SNUG: Self-Supervised Neural Dynamic Garments

Igor Santesteban[1] Miguel A. Otaduy[1] Dan Casas[1]

[1]Universidad Rey Juan Carlos, Spain

first.last@{urjc.es}

http://mslab.es/projects/SNUG

Figure 1. Existing learning-based methods for garment deformations (left) use supervised training schemes that require the expensive computation of large datasets. In contrast, our approach SNUG (right) is a learning-based method that enables the self-supervised training of dynamic neural 3D garments, without requiring any ground-truth data.

Abstract

We present a self-supervised method to learn dynamic 3D deformations of garments worn by parametric human bodies. State-of-the-art data-driven approaches to model 3D garment deformations are trained using supervised strategies that require large datasets, usually obtained by expensive physics-based simulation methods or professional multi-camera capture setups. In contrast, we propose a new training scheme that removes the need for ground-truth samples, enabling self-supervised training of dynamic 3D garment deformations. Our key contribution is to realize that physics-based deformation models, traditionally solved in a frame-by-frame basis by implicit integrators, can be recasted as an optimization problem. We leverage such optimization-based scheme to formulate a set of physics-based loss terms that can be used to train neural networks without precomputing ground-truth data. This allows us to learn models for interactive garments, including dynamic deformations and fine wrinkles, with a two orders of magnitude speed up in training time compared to state-of-the-art supervised methods.

1. Introduction

The efficient modeling of digital garments is an active area of research due to the large number of applications, including fashion design, e-commerce, virtual try-on, and video games. The traditional approach to this problem is through physics-based simulation [38], but the high computational cost required at run time hinders the deployment of these techniques to real-world applications. Recently, learning-based methods [17, 31, 39, 45, 46, 50, 53, 55] have demonstrated that it is possible to closely approximate the accuracy of physics-based solutions. These methods use *supervised* learning strategies to find a function that outputs a deformed garment given an input body descriptor. During the training phase, the supervision is enforced by directly minimizing at a vertex level the difference between the predicted garment and ground-truth 3D meshes. Despite requiring hours of training, learning-based methods are highly-efficient to evaluate at run time, therefore they potentially offer an attractive alternative to traditional physics-based solutions.

However, the need for large datasets in current supervised methods is far from ideal. Ground-truth meshes must be obtained –for each combination of garment, body shape, and pose– via computationally-expensive simulations [37] or complex 3D scanning setups [40], which heavily hinders the scalability of current learning-based methods. We observe that for similar image-based problems, *self-supervised* strategies have shown that it is possible to learn complex tasks without requiring ground-truth data [41, 57]. Unfortunately, self-supervision for dynamic 3D clothing has not been explored.

In this work, we present a self-supervised method to learn dynamic deformations of 3D garments worn by para-

metric human bodies. The key to our success is realizing that the solution to the equations of motion used in current physics-based methods can also be formulated as an optimization problem [34]. More specifically, we show that the per-time-step numerical integration scheme used to update the vertex position (*e.g.,* backward Euler) in physics-based simulators, can be recast as an optimization problem, and demonstrate that the function for this minimization can become the central ingredient of a self-supervised learning scheme. Since this objective function includes both an inertial term and static term directly derived from the equations of motion, we are able to learn time-dependent and pose-dependent deformations *without* any ground-truth data.

The advantages of self-supervision go beyond removing the need for ground-truth data. By reformulating the learning tasks in terms of physics-based intrinsic properties instead of explicit 3D surface similarity, we also mitigate the smoothing artifacts common in supervised methods where L2 losses are used directly at the vertex level [39]. Additionally, self-supervised approaches also generalize better to test sequences outside the distribution of the training set. Finally, we also show how different material models can be easily formulated in our self-supervised framework, bringing the generalization capabilities of physics-based solutions (*i.e.*, deform any material) to learning-based methods, without requiring any precalculation or offline step.

All in all, our main contribution is a novel learning-based method capable of learning to dynamically deform garments using a self-supervised strategy. We demonstrate the superiority of our approach in terms of data requirements, training time, and inference time, and we quantitatively and qualitatively compare our results with state-of-the-art supervised methods.

2. Related Work

Existing methods that model how cloth and garment deform can be categorized into two groups: physics-based models and learning-based models.

Physics-Based Methods. Physics-based simulation of cloth is to date a very mature field. Over the years, many methods have been developed to solve the most relevant challenges. These include the design of deformation models such as in-plane and bending energies [15, 22], robust implicit solvers [4], rich and efficient contact handling [9, 49], or adaptive discretizations [37]. Recent efforts also include the design of differentiable physics simulators [18], including specific problems of cloth simulation such as continuous collision detection and constraint-based solvers [28], and physics-based objectives for tracking and reconstruction of garments [27, 59]. While the majority of the cloth simulation models represent the fabric as a continuum, a recent line of research uses yarn-level representations for high-resolution detail [10, 21]. Some works also show two-way coupling between garments and soft-body avatars [36, 42]. While we do not tackle this level of detail in our paper, more accurate methods could be used to replace our cloth and body models.

Learning-Based Methods. In contrast to physics-based models, which typically require solving large systems of nonlinear equations at each time step, learning-based methods aim at estimating a single function that directly outputs the desired deformation for any input. Inspired by early works on Pose Space Deformation [24], a common strategy is to learn parametric garment deformations, which are added to a mesh template, as a function of pose [16, 56], shape [53], pose-and-shape [5, 45], design [31, 39, 55], or garment size [50].

To this end, state-of-the-art methods for garments use *supervised* strategies that require large datasets of ground-truth data of the specific task to be learned. This methodology has been recently explored for many use cases, including 3D reconstruction [1, 2, 43, 62], garment design [46, 53, 55], animation [7, 17, 19, 31, 39, 56], and virtual try-on [8, 16, 45, 61]. To efficiently tackle the learning task, and depending on the goal of each method, different supervision terms and domains have been used. Most methods use direct 3D supervision at the vertex level [17, 39, 45, 53], but image-based 2D supervision in form of UV maps [20, 23, 46], point clouds [32, 44], or sketches [55] also exist. Very recently, implicit representations have shown impressive results on learning to deform humans [3, 12, 35] and dress avatars [11, 44, 51, 54].

Datasets are a fundamental piece to enable supervision, and most methods [5, 39, 45, 55] opt for synthetic data generated with physics-based simulators such as ARCSim [37] or Argus [25]. Alternatively, other methods [23, 31, 44, 50] use high-quality 3D scans obtained in expensive multi-camera setups [40, 60]. Despite the success of all these supervised methods for learning-based garments, relaying on ground-truth data to train the models is a major limitation due to the associated costs and hinders to create datasets.

Self-supervised strategies are the ideal alternative to circumvent the need for ground-truth data in learning-based methods [48]. Instead of relying on losses that evaluate prediction error based on the difference with respect to ground-truth samples, *self-supervised* methods use implicit properties of the training data (or domain) as a supervision signal [64]. This strategy is nowadays very popular in data-driven methods for image-based problems [26, 41, 63], however, almost all state-of-the-art approaches to learn 3D garment deformations rely on ground-truth data [17, 39, 45]. For 3D deformations tasks not related to garments, many works use physics laws or constraints as a supervision signal [52, 58, 64]. For example, Tompson *et al*. [52] enforce incompressibility constraints to learn to solve the system of equations required in physics-based fluid simulation, Xie *et*

al. [58] enforce temporal coherence of consecutive frames in fluid simulations to enhance detail, and Zhu *et al*. [64] incorporates the governing equations of the physical model (*i.e.*, Partial Differential Equations, PDEs) in the loss to learn image-based flow simulations.

Despite the significant progress in self-supervised learning, no previous works addresses the learning of 3D garments in self-supervised strategy, with just the notable and very recent exception of PBNS [6]. PBNS proposes to learn pose space deformations for garments by enforcing *static* physical consistency during the training of the model. We follow a similar underlying idea, but propose to use a full physics-based deformation scheme recast as an optimization problem to learn, for first time, a model for *dynamic* garment deformations with self-supervision only. Additionally, our approach learns shape-dependent effects and is able to cope with a material model that produces highly-realistic and finer wrinkles.

3. Method

Our goal is to find a function $M()$ that deforms a 3D garment given the underlying body parameters and motion. To this end, in Sec. 3.1, we first describe our garment model used to implement $M()$, which is based on per-vertex dynamic 3D displacements that are added to a rigged template mesh. Then, in Sec. 3.2, we direct our attention to an optimization-based formulation of dynamic deformations. Based on this formulation, in Sec. 3.3, we introduce our main contribution and describe a physics-based deformation model that allows us to train a regressor $R()$ for 3D garment displacements. Importantly, our loss is driven by fundamental physical properties of deformable objects, not by the reconstruction of ground-truth garments, and therefore it enables *self-supervised* learning. In Sec. 3.4 we specify the material model used in the different terms of our loss, and define the relevant energies such as the strain, and bending energies. Finally, in Sec. 3.5 we describe the recurrent architecture used to implement the regressor $R()$. See Figure 2 for an overview of our method.

3.1. Garment Model

Similar to state-of-the-art methods for data-driven garments [5, 17, 39, 45, 53], we leverage and extend existing human body models [13, 30] to encode garment deformations. More specifically, we build our representation on top of the popular SMPL human model [30]. SMPL encodes bodies by deforming a rigged human template according to shape and pose-dependent deformations that are learned from data. Following this idea, we define our garment model as

$$M(\beta, \phi) = W(T(\beta, \phi), J(\beta), \theta, \mathbf{W}_{\mathrm{G}}) \quad (1)$$

$$T(\beta, \phi) = \mathbf{T} + R(\beta, \phi) \quad (2)$$

where W is a skinning function (*e.g.*, linear blend skinning or dual quaternion) with skinning weights $\mathbf{W}_{\mathrm{G}}$, joint locations $J(\beta)$, and motion parameters ϕ that articulate an unposed deformed garment mesh $T(\beta, \phi)$. The latter is computed from a garment template mesh $\mathbf{T}$ deformed by a function $R(\beta, \phi)$ that outputs per-vertex 3D displacements to encode dynamic deformations conditioned to the underlying body shape β and body motion ϕ. The body motion ϕ contains the current body pose θ as well as the global velocity of the root joint.

Assuming that the garment template $\mathbf{T}$ is correctly located on top of the mean SMPL body mesh [30], we define $\mathbf{W}_{\mathrm{G}}$ by borrowing the SMPL skinning weights of closest body vertex in rest pose. In the remainder of this section we introduce our novel strategy to learn the 3D displacement regressor $R(\beta, \phi)$.

3.2. Optimization-Based Dynamic Deformation

Our goal is to learn the 3D displacement regressor $R(\beta, \phi)$ in Equation 2 using a self-supervised strategy. To this end, our first task is to find a set of physics-based properties that describe how cloth behaves. Physics-based simulators traditionally solve dynamics by applying a numerical integration scheme, *e.g.*, backward Euler, to the differential equations of motion, and finding the roots of the resulting nonlinear discrete equations [38]. This formulation is applied independently at each simulation frame, to iteratively update the positions and velocities of garment vertices. Our key observation is to realize that the solution to the equations of motion discretized with backward Euler can also be formulated as an optimization problem [34], and the objective function for this minimization can become the central ingredient of a self-supervised learning scheme. Optimization-based dynamics have been used in the Computer Graphics literature to increase the efficiency and robustness of dynamics solvers, through quasi-Newton schemes and step-size selection [14, 29]. Instead, we propose to leverage such optimization-based formulation to define a loss for training a neural network that generalizes well to any input (*i.e.,* any body shape and motion).

The equations of motion can be discretized with backward Euler as

$$\mathbf{M}\frac{\mathbf{x}^{t+1} - \mathbf{x}^t - \Delta t \mathbf{v}^t}{\Delta t^2} = \mathbf{f}\left(\mathbf{x}^{t+1}, \frac{\mathbf{x}^{t+1} - \mathbf{x}^t}{\Delta t}\right), \quad (3)$$

where $\mathbf{M}$ is the mass matrix, $\mathbf{f}$ are forces, and $\mathbf{x}$ and $\mathbf{v}$ are the positions and velocities of garment nodes. The solution to these equations can be recast as an optimization [14, 34]:

$$\mathbf{x}^{t+1} = \arg\min_{\mathbf{x}} \frac{1}{2\Delta t^2}(\mathbf{x} - \hat{\mathbf{x}})^\top \mathbf{M}(\mathbf{x} - \hat{\mathbf{x}}) + \mathbf{\Phi}, \quad (4)$$

where $\hat{\mathbf{x}} = \mathbf{x}^t + \Delta t \mathbf{v}^t$ is a tentative (explicit) position update, and $\mathbf{\Phi}$ is the potential energy due to internal and external forces $\mathbf{f}$ of the system.

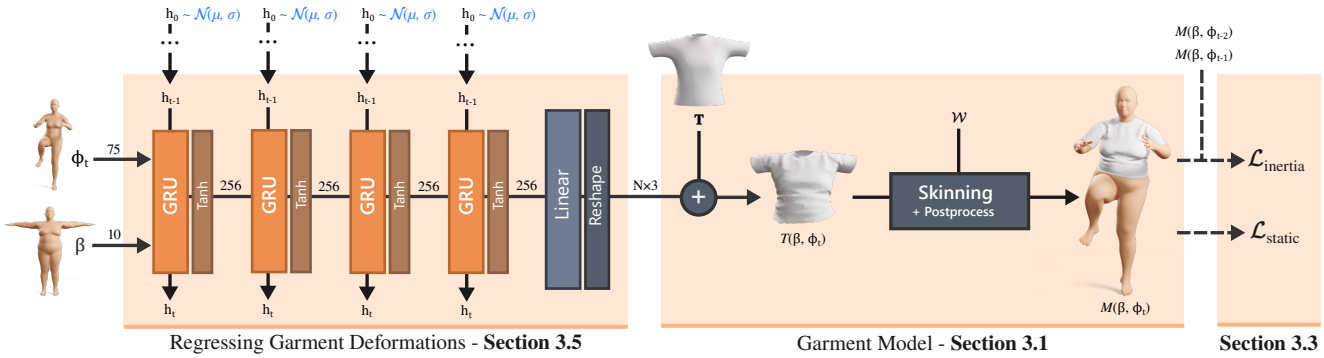


Figure 2. Overview of our method. First, the recurrent regressor predicts per-vertex offsets as a function of body shape and motion. These offsets are added to the garment template which is then skinned to produce the final result. We train the network by optimizing a set of physical properties of the predicted garments, removing need for ground-truth data.

3.3. Turning Dynamics into Self-Supervision

The key to our method is to define a set of losses based on Equation 4 to train the regressor $R()$. To this end, we propose a loss with two terms

$$\mathcal{L} = \mathcal{L}_{\text{inertia}} + \mathcal{L}_{\text{static}}, \tag{5}$$

where $\mathcal{L}_{\text{inertia}}$ models the inertia of the garment and it is defined analogous to the first term of Equation 4

$$\mathcal{L}_{\text{inertia}} = \frac{1}{2\Delta t^2}(\mathbf{x} - \hat{\mathbf{x}})^\top \mathbf{M}(\mathbf{x} - \hat{\mathbf{x}}). \tag{6}$$

Intuitively, this term prevents the change of garment velocities over time, but garment velocities will change anyway due to the underlying body motion, which makes dynamics and wrinkle effects appear.

$\mathcal{L}_{\text{static}}$, the second term of our loss $\mathcal{L}$, models the potential energy $\mathbf{\Phi}$ of Equation 4 which represents the internal and external forces that affect the garment. Inspired by works from cloth simulation literature [37, 47], we define $\mathcal{L}_{\text{static}}$ as the sum of different physics-based terms that model the energies that emerge on deformable solids, including strain, bending, gravity, and collisions

$$\mathcal{L}_{\text{static}} = \mathcal{L}_{\text{strain}} + \mathcal{L}_{\text{bending}} + \mathcal{L}_{\text{gravity}} + \mathcal{L}_{\text{collision}}. \tag{7}$$

This formulation of $\mathcal{L}_{\text{static}}$ is general, and the definition of each term depends on the material model used, which we detail in the next section.

3.4. Material Model

The literature of simulation of elastic solids characterizes materials using equations that relate stimuli (*e.g.*, deformations) to material response (*e.g.*, energies) [47]. Inspired by this, and with the goal of learning physically-correct garment behaviors, we define the terms of our static loss $\mathcal{L}_{\text{static}}$ based on equations of state-of-the-art cloth simulators [37] to model the following energies:

Membrane Strain Energy. The membrane strain term models the response of the material to in-plane deformation. Given a deformed position $x \in \mathbb{R}^3$ and an undeformed position $X \in \mathbb{R}^2$ (*i.e.*, the garment template), it defines an internal energy based on a first-order deformation metric, typically the deformation gradient $\mathbf{F} = \frac{\partial x}{\partial X}$. In our loss we implement it using the Saint Venant Kirchhoff (StVK) elastic material model that defines membrane strain energy as

$$\Psi_{\text{S}} = \frac{\lambda}{2}\text{tr}(\mathbf{G})^2 + \mu\text{tr}(\mathbf{G}^2), \tag{8}$$

where λ and μ are the Lamé constants, and $\mathbf{G} = \frac{1}{2}(\mathbf{F}^\top\mathbf{F} - \mathbf{I})$ is the Green strain tensor. The membrane strain energy of the mesh is computed as

$$\mathcal{L}_{\text{strain}} = \sum_{\text{triangles}} \mathbf{V}\Psi_{\text{S}}, \tag{9}$$

where $\mathbf{V}$ is the volume of each triangle (*i.e.,* area $\times$ thickness).

Bending Energy. The bending term models the energy due to the angle of two adjacent faces and we model it as

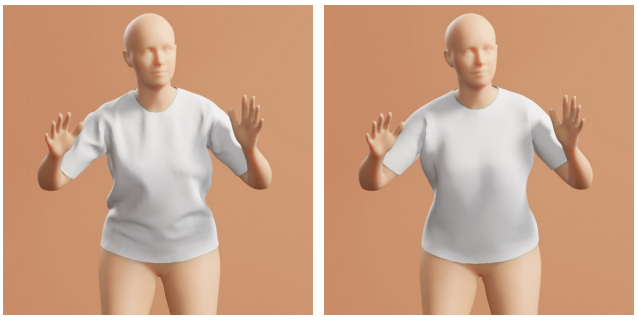
$$\mathcal{L}_{\text{bending}} = \sum_{\text{edges}} \frac{k_{\text{bending}}}{2}\theta^2 \tag{10}$$

where θ is the dihedral angle between the faces and k_{bending} is a bending stiffness.

Gravity. To model the effect of gravity in the learned deformations, we add a loss term with the potential energy of each cloth vertex

$$\mathcal{L}_{\text{gravity}} = \sum_{\text{vertices}} -m\,\mathbf{g}^\top x \tag{11}$$

where m is the vertex mass, and $\mathbf{g}$ is the gravitational acceleration.



(a) StVK (Ours) (b) Mass-spring (PBNS [6])

Figure 3. The material model used is crucial to obtain realistic garment behaviors. We formulate our losses using the Saint Venant Kirchoff (StVK) model, in contrast to simpler alternatives that lead to less expressive deformations.

Collision Penalty. This term is crucial to learn plausible deformations, enforcing the garment to follow the underlying body motion. We implement it as

$$\mathcal{L}_{\text{collision}} = \sum_{\text{vertices}} k_{\text{collision}} \max(\epsilon - d(x), 0)^3 \qquad (12)$$

where $d(x)$ is a function that computes the distance to the body, $k_{\text{collision}}$ is a collision stiffness, and ϵ is a safety margin to prevent the garment from overlapping with the body surface.

To highlight the realism of the proposed material, in Figure 3 we show a ground-truth simulation of our model, and the simpler material model used in PBNS [6] based on a traditional mass-spring formulation. Overall, our model is capable of reproducing more complex behaviors typically present in garments, including wrinkles and folds at different scales.

3.5. Regressing Garment Deformations

With our novel self-supervised loss $\mathcal{L}$ defined in Section 3.3, we are ready to train the garment displacement regressor $R()$ from Equation 2 without requiring ground-truth data. To this end, in order to model the time dependencies of the inertial term $\mathcal{L}_{\text{inertia}}$, we implement the regressor using 4 Gated Recurrent Units (GRU), each with an output of size 256, and `tanh` as the activation function (see Figure 2). However, the recurrent nature of GRUs combined with the lack of ground-truth values to guide the training process make the regressor converge to bad solutions if a naive recurrent training protocol is used. We need to take special care into how the hidden states of the GRUs are initialized and updated.

Intuitively, the model should be able to learn dynamics from just 3 frames, since $\mathcal{L}_{\text{inertia}}$ from Equation 6 depends only on the vertex positions and velocities of the previous step. Therefore, we train our network using sub-sequences of 3 frames. Interestingly, we found that training on longer sub-sequences also minimizes $\mathcal{L}_{\text{inertia}}$ correctly, but the learned deformations do not model true dynamics.

At runtime, the network supports sequences of arbitrary length, but results can degrade noticeably for sequences longer than those used in training if initialization of the GRU hidden states is not well handled. More specifically, we observe that for each training sub-sequence, setting the initial hidden states $\mathbf{h}_0 = 0$ hinders the network to generalize to sequences longer than 3 frames. We address this issue by sampling the initial state $\mathbf{h}_0$ of each GRU from $\mathcal{N}(\mu, \sigma)$ (empirically, $\mu = 0$ and $\sigma = 0.1$), which allows the model to generalize well even for sequences with thousands of frames. Notice that at runtime the state $\mathbf{h}_t$ depends on an arbitrarily large number of previous frames, not just the last 3, hence the use of noise to initialize states on train sub-sequences is fundamental to augment variance in states.

4. Evaluation

4.1. Training

To self-supervise the training process of our regressor $R()$ we need to feed it with human motions and shapes. To this end, we use a set of 52 sequences from the AMASS dataset [33], totaling 6,519 frames, which we split into sub-sequences of 3 frames as described in Section 3.5. We set aside 4 full sequences for validation purposes. To provide body shape variety at train time, each of the sub-sequences is assigned a different body shape β sampled from $\mathcal{U}(-3, 3)$ at each epoch. Notice that, enabled by our self-supervised approach, this strategy allows us to train using thousands of different body shapes, while competitive supervised methods are limited to a dramatically smaller shape sample ([39] uses 9 shapes, [45] uses 17) due to the computational restrictions caused by the need for a ground-truth database.

Regarding the network hyper-parameters, we use a batch size of 16, initially train for 10 epochs using a learning rate of 0.001, and then resume the learning with a learning rate of 0.0001 until it converges. This approach is fast, works for all garments, and avoids erroneous states. The rest of the material and training parameters do not affect stability. Larger learning rates can introduce instabilities due to energy spikes that make the training struggle to recover (*i.e.* the predicted mesh has collisions that are too large to be resolved). Small body-garment collisions are not a problem – *e.g.*, we can handle pants despite self-collisions in the legs on some poses.

Our approach does not require balancing loss terms, we just need to set the material properties of the garment. To this end, we tune material parameters to produce a desired fabric behavior, hence the parameters of the loss have a physical meaning – they are not arbitrary hyperparame-

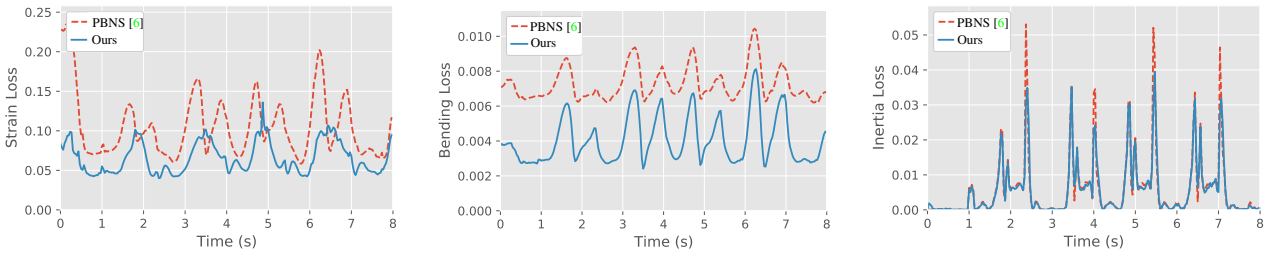


Figure 4. Quantitative evaluation of our approach. We evaluate the minimization error in different the physics-based terms used in our loss, in the test sequence `01_01` of AMASS [33]. Sudden motion changes (*e.g.*, jumps) naturally produce peaks in the inertial term, due to drastic changes in the velocity of the garment. Intuitively, cloth dynamics arise when the garment resists to those changes enforced by the body, therefore lower inertial values indicates that our model better learns time-dependent effects than PBNS [6]. See the supplementary video for qualitative results of this evaluation. The significantly improved realism of our method is better appreciated looking into sequences of deformed garments.

	Strain	Bending	Gravity	Inertia
PBNS [6]	0.111	0.007	0.044	0.0035
SNUG (Ours)	**0.064**	**0.004**	**0.028**	**0.0034**

Table 1. To quantitative evaluate our method we compute the physics-based loss terms of our trained model, in unseen sequences, and compare to PBNS. We produce lower errors in all terms, indicating that our approach results in deformations that better match physics-based simulators.

	W/o bending	W/o strain	W/o gravity	W/o inertia	Full
Mean-curvature error	17.3	19.1	7.5	2.8	**2.7**

Table 2. Quantitative ablation study. Each term of our loss contributes to the accuracy of the final result.

ters. To compute the mass matrix $\mathbf{M}$ we use real measurements of the thickness and density of 100% cotton fabric (0.47 mm and 426 kg/m^3 respectively). The rest of the material parameters have the following values: the Lamé constants are set to $\lambda = 20.9$ and $\mu = 11.1$, the bending stiffness $k_{\text{bending}} = 3.96\text{e}{-5}$, the collision stiffness $k_{\text{collision}} = 250$, and the collision margin $\epsilon = 2$ mm. We use the same parameters for all our garments.

To thoroughly validate our model, in addition to comparisons to SOTA methods, in this section we also include ablations and comparisons that use a ground-truth simulated dataset. For as fair as possible evaluations, such dataset is created using the same motions, and the same train-test split, that we use to train SNUG.

We implement our method in a regular desktop PC equipped with an AMD Ryzen 7 2700 CPU, an Nvidia GTX 1080 Ti GPU, and 32GB of RAM.

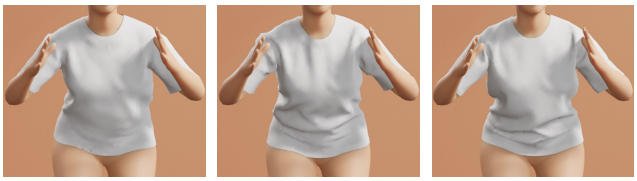
	Data generation	Train	Runtime	Memory
TailorNet [39]	29 h	6.5 h	10.1 ms	2114 MB
Santesteban [45]	180 h	17 h	2.5 ms	109 MB
SNUG (Ours)	**0 h**	**2 h**	**2.2 ms**	**19 MB**

Table 3. Timings, memory requirements, and performance of state-of-the-art methods. Our self-supervised approach avoids the expensive cost of data generation, while also achieving significantly lower training times.

4.2. Quantitative Evaluation

To quantitatively evaluate our approach, we measure the physics-based terms of our loss $\mathcal{L}$ in test motions and compare it with the predictions of PBNS [6]. Notice that the original PBNS method uses a different (and simpler) material model but, in order to get a meaningful quantitative comparison, we extended and re-trained the publicly available PBNS implementation with our material model defined in Section 3.4. Also, notice that we cannot provide this comparison for supervised state-of-the-art methods (*e.g.*, [17, 39, 45]) because the simulation schemes, material models, and parameters used to build their datasets are different and, therefore, the ground-truth physics properties (*i.e.*, our loss terms) might differ significantly.

Figure 4 shows the quantitative evaluation for the most important terms of our loss, and compares it with the extended implementation of PBNS [6] using our material, in the test sequence `01_01` of AMASS [33]. Notice how our method consistently produces lower error values across all terms (strain, bending, and inertia), indicating that test samples processed with SNUG better match the behavior of physics-based solutions (*i.e.*, the minimization of the terms). Table 1 presents a quantitative evaluation of both methods in our full test set (4 sequences, 598 frames unseen



(a) Supervised (b) Ours (c) Simulation

Figure 5. When trained using same motions and same architecture, direct supervision at the vertex level leads to smoothing artifacts (a). In contrast, our physics-based loss is able to learn more realistic details (b), as shown in this frame from a test sequence.

at train time), which further demonstrates that our approach improves upon the method of PBNS.

To validate each term of our formulation, in Table 2 we show an ablation of the mean-curvature error, evaluated in the test set of our ground truth simulated dataset, when leaving out some of the terms.

Finally, in Table 3 we also evaluate the memory requirements, training time, and runtime performance of our approach and compare to existing state-of-the-art supervised methods. Even if these methods do no address exactly the same problem (*e.g.* TailorNet [39] models garment variations and SNUG does not, but the latter models dynamics), SNUG outperforms supervised methods by a large margin in all metrics, resulting in a compact model, only 19MB, trained in just 2h, which opens the door to scalable learning-based garment models.

4.3. Qualitative Evaluation

We qualitatively evaluate our method in Figure 7 and, more extensively, in the supplementary video. To this end, notice that we always use body shapes and motions unseen during training. Additionally, we provide comparisons to the state-of-the-art *supervised* methods of Santesteban *et al.* [45] and TailorNet [39], as well as to the recent work PBNS [6] that uses physics-constraints as supervision. To ease the assessment of the realism of each method, we also show results computed with a physics-based simulator [37], but notice that this is a traditional offline method, several orders of magnitude slower.

These results demonstrate that our self-supervised method SNUG produces garment deformations that are, at least, on par with the state-of-the-art *supervised* methods [39, 45], while we do not require *any* ground-truth dataset. For PBNS [6], we use a mean body shape because it does not generalize to different bodies. Because PBNS does not model an inertial term and it is limited to a simpler material model, the garment deformations are generally more stiff, less realistic, and do not change naturally as a function of body pose. This is visible in rows 1 and 3 for PBNS in Figure 7, where the overall wrinkles are the same despite the significant change in body pose.

To further validate our model, we use the ground-truth simulated dataset (described in Section 4.1, used for validation purposes only) to retrain our neural network in a per-vertex supervised manner. In Figure 5 and in the supplementary we qualitatively demonstrate that the self-supervised method learns more detailed wrinkles than the supervised counterpart trained with exactly the same motions.

Additionally, in Figure 6 we show more results for a variety of garments learned with our approach, including t-shirts, tops, sleeveless shirts, pants, and shorts, worn by different body shapes. Notice how our approach produces different wrinkles for each garment type, pose, and shape combination, demonstrating the generalization capabilities of our self-supervised approach. For this figure, we trained one regressor for each garment type. In the supplementary video you can see animated results of these garments, showcasing for the first time realistic dynamic deformations of self-supervised learning-based garments.

Finally, in the supplementary video we also show an ablation study of the influence of each term of our loss function. We demonstrate that all terms of Equation 5 contribute to improve the realism of our predicted garments.

5. Limitations and Conclusion

We believe SNUG makes an important step towards efficient learning-based models for 3D garments. To improve the state-of-the-art, instead of following the standard route of training with more data, adding more explicit supervision, or designing more complex architectures, we show that self-supervision based on physical properties of deformable solids leads to simpler and smaller yet highly-realistic models.

While our physics-based loss terms are the fundamental key to self-supervision, we also want to point out that our strategy of exploiting optimization-based schemes (originally derived for simulation problems) to train a neural network carries a few weaknesses and important considerations to take into account.

Specifically, we notice that the self-supervised network tends to converge to simpler solutions than a traditional simulator. For example, although our approach is capable of learning pose- and shape-dependent wrinkles and overall dynamics, we struggle to predict fine-level dynamics. We hypothesize that this limitation arises from a fundamental difference in how our method works: while standard simulators solve physics for one frame at a time, our model optimizes thousands of frames simultaneously during training. This makes our approach more prone to converge to simpler local minima. Nevertheless, we want to highlight that, despite this limitation, the cloth dynamics learned by our method are on par with other data-driven approaches.

Another aspect open to future research is the collision

Figure 6. Qualitative results of our self-supervised method, in validation body shapes and poses unseen during training. SNUG successfully learns highly-realistic garment deformations, including fine wrinkles, as a function of body shape and motion.

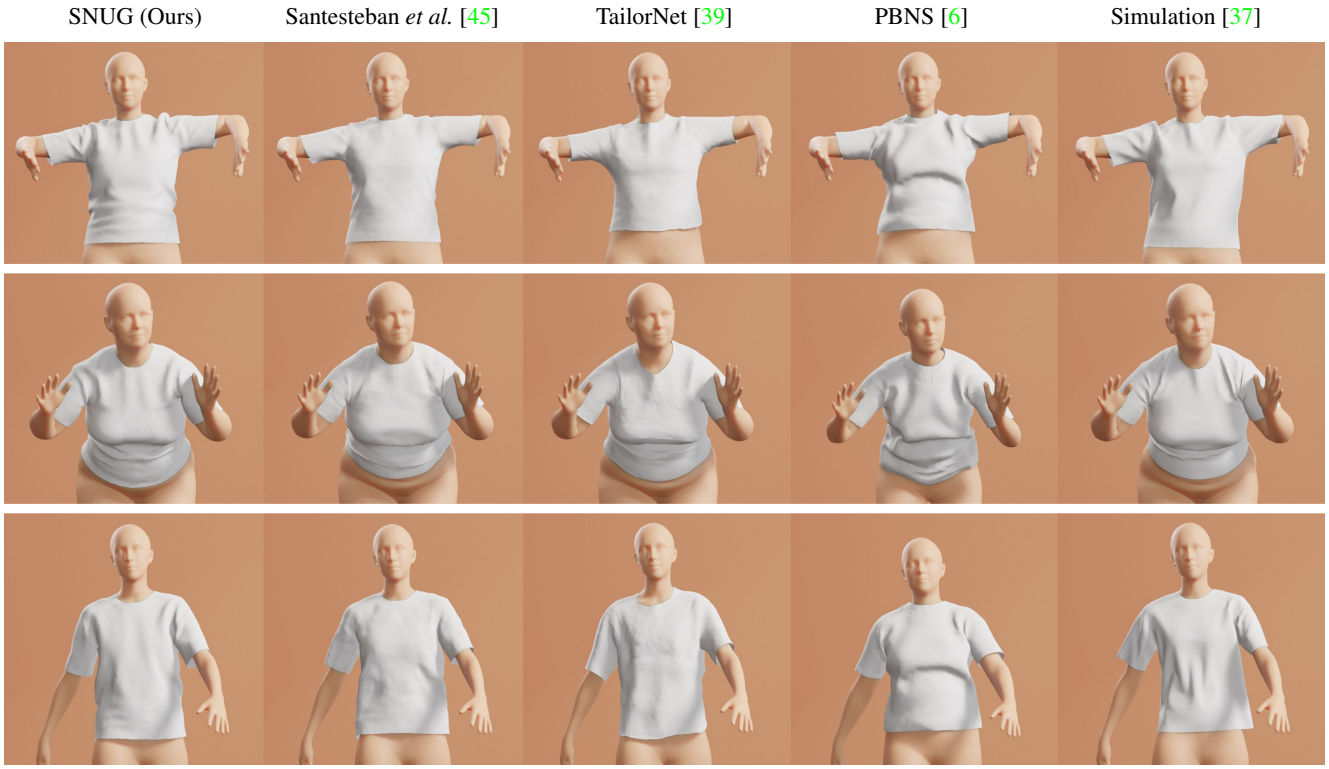


Figure 7. Qualitative comparison with state-of-the-art methods. SNUG generalizes well to unseen body shapes and motions, and produces detailed folds and wrinkles. The results of SNUG are, at least, on par with the realism of *supervised* methods that require large datasets [39, 45], and close to state-of-the-art *offline physics-based* simulation [37].

handling. Although our loss penalizes collisions between the garment and the body in train samples, we found noticeable collisions in test motions. Although these collisions can be efficiently solved with a postprocessing step, we believe it would be valuable to explore ways to enforce this constraint on the network. Addressing self-collisions of the garment is another aspect that would be worth taking into consideration.

To foster future research on the field, our trained models and the code to run them are available on http://mslab.es/projects/SNUG.

Acknowledgments. The work was funded in part by the European Research Council (ERC Consolidator Grant no. 772738 TouchDesign) and Spanish Ministry of Science (RTI2018-098694-B-I00 VizLearning).

References

[1] Thiemo Alldieck, Marcus Magnor, Bharat Lal Bhatnagar, Christian Theobalt, and Gerard Pons-Moll. Learning to Reconstruct People in Clothing from a Single RGB Camera. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[2] Thiemo Alldieck, Marcus Magnor, Weipeng Xu, Christian Theobalt, and Gerard Pons-Moll. Detailed Human Avatars from Monocular Video. In *Proc. of International Conference on 3D Vision (3DV)*, pages 98–109. IEEE, 2018. 2

[3] Thiemo Alldieck, Hongyi Xu, and Cristian Sminchisescu. imGHUM: Implicit Generative Models of 3D Human Shape and Articulated Pose. In *Proc. of IEEE International Conference on Computer Vision (ICCV)*, 2021. 2

[4] David Baraff and Andrew Witkin. Large Steps in Cloth Simulation. In *Proc. of Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, page 43–54, 1998. 2

[5] Hugo Bertiche, Meysam Madadi, and Sergio Escalera. CLOTH3D: Clothed 3D Humans. In *Proc. of European Conference on Computer Vision (ECCV)*, 2020. 2, 3

[6] Hugo Bertiche, Meysam Madadi, and Sergio Escalera. PBNS: Physically Based Neural Simulation for Unsupervised Garment Pose Space Deformation. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 40(6), dec 2021. 3, 5, 6, 7, 8

[7] Hugo Bertiche, Meysam Madadi, Emilio Tylson, and Sergio Escalera. DeePSD: Automatic Deep Skinning and Pose Space Deformation for 3D Garment Animation. In *Proc. of IEEE International Conference on Computer Vision (ICCV)*, 2021. 2

[8] Bharat Lal Bhatnagar, Garvita Tiwari, Christian Theobalt, and Gerard Pons-Moll. Multi-garment net: Learning to Dress 3D People from Images. In *Proc. of IEEE International Conference on Computer Vision (ICCV)*, pages 5420–5430, 2019. 2

[9] Robert Bridson, Ronald Fedkiw, and John Anderson. Robust treatment of collisions, contact and friction for cloth animation. *ACM Trans. Graph.*, 21(3):594–603, 2002. 2

[10] Gabriel Cirio, Jorge Lopez-Moreno, David Miraut, and Miguel A Otaduy. Yarn-level simulation of woven cloth. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 33(6):1–11, 2014. 2

[11] Enric Corona, Albert Pumarola, Guillem Alenyà, Gerard Pons-Moll, and Francesc Moreno-Noguer. SMPLicit: Topology-aware Generative Model for Clothed People. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[12] Boyang Deng, John P Lewis, Timothy Jeruzalski, Gerard Pons-Moll, Geoffrey Hinton, Mohammad Norouzi, and Andrea Tagliasacchi. NASA: Neural Articulated Ahape Approximation. In *Proc. of European Conference on Computer Vision (ECCV)*, 2020. 2

[13] Andrew Feng, Dan Casas, and Ari Shapiro. Avatar Reshaping and Automatic Rigging Using a Deformable Model. In *Proc. of ACM SIGGRAPH Conference on Motion in Games (MIG)*, pages 57–64, 2015. 3

[14] Theodore F Gast, Craig Schroeder, Alexey Stomakhin, Chenfanfu Jiang, and Joseph M Teran. Optimization Integrator for Large Time Steps. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 21(10):1103–1115, 2015. 3

[15] Eitan Grinspun, Anil N. Hirani, Mathieu Desbrun, and Peter Schröder. Discrete Shells. In *Proc. of ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA)*, page 62–67, 2003. 2

[16] Peng Guan, Loretta Reiss, David A Hirshberg, Alexander Weiss, and Michael J Black. DRAPE: DRessing Any PErson. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 31(4), 2012. 2

[17] 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 *Proc. of IEEE International Conference on Computer Vision (ICCV)*, 2019. 1, 2, 3, 6

[18] Yuanming Hu, Luke Anderson, Tzu-Mao Li, Qi Sun, Nathan Carr, Jonathan Ragan-Kelley, and Frédo Durand. DiffTaichi: Differentiable Programming for Physical Simulation. *ICLR*, 2020. 2

[19] Zeng Huang, Yuanlu Xu, Christoph Lassner, Hao Li, and Tony Tung. ARCH: Animatable Reconstruction of Clothed Humans. In *Proc. of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3093–3102, 2020. 2

[20] Ning Jin, Yilin Zhu, Zhenglin Geng, and Ron Fedkiw. A Pixel-Based Framework for Data-Driven Clothing. *Computer Graphics Forum (Proc. of SCA)*, 2020. 2

[21] Jonathan M. Kaldor, Doug L. James, and Steve Marschner. Simulating knitted cloth at the yarn level. *ACM Trans. Graph.*, 27(3):1–9, 2008. 2

[22] Theodore Kim. A finite element formulation of baraff-witkin cloth. *Computer Graphics Forum*, 39(8):171–179, 2020. 2

[23] Zorah Lahner, Daniel Cremers, and Tony Tung. DeepWrinkles: Accurate and Realistic Clothing Modeling. In *Proc. of European Conference on Computer Vision (ECCV)*, 2018. 2

[24] John P Lewis, Matt Cordner, and Nickson Fong. Pose Space Deformation: A Unified Approach to Shape Interpolation and Skeleton-Driven Deformation. In *Proc. of the Annual Conference on Computer Graphics and Interactive Techniques*, pages 165–172, 2000. 2

[25] Jie Li, Gilles Daviet, Rahul Narain, Florence Bertails-Descoubes, Matthew Overby, George E Brown, and Laurence Boissieux. An Implicit Frictional Contact Solver for Adaptive Cloth Simulation. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 37(4):1–15, 2018. 2

[26] Xueting Li, Sifei Liu, Kihwan Kim, Shalini De Mello, Varun Jampani, Ming-Hsuan Yang, and Jan Kautz. Self-supervised single-view 3d reconstruction via semantic consistency. In *Proc. of European Conference on Computer Vision (ECCV)*, 2020. 2

[27] Yue Li, Marc Habermann, Bernhard Thomaszewski, Stelian Coros, Thabo Beeler, and Christian Theobalt. Deep Physics-aware Inference of Cloth Deformation for Monocular Human Performance Capture. In *Proc. of International Conference on 3D Vision (3DV)*, 2021. 2

[28] Junbang Liang, Ming Lin, and Vladlen Koltun. Differentiable Cloth Simulation for Inverse Problems. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 771–780, 2019. 2

[29] Tiantian Liu, Sofien Bouaziz, and Ladislav Kavan. Quasi-Newton Methods for Real-Time Simulation of Hyperelastic Materials. *ACM Trans. Graph.*, 36(4), 2017. 3

[30] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 34(6):1–16, 2015. 3

[31] Qianli Ma, Jinlong Yang, Anurag Ranjan, Sergi Pujades, Gerard Pons-Moll, Siyu Tang, and Michael J. Black. Learning to Dress 3D People in Generative Clothing. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, 2020. 1, 2

[32] Qianli Ma, Jinlong Yang, Siyu Tang, and Michael J Black. The power of points for modeling humans in clothing. In *Proc. of IEEE International Conference on Computer Vision (ICCV)*, 2021. 2

[33] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of Motion Capture as Surface Shapes. In *Proc. of IEEE International Conference on Computer Vision (ICCV)*, pages 5442–5451, Oct. 2019. 5, 6

[34] Sebastian Martin, Bernhard Thomaszewski, Eitan Grinspun, and Markus Gross. Example-Based Elastic Materials. *ACM Trans. Graph.*, 30(4), 2011. 2, 3

[35] Marko Mihajlovic, Yan Zhang, Michael J Black, and Siyu Tang. LEAP: Learning Articulated Occupancy of People. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[36] Juan Montes, Bernhard Thomaszewski, Sudhir Mudur, and Tiberiu Popa. Computational Design of Skintight Clothing. *ACM Trans. Graph.*, 39(4), 2020. 2

[37] Rahul Narain, Armin Samii, and James F O'brien. Adaptive Anisotropic Remeshing for Cloth Simulation. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 31(6):1–10, 2012. 1, 2, 4, 7, 8

[38] Andrew Nealen, Matthias Müller, Richard Keiser, Eddy Boxerman, and Mark Carlson. Physically Based Deformable Models in Computer Graphics. *Computer Graphics Forum*, 25(4):809–836, 2006. 1, 3

[39] Chaitanya Patel, Zhouyingcheng Liao, and Gerard Pons-Moll. The Virtual Tailor: Predicting Clothing in 3D as a Function of Human Pose, Shape and Garment Style. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, 2020. 1, 2, 3, 5, 6, 7, 8

[40] Gerard Pons-Moll, Sergi Pujades, Sonny Hu, and Michael J. Black. ClothCap: Seamless 4D clothing capture and retargeting. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 36(4), 2017. 1, 2

[41] Amit Raj, Patsorn Sangkloy, Huiwen Chang, Jingwan Lu, Duygu Ceylan, and James Hays. SwapNet: Image Based Garment Transfer. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 666–682, 2018. 1, 2

[42] Cristian Romero, Miguel A. Otaduy, Dan Casas, and Jesus Perez. Modeling and Estimation of Nonlinear Skin Mechanics for Animated Avatars. *Computer Graphics Forum (Proc. Eurographics)*, 39(2), 2020. 2

[43] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. PIFu: Pixel-Aligned Implicit Function for High-Resolution Clothed Human Digitization. In *Proc. of IEEE International Conference on Computer Vision (ICCV)*, 2019. 2

[44] Shunsuke Saito, Jinlong Yang, Qianli Ma, and Michael J. Black. SCANimate: Weakly supervised learning of skinned clothed avatar networks. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[45] Igor Santesteban, Miguel A. Otaduy, and Dan Casas. Learning-Based Animation of Clothing for Virtual Try-On. *Computer Graphics Forum (Proc. Eurographics)*, 38(2), 2019. 1, 2, 3, 5, 6, 7, 8

[46] Yu Shen, Junbang Liang, and Ming C. Lin. GAN-based Garment Generation Using Sewing Pattern Images. In *Proc. of European Conference on Computer Vision (ECCV)*, 2020. 1, 2

[47] Eftychios Sifakis and Jernej Barbic. FEM simulation of 3D deformable solids: a practitioner's guide to theory, discretization and model reduction. In *SIGGRAPH 2012 Courses*, pages 1–50. ACM, 2012. 4

[48] Russell Stewart and Stefano Ermon. Label-Free Supervision of Neural Networks with Physics and Domain Knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, page 2576–2582, 2017. 2

[49] Min Tang, Tongtong Wang, Zhongyuan Liu, Ruofeng Tong, and Dinesh Manocha. I-Cloth: Incremental Collision Handling for GPU-Based Interactive Cloth Simulation. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 37(6), 2018. 2

[50] 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 *Proc. of European Conference on Computer Vision (ECCV)*, 2020. 1, 2

[51] Garvita Tiwari, Nikolaos Sarafianos, Tony Tung, and Gerard Pons-Moll. Neural-GIF: Neural Generalized Implicit Functions for Animating People in Clothing. In *Proc. of IEEE International Conference on Computer Vision (ICCV)*, 2021. 2

[52] Jonathan Tompson, Kristofer Schlachter, Pablo Sprechmann, and Ken Perlin. Accelerating Eulerian Fluid Simulation With Convolutional Networks. In *International Conference on Machine Learning (ICML)*, pages 3424–3433, 2017. 2

[53] Raquel Vidaurre, Igor Santesteban, Elena Garces, and Dan Casas. Fully Convolutional Graph Neural Networks for Parametric Virtual Try-On. *Computer Graphics Forum (Proc. SCA)*, 39(8), 2020. 1, 2, 3

[54] Shaofei Wang, Marko Mihajlovic, Qianli Ma, Andreas Geiger, and Siyu Tang. MetaAvatar: Learning Animatable Clothed Human Models from Few Depth Images. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 2

[55] Tuanfeng Y Wang, Duygu Ceylan, Jovan Popović, and Niloy J Mitra. Learning a Shared Shape Space for Multimodal Garment Design. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 37(6), 2018. 1, 2

[56] 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 (Proc. SIGGRAPH Asia)*, 38(6), 2019. 2

[57] Zhonghua Wu, Guosheng Lin, Qingyi Tao, and Jianfei Cai. M2E-Try On Net: Fashion from Model to Everyone. In *Proc. of ACM International Conference on Multimedia*, 2019. 1

[58] You Xie, Erik Franz, Mengyu Chu, and Nils Thuerey. tempoGAN: A Temporally Coherent, Volumetric GAN for Super-resolution Fluid Flow. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 37(4), 2018. 2, 3

[59] Tao Yu, Zerong Zheng, Yuan Zhong, Jianhui Zhao, Qionghai Dai, Gerard Pons-Moll, and Yebin Liu. Simulcap: Single-view human performance capture with cloth simulation. In *Proc. of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[60] Chao Zhang, Sergi Pujades, Michael J Black, and Gerard Pons-Moll. Detailed, accurate, human shape estimation from clothed 3d scan sequences. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, pages 4191–4200, 2017. 2

[61] Fuwei Zhao, Zhenyu Xie, Michael C. Kampffmeyer, Haoye Dong, Songfang Han, Tianxiang Zheng, Tao Zhang, and Xiaodan Liang. M3D-VTON: A Monocular-to-3D Virtual Try-On Network. In *Proc. of IEEE International Conference on Computer Vision (ICCV)*, 2021. 2

[62] Heming Zhu, Yu Cao, Hang Jin, Weikai Chen, Dong Du, Zhangye Wang, Shuguang Cui, and Xiaoguang Han. Deep Fashion3D: A Dataset and Benchmark for 3D Garment Reconstruction from Single Images. In *Proc. of European Conference on Computer Vision (ECCV)*, 2020. 2

[63] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks . In *Proc. of IEEE International Conference on Computer Vision (ICCV)*, pages 2223–2232, 2017. 2

[64] Yinhao Zhu, Nicholas Zabaras, Phaedon-Stelios Koutsourelakis, and Paris Perdikaris. Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data. *Journal of Computational Physics*, 394:56–81, 2019. 2, 3