practical applications and represents a promising direction for exploring the generality of neural simulations in clothing deformation. GAPS demonstrates robust generalization to novel motions but falls short in its ability to process diverse garments within a unified framework.

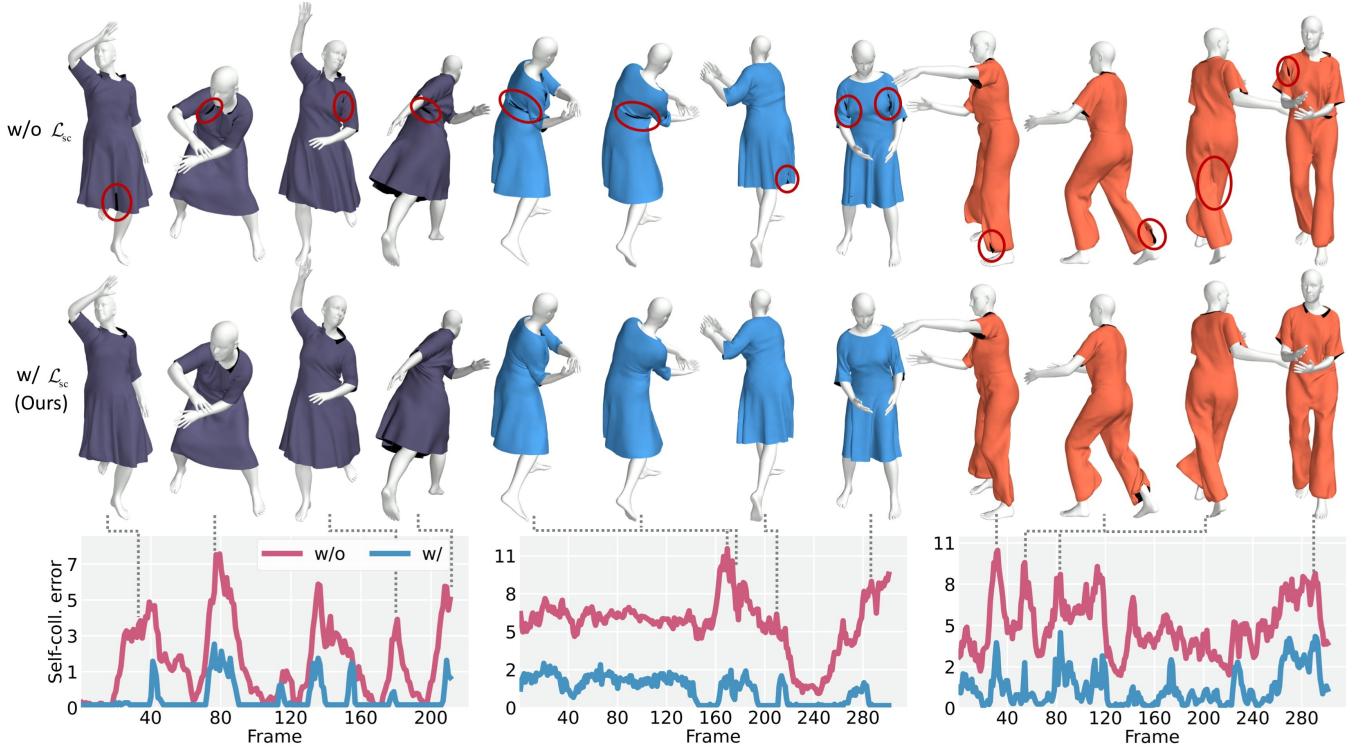


Fig. 8: Qualitative and quantitative evaluations on self-collision loss.

In terms of wrinkle generation, the integration of physics-aware losses has a significant impact on the diversity of deformations. SwinGar relies solely on supervised losses for deformation learning, which confine its output largely within the bounds of its training dataset. This limitation often leads to an unnatural appearance and reduced diversity in deformation results.

Our method differentiates itself from previous studies particularly during the training phase, where it leverages the ability to adjust weights across different frequency bands, allowing for a targeted emphasis on the information-rich middle-frequency band crucial for capturing subtle and complex fabric details. In contrast, conventional supervised losses typically focus primarily on low-frequency information due to their reliance on using complete vertex coordinates, which naturally capture base deformations and result in over-smoothing outputs. Consequently, other methods are unable to accurately render complex wrinkle patterns that are essential for realistic clothing simulations.

Regarding inference speed, both SwinGar and our method use graph networks. Despite their complex structures, these deep learning frameworks manage to achieve efficient multi-frame parallel processing, enabling real-time performance. In contrast, HOOD, although capable of handling different garments and motions, requires velocity and acceleration data from previous frame to inform deformations. This requirement makes multi-frame parallel processing challenging and slows down the inference process. On the other hand, the architecture of the GAPS network is relatively simple, focusing primarily on linear layers and gated recurrent units. This simplicity allows for high-speed inference and supports real-time per-

formance, making it highly efficient. However, when dealing with topology-varying clothing, the requirement to train each object separately still leads to considerable time consumption.

**Deformation results** are shown in Fig. 9. SwinGar’s deformation learning is driven by simulation data, *i.e.*, the deformations that are mere extrapolations from the training data distribution, leading to issues such as unnatural hemlines in the apricot dress and strip-like wrinkles in the abdomen of a purple jumpsuit (the second row of Fig. 9). GAPS, while generally producing realistic deformation results, suffers from a lack of detail, and emphasizing wrinkle-related clothing model losses can destabilize inertial force convergence, presenting a challenge in balancing multiple losses to achieve optimal results. Different training modes significantly impact the results. HOOD stands out for its temporal consistency and stability, implementing dynamic deformations using velocity and acceleration predictions, though its reliance on various unsupervised physics-informed losses occasionally results in crumpled appearances (the fourth row of Fig. 9). Our method, by improving information propagation within the graph and adjusting the information in different frequency bands, effectively mitigates the crumpled issues and over-smoothing, showing the fine-grained details in qualitative results. Moreover, it achieves better accuracy and aligns more closely with the ground truth. Compared to the spectrum-inspired SwinGar, our method reduces the average vertex distance  $E_{verts}$  by approximately 0.2 cm and the average angular deviation of vertex normals  $E_{vnorm}$  by  $3.69^\circ$ . Additionally, with the introduction of self-collision unsupervised losses, our method lowers the self-collision rate by 0.31% compared to SwinGar.

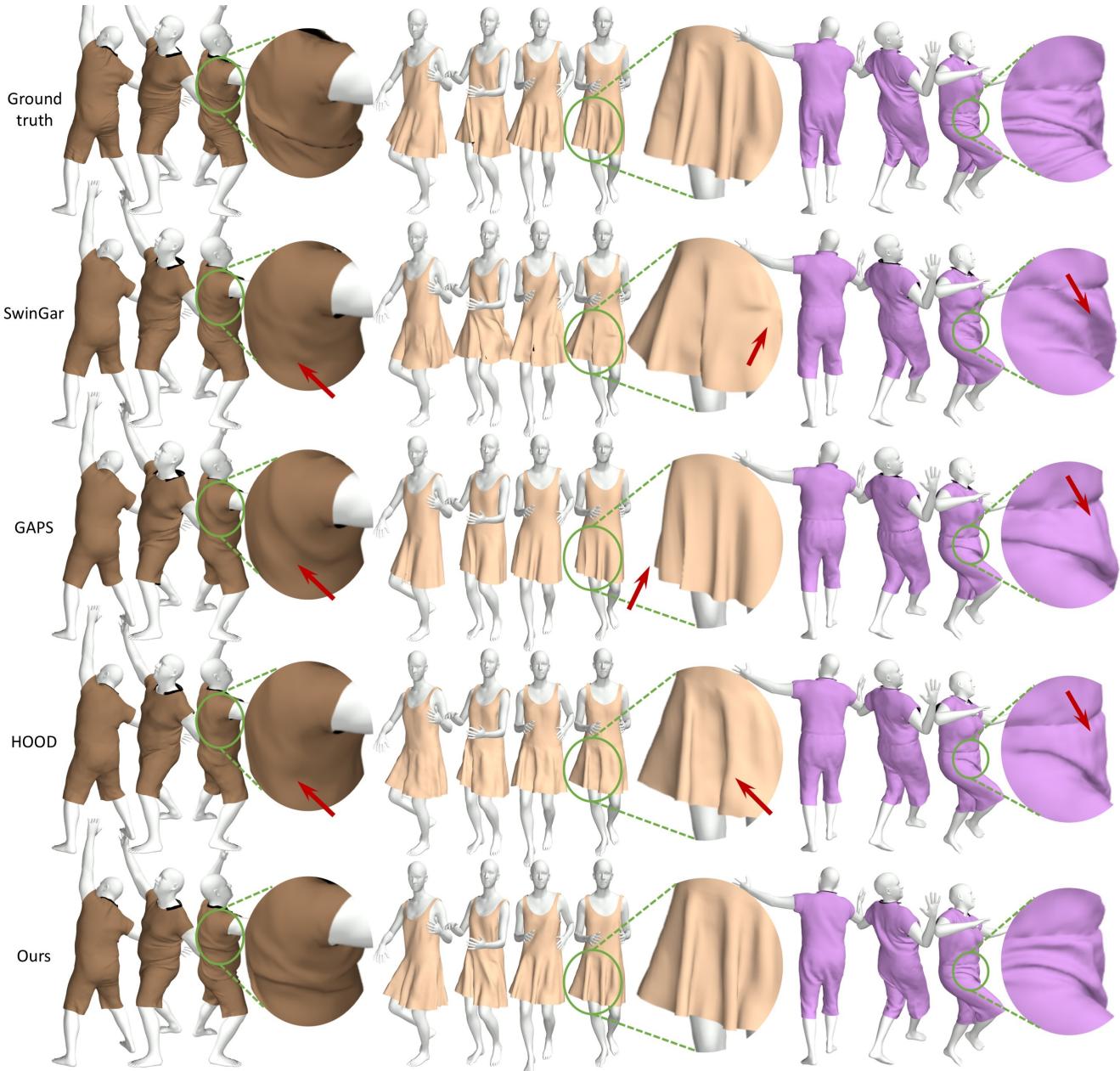


Fig. 9: Comparison with state-of-the-art neural cloth simulation methods. Observe the natural and fine-grained details in our generated deformations.

## V. CONCLUSION

We have developed a novel solution for simulating the clothing behavior of arbitrary meshes with an enhanced degree of realism. Leveraging the generalization abilities of graph neural networks, which are good at processing irregular data, we introduce a flexible graph information propagation mechanism for the task of 3D mesh deformation. This mechanism greatly improves the expressiveness of our model. To facilitate effective model learning, we propose a frequency-division loss to optimize individual frequency components directly, thereby ensuring the generation of rich details. Moreover, we incorporate self-collision and several unsupervised loss terms based on physical properties together to further refine the

model’s performance and enhance the deformation plausibility. Extensive evaluations demonstrate the effectiveness of our approach. We believe our method makes an important step in the field of deformation learning, achieving high fidelity and efficiency in clothing simulation.

Our work still has a few limitations. First, our self-collision constraint does not entirely prevent intersections among garment vertices. Specifically, it may not be effective in avoiding certain edge-to-edge intersections or collisions that occur under extreme poses. These issues might require additional post-processing steps to achieve optimal results. Furthermore, while our current framework is able to handle garments with varying topologies, it does not address other types of body objects that are not represented through SMPL parameters. There is

potential to expand our existing body feature representation to a more generalized form, thereby increasing the applicability of the method.

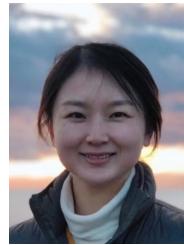
### ACKNOWLEDGMENTS

This work is partially supported by the National Natural Science Foundation of China (62402021, 62403017), the Beijing Natural Science Foundation (4244088, 4232017) and JSPS KAKENHI (22K12331).

### REFERENCES

- [1] N. Magnenat-Thalmann, R. Laperrière, and D. Thalmann, “Joint-dependent local deformations for hand animation and object grasping,” in *Proc. Graph. Interface*, 1989, pp. 26–33.
- [2] L. Kavan, S. Collins, J. Zára, and C. O’Sullivan, “Skinning with dual quaternions,” in *Proc. Symp. Interactive 3D Graph. Games*, 2007, pp. 39–46.
- [3] J. P. Lewis, M. Cordner, and N. Fong, “Pose space deformation: A unified approach to shape interpolation and skeleton-driven deformation,” in *Proc. Annu. Conf. Comput. Graph. Interact. Tech.*, 2000, pp. 165–172.
- [4] A. Nealen, M. Müller, R. Keiser, E. Boxerman, and M. Carlson, “Physically based deformable models in computer graphics,” *Comput. Graph. Forum*, vol. 25, no. 4, pp. 809–836, 2006.
- [5] R. Narain, A. Samii, and J. F. O’Brien, “Adaptive anisotropic remeshing for cloth simulation,” *ACM Trans. Graph.*, vol. 31, no. 6, 2012.
- [6] M. Macklin, M. Müller, and N. Chentanez, “XPBD: Position-based simulation of compliant constrained dynamics,” in *Proc. Int. Conf. Motion Games*, 2016, pp. 49–54.
- [7] M. Zhang, T. Y. Wang, D. Ceylan, and N. J. Mitra, “Dynamic neural garments,” *ACM Trans. Graph.*, vol. 40, no. 6, 2021.
- [8] X. Pan, J. Mai, X. Jiang, D. Tang, J. Li, T. Shao, K. Zhou, X. Jin, and D. Manocha, “Predicting loose-fitting garment deformations using bone-driven motion networks,” in *ACM SIGGRAPH*, 2022.
- [9] R. Shi, T. Li, L. Zhang, and Y. Yamaguchi, “Visualization comparison of vision transformers and convolutional neural networks,” *IEEE Trans. Multimedia*, vol. 26, pp. 2327–2339, 2024.
- [10] R. Shi, T. Li, Y. Yamaguchi, and L. Zhang, “Traffic scene-informed attribution of autonomous driving decisions,” *IEEE Trans. Intell. Transp. Syst.*, pp. 1–12, 2025.
- [11] R. Vidaurre, I. Santesteban, E. Garces, and D. Casas, “Fully convolutional neural networks for parametric virtual try-on,” *Comput. Graph. Forum*, vol. 39, no. 8, pp. 145–156, 2020.
- [12] M. Zhang, D. Ceylan, and N. J. Mitra, “Motion guided deep dynamic 3D garments,” *ACM Trans. Graph.*, vol. 41, no. 6, 2022.
- [13] Y. D. Li, M. Tang, X. R. Chen, Y. Yang, R. F. Tong, B. L. An, S. C. Yang, Y. Li, and Q. L. Kou, “D-Cloth: Skinning-based cloth dynamic prediction with a three-stage network,” *Comput. Graph. Forum*, vol. 42, no. 7, p. e14937, 2023.
- [14] T. Li, R. Shi, Z. Li, T. Kanai, and Q. Zhu, “Efficient deformation learning of varied garments with a structure-preserving multilevel framework,” *Proc. ACM Comput. Graph. Interact. Tech.*, vol. 7, no. 1, may 2024.
- [15] Y. Li, M. Tang, Y. bo Yang, Z. Huang, R. Tong, S. Yang, Y. Li, and D. Manocha, “N-Cloth: Predicting 3D cloth deformation with mesh-based networks,” *Comput. Graph. Forum*, vol. 41, no. 2, pp. 547–558, 2022.
- [16] T. Li, R. Shi, and T. Kanai, “Detail-aware deep clothing animations infused with multi-source attributes,” *Comput. Graph. Forum*, vol. 42, no. 1, pp. 231–244, 2023.
- [17] P. Li, T. Y. Wang, T. L. Kesdogan, D. Ceylan, and O. Sorkine-Hornung, “Neural garment dynamics via manifold-aware transformers,” *Comput. Graph. Forum*, vol. 43, no. 2, p. e15028, 2024.
- [18] H. Bertiche, M. Madadi, and S. Escalera, “PBNS: Physically based neural simulation for unsupervised garment pose space deformation,” *ACM Trans. Graph.*, vol. 40, no. 6, 2021.
- [19] I. Santesteban, M. A. Otaduy, and D. Casas, “SNUG: Self-supervised neural dynamic garments,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 8130–8140.
- [20] H. Bertiche, M. Madadi, and S. Escalera, “Neural cloth simulation,” *ACM Trans. Graph.*, vol. 41, no. 6, 2022.
- [21] A. Grigorev, B. Thomaszewski, M. J. Black, and O. Hilliges, “HOOD: Hierarchical graphs for generalized modelling of clothing dynamics,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 16965–16974.
- [22] T. Li, R. Shi, Q. Zhu, and T. Kanai, “SwinGar: Spectrum-inspired neural dynamic deformation for free-swinging garments,” *IEEE Trans. Vis. Comput. Graphics*, pp. 1–16, 2023.
- [23] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” in *Proc. Int. Conf. Learn. Representations*, 2017. [Online]. Available: <https://openreview.net/forum?id=SJU4ayYgl>
- [24] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, “Graph attention networks,” in *Proc. Int. Conf. Learn. Representations*, 2018. [Online]. Available: <https://openreview.net/forum?id=rJXMpikCZ>
- [25] S. Brody, U. Alon, and E. Yahav, “How attentive are graph attention networks?” in *Proc. Int. Conf. Learn. Representations*, 2022. [Online]. Available: <https://openreview.net/forum?id=F72ximsx7C1>
- [26] C. Patel, Z. Liao, and G. Pons-Moll, “TailorNet: Predicting clothing in 3D as a function of human pose, shape and garment style,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 7363–7373.
- [27] D. Baraff and A. Witkin, “Large steps in cloth simulation,” in *ACM SIGGRAPH*, 1998, pp. 43–54.
- [28] C. Jiang, T. Gast, and J. Teran, “Anisotropic elastoplasticity for cloth, knit and hair frictional contact,” *ACM Trans. Graph.*, vol. 36, no. 4, 2017.
- [29] M. Tang, T. Wang, Z. Liu, R. Tong, and D. Manocha, “I-Cloth: Incremental collision handling for gpu-based interactive cloth simulation,” *ACM Trans. Graph.*, vol. 37, no. 6, 2018.
- [30] M. Li, D. M. Kaufman, and C. Jiang, “Codimensional incremental potential contact,” *ACM Trans. Graph.*, vol. 40, no. 4, 2021.
- [31] K.-J. Choi and H.-S. Ko, “Stable but responsive cloth,” *ACM Trans. Graph.*, vol. 21, no. 3, pp. 604–611, 2002.
- [32] P. Volino, N. Magnenat-Thalmann, and F. Faure, “A simple approach to nonlinear tensile stiffness for accurate cloth simulation,” *ACM Trans. Graph.*, vol. 28, no. 4, 2009.
- [33] J. Liang, M. Lin, and V. Koltun, “Differentiable cloth simulation for inverse problems,” in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019.
- [34] Y. Li, T. Du, K. Wu, J. Xu, and W. Matusik, “Diffcloth: Differentiable cloth simulation with dry frictional contact,” *ACM Trans. Graph.*, vol. 42, no. 1, 2022.
- [35] G. Cirio, J. Lopez-Moreno, D. Miraut, and M. A. Otaduy, “Yarn-level simulation of woven cloth,” *ACM Trans. Graph.*, vol. 33, no. 6, 2014.
- [36] G. Sperl, R. M. Sánchez-Banderas, M. Li, C. Wojtan, and M. A. Otaduy, “Estimation of yarn-level simulation models for production fabrics,” *ACM Trans. Graph.*, vol. 41, no. 4, 2022.
- [37] L. Wu, B. Wu, Y. Yang, and H. Wang, “A safe and fast repulsion method for gpu-based cloth self collisions,” *ACM Trans. Graph.*, vol. 40, no. 1, 2020.
- [38] C. Li, M. Tang, R. Tong, M. Cai, J. Zhao, and D. Manocha, “P-Cloth: Interactive complex cloth simulation on multi-gpu systems using dynamic matrix assembly and pipelined implicit integrators,” *ACM Trans. Graph.*, vol. 39, no. 6, nov 2020.
- [39] H. Wang, “GPU-based simulation of cloth wrinkles at submillimeter levels,” *ACM Trans. Graph.*, vol. 40, no. 4, 2021.
- [40] Z. Chen, H.-Y. Chen, D. M. Kaufman, M. Skouras, and E. Vouga, “Fine wrinkling on coarsely meshed thin shells,” *ACM Trans. Graph.*, vol. 40, no. 5, 2021.
- [41] S. Bouaziz, S. Martin, T. Liu, L. Kavan, and M. Pauly, “Projective dynamics: fusing constraint projections for fast simulation,” *ACM Trans. Graph.*, vol. 33, no. 4, 2014.
- [42] M. Ly, J. Jouve, L. Boissieux, and F. Bertails-Descoubes, “Projective dynamics with dry frictional contact,” *ACM Trans. Graph.*, vol. 39, no. 4, 2020.
- [43] T. Y. Wang, T. Shao, K. Fu, and N. J. Mitra, “Learning an intrinsic garment space for interactive authoring of garment animation,” *ACM Trans. Graph.*, vol. 38, no. 6, 2019.
- [44] H. Bertiche, M. Madadi, and S. Escalera, “Cloth3D: Clothed 3D humans,” in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 344–359.
- [45] T. Y. Wang, D. Ceylan, J. Popović, and N. J. Mitra, “Learning a shared shape space for multimodal garment design,” *ACM Trans. Graph.*, vol. 37, no. 6, 2018.
- [46] G. Tiwari, B. L. Bhatnagar, T. Tung, and G. Pons-Moll, “SIZER: A dataset and model for parsing 3D clothing and learning size sensitive 3D clothing,” in *Proc. Eur. Conf. Comput. Vis.*, vol. 12348, 2020, pp. 1–18.
- [47] Z. Zheng, T. Wang, Q. Feng, Z. Pan, X. Gao, and K. Wu, “Proxy asset generation for cloth simulation in games,” *ACM Trans. Graph.*, vol. 43, no. 4, Jul. 2024.
- [48] J. Yu and Z. Wang, “Super-resolution cloth animation with spatial and temporal coherence,” *ACM Trans. Graph.*, vol. 43, no. 4, Jul. 2024.

- [49] Z. Lähner, D. Cremers, and T. Tung, “Deepwrinkles: Accurate and realistic clothing modeling,” in *Proc. Eur. Conf. Comput. Vis.*, 2018, p. 698–715.
- [50] N. Jin, Y. Zhu, Z. Geng, and R. Fedkiw, “A pixel-based framework for data-driven clothing,” *Comput. Graph. Forum*, vol. 39, no. 8, pp. 135–144, 2020.
- [51] J. Wu, Y. Jin, Z. Geng, H. Zhou, and R. Fedkiw, “Recovering geometric information with learned texture perturbations,” *Proc. ACM Comput. Graph. Interact. Tech.*, vol. 4, no. 3, Sep. 2021.
- [52] Z. Geng, D. Johnson, and R. Fedkiw, “Coercing machine learning to output physically accurate results,” *J. Comput. Phys.*, vol. 406, p. 109099, 2020.
- [53] J. Wu, Z. Geng, H. Zhou, and R. Fedkiw, “Skinning a parameterization of three-dimensional space for neural network cloth,” *CoRR*, vol. abs/2006.04874, 2020. [Online]. Available: <https://arxiv.org/abs/2006.04874>
- [54] L. De Luigi, R. Li, B. Guillard, M. Salzmann, and P. Fua, “Drapenet: Garment generation and self-supervised draping,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 1451–1460.
- [55] R. Li, C. Dumery, B. Guillard, and P. Fua, “Garment recovery with shape and deformation priors,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 1586–1595.
- [56] Y. Li, H.-Y. Chen, E. Larionov, N. Sarafianos, W. Matusik, and T. Stuyck, “Diffavatar: Simulation-ready garment optimization with differentiable simulation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 4368–4378.
- [57] L. Liu, Y. Zheng, D. Tang, Y. Yuan, C. Fan, and K. Zhou, “NeuroSkinning: Automatic skin binding for production characters with deep graph networks,” *ACM Trans. Graph.*, vol. 38, no. 4, 2019.
- [58] Z. Xu, Y. Zhou, E. Kalogerakis, C. Landreth, and K. Singh, “RigNet: Neural rigging for articulated characters,” *ACM Trans. Graph.*, vol. 39, no. 4, 2020.
- [59] T. Li, R. Shi, and T. Kanai, “DenseGATs: A graph-attention-based network for nonlinear character deformation,” in *Proc. Symp. Interactive 3D Graph. Games*, 2020, pp. 5:1–5:9.
- [60] ———, “MultiResGNet: Approximating nonlinear deformation via multi-resolution graphs,” *Comput. Graph. Forum*, vol. 40, no. 2, pp. 537–548, 2021.
- [61] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, “SMPL: A skinned multi-person linear model,” *ACM Trans. Graph.*, vol. 34, no. 6, 2015.
- [62] E. Gundogdu, V. Constantin, S. Parashar, A. Seifoddini, M. Dang, M. Salzmann, and P. Fua, “GarNet++: Improving fast and accurate static 3D cloth draping by curvature loss,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 1, pp. 181–195, 2022.
- [63] R. Chen, L. Chen, and S. Parashar, “GAPS: Geometry-aware, physics-based, self-supervised neural garment draping,” in *Int. Conf. on 3D Vis.*, 2024, pp. 116–125.
- [64] H. Bertiche, M. Madadi, E. Tyson, and S. Escalera, “DeePSD: Automatic deep skinning and pose space deformation for 3D garment animation,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 5451–5460.
- [65] M. Tancik, P. P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. T. Barron, and R. Ng, “Fourier features let networks learn high frequency functions in low dimensional domains,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2020. [Online]. Available: <https://proceedings.neurips.cc/paper/2020/hash/55053683268957697aa39fba6f231c68-Abstract.html>
- [66] W. Xie, D. Song, C. Xu, C. Xu, H. Zhang, and Y. Wang, “Learning frequency-aware dynamic network for efficient super-resolution,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 4288–4297.
- [67] F. Fang, L. Hu, J. Liu, Q. Yi, T. Zeng, and G. Zhang, “HFGN: High-frequency residual feature guided network for fast mri reconstruction,” *Pattern Recognit.*, vol. 156, p. 110801, 2024.
- [68] V. Sitzmann, J. N. P. Martel, A. W. Bergman, D. B. Lindell, and G. Wetzstein, “Implicit neural representations with periodic activation functions,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2020.
- [69] X. Chen, T. H. Li, R. Zhang, and G. Li, “Frequency-aware self-supervised monocular depth estimation,” in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2023, pp. 5797–5806.
- [70] I. Santesteban, M. Otaduy, N. Thuerey, and D. Casas, “ULNeF: Untangled layered neural fields for mix-and-match virtual try-on,” in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 12110–12125.
- [71] R. Li, B. Guillard, and P. Fua, “ISP: multi-layered garment draping with implicit sewing patterns,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2023.
- [72] Y. Shao, C. C. Loy, and B. Dai, “Towards multi-layered 3D garments animation,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2023, pp. 14315–14324.
- [73] D. Lee, H. Kang, and I.-K. Lee, “ClothCombo: Modeling inter-cloth interaction for draping multi-layered clothes,” *ACM Trans. Graph.*, vol. 42, no. 6, 2023.
- [74] Carnegie-Mellon, “CMU graphics lab motion capture database,” <http://mocap.cs.cmu.edu/>, 2010, accessed: 2023.
- [75] M. Atzmon and Y. Lipman, “SAL: Sign agnostic learning of shapes from raw data,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 2562–2571.
- [76] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1026–1034.
- [77] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Proc. Int. Conf. Learn. Representations*, 2015. [Online]. Available: <http://arxiv.org/abs/1412.6980>
- [78] T. Möller and B. Trumbore, “Fast, minimum storage ray-triangle intersection,” *J. Graph. Tools*, vol. 2, no. 1, pp. 21–28, 1997.



**Tianxing Li** is a lecturer in the College of Computer Science, Beijing University of Technology, Beijing, China. Her current research interests include computer animation, visualization, and pattern recognition. She received her Ph.D. degree in graphic and computer sciences from the University of Tokyo, Tokyo, Japan, in 2021.



**Rui Shi** is a lecturer in the School of Information Science and Technology, Beijing University of Technology, Beijing, China. His current research interests include explainable artificial intelligence, neural network, and visualization. He received his Ph.D. degree in graphic and computer sciences from the University of Tokyo, Tokyo, Japan, in 2022. He served as a visiting researcher in the Department of General Systems Studies, the University of Tokyo.



**Takashi Kanai** is a professor in the Graduate School of Arts and Sciences, the University of Tokyo. His research interests include geometry processing and physics-based animation in computer graphics. He received his Ph.D. degree in engineering from the University of Tokyo in 1998. He is a member of ACM, ACM SIGGRAPH, IIEEJ (the Institute of Image Electronics Engineers of Japan), and IPSJ (Information Processing Society of Japan).



**Qing Zhu** is a professor in the College of Computer Science, Beijing University of Technology, Beijing, China. Her research interests include multimedia information processing technology, virtual reality technology, and information integration technology. She received her Ph.D. degree in electronic information and communication from Waseda University, Tokyo, Japan, in 2000.