



HAL
open science

GAPS: Geometry-Aware, Physics-Based, Self-Supervised Neural Garment Draping

Ruochen Chen, Liming Chen, Shaifali Parashar

► **To cite this version:**

Ruochen Chen, Liming Chen, Shaifali Parashar. GAPS: Geometry-Aware, Physics-Based, Self-Supervised Neural Garment Draping. 3DV, Mar 2024, Davos (Suisse), Switzerland. hal-04372245

HAL Id: hal-04372245

<https://hal.science/hal-04372245v1>

Submitted on 25 Nov 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

GAPS: Geometry-Aware, Physics-Based, Self-Supervised Neural Garment Draping

Ruochen Chen

Liming Chen

Shaifali Parashar

CNRS, Université Claude Bernard Lyon 1, INSA Lyon, École Centrale de Lyon, Université Lumière Lyon 2, LIRIS UMR5205

{ruochen.chen, liming.chen, shaifali.parashar}@liris.cnrs.fr

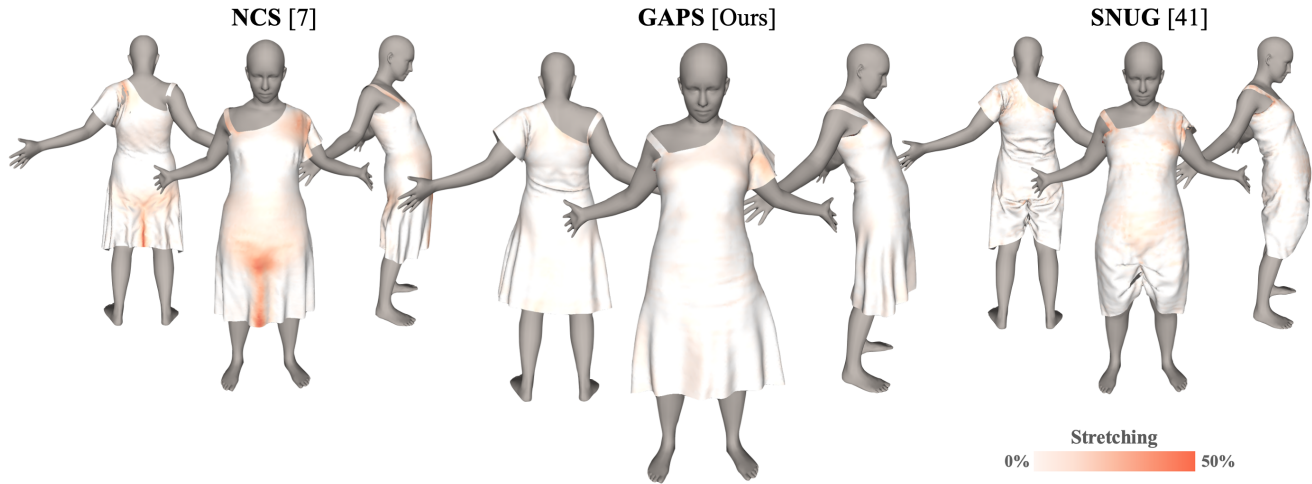


Figure 1. SNUG [41] and NCS [7] perform self-supervised draping of garments. However, they are prone to unrealistic stretching arising due to their efforts to prevent garment-body collisions. To overcome this issue, we propose GAPS which performs geometrically aware mechanism to drape a garment by stretching only if necessary. This controls body-garment collisions automatically without any post-processing [41] or restrictive measures [7]. Furthermore, our geometry-aware skinning allows better handling of loose garments.

Abstract

Recent neural, physics-based modeling of garment deformations allows faster and visually aesthetic results as opposed to the existing methods. Material-specific parameters are used by the formulation to control the garment inextensibility. This delivers unrealistic results with physically implausible stretching. Oftentimes, the draped garment is pushed inside the body which is either corrected by an expensive post-processing, thus adding to further inconsistent stretching; or by deploying a separate training regime for each body type, restricting its scalability. Additionally, the flawed skinning process deployed by existing methods produces incorrect results on loose garments.

In this paper, we introduce a geometrical constraint to the existing formulation that is collision-aware and imposes garment inextensibility wherever possible. Thus, we obtain realistic results where draped clothes stretch only while covering bigger body regions. Furthermore, we propose

a geometry-aware garment skinning method by defining a body-garment closeness measure which works for all garment types, especially the loose ones. Our code is publicly available at <https://github.com/Simonhfls/GAPS>.

1. Introduction

Modeling digital garments for real-time applications, such as video game simulations, fashion design, and e-commerce virtual try-on systems is a challenging yet important problem to solve. Traditionally, clothes are modeled using Physics-Based Simulation (PBS) [1, 25, 26] of continuum fabric [22] or highly detailed yarn [10] as per Newton’s laws of motion. While these techniques are capable of delivering precise simulations, their high computational complexity makes them unsuitable for real-time and web-based applications.

With the success of deep learning in major 3D learning tasks [27, 35–37, 42], supervised learning of garment drap-

ing [11, 15, 30, 33, 43–45] gained interest. They used a neural network fusing a given garment and a body (a joint pose and shape representation generally parametrized using SMPL [19]) to produce drapings similar to PBS, but at a faster rate.

A huge downside is that the supervision requires high quality 3D meshes of draped garments over various bodies which not only requires expensive setup (3D scanners and multi-view cameras) but also a huge amount of manual intervention. In addition, all these methods leave significant smoothing artefacts that often result in visually unaesthetic drapings.

Recently, [6] proposed a self-supervised learning of garment draping by enforcing a static physical consistency of garments while training. Inspired by [24], [7, 41] reformulated motion constraints in PBS as an optimization problem and enforced physical consistency on both static and dynamic garment deformations. Not only such a formulation is computationally more efficient than PBS and supervised methods but it also does not incur smoothing artefacts. However, a major shortcoming of [7, 41] is the assumption that the plasticity or elasticity of worn clothes depends solely on material properties. When a garment fits onto a body, it remains inextensible in most parts and stretches only when wrapped over bigger body regions. Current models use only material parameters and limit the strain and bending forces to allow only small changes. This causes two major problems: 1) Unrealistic fittings with abnormal local stretches that are incoherent with the physical nature of clothes, see Fig. 1. 2) Unsolvable body-garment collisions due to the naive strain/bending minimization. To overcome 2), [41] relies on heavy post-processing that forcefully pushes the collided garment vertex outside the body. This impairs run-time performance and contributes to further incoherent stretching. [7] uses data augmentation to reduce collisions and limits the training to single body shape. This is impractical as it requires to re-train for each body shape. It leads to longer training duration (reported up to 24 hours for loose garments such as dresses). [41] simultaneously trains to drape a garment over several body types. However, it does not handle cloth dynamics well. In addition, current methods use flawed garment-skinning whose accuracy is dependent on the tightness of the body which degrades their performance poorly while draping loose garments as seen Fig. 1.

In this paper, we propose **GAPS**: a geometry-aware, physics-based, self-supervised neural garment draping method to fix the above-mentioned problems. Geometry-based deformation modeling is widely popular in many computer vision problems such as 3D reconstruction [2, 28, 29, 32] and shape matching [3, 13]. To our best knowledge, **GAPS** is the first neural method that combines physics-based formulation with geometrical constraints on deforma-

tions to force collision-aware garment inextensibility. It is built upon [41] and imposes inextensibility by preserving local distances and areas, which is only progressively relaxed in case of body-garment collisions. Consequently, it replicates the true cloth behaviour: maintains inextensibility, stretches only over bigger body regions and does not penetrate the body. Furthermore, we propose a novel, geometry-aware skinning method which is suitable for all garment types as seen Fig. 1. Existing point-based methods [9, 38] learn a backward transformation field between posed body and canonical pose which fails on loose garments such as skirts. More advanced strategies such as [21] learn implicit neural garment body interactions but they don't generalise well. Contrary to these works, our proposed skinning is rather simple and able to generalise well. Our experiments demonstrate a significant qualitative and quantitative improvement over state-of-the-art methods.

2. Related Work

We classify existing cloth deformation methods into physics-based and learning-based methods.

Physics-based methods. Elastic continuum models [1] have been widely popular for modeling cloth deformations. However, their computational complexity has been an issue. Many simplifications such as reformulation of elastic model with Finite Element Method [16], faster simulations using mass-spring model [18], motion constraints [22] have been proposed to reduce the computational complexity at a slightly compromised accuracy. To maintain cloth inextensibility, [4] proposed an impulse-based constraint to minimize changes in lengths. However, this constraint was posed everywhere on the cloth and did not yield a significant improvement. [8, 34] proposed an iterative refinement as post-processing to minimize changes in edge lengths taking into account the collisions, friction and contacts. Unlike [4], we impose inextensibility into the geometrical structure through local covariance matrices, which are widely used to compute surface normals and variations [31]. We allow a gradual relaxation on the constraint whenever the draped garment finds itself inside the body; thus imposing it locally.

Learning-based methods. Despite several improvements, the computational complexity of PBS methods remains unsuitable for fast or real-time applications. Learning-based methods, on the other hand, yield a fast inference. Most methods focus on supervised learning. From a large dataset of a garment draped over several bodies, [5, 20, 30, 39, 44, 45] learn a parametric garment deformation. Instead of solving for the garment vertex positions at each time step between the garment at rest and the desired pose, they use ground truth supervision to estimate a single function that directly estimates the positioning of draped garment over the input body by taking into account the local and global

characteristics of the body and the garment. More advanced versions [14, 15] of these architectures explicitly model the skin-cloth interactions and improve the performance by incorporating local feature pooling. Since these methods are designed to learn a global representation, finer details such as wrinkles are barely produced which leads to plastic-like garment drapings. Most importantly, gathering such a large amount of data is a tedious task: be it from a PBS described above or a multi-camera setup. In addition, training on these large datasets is computationally expensive.

Recently, [6, 7, 41] proposed self-supervised learning of garment deformations which do not suffer from the drawbacks of the above-mentioned methods. While [6] can model only static deformations, [7, 41] can model both static and dynamic cloth deformations by imposing PBS as an optimization problem. Consequently, the error margin is much higher on these methods and only material-based control over strain and bending forces produces unrealistic stretches. [6] proposed a cloth consistency loss to minimize edge differences and maintain smoothness by minimizing Laplacians in local neighbourhood. [7] also used this cloth consistency model but it is uniformly applied everywhere on the garment even in the areas where the garment needs to stretch in order to fit. Instead, we control garment inextensibility realistically by stretching progressively, only when necessary. As a result, we do not require to impose any schemes to avoid body-garment collisions. While [41] fails on most loose-fitted garments such as dresses due to its flawed skinning, our proposed geometry-aware skinning allows to handle all types of garments easily. We will discuss it further in the methodology section.

3. Geometry-Aware Modeling

We now discuss our main contributions. First, we describe geometry-based deformation modeling which together with the existing physics-based modeling allows us to learn true cloth behaviour. Then, we describe the geometry-aware method to compute body-participation in garment dynamics to compute garment skinning. This allows efficient draping of all types of garments.

3.1. Garment Deformation Modeling

Even if the garments are made of highly stretchable material, they stretch only while fitting over larger body regions. Depending on the body shape and the dynamic motion, the stretching is generally localized to small areas while the majority of the draped garment remains inextensible; thus preserving local lengths, angles and areas. Geometrically, surfaces that deform while preserving these local properties are defined using *isometry* [17]: a geodesic-preserving map on smooth manifolds. It is imposed by preserving *metric tensors* which are locally defined using surface jacobians. While this differential modeling delivers promising results,

the computation of local jacobians and Hessians leads to slow neural networks [2, 3, 12], which is not desirable. We represent garments using meshes which are a discrete representation of a continuous space. Therefore, we relax isometry by enforcing inextensibility locally on each mesh vertex. Given $\mathcal{X} = (x_i)_{i=1}^N$, a mesh containing N vertices x_i that transforms into $\mathcal{Y} = (y_i)_{i=1}^N$, a naive inextensibility imposition would be force each edge around y_i within its local neighborhood $y_j, j \in \mathcal{N}_i$ to comply with the corresponding one around x_i in terms of length. So, we force that

$$D_i = \sum_{j \in \mathcal{N}_i} (\|x_i - x_j\| - \|y_i - y_j\|)^2 = 0. \quad (1)$$

While imposing this constraint minimizes inextensibility, it delivers jittery results. This is because under this scheme, each edge length will be optimized independently causing slight jitters.

Instead, we impose a restriction on covariance matrix. For x_i , it is given by

$$\mathbf{C}_{x_i} = \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} (x_j - \bar{x}_i)(x_j - \bar{x}_i)^\top, \quad (2)$$

where \bar{x}_i is the mean of x_j . We impose inextensibility as local rigidity around each vertex. Given that $y_i = \mathbf{R}x_i + \mathbf{T}$, where \mathbf{R} and \mathbf{T} are a 3D rotation and a translation respectively, we assume that $y_j = \mathbf{R}x_j + \mathbf{T}$. Thus, we obtain

$$\mathbf{C}_{y_i} = \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} (\mathbf{R}x_j - \mathbf{R}\bar{x}_i)(\mathbf{R}x_j - \mathbf{R}\bar{x}_i)^\top = \mathbf{R}\mathbf{C}_{x_i}\mathbf{R}^\top. \quad (3)$$

Using SVD, we can write $\mathbf{C}_{x_i} = \mathbf{U}\mathbf{S}_{x_i}\mathbf{U}^\top$. It can be easily verified that $\mathbf{C}_{y_i} = \mathbf{R}\mathbf{U}\mathbf{S}_{x_i}\mathbf{U}^\top\mathbf{R}^\top = \mathbf{V}\mathbf{S}_{y_i}\mathbf{V}^\top$. Thus inextensibility implies preservation of singular values of corresponding covariance matrix. It implies that $\sigma_i \in \mathbf{S}_{x_i} = \text{diag}(\sigma_1, \sigma_2, \sigma_3)$ will satisfy the characteristic polynomial of \mathbf{C}_{y_i} , i.e.,

$$\det(\mathbf{C}_{y_i} - \alpha\sigma_i\mathbf{I}_{3 \times 3}) = 0, \quad (4)$$

where $\mathbf{I}_{3 \times 3}$ is identity and $\alpha = 1$. We use the above constraint to enforce realistic cloth behaviour. It has 3 benefits: 1) σ_i are derived from template garment and therefore, can be precomputed. 2) We avoid computing singular values of the garment in current timestep during optimization, which is expensive and may suffer from numerical instabilities due to vanishing gradients. 3) We can control the degree of stretching just by changing α . In section 4, we will explain how α can be progressively changed to stretch the garment in order to avoid body collisions.

3.2. Body-Participation In Garment Dynamics

In order to dynamically fit a garment over a body, it is required to compute entire body's participation towards the

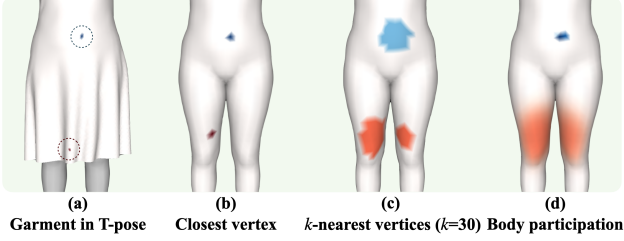


Figure 2. Body-participation computation for garment vertices close (in blue) or far (in red) to the body.

movement of each garment vertex. As described in [19]: for each body vertex, it can be computed as a weighted blend of body-joints, with weights describing the impact of body-joints towards the motion at a given vertex. Assuming that the garments follow body motions, [6, 7, 41] assign each garment vertex the blending weights of its nearest body vertex directly. While this may be effective for tightly-fit garments such as T-shirts, it is extremely suboptimal for looser garments like dresses. In order to compensate, [7] applies iterative Laplacian smoothing of the weights of each garment vertex w.r.t. its neighbours. On a loose garment, there is a high possibility for a garment vertex and its neighbours to match to the same body vertex, causing undesired artefacts.

To overcome these drawbacks, we introduce a simple, geometry-aware computation of blend weights premised on two fundamental observations: 1) For a given garment vertex, the influence of the blend weights of a body vertex is negatively correlated with their respective Euclidean distances. 2) On loose garments, the blend weights are less likely to depend on the proximate body vertex. We use the Gaussian Radial Basis Function (RBF) to compute the participation of each body vertex to a specific garment vertex. We write a Gaussian RBF kernel as

$$\varphi(r) = e^{-\frac{r^2}{km_i^2}}, \quad (5)$$

where m_i is the Euclidean distance between a garment vertex $g_i \in \mathcal{G}$ and its nearest body vertex. Then, we define a participation matrix \mathcal{P} where each element p_{ij} signifies the degree of participation of body vertex $b_j \in \mathcal{B}$ in g_i as

$$\mathcal{P} = \begin{bmatrix} \varphi(d_{11} - m_1) & \dots & \varphi(d_{1N_b} - m_1) \\ \vdots & \ddots & \vdots \\ \varphi(d_{N_g1} - m_{N_g}) & \dots & \varphi(d_{N_gN_b} - m_{N_g}) \end{bmatrix} \quad (6)$$

where d_{ij} represents the Euclidean distance between g_i and b_j , N_g and N_b are the number of elements in \mathcal{G} and \mathcal{B} respectively and $k = 0.5$. The new garment blend weights, $\tilde{\mathcal{W}}$, are then given by

$$\tilde{\mathcal{W}} = \text{normalize}(\mathcal{P} * \mathcal{W}), \quad (7)$$

where \mathcal{W} are the blend weights described in [19]. The normalization makes the cumulative sum of each row to 1.

Fig. 2 shows the body participation measures for garment vertices that drape close (in blue) or far (in red) to the body. Picking the closest body vertex and considering its blend weights might work for garment vertices close to the body. However, for the vertices on looser regions, it is clear that motion dynamics is dependent on a larger body region. Naively choosing k-nearest neighbours might work for vertices on looser regions but not on the tight ones. Our simple and effective body-participation method can automatically find the best-fitting region for any type of garment, irrespective of its topology which allows flawless garment skinning.

4. Method

We integrate our geometric formulations with PBS principles to improve the performance of learning-based cloth simulations.

4.1. Model

Our model is developed within an unsupervised learning framework proposed in [41]. It incorporates a Gated Recurrent Unit (GRU) module R outputting vertex displacement relative to the template garment \mathcal{T} (unposed and undeformed). This GRU module consists of four layers, each yielding an output size of 256, with tanh activation functions. After combining it with \mathcal{T} to obtain the unposed, deformed garment (\mathcal{M}_t), we subsequently apply the skinning function S which articulates \mathcal{M}_t . The garment blend weights $\tilde{\mathcal{W}}$ for skinning are estimated using Body-Participation module, described in section 3.2. The network is trained by optimizing a composite loss function, which includes our newly proposed inextensibility loss (described in section 3.1) in addition to the PBS losses. The architecture of our model is depicted in Fig. 3.

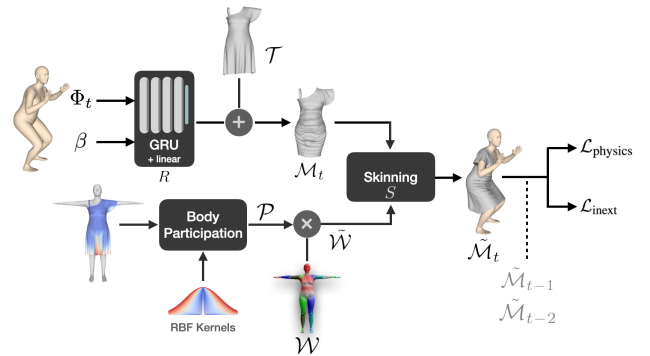


Figure 3. Overview of our method. We added RBF based Body Participation module on [41] to estimate blend weights based on relative garment-body geometry.

The garment is modeled as

$$\begin{aligned}\mathcal{M}_t &= \mathcal{T} + R(\Phi_t, \beta), \\ \tilde{\mathcal{M}}_t &= S(\mathcal{M}_t, J(\beta), \theta_t, \tilde{\mathcal{W}}),\end{aligned}\quad (8)$$

where the regressor R takes as input the body shape β and the motion parameter $\Phi_t = \{\theta_t, v_t\}$ containing the current body pose θ_t and global velocity v_t of the root joint. The skinning function S articulates \mathcal{M}_t based on joint locations $J(\beta)$, current body pose and the estimated garment blend weights $\tilde{\mathcal{W}}$ using the Body-Participation module.

In order to learn the garment deformations, we know that the physical equilibrium of forces related to strain, gravity, collision and bending (defined in PBS) should be achieved. Like [41], we enforce them as losses to be minimized. We express them as

1) *Strain*. We use the Saint-Venant–Kirchhoff (StVK) material model for simulating deformation. It is given by

$$\mathcal{L}_{\text{strain}} = \mu \|\epsilon_{\text{stvk}}\|_{\text{Fro}}^2 + \frac{\lambda}{2} \text{tr}(\epsilon_{\text{stvk}})^2, \quad (9)$$

where $\epsilon_{\text{stvk}} = \frac{1}{2}(\mathbf{F}^\top \mathbf{F} - \mathbf{I})$ represents the Green-Lagrangian strain tensor, where \mathbf{F} is the deformation gradient, and \mathbf{I} is the identity matrix. μ and λ are the Lamé coefficients reflecting the material properties.

2) *Gravity*. We incorporate gravity by minimizing the potential energy of the garment, given by

$$\mathcal{L}_{\text{gravity}} = \sum_{\text{vertices}} -mg^\top x, \quad (10)$$

where m is the particle mass and g is the gravitational acceleration.

3) *Collision*. It penalizes penetration between the body and the garment. For each garment vertex, it is given by

$$\mathcal{L}_{\text{collision}} = \sum_{\text{vertices}} k_c * d_c^3, \quad (11)$$

where k_c is a balancing factor, and $d_c = \max(\epsilon - d(x), 0)$ quantifies the degree of interpenetration. $d(x)$ is the signed distance between garment vertex and body surface, and ϵ is a small positive constant introduced to enhance stability.

4) *Bending*. We use a different formulation than [7, 41]. We express it as a balance between smoothness in local garment areas and global coherence with the original template (to retain natural drape characteristics). It is given by

$$\mathcal{L}_{\text{bending}} = \sum_{\text{edges}} k_b \frac{l^2}{8a} (\alpha(\theta_t - \theta_r)^2 + (1 - \alpha)\theta_t^2), \quad (12)$$

where k_b is the bending stiffness, θ_t denotes the dihedral angle of the deformed garment, θ_r signifies the corresponding dihedral angle in the rest state of the template garment, l is the common edge length and a is the summation of the

areas of both faces. The balancing coefficient $\alpha \in [0, 1]$ is proportional to the distance between the garment and the current edge in the rest pose.

5) *Inertia*. In order to convert PBS to an optimisation problem, [41] proposed the inertia loss. It is given by

$$\mathcal{L}_{\text{inertia}} = \sum_{\text{vertices}} \frac{1}{2\Delta t^2} m(x - x^{[t-1]} - \Delta t v^{[t-1]})^2, \quad (13)$$

where Δt is the simulation time step, $x^{[t]}$ and $x^{[t-1]}$ specify the particle’s position at times t and $t - 1$, respectively. As suggested in [7], we do not to back-propagate $\mathcal{L}_{\text{inertia}}$ through previous frames.

Together, all these physical losses are given by

$$\mathcal{L}_{\text{physics}} = \mathcal{L}_{\text{strain}} + \mathcal{L}_{\text{bending}} + \mathcal{L}_{\text{collision}} + \mathcal{L}_{\text{gravity}} + \mathcal{L}_{\text{inertia}}. \quad (14)$$

Inextensibility Loss. Our novel geometric loss that enforces inextensibility (as per eq. 4) within the one-ring local area associated with each vertex, is given by

$$\mathcal{L}_{\text{inext}} = k_i \sum_{\text{vertices}} \sum_{j=1}^3 |\det(\mathbf{C} - k_{\text{ext}} \sigma_j \mathbf{I}_{3 \times 3})|, \quad (15)$$

where k_i is a balancing factor, \mathbf{C} is the covariance matrix within 1-ring neighbourhood and σ_j are the singular values of covariance matrix of the 1-ring neighbourhood around the corresponding point in garment template \mathcal{T} . k_{ext} decides the degree of extension: $k_{\text{ext}} = 1$ indicates inextensibility and $k_{\text{ext}} > 1$ indicates stretching. Given that garment stretches only while covering large body regions, k_{ext} should be 1 and changed only if the body-garment collision is detected. Moreover, to not be affected by slight perturbations, the garment should only be gradually stretched upon collision detection. Considering this, we write

$$k_{\text{ext}} = 1 + \min(10d_c, 0.03) \times \min(e, 100), \quad (16)$$

where e is the current epoch. We first allow the network to stabilize through 100 epochs and then gradually stretch the garment locally in case of body-garment collisions.

The overall loss is

$$\mathcal{L} = \mathcal{L}_{\text{physics}} + \mathcal{L}_{\text{inext}}. \quad (17)$$

5. Experiments

Training. During the training phase, the Gated Recurrent Unit (GRU) layers are initialized with random hidden states. The batch size is 128, the learning rate is set to $1e-3$ for the initial 10 epochs and $1e-4$ for the rest. The balancing weight for inextensibility loss k_i is set to $2e8$. All other hyperparameters as well as material parameters are set according to [41]. We randomly sample the shape parameters from $\mathcal{U}(-2, 2)$ for each batch. We select 60 train sequences



Figure 4. T-shirt, Dress, Skirt, and Gown draped using **GAPS**.

and 5 validation sequences from AMASS dataset [23] containing around 11000 poses. We compare with **SNUG** [41] and **NCS** [7]: the only state-of-the-art methods closest to ours. For a fair comparison, we train them on the same motion sequences as ours. **GAPS** and **SNUG** are trained on variable bodies whereas **NCS** is trained on a single body. The training code for **SNUG** is unavailable so we have implemented it. Our training losses are slightly better than the reported numbers in the paper.

Error metrics. We report ε_e and ε_a as the mean difference (in %) between the edge lengths and areas between the template and the draped garment. We report ε_c as the % of draped garment vertices colliding with the body.

Quantitative Evaluation. We consider the validation sequence AMASS 86_07 on a single body and drape four garments shown in Fig. 4. In general, each garment should be able to fit on this body without stretching. However, due to the dynamic motion of the body, stretching is inevitable. Tab. 1 shows the results. **GAPS** shows the best performance on all datasets. The drapings are obtained with minimal stretch and collisions, thanks to our collision-aware imposition of inextensibility and geometry-aware skinning. We remind that **NCS** performs body-specific draping. It is trained only on the body that we used in this experiment while others are trained on variable bodies. It performs well (close to ours) on T-shirt which is a simple, tight-fitting garment. On other garments, which are all more challenging than T-shirt, it degrades significantly. **SNUG** performs a post-processing to fix body-garment collisions which makes $\varepsilon_c = 0$ but it considerably increases ε_a and ε_e . We compute all metrics before post-processing, thus evaluating only the method’s performance. Fig. 5 shows the ε_a and ε_c on the entire AMASS 86_07 sequence with a Dress draped on it. Unlike **NCS**, the stretches produced by **GAPS** seem to be quite realistic with minimal stretching in loose garment areas.

Performance on Diverse Bodies. Both **GAPS** and **SNUG** are trained on variable bodies. We compare their scalability. Fig. 6 shows diverse body shapes where **SNUG** shows strong body-garment penetrations. It heavily depends on post-processing to generate visually comparable results. In contrast, **GAPS**’ collision-aware inextensibility loss allows visually remarkable results. We have omitted results for ex-

remely thin bodies as they are similar for all methods.

Performance on Loose Garments. We revisit dress to compare the performance on loose garments. **SNUG** assumes that garment motion is close to the body’s and uses the blend weights proposed in [19] for garment vertices and therefore, fails while fitting loose garments. **NCS** uses a Laplacian smoothing to estimate the blend weights better in order to fit loose garments. However, this requires an iteration parameter to be set. We fine-tune this parameter (set to 50) in order to get the best possible performance on the dress. Fig. 7 shows the results. **SNUG** can not deal with loose garments. On looser areas, we observed a big spike in strain loss which hinders convergence. **NCS** can not produce realistic results despite the careful selection of the smoothing iterations; the results are similar to **SNUG**. In contrast, **GAPS**’ geometry-aware skinning is robust, generates realistic results quite close to the PBS (used in [40]). Fig. 8 compares skinning of the template garment onto a posed body using different skinning methods. The closest vertex based skinning shows large discontinuities between the legs. The use of K-nearest vertices reduces these discontinuities, however with an inevitable compromise with accuracy due to the fixed number of neighbours. The Laplacian smoothing based skinning of **NCS** is similar to the k-nearest vertices one, which explains its degraded performance on loose garments.

Ablation Study. We study the individual impact of various components within the **GAPS** framework.

1) *Collision-awareness.* By fixing $k_{\text{ext}} = 1$ in $\mathcal{L}_{\text{inext}}$ (see eq. (15)), we impose the garment to remain intextensible everywhere. Tab. 2 shows the % of vertices in collision, ε_c . We see that incorporated collision-awareness in **GAPS** is crucial for it to drape realistically.

2) *Inextensibility.* Tab. 3 shows that explicit enforcement of inextensibility loss $\mathcal{L}_{\text{inext}}$ on top of physics-based losses $\mathcal{L}_{\text{physics}}$ forces **GAPS** to maintain inextensibility especially while draping over smaller bodies which are expected to be as inextensible as possible. This shows the complementary nature of strain loss, eq (9) and $\mathcal{L}_{\text{inext}}$ (15).

3) *RBF-based skinning.* As demonstrated in Fig. 9, the RBF-based skinning method employed in **GAPS** enhances realism for loose garments and generalizes well to tight garments. Note that draping results for tight garments, such as T-shirt, are similar whether the RBF-based skinning method is employed or not.

Timing Performance. **SNUG** takes less than 1 hour to converge for tight garments with less than 10k vertices but will take a significantly longer time (up to 24 hours) for looser garments because of the spike in strain loss. **NCS** takes 1-24 hours. Our method takes 1 hour for tight garments and up to 8 hours for looser garments. All training is carried out on 8 NVIDIA V100 GPUs.

As for run-time performance, we measure the process-

	T-shirt			Dress			Skirt			Gown		
	ϵ_e	ϵ_a	ϵ_c	ϵ_e	ϵ_a	ϵ_c	ϵ_e	ϵ_a	ϵ_c	ϵ_e	ϵ_a	ϵ_c
SNUG	7.815	13.796	5.631 (± 2.453)	4.833	6.084	3.113 (± 1.496)	3.062	4.063	0.365 (± 0.788)	5.906	7.434	3.515 (± 3.376)
NCS	3.615	5.311	0.522 (± 0.513)	4.164	5.391	0.368 (± 0.556)	4.386	5.677	2.076 (± 0.668)	4.782	6.686	0.633 (± 1.041)
GAPS	3.343	5.290	0.116 (± 0.420)	3.051	3.825	0.196 (± 0.308)	2.337	3.019	0.026 (± 0.131)	3.639	4.249	0.123 (± 0.225)

Table 1. Inextensibility and Collision metrics on AMASS 86_07. **GAPS** fits all the garments with minimal collisions and minimal garment extensions.

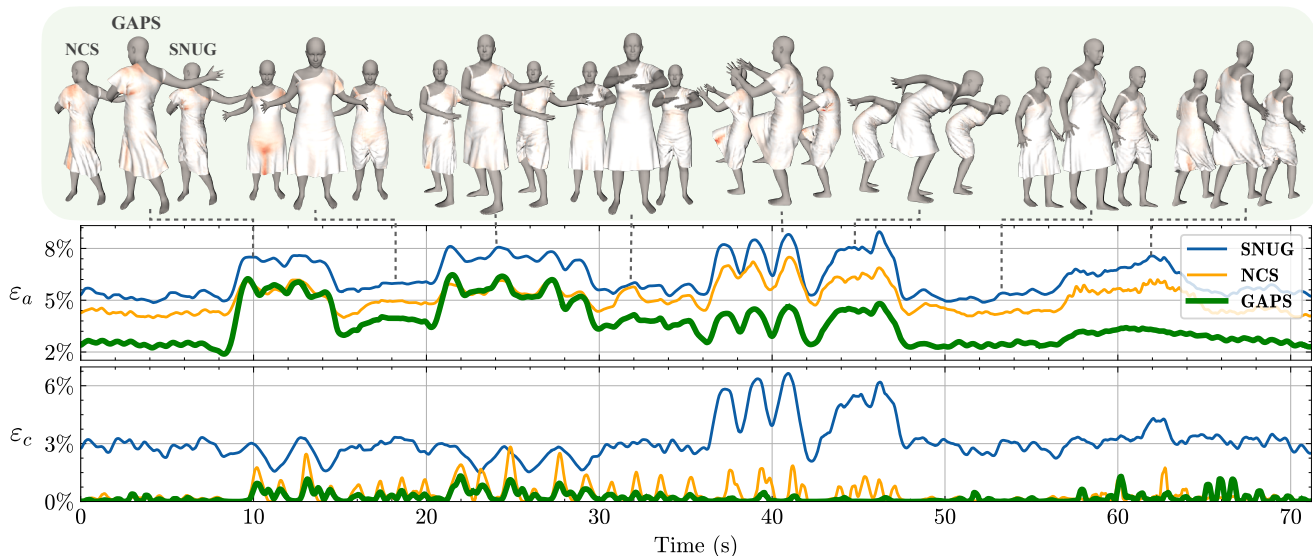


Figure 5. Results on AMASS sequence 86_07 with Dress. **GAPS** shows the best performance with minimal collision and realistic stretching (indicated in orange).

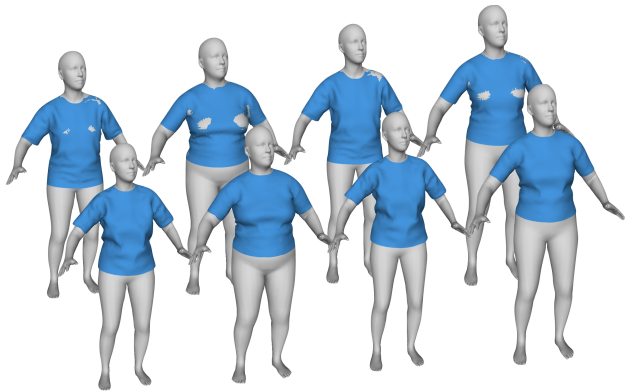


Figure 6. Results on diverse bodies. **SNUG** (top) shows significant garment penetration while **GAPS** (bottom) shows remarkable performance.

	T-shirt	Dress	Skirt	Gown
GAPS	0.116	0.196	0.026	0.123
GAPS ($k_{\text{ext}} = 1$)	5.191	5.572	2.992	7.027
GAPS without $\mathcal{L}_{\text{inext}}$	5.344	5.924	3.105	7.580

Table 2. ϵ_c across various configurations shows that collision-awareness is crucial to **GAPS**.

	Avg body		Chubby		Obese	
	ϵ_e	ϵ_a	ϵ_e	ϵ_a	ϵ_e	ϵ_a
GAPS	3.05	3.83	4.32	5.06	6.41	6.44
No $\mathcal{L}_{\text{inext}}$	3.84	5.13	4.38	5.54	6.43	6.94

Table 3. ϵ_e and ϵ_a evaluated on Dress across various body types shows the importance of inextensibility loss.

ing duration from the stage of raw body pose data to the final body and garment meshes. For a fair comparison, we use a CMU motion sequence comprising 2,175 frames to evaluate the runtime. The tests are executed on an Intel Xeon Gold 6146 and NVIDIA V100. For **SNUG**, we used the checkpoint and associated script provided by the author. For **NCS**, we use the author’s script for training and prediction. Tab. 4 shows the comparison. **SNUG** takes greater runtime due to the post-processing. The authors suggested the possibility of parallelizing the collision post-processing component on the GPU to enhance runtime performance. However, their runtime GPU code is not publicly available. Our approach achieves the fastest performance without requiring a GPU.

Summary of methods. **SNUG** is trained on various body shapes; it is scalable but relies on post-processing to overcome body-garment collisions, which are significant. It

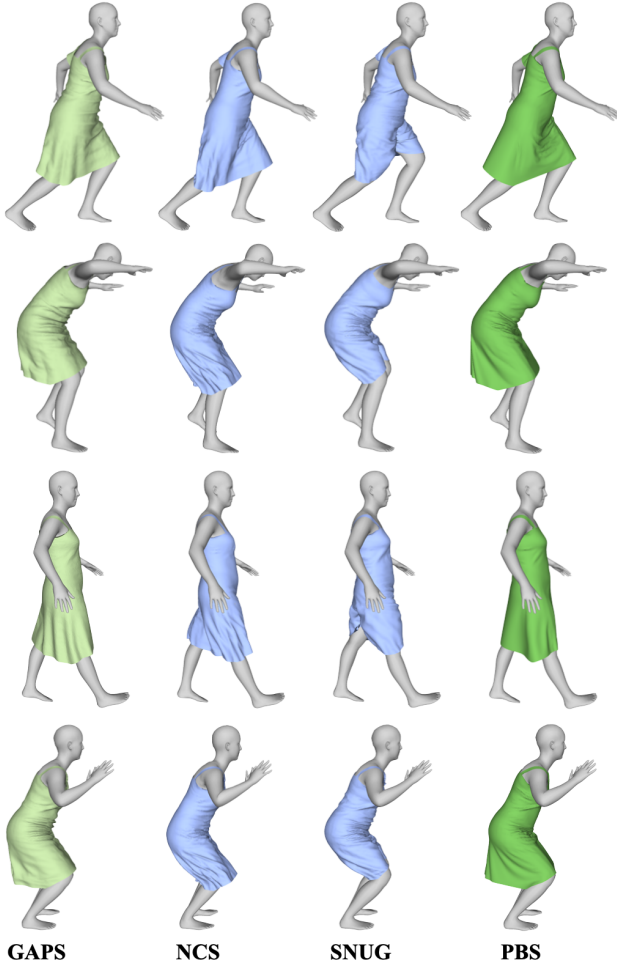


Figure 7. Results on Dress. **GAPS** shows realistic results quite close to the PBS.

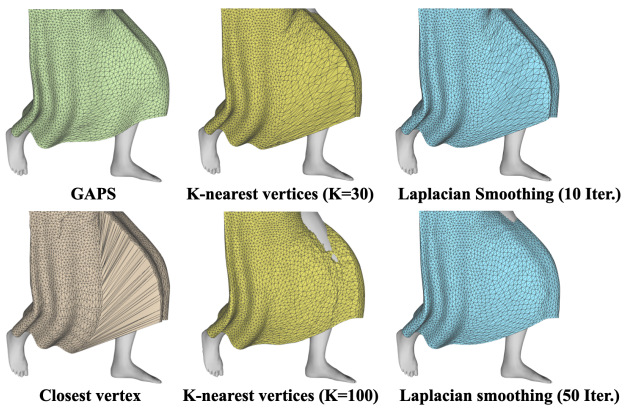


Figure 8. Comparing the geometry-aware, RBF-based skinning in **GAPS** with others.

cannot handle loose garments. **NCS** is body-specific and trained on a single body; it is highly restrictive but it fits decently well with lesser collisions and smaller extensions

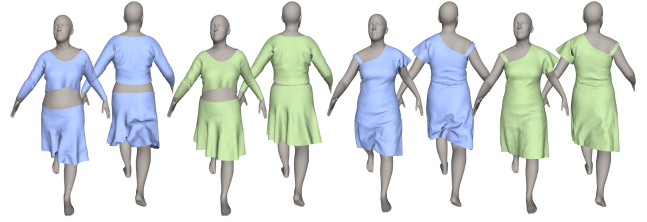


Figure 9. **GAPS** with (in green) RBF-based skinning and without it (in blue).

	Train	Runtime
SNUG	1 - 24 h	23.7 ms
NCS	1 - 24 h	3.3 ms
GAPS	1-8 h	2.7 ms

Table 4. Timing performance.

than **SNUG**. It produces some unrealistic stretches and does not drape loose clothes well, due to its flawed skinning. **GAPS** is scalable, minimizes body-garment collisions and garment extensions. It produces realistic stretches and drapes all types of garments well.

6. Conclusion

We presented **GAPS**: a geometry-aware, physics-based, self-supervised garment draping method. It incorporates explicit collision-aware inextensibility enforcement which encourages realistic drapings where the garment stretches only while fitting over larger body regions. Furthermore, the collision-awareness module significantly reduces body-garment collisions, eliminating the need for expensive post-processing or restrictive training. In addition, we proposed a geometry-aware skinning approach which automatically computes the right body-participation to a garment dynamics and therefore, can handle a variety of garments. This allows us to obtain significantly better draping results, even on loose garments. Most importantly, our modeling is generic and can be easily incorporated with others.

Limitations & future work Though temporal neural networks have been incorporated in our approach, there is still a necessity for better handling of the cloth dynamics. Additionally, while the implementation of our collision-aware inextensibility loss has substantially mitigated body-garment collision issues, challenges persist in extreme body poses and shapes. Furthermore, our current framework does not yet address cloth self-collision and cannot deal with multi-layer and topology-varying clothing settings. As a future work, we would explore the possibility of imposing our collision-aware inextensibility as a hard constraint to ensure zero collisions with minimal garment extensions.

Acknowledgements This research has received funding from the project RHINO, an ANR-JCJC research grant.

References

- [1] David Baraff and Andrew Witkin. Large steps in cloth simulation. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, page 43–54, 1998. [1](#), [2](#)
- [2] Jan Bednarik, Shaifali Parashar, Erhan Gundogdu, Mathieu Salzmann, and Pascal Fua. Shape reconstruction by learning differentiable surface representations. In *Conference on Computer Vision and Pattern Recognition*, 2020. [2](#), [3](#)
- [3] Jan Bednarik, Vladimir G. Kim, Siddhartha Chaudhuri, Shaifali Parashar, Mathieu Salzmann, Pascal Fua, and Noam Aigerman. Temporally-Coherent Surface Reconstruction via Metric-Consistent Atlases. In *International Conference on Computer Vision*, 2021. [2](#), [3](#)
- [4] Jan Bender and Daniel Bayer. Parallel Simulation of Inextensible Cloth. In *Workshop in Virtual Reality Interactions and Physical Simulation*, 2008. [2](#)
- [5] Hugo Bertiche, Meysam Madadi, and Sergio Escalera. CLOTH3D: clothed 3d humans. In *European Conference on Computer Vision*, 2020. [2](#)
- [6] Hugo Bertiche, Meysam Madadi, and Sergio Escalera. Pbn: Physically based neural simulation for unsupervised garment pose space deformation. *ACM Transactions on Graphics*, 40(6), 2021. [2](#), [3](#), [4](#)
- [7] Hugo Bertiche, Meysam Madadi, and Sergio Escalera. Neural cloth simulation. *ACM Transactions on Graphics*, 41(6), 2022. [1](#), [2](#), [3](#), [4](#), [5](#), [6](#)
- [8] Robert Bridson, Ronald Fedkiw, and John Anderson. Robust treatment of collisions, contact and friction for cloth animation. *ACM SIGGRAPH*, pages 594–663, 2002. [2](#)
- [9] Xu Chen, Yufeng Zheng, Michael J Black, Otmar Hilliges, and Andreas Geiger. Snarf: Differentiable forward skinning for animating non-rigid neural implicit shapes. In *International Conference on Computer Vision (ICCV)*, 2021. [2](#)
- [10] Gabriel Cirio, Jorge Lopez-Moreno, David Miraut, and Miguel A. Otaduy. Yarn-level simulation of woven cloth. *ACM Transactions on Graphics*, 33(6), 2014. [1](#)
- [11] Enric Corona, Albert Pumarola, Guillem Alenyà, Gerard Pons-Moll, and Francesc Moreno-Noguer. Smplicit: Topology-aware generative model for clothed people. In *Conference on Computer Vision and Pattern Recognition*, 2021. [2](#)
- [12] Sagnik Das, Ke Ma, Zhixin Shu, and Dimitris Samaras. Learning an isometric surface parameterization for texture unwrapping. In *European Conference on Computer Vision*, 2022. [3](#)
- [13] Nicolas Donati, Abhishek Sharma, and Maks Ovsjanikov. Deep Geometric Functional Maps: Robust Feature Learning for Shape Correspondence. In *Conference on Computer Vision and Pattern Recognition*, 2020. [2](#)
- [14] Erhan Gundogdu, Victor Constantin, Amrollah Seifoddini, Minh Dang, Mathieu Salzmann, and Pascal Fua. Garnet: A Two-Stream Network for Fast and Accurate 3D Cloth Draping. In *International Conference on Computer Vision*, 2019. [3](#)
- [15] Erhan Gundogdu, Victor Constantin, Shaifali Parashar, Amrollah Seifoddini, Minh Dang, Mathieu Salzmann, and Pascal Fua. Garnet++: Improving Fast and Accurate Static 3D Cloth Draping by Curvature Loss. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(1), 2020. [2](#), [3](#)
- [16] Theodore Kim. A finite element formulation of baraff-witkin cloth. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, Goslar, DEU, 2020. Eurographics Association. [2](#)
- [17] John M. Lee. *Introduction to Smooth Manifolds*. Springer, 2003. [3](#)
- [18] Tiantian Liu, Adam W. Bargteil, James F. O’Brien, and Ladislav Kavan. Fast simulation of mass-spring systems. *ACM Transactions on Graphics*, 32(6):1–7, 2013. [2](#)
- [19] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM SIGGRAPH Asia*, 34(6):248:1–248:16, 2015. [2](#), [4](#), [6](#)
- [20] Zorah Löhner, Daniel Cremers, and Tony Tung. Deep-Wrinkles: Accurate and Realistic Clothing Modeling. In *European Conference on Computer Vision*, pages 698–715. Springer International Publishing, 2018. [2](#)
- [21] Q. Ma, J. Yang, M. J. Black, and S. Tang. Neural point-based shape modeling of humans in challenging clothing. In *2022 International Conference on 3D Vision (3DV)*, pages 679–689, Los Alamitos, CA, USA, 2022. IEEE Computer Society. [2](#)
- [22] Miles Macklin, Matthias Müller, and Nuttapon Chentanez. XPBD: Position-Based Simulation of Compliant Constrained Dynamics. In *Proceedings of the 9th International Conference on Motion in Games*, page 49–54, 2016. [1](#), [2](#)
- [23] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *International Conference on Computer Vision*, pages 5442–5451, 2019. [6](#)
- [24] Sebastian Martin, Bernhard Thomaszewski, Eitan Grinspun, and Markus Gross. Example-based elastic materials. In *ACM SIGGRAPH*, 2011. [2](#)
- [25] Rahul Narain, Armin Samii, and James F. O’Brien. Adaptive anisotropic remeshing for cloth simulation. *ACM Transactions on Graphics*, 31(6):147:1–10, 2012. Proceedings of ACM SIGGRAPH Asia 2012, Singapore. [1](#)
- [26] Andrew Nealen, Matthias Mueller, Richard Keiser, Eddy Boxerman, and Mark Carlson. Physically Based Deformable Models in Computer Graphics. *Computer Graphics Forum*, 25(4), 2006. [1](#)
- [27] Mohamed Omran, Christoph Lassner, Gerard Pons-Moll, Peter V. Gehler, and Bernt Schiele. Neural body fitting: Unifying deep learning and model-based human pose and shape estimation. In *International Conference on 3D Vision*, 2018. [1](#)
- [28] Shaifali Parashar, Mathieu Salzmann, and Pascal Fua. Local Non-Rigid Structure-From-Motion from Diffeomorphic Mappings. In *Conference on Computer Vision and Pattern Recognition*, 2020. [2](#)
- [29] Shaifali Parashar, Adrien Bartoli, and Daniel Pizarro. Robust Isometric Non-Rigid Structure-From-Motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6409–6423, 2021. [2](#)

- [30] Chaitanya Patel, Zhouyingcheng Liao, and Gerard Pons-Moll. TailorNet: Predicting Clothing in 3D as a Function of Human Pose, Shape and Garment Style. In *Conference on Computer Vision and Pattern Recognition*, 2020. 2
- [31] Mark Pauly, Markus Gross, and P. Leif Kobbelt. Efficient simplification of point-sampled surfaces. In *IEEE Conference on Visualization*, 2002. 2
- [32] Mathieu Perriollat, Richard Hartley, and Adrien Bartoli. Monocular Template-Based Reconstruction of Inextensible Surfaces. In *British Machine Vision Conference*, 2008. 2
- [33] Tobias Pfaff, Meire Fortunato, Alvaro Sanchez-Gonzalez, and Peter Battaglia. Learning mesh-based simulation with graph networks. In *International Conference on Learning Representations*, 2021. 2
- [34] Xavier Provot. Deformation constraints in a mass-spring model to describe rigid cloth behaviour. In *Graphics interface*, 1995. 2
- [35] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In *Conference on Computer Vision and Pattern Recognition*, 2017. 1
- [36] Charles R. Qi, Li Yi, Hao Su, and Leonidas J. Guibas. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In *NeurIPS*, 2017.
- [37] Elad Richardson, Matan Sela, and Ron Kimmel. 3d face reconstruction by learning from synthetic data. 2016. 1
- [38] Shunsuke Saito, Jinlong Yang, Qianli Ma, and Michael J. Black. SCANimate: Weakly supervised learning of skinned clothed avatar networks. In *Proceedings IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2
- [39] Igor Santesteban, Miguel A. Otaduy, and Dan Casas. Learning-Based Animation of Clothing for Virtual Try-On. *Computer Graphics Forum*, 38(2):355–366, 2019. 2
- [40] Igor Santesteban, Nils Thuerey, Miguel A Otaduy, and Dan Casas. Self-Supervised Collision Handling via Generative 3D Garment Models for Virtual Try-On. 2021. 6
- [41] Igor Santesteban, Miguel A Otaduy, and Dan Casas. SNUG: Self-Supervised Neural Dynamic Garments. In *Conference on Computer Vision and Pattern Recognition*, 2022. 1, 2, 3, 4, 5, 6
- [42] Soltani, Huang, Wu, Kulkarni, and Tenenbaum. Synthesizing 3d shapes via modeling multi-view depth maps and silhouettes with deep generative networks. In *Conference on Computer Vision and Pattern Recognition*, 2017. 1
- [43] Garvita Tiwari, Bharat Lal Bhatnagar, Tony Tung, and Gerard Pons-Moll. SIZER: A Dataset and Model for Parsing 3D Clothing and Learning Size Sensitive 3D Clothing. In *European Conference on Computer Vision*, 2020. 2
- [44] Tuanfeng Y. Wang, Tianjia Shao, Kai Fu, and Niloy J. Mitra. Learning an intrinsic garment space for interactive authoring of garment animation. *ACM Transactions on Graphics*, 38(6), 2019. 2
- [45] Meng Zhang, Tuanfeng Y. Wang, Duygu Ceylan, and Niloy J. Mitra. Dynamic neural garments. *ACM Transactions on Graphics*, 40(6), 2021. 2