Self-supervised Garment Dynamics with Persistent Wrinkles

Xiaoyuan Yang[1], Deshan Gong[3], Taku Komura[3], and He Wang⋆[1,2]

[1]University of Leeds, [2]University College London, [3]The University of Hong Kong

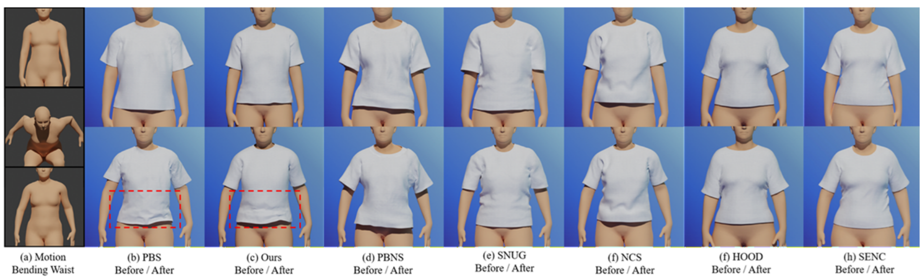


Fig. 1: Given the bending waist motion (a), the physics-based cloth simulator (PBS) can simulate persistent wrinkles around the abdomen area (b) caused by the material plasticity of fabrics. However, existing self-supervised cloth simulators (PBNS [5], SNUG [33], NCS [6], HOOD [14], and SENC [23]) naively model cloth as elastic material and cannot simulate persistent wrinkles (d, e, f, g, h). Our novel self-supervised simulator models material plasticity to simulate persistent wrinkles as the PBS (c).

Abstract. Self-supervised neural garment simulation has become popular due to its computational efficiency, good visual realism, and no reliance on training data. However, existing methods greatly simplify the mechanical properties of fabrics, ignoring persistent wrinkles caused by plasticity. Although this simplification allows for modeling of purely elastic material and simple training via energy minimization, the lack of believable wrinkles adversely affects the visual realism. Therefore, we introduce the first self-supervised neural garment simulator that explicitly models persistent wrinkles. This is accomplished through a novel physics-inspired loss function, which turns learning into a moving energy minimization problem to mimic plasticity. However, this requires learning to use a changing loss function, which causes difficulties in training because the loss function changes during optimization. To this end, we propose a new physics-inspired curriculum learning scheme where the target material for learning gradually changes from pure elasticity to elasto-plasticity, allowing the loss function and the learnable parameters to jointly converge. Through a comprehensive evaluation, we show that for the first time, self-supervised learning models can generate natural persistent wrinkles, outperforming existing methods on a variety of garments, body shapes, and body motions, according to a range of metrics. Our code is publicly available at *https://github.com/realcrane/EPNet*

⋆ corresponding author, he_wang@ucl.ac.uk

Keywords: Cloth simulation · Self-supervised learning · Persistent wrinkles · Curriculum learning

1 Introduction

Virtual garments are important digital assets for computer games, mixed reality, online marketing, *etc*., where generating realistic virtual garments is universally required but labor-intensive. High-fidelity virtual garments are generated from physics-based simulation with the visual quality as the main goal [37], but it is time-consuming and not suitable for interactive or real-time applications. In contrast, data-driven methods can quickly generate plausible garments, but they are not only less realistic but also strongly dependent on large amounts of training data [16, 17]. Recently, self-supervised neural garment simulators have emerged as a midway solution, where data reliance is eliminated and the time-consuming solution of physical systems is replaced by faster inference on neural networks.

Existing self-supervised learning (SSL) garment models focus on draping, cloth dynamics, collision handling, contacts [5,6,33]. Despite being designed for various purposes, these methods use a similar strategy for modeling bending. They often formulate a bending energy which is minimized during training. This strategy essentially treats the garment as an elastic material (at least in bending behaviors) and aims to train the network to predict meshes with a small bending energy. The formulation usually assumes a rest bending (RB) term, which is defined as the dihedral angle between edge-adjacent triangles [15], and the deviation from this RB yields the bending potential energy. One example is $L_{bending} = \sum_{edges} k(\theta - \theta_{RB})^2$ [6] where k is a scalar, θ is the predicted dihedral angle, and θ_{RB} is the dihedral angle in RB state. In existing methods, RB is often set manually. When setting RB to zero, it completely ignores plasticity, so the wrinkles generated by these methods are essentially left-over bending energy, and no sharp and persistent wrinkles can be generated; when setting RB to non-zero, it becomes *dynamic*, which requires manual tuning for every edge in every frame. Furthermore, this dynamic change of RB should be physically based instead of being arbitrarily pre-defined, because plasticity on an edge can happen multiple times as the bending increases, manifested as *multi-phasic plasticity* [11]. Consequently, the simplification of bending energy in existing methods leads to unrealistic garment deformation, failing to reproduce persistent wrinkles, a common phenomenon in garments.

We propose the first SSL neural garment simulator that can generate natural persistent wrinkles. Given that wrinkles are mainly formed due to bending [11, 29], we propose a new bending energy term based on dynamic RB. For every edge in the garment mesh, its RB is a spatio-temporal function of the deformation of its neighboring edges and its own deformation in the past. The design is inspired by that, spatially, existing methods can already predict the bending angles close to a pre-defined *static* RB [5,6,33]. This gives us the confidence that neural networks can perform similarly well for dynamic RBs if they are known. However, since dynamic RBs are not known *a priori*, we unsupervisingly learn the RB as a time series with the correct dynamics, to mimic the multi-phasic plastic behavior of real fabrics.

Despite the conceptual simplicity of our idea, training a neural network with dynamic RB presents an underexplored learning paradigm. Previously, the learning was a fixed energy minimization problem as the RB is pre-defined. Now it becomes a moving energy minimization problem as the RB is not known in the beginning of the training. This creates a chicken-egg problem: not knowing the RB makes it difficult to specify a loss function to train the model, but we can only compute the correct RB from the garment deformation predicted by a trained model. To this end, we propose a physics-inspired curriculum learning scheme which enables us to jointly learn the RB values and the model parameters.

We qualitatively and quantitatively evaluate our simulator and compare it with the state-of-the-art methods across various body motions, body shapes, and garment types, on a wide range of metrics. We demonstrate that, for the first time, persistent wrinkles can be generated from SSL models, which are both visually believable and physically plausible. Our contributions include:

1. the first SSL neural garment model that can generate physically plausible and natural persistent wrinkles.
2. a novel neural network, trained by energy minimization, containing a new dynamic bending energy term to mimic multi-phasic plasticity.
3. a physics-inspired curriculum learning scheme to enable the training with changing loss functions.

2 Related Work

Physics-Based Cloth Simulation Physics-based cloth simulation has been extensively studied since early deformable models [35], with continuous advances toward visually realistic garment behavior [10, 12, 21, 22, 34, 39]. Modern simulators incorporate sophisticated material models and fine-grained behaviors, including yarn-level effects and plastic deformation [8, 9, 11, 18, 29]. While these methods can accurately model time-dependent and plastic phenomena through explicit physical integration, they are computationally expensive and unsuitable for real-time applications.

Data-driven Garment Simulation Data-driven approaches replace explicit simulation with neural networks trained on simulation data [17,31,32,36,40]. By learning deformation patterns from large datasets, these methods achieve fast runtime inference and can implicitly capture complex material behaviors present in the training data. However, they rely on extensive paired simulation datasets and their generalization ability is strongly tied to dataset coverage and diversity.

Self-Supervised Garment Simulation Self-supervised garment simulators remove the dependency on data by encoding physical priors into energy-based loss functions. PBNS [5] introduced the first self-supervised framework, formulating training as minimizing internal and external energy terms. Subsequent works improved architectural design and loss formulations: SNUG [33] incorporated inertial effects, Neural Cloth Simulation [6] decoupled static and dynamic feature learning, HOOD [14] modeled local garment-body interactions with graph networks, ContourCraft [13] introduced contour-based interaction losses, SENC [23] addressed self-collision through global intersection analysis, ESLR-Sim [24] enhanced long-range information propagation, while Pb4U-GNet [25] addressed cross-resolution generalization. Despite the advances, these methods cannot simulate natural persistent wrinkles, which limits their visual realism. Our work introduces the first self-supervised neural garment simulator that generates natural persistent wrinkles.

Wrinkle Simulation Real-world cloth develops wrinkles when deformed, and realistically simulating wrinkles is essential for visual plausibility [4]. Realistic wrinkle simulation has long been a goal in computer graphics. Cloth wrinkles are commonly categorized as dynamic or static [20]. Dynamic wrinkles refer to transient folds that form and disappear during motion. In contrast, static wrinkles are permanent deformations that persist even after the cloth is no longer under load. Early efforts in simulating dynamic wrinkles employed geometric models to define cloth fold propagation [2]. To achieve greater physical realism, later research integrated physics into wrinkle formation modeling [7].

More recently, data-driven approaches have learned wrinkles from data [38] or use generative models to add folds to garments [19]. In contrast, research on simulating static wrinkles has been underexplored. To model permanent wrinkles, [29] employed a hardening plastic model. [28] used the Dahl friction model to capture hysteresis and simulate unrecoverable wrinkles. More recently, [11] introduced a time-dependent friction and plastic model to simulate time-dependent wrinkles. Our work focuses on static wrinkle simulation. Unlike prior work, our model is a self-supervised neural simulator that enables fast inference without requiring training data.

3 Method

Our simulator is designed for simulating on-body garments. It models a garment as a thin-shell which is discretized into a triangle mesh of N vertices and D edges. Its state $\mathcal{S}$ consists of vertex positions $\mathbf{x} \in \mathbb{R}^{3N}$ and velocities $\dot{\mathbf{x}} \in \mathbb{R}^{3N}$. Garment motions are denoted by $\{\mathcal{S}^{(t)} | t \in \mathbb{Z}, t \in [0, T]\}$, where $\mathcal{S}^{(t)}$ is the garment state at time step t. For the body skeleton with j joints, we denote its state by $\mathcal{B} \in \mathbb{R}^{9j}$ where each joint has a feature vector in $\mathbb{R}^9$ denoting its location and orientation [6]. To represent the human body surface with respect the skeleton, we use SMPL [26].

To simulate a garment, our simulator predicts the future garment states, given its initial state and a body motion: $\mathcal{S}^{(1:T)} = f_{sim}(\mathcal{S}^{(0)}, \mathcal{B}^{(0:T)}; \boldsymbol{\Theta})$ where f_{sim} denotes our self-supervised garment simulator with learnable parameter $\boldsymbol{\Theta}$. $\boldsymbol{\Theta}$ is learned by minimizing a physics-based loss function:

$$\mathcal{L} = \sum_{t}^{T} \frac{h^2}{2} \Big(\frac{\frac{\mathbf{x}^{(t+1)} - \mathbf{x}^{(t)}}{h} - \dot{\mathbf{x}}^{(t)}}{h}\Big)^{\top} \mathbf{M} \Big(\frac{\frac{\mathbf{x}^{(t+1)} - \mathbf{x}^{(t)}}{h} - \dot{\mathbf{x}}^{(t)}}{h}\Big) + W^{(t)} \tag{1}$$

where $\mathbf{M}$ is the garment lumped mass matrix, W denotes the potential energy due to, *e.g.*, gravity, garment stretching energy, bending energy, *etc.*, and h is the time step size. This minimization is equivalent to using implicit Euler [3] to solve the equation defined by the Newton's Second law: $\mathbf{M}\ddot{\mathbf{x}} = \mathbf{f} = -\nabla W$, where $\mathbf{f}$ is the resultant force imposed on the garment [27]. The potential energy W contains several terms:

$$W = w_b W_{bend} + w_{st} W_{stretch} + w_{sh} W_{shear} + w_c W_{collision} + W_{gravity} \tag{2}$$

where W_{bend}, $W_{stretch}$, W_{shear}, $W_{collision}$, and $W_{gravity}$ denote the bending, stretching, shear, collision, and gravitational energy terms. Among them, W_{bend} determines the bending mechanics which dictates the persistent wrinkle formation. Therefore W_{bend} is the key.

3.1 Physics-inspired Bending Energy for Wrinkles

In physics-based garment simulations, a common way to simulate persistent wrinkles is modeling garments as elasto-plastic material. In this case, the total bending is decomposed into elastic and plastic deformation: $\varepsilon = \varepsilon_e + \varepsilon_p$, where ε, ε_e and ε_p denote the total bending strain, elastic bending strain, and plastic bending strain, respectively. The elastic bending stress always tends to keep the garment in the state where $\varepsilon_e = 0$. Therefore, when $\varepsilon_p = 0$, the elastic bending stress keeps the garment in the shape with zero strain. ε_p is only updated when ε_e becomes greater than the yield strain, *i.e.*, $|\varepsilon_e| > \varepsilon_y$ where ε_y is the yield strain serving as a threshold.

ε_p is the RB in existing methods which is predefined and static. Unlike existing methods, our RB is self-supervisedly learned and dynamic, by adopting the perfect plastic model [30] where the part of the elastic strain exceeding the yield strain is immediately converted to the plastic strain:

$$\varepsilon_p \to \varepsilon_p + \mathrm{sign}(\varepsilon_e)(|\varepsilon_e| - \varepsilon_y) \tag{3}$$

Note ε_p can change over time and is a function of the past plastic strains for an edge. In practice, we compute Eq. (3) by:

$$\Delta\varepsilon^{(t)} = \varepsilon^{(t)} - \varepsilon_p^{(t-1)} - \varepsilon_y \tag{4}$$

$$\varepsilon_p^{(t)} = \varepsilon_p^{(t-1)} + \mathrm{sigmoid}(k_p \Delta\varepsilon^{(t)})\Delta\varepsilon^{(t)} \tag{5}$$

where $\varepsilon^{(t)}$ is the total strain of an edge at time t. $\varepsilon_p^{(t)}$ is the plastic strain at time t. k_p is a coefficient. Finally, our bending energy is computed as:

$$W_{bend} = \frac{1}{TD}\sum_{t}^{T}\sum_{d}^{D} k_b \frac{l^2}{8a}(\varepsilon_d^{(t)} - \varepsilon_{p,d}^{(t)})^2 \tag{6}$$

where k_b is the bending stiffness, l is the length of the edge, a is the sums of the areas of the two triangles, d is the edge index. $\varepsilon_d^{(t)}$ and $\varepsilon_{p,d}^{(t)}$ are the total strain and the RB at time t for edge d. $\varepsilon_{p,d}^{(t)}$ is defined as the bending angle at which the bending energy is zero, corresponding to the dihedral angle between two adjacent triangles.

The introduction of the new W_{bend} creates a chicken-egg problem due to that the RB $\varepsilon_{p,d}^{(t)}$ is unknown. In previous methods, $\varepsilon_{d}^{(t)}$ is predicted by the model and $\varepsilon_{p,d}^{(t)}$ is predefined, so the learning is a fixed energy minimization problem. However, in our method, $\varepsilon_{p,d}^{(t)}$ needs to be computed based on Eqs. (4) and (5) which needs $\varepsilon_{d}^{(t)}$, but the model needs to be trained well to predict $\varepsilon_{d}^{(t)}$. The inter-dependence between $\varepsilon_{d}^{(t)}$ and $\varepsilon_{p,d}^{(t)}$ requires us to solve a moving energy minimization problem.

3.2 Physics-inspired Curriculum Learning

To solve this chicken-egg problem, we introduce two networks: an Elastic Network (E-Net) and a Plastic Network (P-Net). E-Net simulates garment elastic dynamics, and P-Net predicts the evolving RB to model the plastic deformations formed in motions.

$$\mathcal{S}^{(1:T)} = \text{E-Net}(\mathcal{S}^{(0)}, \mathcal{B}^{(0:T)}, \varepsilon_{p,d}^{(0:T)}); \quad \varepsilon_{p,d}^{(t)} = \text{P-Net}(\mathcal{S}^{(t-1)}, \varepsilon_{p,d}^{(t-1)}) \tag{7}$$

E-Net and P-Net are mutually dependent: the $\varepsilon_{p,d}^{(0:T)}$ of the E-Net in Eq. (7) dependents on output of P-Net, and the input of the P-Net, $\varepsilon_{p,d}^{(t-1)}$, depends on the output of E-Net. Theoretically, to train our simulator, we could initialize both E-Net and P-Net and rely on optimization to lead to a convergence. In practice, the convergence proves to be difficult. So we break the tie, and train E-Net and P-Net alternatively. Furthermore, since we need to enforce Eqs. (4) and (5) during training, *i.e.*, the P-Net prediction needs to satisfy both equations between frames, the design of the alternative training scheme needs special care. To this end, inspired by curriculum learning [41], we design a curriculum where the physical behaviors of the garment change from simple to complex, which corresponds to changing from purely elastic to elasto-plastic material.

We denote the current iteration between E-Net and P-Net by another superscript i and represent the RB of an edge in time t as $\varepsilon_{p}^{(t,i)}$. We keep both t and i to distinguish between the frame and the training iteration. The whole curriculum learning is shown in Algorithm 1. At the high level, we first use a near-zero RB to train E-Net as if the garment is purely elastic. Then we update the RB from the E-Net prediction via Eqs. (4) and (5). Next, the updated RB is used to train P-Net as the target RB. The training is done by minimizing the mean squared error between the predicted RB by P-Net and the target RB. Then the whole process repeats itself until convergence. As the RB accumulates during training when large deformation is present, the material gradually changes from purely elastic to elasto-plastic.

Algorithm 1 Curriculum Learning of Self-supervised Elasto-Plastic Material

Initialize $\varepsilon_p^{(t,0)}, \varepsilon_p \sim \mathbf{N}(\mathbf{0}, \sigma)$, $\varepsilon_y =$ *user defined*
for $i = 1, 2, \ldots, I$ **do**
    Train E-Net via Eq. (1) with ε_p as the RB
    Predict $\varepsilon^{(t)}$ for every edge using E-Net
    for $t = 1, 2, \ldots, T$ **do**
        Compute $\varepsilon_p^{(t,i)}$ for every edge via Eqs. (4) and (5).
    end for
    $\varepsilon_p \leftarrow \varepsilon_p^{(t,i)}$
    Train P-Net with ε_p as the target RB
    Predict ε_p' using P-Net
    $\varepsilon_p \leftarrow \varepsilon_p'$
end for
return trained model parameters $\boldsymbol{\Theta}$

3.3 Neural Elastic and Plastic Model

Our novel curriculum learning method is highly flexible and imposes no architectural restrictions on the design of E-Net and P-Net. While the specific instantiations of E-Net and P-Net used in this work are inspired by prior self-supervised simulators for their simplicity and ability to capture garment mechanics, our learning method is theoretically architecture-agnostic. It can be used in a plug-and-play manner to simulate elasto-plastic garments that exhibit persistent wrinkles.

Elastic Net Similar to [6], we adopt a recurrent encoder-decoder to model body motion in two subspaces: static subspace and dynamic subspace, shown in Fig. 2. We use the Static Encoder to encode the body static pose and the Dynamic Encoder to handle body motions. The outputs of the two encoders are combined and concatenated with the RB, which is predicted by P-Net (on the first training iteration, RB is initialized to a near-zero value). The combined features are then fed into the Decoder to predict garment deformations for all frames. The predicted deformations are applied through skinning to fit the garment onto the body. Based on the simulated deformation, RB is updated by using Eqs. (4) and (5), which is used as the training target RB for P-Net.

Plastic Net P-Net adopts an encode–process–decode message-passing architecture [14] defined on an intrinsic hinge graph. Its goal is to predict the evolving RB, which captures the accumulated plastic deformation in the garment. Persistent wrinkles exhibit both spatial and temporal correlations: spatially, wrinkles propagate across neighboring mesh regions, while temporally plastic deformation accumulates over successive bending motions. We model these correlations using a hinge-graph message-passing network with incremental updates over time.

To capture spatial correlations of plastic deformation, we construct a hinge graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ derived from the garment mesh. Each node $v \in \mathcal{V}$ corresponds to an interior mesh hinge (a pair of adjacent triangles sharing an edge), and edges $(v_i, v_j) \in \mathcal{E}$ represent adjacency between neighboring hinges. Bending angles are

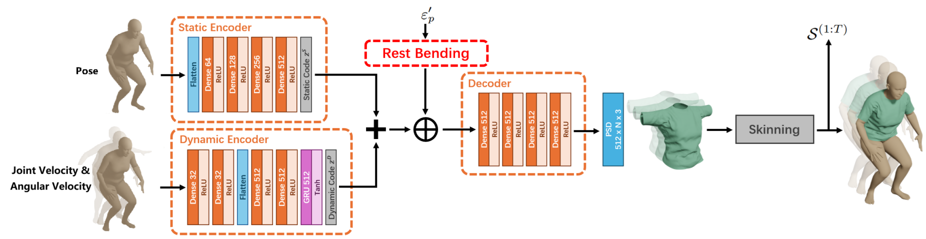


Fig. 2: Elastic Net (E-Net): The input consists of a pose and its first-order time derivative. After passing through the encoder, the RB predicted by P-Net is used as a conditional input to the decoder, which then predicts the final garment deformation $\mathcal{S}^{(1:T)}$. Based on the predicted deformation, the RB is computed and subsequently used for P-Net training.

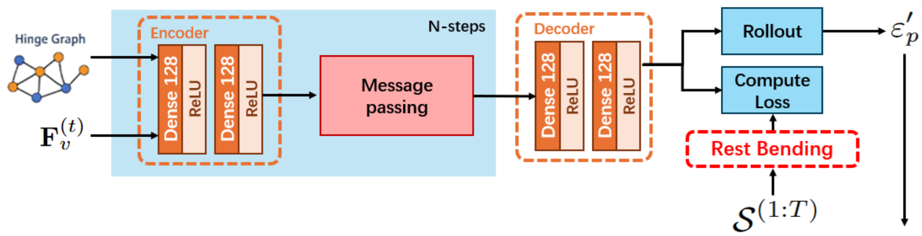


Fig. 3: Plastic Net (P-Net): Hinge-level node and edge features at time step t are first embedded by MLP encoders. We then apply 3 message-passing steps, where edge features are updated using an edge MLP $f_{v\to e}(e_{ij}, v_i, v_j)$ and node features are updated using a node MLP $f_{e\to v}(v_i, \text{mean}_j(e_{ij}))$. After decoding, the Rest Bending (RB) computed by Eqs. (4) and (5) from the E-Net predicted deformation $\mathcal{S}^{(1:T)}$ as the training target for learning. By rolling out P-Net over time produces the predicted ε'_p which is used as RB for condition E-Net.

defined per hinge, and message passing is performed on this intrinsic hinge graph. As shown in Fig. 3, at each time step P-Net takes hinge-level node and edge features $\mathbf{F}_v^{(t)}$ derived from local geometry (deformed and rest configurations), timestep information, and material parameters. These features are embedded by MLP encoders and processed using three steps of message passing. The MLP decoder outputs a scalar increment for each hinge, representing the predicted update of the RB.

Persistent wrinkles form when the bending along an edge exceeds a certain threshold, establishing a non-zero RB. Once formed, the fabric tends to return to this RB configuration rather than the flat state. If further deformation occurs and exceeds another threshold, RB increases and the wrinkle becomes sharper. Consequently, the RB of each hinge depends on its previously accumulated state. To model this temporal evolution, P-Net predicts incremental plastic updates at each time step (via Eqs. (4) and (5)). Rolling out these updates over time produces the predicted RB sequence, which is then used to condition E-Net during the next training iteration. Unlike existing self-supervised garment simulators, which primarily model transient garment dynamics through recurrent state propagation, our method explicitly maintains and updates a persistent rest-bending state, enabling the accumulation of irreversible wrinkle effects over long-term deformation histories.

4 Experiments

Data We use the widely employed AMASS dataset [1], which includes the SMPL body model, the MANO hand model (52 joints, 6,890 vertices), and 252 motion sequences with approximately $188,928$ poses. Following the same setting of existing methods [6], we use 90% data for training, 10% for testing. Before training, we extract the joints for representing body poses [6].
Baselines We employ a PBS with a perfect plastic model [29] as the ground-truth. To highlight our model's unique ability to simulate persistent wrinkles, we also compare our model with the state-of-the-art self-supervised neural cloth simulators: PBNS [5], SNUG [33], NCS [6], HOOD [14], and SENC [13].
Evaluation Motions Since there is no ground-truth in self-supervised learning, we employ PBS with plasticity as reference for evaluation. However, we found that not all motions in AMASS are suitable for evaluation. To ensure meaningful wrinkle assessment, we exclude motions that produce negligible deformation which does not lead to persistent wrinkles. We also exclude motions that produce unresolvable penetrations in PBS. At the end, we choose six representative motions (bending waist, ascending stairs, dancing, dodging, jumping, raising hand) for physical simulation to generate reference garments.
Metrics We adopt Bending Error Distribution (BED), Plastic Error (PE), Bending Error (BE), Mean Euclidean Distance (MED), Chamfer Distance (CD), Bending Energy (BEN) for comparison (all lower is better), and garment-body collision (COL). BED is computed on the top 10% edges selected by Segmented Maximum Analysis based on peak plastic bending over the sequence, focusing the evaluation on wrinkle-relevant regions. PE is the per-edge L1 error of the ε_p' predicted by P-Net. Note that PBNS, SNUG, NCS, HOOD, and SENC do not have RB so their RB is effectively zero. BE is per-edge L1 error of the predicted ε from E-Net. MED and CD is the mean Euclidean distance and Chamfer Distance between the predicted vertex locations and the ground-truth. BEN is the mean per-edge bending energy. COL is the mean garment-body penetration depth. Full metric definitions are provided in the supplementary material.

4.1 Persistent Wrinkles on Unseen Motions

We show a visual comparison in Fig. 1. Despite all method generate wrinkles, our method is the most similar to the PBS, especially on the front of the t-shirt. PBNS, HOOD, and SENC generates few wrinkles. SNUG and NCS generate sharp wrinkles but in the wrong places. Most of the sharp persistent wrinkles should be around the waist as the motion is waist bending as shown in Fig. 1 (a). Furthermore, our method captures the causality clearly. Before bending, wrinkles should not appear around the abdomen area while sharp persistent wrinkles

Table 1: Numerical comparison on unseen motions. Metrics are averaged across all testing motions.

Base/Mtr	BED(rad)↓	PE(rad)↓	BE(rad)↓	MED(m)↓	CD(m^2)↓	BEN↓	COL(m)↓
PBNS	0.60223	0.12506	0.18733	0.04038	0.00054	0.04565	**0.00095**
SNUG	0.64567	0.12506	0.20327	0.03191	0.00061	0.05976	0.00136
NCS	0.76556	0.12506	0.17611	0.03210	0.00040	0.03412	0.00129
HOOD	0.53553	0.12506	0.20948	0.06081	0.00118	0.13150	0.00210
SENC	0.52529	0.12506	0.19785	0.05486	0.00074	0.13657	0.00160
Ours	**0.28714**	**0.03206**	**0.17118**	**0.02355**	**0.00027**	**0.02999**	0.00129

should appear after the bending. However, PBNS, SNUG, NCS, HOOD, and SENC do not show this causality and generate wrinkles even before bending.

We further show numerical comparison in Tab. 1. The smaller BED, PE, BE, and BEN directly demonstrate that our method is closer to PBS when forming persistent wrinkles. The learned plasticity reduces the potential energy stored in the deformation hence lower energy and error. This also leads to more accurate geometries indicated by smaller MED and CD. Our method is universally better across metrics. For each metric, comparing with the best-performing baseline, our approach improves the performance by 45.34% in BED, 74.36% in PE, 2.79% in BE, 26.19% in MED, 32.50% in CD, and 12.10% in BEN, while achieving the second-best COL, indicating that the improved wrinkle modeling does not compromise collision handling.

4.2 Wrinkles on Unseen Bodies and Motions

Once trained, our P-Net can generalize to different bodies, as it can predict the RB based on motions and garment itself. Fig. 4 compares the wrinkles simulated by our model and PBS when the t-shirt is on three different bodies: Normal, Slim, and Obese, where the latter two are not used in training. Also, all the motions are unseen at the training stage either. Trained well on the Normal body Fig. 4 (b), the wrinkles on the Slim and Obese body (Fig. 4 (c-f)) also look similar to the PBS results. These wrinkles are largely in the armpits, abdomen, and chest area highlighted by the red rectangles.

On the high level, the generalization of P-Net stems from the fact that it learns to predict the RB based on the garment hinge-graph GNN, the initial deformation, and the body motion. The initial deformation is influenced by the initial body shape. But this is somewhat counter-intuitive as wrinkle formation should be heavily influenced by the body deformation too, rather than the motion and the initial body shape only. However, in our method, the body deformation is handled by the E-Net. In other words, P-Net predicts the likely plasticity for the whole motion and E-Net adds elasticity caused by the body deformation.

Tab. 2 shows the quantitative results on unseen body shapes. Our method achieves the best performance across all evaluated metrics, demonstrating strong

Fig. 4: Visual comparison between our simulator and PBS across unseen bodies and motions: slim, and obese. Trained on the normal body, our simulator can generate wrinkles that are closely match PBS (a,b). Furthermore, it can generate plausible wrinkles on the slim and obese body which are unseen in training. The positions of the formed wrinkles are similar to PBS as well (c,d,e,f).

Table 2: Numerical comparison on unseen bodies. Metric are averages across all bodies.

Base/Mtr	BED(rad)↓	PE(rad)↓	BE(rad)↓	MED(m)↓	CD(m^2)↓	BEN↓	COL(m)↓
PBNS	0.66252	0.14416	0.20414	0.05169	0.00078	0.03053	**0.00626**
SNUG	0.66125	0.14416	0.21759	0.02773	0.00040	0.03657	0.00773
NCS	0.74892	0.14416	0.19755	0.03444	0.00032	0.02778	0.00756
HOOD	0.69630	0.14416	0.21796	0.04612	0.00090	0.07248	0.00654
SENC	0.87741	0.14416	0.19979	0.03686	0.00066	0.05565	0.00652
Ours	**0.46630**	**0.09690**	**0.19593**	**0.02402**	**0.00024**	**0.02252**	0.00754

generalization beyond the body shapes used during training. For each metric, comparing with the best-performing baseline, our approach improves the performance by 29.48% in BED, 32.78% in PE, 0.82% in BE, 13.38% in MED, 25.00% in CD, and 18.93% in BEN, while maintaining competitive COL performance. In particular, we obtain the lowest BED, BE, and BEN, indicating that our predicted bending behavior and wrinkle distribution align most closely with the PBS under body-shape variation.

Despite the overall good results, we still notice visual differences between our results and PBS. The primary limitation stems from the deformation pipeline: our method, similar to prior self-supervised simulators such as NCS and SNUG, relies on skinning-based garment attachment to the body. As a result, garment deformation is implicitly constrained to follow the body surface, leading to tight fitting behaviors. Although gravity is included as an energy term in

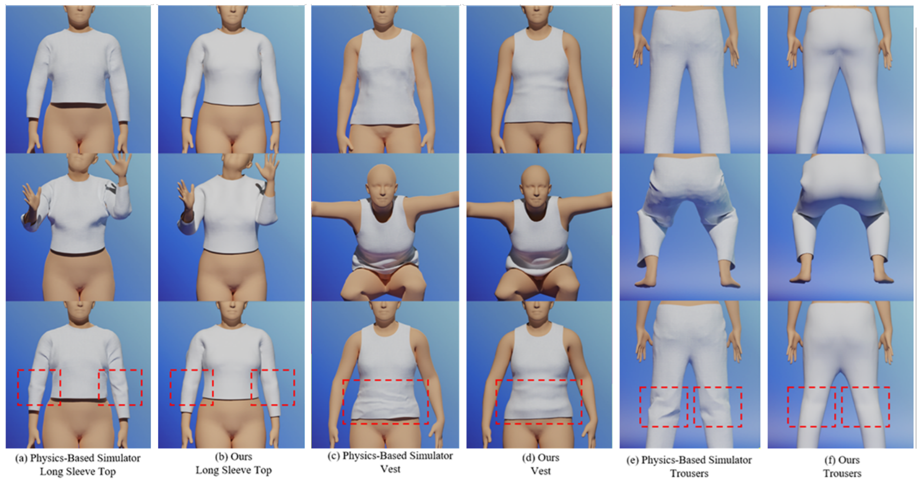


Fig. 5: A comparison of wrinkle patterns across different garment types. (a), (c), and (e) present the PBS simulation results for the long-sleeve top, vest, and T-shirt, respectively, whereas (b), (d), and (f) show the corresponding predictions generated by our model.

these methods, gravity alone is insufficient to produce realistic loose draping. In the absence of explicit contact resolution and inequality-based collision constraints, the energy-based formulation cannot create stable garment–body separation. The garment is encouraged to minimize penetration, but it is not allowed to detach and re-drape freely under gravity. In contrast, PBS explicitly solves body–garment collisions and contact forces at each time step, enabling garments to be lifted, separated from the body, and naturally draping under gravity.

4.3 Persistent Wrinkles on Unseen Garments

We further evaluate our method on unseen garment types, including long-sleeve top, vest, and trousers. Although P-Net is trained only on the t-shirt, it generalizes effectively to new garment geometries. As shown in Fig. 5, our method produces wrinkle distributions (b, d, f) that closely resemble those generated by the PBS (a, c, e), demonstrating visually consistent persistent wrinkle formation across different garment structures.

The quantitative comparison in Tab. 3 further supports this observation. Our method achieves the best performance across all reported metrics. For each metric, comparing with the best-performing baseline, our approach improves the performance by 5.13% in BED, 36.23% in PE, 12.94% in BE, 5.63% in MED, 20.00% in CD, and 12.65% in BEN, without degrading COL performance. The reduced PE confirms that the learned plastic strain evolution transfers across mesh topologies, while the improvements in bending and geometric errors indicate more accurate wrinkle localization and overall deformation fidelity.

Table 3: Numerical comparison on unseen garments. Metrics are averaged across all testing garments.

Base/Mtr	BED(rad)↓	PE(rad)↓	BE(rad)↓	MED(m)↓	CD(m^2)↓	BEN↓	COL(m)↓
PBNS	0.68964	0.17022	0.22727	0.03946	0.00061	0.04194	**0.00022**
SNUG	0.55103	0.17022	0.20903	0.03447	0.00068	0.07346	0.00255
NCS	0.70959	0.17022	0.18726	0.03488	0.00060	0.03454	0.00241
HOOD	0.45466	0.17022	0.21034	0.05640	0.00102	0.15023	0.00353
SENC	0.44631	0.17022	0.20987	0.05788	0.00101	0.16064	0.00114
Ours	**0.42340**	**0.10855**	**0.16303**	**0.03253**	**0.00048**	**0.03017**	0.00237

These results suggest that modeling plasticity on an intrinsic hinge graph enables topology-aware generalization, allowing the simulator to adapt to diverse garment configurations without relying on garment-specific templates.

4.4 Simulating User-specified Materials

An important property of garment simulators is the ability to control material behavior. In physics-based methods, this is achieved by directly adjusting constitutive parameters, whereas existing self-supervised neural simulators mainly tune the weights of elastic energy terms. In contrast, our formulation explicitly models plasticity and introduces a physically meaningful yield threshold ε_y in Eq. (4), which controls the onset of plastic deformation. A smaller ε_y makes plastic accumulation easier, leading to more pronounced and persistent wrinkles. As shown in Fig. 6, setting ε_y to 30°, 60°, and 90° progressively increases the resistance to wrinkle formation under the same motion. When $\varepsilon_y = 30°$, plastic deformation is triggered early, resulting in clearly visible persistent folds. In contrast, when $\varepsilon_y = 90°$, significantly larger bending is required before plastic accumulation occurs, and the garment behaves more elastically. Importantly, this qualitative change in wrinkle persistence cannot be reproduced by merely adjusting other parameters such as the bending stiffness k_b or the plastic gain k_p, which only affect stiffness or update smoothness but do not alter the yield mechanism itself. Notably, even when $\varepsilon_y = 30°$, the accumulated RB may exceed 30°, as the threshold controls the onset of plasticity rather than its upper bound.

4.5 Ablation Study

P-Net is the core component responsible for modeling the dynamics of the evolving RB. To justify our architectural design, we conduct ablation studies comparing different alternatives for spatial–temporal modeling of the internal plastic strain. Specifically, we evaluate GCN plus recurrent neural networks (RNN), long short-term memory (LSTM), Transformer, and a hinge-graph-based GNN with recurrent message passing. As shown in Tab. 4, the Transformer achieves competitive accuracy compared to the proposed GNN, particularly on geometric

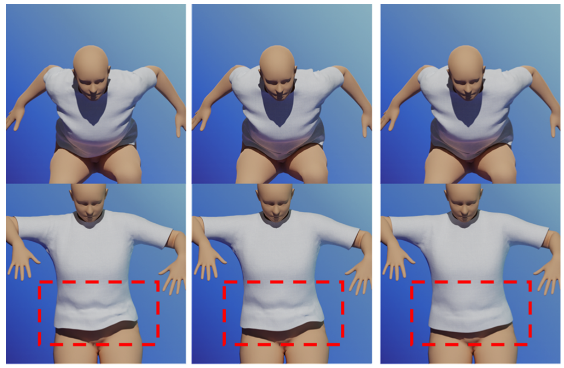


Fig. 6: Our simulator allows users to tweak the fabric plastic properties. (a) Fabric tends to have wrinkles. By setting ε_y to 30°, the simulated garment, like paper, is very likely to form wrinkles after a bending motion. (b) Fabric is less likely to have wrinkles. After increasing ε_y to 60°, the garment forms fewer wrinkles. (c) Fabric is not likely to have wrinkles. As ε_y is set to 90°, the garment becomes more wrinkle-resistant.

Table 4: Ablation Study. Alternative networks for learning the RB dynamics.

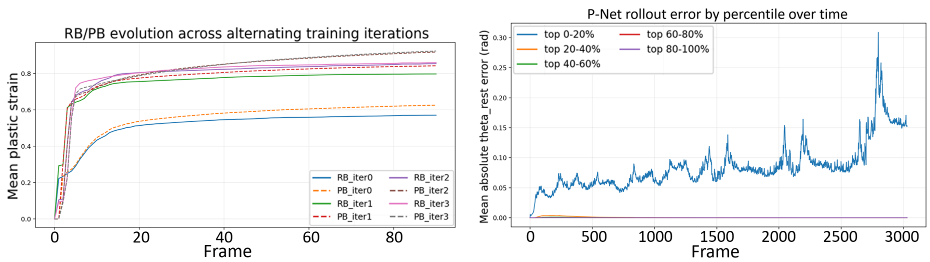
Base/Mtr	BED(rad)↓	PE(rad)↓	BE(rad)↓	MED(m)↓	CD(m^2)↓	BEN↓	GPU(GB) ↓	FPS↑
RNN	0.86971	0.09968	**0.17048**	0.03274	0.00043	**0.02937**	9.1	1.40
LSTM	0.84272	0.07535	0.17149	0.03251	0.00042	0.02940	8.5	1.61
Transformer	0.52095	0.05263	0.17482	0.02460	0.00029	0.03027	9.2	0.17
GNN	**0.28714**	**0.03206**	0.17118	**0.02355**	**0.00027**	0.02999	**0.6**	**9.43**

metrics. However, it incurs substantially higher computational cost and significantly lower inference speed. In contrast, the hinge-graph GNN with message passing not only achieves the best overall performance on key metrics, but also requires dramatically less GPU memory and delivers an order-of-magnitude improvement in frames per second (FPS). These results indicate that hinge-level message passing is both more efficient and better aligned with the intrinsic mesh structure of plastic strain update. Therefore, we adopt the GNN with message passing as the default architecture for P-Net.

Our model is trained using 4 alternating training cycles. As shown in Fig. 8a, curriculum learning gradually evolves the plastic state from near-elastic behavior toward a converged elasto-plastic solution, and from Fig. 7, we observe that increasing alternating cycles beyond four provides small improvement.

We additionally tested 10 randomly selected long motion sequences. These sequences were randomly sampled from the evaluation set. At each frame, samples are ranked by error and divided into five percentile groups (top 0–20% to top 80–100%). As shown in Fig. 8b, the average top-20% error remains around 0.05 for the first 700 frames and around 0.1 up to approximately 2500 frames, indicating stable long-term generation. Significant drift only appears after ex-

Fig. 7: Convergence behavior under different alternating cycle settings.



(a) (b)

Fig. 8: Training dynamics and long-term stability of our method. (a) Rest Bending (RB) and Predicted Rest Bending (PB) evolution across alternating training iterations. (b) Long-term autoregressive error over rollout frames.

tremely long horizons (∼3000 frames). Most edges maintain low prediction error throughout the rollout.

5 Discussions and Conclusions

We have proposed the first self-supervised neural garment simulator that is capable of simulating persistent wrinkles. We model garments as an elasto-plastic material and design two networks, E-Net and P-Net, correspondingly. P-Net predicts the RB, enabling E-Net to keep the garment in non-flat state and generating persistent wrinkles. We have also proposed a novel curriculum learning method to address the difficulty in learning caused by the moving energy minimization problem. This curriculum learning strategy is physics-inspired and achieves convergence with a few cycles. We have demonstrated that our simulator can more closely mimic physics-based simulators in persistent wrinkle formation in unseen motions, bodies, and garments.

Our simulator has limitations. Our model cannot simulate wrinkles being flattened by stretching. Moreover, we do not evaluate on loose or multi-layered garments, which are orthogonal to our research. In future work, we will address these limitations. Moreover, we will incorporate other existing SSL models as instantiation of our E-Net, to make our model a plug-and-play component.

6 Acknowledge

This work was partially funded by the HKSAR ITC under ITSP-Platform grants (Ref: ITS/335/23FP, ITS/469/24FP).

References

1. Advanced Computing Center for the Arts and Design: ACCAD MoCap Dataset, https://accad.osu.edu/research/motion-lab/mocap-system-and-data
2. Aono, M.: A wrinkle propagation model for cloth. In: CG International'90: Computer Graphics Around the World, pp. 95–115. Springer (1990)
3. Baraff, D., Witkin, A.: Large steps in cloth simulation. In: Seminal Graphics Papers: Pushing the Boundaries, Volume 2, pp. 767–778 (2023)
4. Benusiglio, A., Mansard, V., Biance, A.L., Bocquet, L.: The anatomy of a crease, from folding to ironing. Soft Matter **8**(12), 3342–3347 (2012)
5. Bertiche, H., Madadi, M., Escalera, S.: Pbns: physically based neural simulation for unsupervised garment pose space deformation. ACM Trans. Graph. **40**(6) (Dec 2021)
6. Bertiche, H., Madadi, M., Escalera, S.: Neural cloth simulation. ACM Transactions on Graphics (TOG) **41**(6), 1–14 (2022)
7. Bridson, R., Marino, S., Fedkiw, R.: Simulation of clothing with folds and wrinkles. In: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation. pp. 28–36 (2003)
8. Cirio, G., Lopez-Moreno, J., Miraut, D., Otaduy, M.A.: Yarn-level simulation of woven cloth. ACM Transactions on Graphics (TOG) **33**(6), 1–11 (2014)
9. Cirio, G., Lopez-Moreno, J., Otaduy, M.A.: Yarn-level cloth simulation with sliding persistent contacts. IEEE transactions on visualization and computer graphics **23**(2), 1152–1162 (2016)
10. Gong, D., Mao, N., Wang, H.: Bayesian differentiable physics for cloth digitalization. In: The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2024)
11. Gong, D., Yang, Y., Shao, T., Wang, H.: Cloth animation with time-dependent persistent wrinkles. In: Computer Graphics Forum. p. e70031. Wiley Online Library (2025)
12. Gong, D., Zhu, Z., Bulpitt, A.J., Wang, H.: Fine-grained differentiable physics: a yarn-level model for fabrics. In: The International Conference on Learning Representations (ICLR) (2022)
13. Grigorev, A., Becherini, G., Black, M., Hilliges, O., Thomaszewski, B.: Contourcraft: Learning to resolve intersections in neural multi-garment simulations. In: ACM SIGGRAPH 2024 conference papers. pp. 1–10 (2024)
14. Grigorev, A., Black, M.J., Hilliges, O.: Hood: Hierarchical graphs for generalized modelling of clothing dynamics. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 16965–16974 (2023)
15. Grinspun, E., Hirani, A.N., Desbrun, M., Schröder, P.: Discrete shells. In: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation. pp. 62–67 (2003)
16. Gundogdu, E., Constantin, V., Parashar, S., Seifoddini, A., Dang, M., Salzmann, M., Fua, P.: Garnet++: Improving fast and accurate static 3d cloth draping by curvature loss. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(1), 181–195 (2020)
17. Gundogdu, E., Constantin, V., Seifoddini, A., Dang, M., Salzmann, M., Fua, P.: Garnet: A two-stream network for fast and accurate 3d cloth draping. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 8739–8748 (2019)

18. Kaldor, J.M., James, D.L., Marschner, S.: Simulating knitted cloth at the yarn level. ACM Transactions on Graphics (TOG) **27**(3), 1–9 (2008)
19. Lahner, Z., Cremers, D., Tung, T.: Deepwrinkles: Accurate and realistic clothing modeling. In: Proceedings of the European conference on computer vision (ECCV). pp. 667–684 (2018)
20. Larboulette, C., Cani, M.P.: Real-time dynamic wrinkles. In: Proceedings Computer Graphics International, 2004. pp. 522–525. IEEE (2004)
21. Li, Y., Du, T., Wu, K., Xu, J., Matusik, W.: Diffcloth: Differentiable cloth simulation with dry frictional contact. ACM Transactions on Graphics (TOG) **42**(1), 1–20 (2022)
22. Liang, J., Lin, M., Koltun, V.: Differentiable cloth simulation for inverse problems. Advances in neural information processing systems **32** (2019)
23. Liao, Z., Wang, S., Komura, T.: Senc: Handling self-collision in neural cloth simulation. In: European Conference on Computer Vision. pp. 385–402. Springer (2024)
24. Liu, A., Hu, K., Mo, C.A., Li, C., Wang, Z.: Extended short- and long-range mesh learning for fast and generalised garment simulation. 2025 IEEE International Conference on Multimedia and Expo (ICME) pp. 1–6 (2025)
25. Liu, A., Hu, K., Mo, C.A., Wu, Q., Kang, W., Wang, Z.: Pb4u-gnet: Resolution-adaptive garment simulation via propagation-before-update graph network. In: Proceedings of the AAAI Conference on Artificial Intelligence (AAAI). vol. 40, pp. 7060–7068 (2026)
26. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: Smpl: a skinned multi-person linear model. ACM Transactions on Graphics (TOG) **34**(6), 1–16 (2015)
27. Martin, S., Thomaszewski, B., Grinspun, E., Gross, M.: Example-based elastic materials. ACM Transactions on Graphics (TOG) **30**(4), 1–8 (2011)
28. Miguel, E., Tamstorf, R., Bradley, D., Schvartzman, S.C., Thomaszewski, B., Bickel, B., Matusik, W., Marschner, S., Otaduy, M.A.: Modeling and estimation of internal friction in cloth. ACM Transactions on Graphics (TOG) **32**(6), 1–10 (2013)
29. Narain, R., Pfaff, T., O'Brien, J.F.: Folding and crumpling adaptive sheets. ACM Transactions on Graphics (TOG) **32**(4), 1–8 (2013)
30. O'Brien, J.F., Bargteil, A.W., Hodgins, J.K.: Graphical modeling and animation of ductile fracture. ACM Transactions on Graphics **21**(3), 291–294 (2002)
31. Patel, C., Liao, Z., Pons-Moll, G.: Tailornet: Predicting clothing in 3d as a function of human pose, shape and garment style. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7365–7375 (2020)
32. Santesteban, I., Otaduy, M.A., Casas, D.: Learning-based animation of clothing for virtual try-on. In: Computer Graphics Forum. vol. 38, pp. 355–366. Wiley Online Library (2019)
33. Santesteban, I., Otaduy, M.A., Casas, D.: Snug: Self-supervised neural dynamic garments. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8140–8150 (2022)
34. Stuyck, T.: Cloth simulation for computer graphics. Springer Nature (2022)
35. Terzopoulos, D., Platt, J., Barr, A., Fleischer, K.: Elastically deformable models. In: Proceedings of the 14th annual conference on Computer graphics and interactive techniques. pp. 205–214 (1987)
36. Tiwari, L., Bhowmick, B.: Deepdraper: Fast and accurate 3d garment draping over a 3d human body. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1416–1426 (2021)

37. Wang, H.: Gpu-based simulation of cloth wrinkles at submillimeter levels. ACM Transactions on Graphics (TOG) **40**(4), 1–14 (2021)
38. Wang, H., Hecht, F., Ramamoorthi, R., O'Brien, J.F.: Example-based wrinkle synthesis for clothing animation. ACM Transactions on Graphics (TOG) **29**(4), 1–8 (2010)
39. Wang, H., O'Brien, J.F., Ramamoorthi, R.: Data-driven elastic models for cloth: modeling and measurement. ACM transactions on graphics (TOG) **30**(4), 1–12 (2011)
40. Wang, T.Y., Shao, T., Fu, K., Mitra, N.J.: Learning an intrinsic garment space for interactive authoring of garment animation. ACM Transactions on Graphics (TOG) **38**(6), 1–12 (2019)
41. Wang, X., Chen, Y., Zhu, W.: A survey on curriculum learning. IEEE transactions on pattern analysis and machine intelligence **44**(9), 4555–4576 (2021)