DOCTORAL THESIS

Triangulation-Agnostic Data-Driven Modeling and Animation of 3D Clothing

Author:

Raquel Vidaurre Gallart

Supervisors:

Dan Casas Guix
Elena Garcés García

Doctoral Program in Information and Communication Technologies

International Doctoral School

Madrid, 2024

Agradecimientos

El camino de esta tesis ha sido largo y duro, y jamás habría llegado hasta aquí sin el apoyo de muchas personas.

Gracias a Dan por su supervisión y apoyo. Tu actitud positiva me ha levantado el ánimo en incontables ocasiones en las que yo pensaba que todo estaba fatal. Y a Elena, gracias por guiarme y poner tus conocimientos al servicio de esta investigación. He aprendido mucho con vosotros. Gracias también a Jorge, que fue quien me dio la primera oportunidad de entrar a investigar en este grupo, y a Miguel Ángel, que, como jefe del grupo, ha gestionado eventos, *reading groups*, viajes y papeleos.

Gracias a todos los compañeros que me he ido encontrando de GMRV y MSLab, gente con la que he tenido la suerte de compartir despacho, cafés, comidas, cenas, excursiones y cervezas. Todos habéis hecho del trabajo en la uni algo mucho más agradable, y estoy encantada de haberme encontrado con vosotros. Unos cuantos os habéis convertido en amigos y espero que sigáis formando parte de mi vida mucho tiempo.

Gracias a todos mis amigos. Tengo la suerte de tener tantos, que me da miedo nombrarlos y dejarme alguno. Amigos en Madrid, en Valencia y esparcidos por el mundo, amigos de hace 30 años y de hace 2... Los buenos ratos con todos vosotros han sido gasolina para tirar cuando la vida se me ha hecho muy bola. Gracias por formar parte de mi vida. Muchos de ellos, además, son doctores y no sólo han sido una referencia, sino que se han esforzado en aconsejarme, motivarme y acompañarme en este proceso. Gracias!

Gracias al perreo, el compañero más inesperado de esta época. Quién me iba a decir que, "sin ser nada de eso yo", bailar moviendo el culo se iba a convertir en mi mayor vía de escape. Jamás habría creído que esto podría pasar, pero, mira tú por dónde,

no sólo me lo paso genial, también ha cambiado completamente mi relación con mi cuerpo. Gracias también a las perras por los ratitos buenos.

Gracias a Wid, que es negacionista del trabajo y mientras escribo estas líneas me muerde un pie porque quiere que le haga caso. Puede ser que no haya ayudado mucho a que lleve la tesis a término, atrapándome cuando tenía que trabajar o intentando tumbarse encima del portátil... Pero me hizo una compañía increíble cuando la oficina pasó a estar en mi salón y le quiero a rabiar.

Gracias especial también a todas las personas con las que he tenido la suerte de compartir la que había sido a mi casa y pasó a ser nuestra. Colores, Kala, Fuetlovers, habéis sido hogar en tiempos complicados. Mil millones de gracias.

Gracias a ~~Gloria~~ amor, que no sé muy bien qué pasó ahí, pero en cuestión de 4 días le diste un vuelco a mi vida y pasaste a ser una parte esencial. Muchísimas gracias por haberme dado tanto apoyo logístico y emocional en la última etapa de la tesis, pero, sobre todo, por hacerme cada día más bonito. Te quiero vuchísimo.

Gracias a Isa que, bueno, se puede dar por aludida porque estaba indirectamente mencionada en prácticamente todos los párrafos anteriores. Has formado parte del desarrollo de esta tesis desde antes de empezar, en los cafés, el yoga y los patitos de goma, hasta ahora, que estando muy lejos has seguido estando ahí. Teta, qué suerte tenerte, no sólo eres una hermana, eres una amiga, una compañera y una referenta. ¡Te quiero! #loheintentadoloheconseguido :p

Gracias a mis padres, que han apostado siempre por nosotras, por nuestra formación y por nuestro bienestar. Sois los que más fuerte creéis en mis capacidades y estáis siempre ahí cuando os necesito. Mamá, gracias por ser mi *couchin* favorita. Eres, sin duda, la persona a la que más veces le he dicho que iba a abandonar la tesis... y eres también la persona por la que no lo he hecho. Os quiero. <3

Abstract

Clothing is fundamental in society, serving as both physical protection and a means of communication, impacting how we are perceived and interact with others. The fashion industry, a major global economic force, benefits from the rise of digital fashion, which allows for the design of virtual garments for avatars and enhances user experiences in entertainment. Digital cloth simulation can revolutionize the industry by enabling designers to visualize and modify garments in virtual space, thus speeding up the design process and reducing waste. Virtual try-on applications offer significant advantages for online shopping, providing convenience, personalized recommendations, and reducing returns, contributing to a more sustainable fashion industry.

In this context, animating clothing remains a longstanding goal in Computer Graphics. While traditional physics-based simulations achieve realistic results, they are computationally expensive. Recent advances in data-driven models offer faster alternatives but face challenges in handling 3D garment representations. This thesis addresses these limitations by exploring the use of deep learning architectures for cloth animation that are agnostic to mesh discretization and that generalize across different designs and body shapes.

To this end, we propose two novel approaches to generate cloth deformations. Our first contribution provides a fully-convolutional pipeline that fits garments from a parametric design space to a target body shape. The convolutional nature of the method enables the optimized generation of draped garments to various body types and designs without predefined topologies. Our second contribution takes advantage of successful generative models that work on image domains to generate displacement maps encoding deformations, which enables the creation of high-frequency details. This model, conditioned on body shape, pose, and design parameters, produces temporally-coherent deformations for animation sequences.

Our methods show the potential of data-driven models to generalize to different garment designs. Such technologies offer a solution for scalable and efficient simulation of 3D clothing.

Contents

Figures

1 Introduction

Clothing plays a crucial role in our society. Although it started as a form of physical protection and comfort, it soon developed into a way of communication. How we dress substantially impacts how we are perceived, and consequently how we interact and connect. From a single glance at a person, thanks to how the person is dressed, we can identify, for example, status, intentions, cultural background, profession, and even creativity. Altogether, our clothes are a way to express our individual and social identity, and they are a key ingredient of our interactions with our community.

Given the importance of clothing in our society, it is to be expected that the fashion industry is a significant part of the global economy. It is not casual, that some of the wealthiest people in the world started their fortune as clothing retailers. The production, distribution, and consumption of clothing contribute to economic growth and employment opportunities worldwide.

Virtual clothing refers to a digital representation of cloth. The virtual garment doesn't necessarily exist as a physical object but as a visual representation created with computer graphics. Given the impact of clothing and fashion in our world, we can only imagine the potential benefit of virtual clothing.

Digital fashion is emerging as the design of clothes to dress virtual avatars, allowing personal expression and identity in the online world. In the field of entertainment, virtual garments can significantly enrich the user experience. Not only does it improve the immersion, but it can also be very decisive in the creation of believable characters, by providing them with a context and an identity. In movies and video games, costumes are a crucial part of the setting, and they greatly affect the viewer's feeling of immersion. Therefore, the design and simulation of virtual clothing are very important for games, animation, and VFX production. For animated films, it is paramount to get visually pleasing and realistic simulations, while video games will look for scalability, responsiveness, and interactivity. Efficiency in the simulation

is then crucial in some environments, especially in virtual and augmented reality scenarios.

Digital cloth simulation has the potential to revolutionize the fashion industry, as it allows designers to visualize garments in virtual space. Prototyping with physical clothes can be time- and resources-consuming. At the same time, with simulations, they can tweak fabrics, fits, and cuts and see how it will affect the final product in almost real-time, thus accelerating the design process and reducing waste. Accuracy is crucial in this kind of application, as simulations need to be as close to real-world fabrics as possible.

On the other end of the fashion industry, with the rise of online shopping, virtual try-on applications allow customers to see how clothes fit their bodies. Physical try-on requires customers to visit stores and involves searching for garments with their style and size, changing between the store and the fitting rooms, and often waiting in lines. While some customers enjoy it, this process can be time-consuming and inconvenient, especially for people living in remote areas, or with uncommon sizes or necessities, that may have difficulties finding clothes that fit them. Meanwhile, virtual try-on allows customers to try garments from the comfort of their homes, eliminating the need to travel to the shop, thus saving time and money. It also has the potential to create recommendations based on style and shape, giving a more personal experience, and simultaneously encouraging the users to experiment with their styles. Seeing the actual fit of the garment increases confidence in online shopping and greatly reduces returns, benefiting both customers and companies. Besides, having a unified inventory reduces the need for physical samples and the waste of unsold garments, thus helping make the fashion industry a little bit more sustainable.

The simulation of virtual clothing has been a longstanding goal in the field of Computer Graphics, due to its wide range of applications. For some applications, like virtual try-on, the simulation needs to be fast, adaptable, and accurate.

The traditional approach to tackle this problem is physics-based simulation (PBS). The idea behind this approach is to discretize the objects in the scene and time and use physical laws to predict, at each timestep, how the system will evolve. Physical simulation of cloth can range from a simple mass-spring system to a yarn-level simulation. These solutions have been shown to achieve incredibly realistic results,

but they also have some drawbacks. On one hand, the physical parameters that determine the material behavior are difficult to tune and need expert supervision. On the other hand, the quality and realism of a simulation increase with its degrees of freedom, and so does its computational cost. Plus, the simulation must be run again to adapt to any change in the scene.

Recently, data-driven models have emerged as a faster alternative. Instead of explicitly programming the task, these methods receive large amounts of data and learn patterns and relationships from them. They usually require the most resources for data collection and preparation, as well as in training and tuning the model. However, once trained, they can potentially perform very complex tasks more efficiently than physics-based methods. Such benefits have led the scientific community to the exploration of using deep learning methods to approach all kinds of tasks, allowed by the increasing amount of data and computational power. Cloth modeling is not an exception. Deep learning models can register, reconstruct from images, and simulate clothes, but they present some notorious downfalls. While they work perfectly in structured data like Euclidean domains (2D/3D grids, fixed size embedding vectors, sequences,...), they struggle with generalizing to new unseen structures. Unfortunately, 3D garments are represented as 3D meshes with irregular sampling and different connectivity, which hinders the adoption of data-driven models for 3D clothing. Some methods try to circumvent this issue with alternative representations, like implicit fields [Tiw*21; San*22], point clouds [Ma*21a; Zak*21], or displacement maps [LCT18], but these methods often struggle with obtaining consistent meshes and fine wrinkles. Alternatively, other approaches that work directly on meshes are generally limited to a single garment or discretization.

The main goal of this thesis is to address these limitations, by exploring ways to extend data-driven methods, so they can cope with a diverse range of garment designs and triangulations. We will investigate how to extend the convolutional networks to work with unstructured data and to solve their limitations in the regression of deformations.

1.1 Objectives and Contributions

This thesis is aimed at developing a data-driven framework for accurate 3D garment draping. We see a significant potential in Machine Learning as a fast and expressive tool for cloth modeling. Compared to PBS, Deep Learning methods are in general faster at runtime and have the capability of scaling better. We strongly believe that the generalization capabilities of convolutional filters and generative methods can be leveraged to improve data-driven cloth deformations.

At the beginning of this research, our first observation was that most methods that applied Deep Learning on 3D meshes werebased on fully-connected architectures. Fully-connected networks are a type of artificial neural networks that consist of the concatenation of densely connected layers (meaning that each neuron in one layer is connected to every neuron in the next layer). These networks can approximate any function, but they have a high number of weights and they tend to overfit. They also require fixed size input, which, in the context of 3D cloth, means that these models cannot generalize to garments with different triangulations (*e.g.* different surface discretizations). Alternatively, convolutional models are designed to handle spatial data. In the image domain, their local connectivity and shared weights make them more efficient and effective at capturing features such as edges or patterns, leading to better results. Additionally, they require fewer parameters and don't need a fixed size input. For these reasons, we decided to address these limitations by exploring the extension of convolutional filters to the mesh domain.

First, the thesis introduces a novel approach that uses fully-convolutional graph neural networks to model 3d clothing. This method introduces a parametric garment space, generating a range of garments, and manages to deform them to fit different body shapes, without relying on a given topology (in the context of this thesis, by *topology* we mean a given surface discretization). To do so, we decouple three sources of deformation: garment design, body shape, and garment material. At each step of the pipeline, a different network is trained to predict the overall garment shape, a smooth fit, and the material-specific wrinkles respectively. Altogether, we get a full framework that returns a fitted garment, given the design and shape parameters, some examples are shown in Figure 1.1. Note that the resulting mesh has a different optimized topology for each garment and that the method works for new and unseen designs.

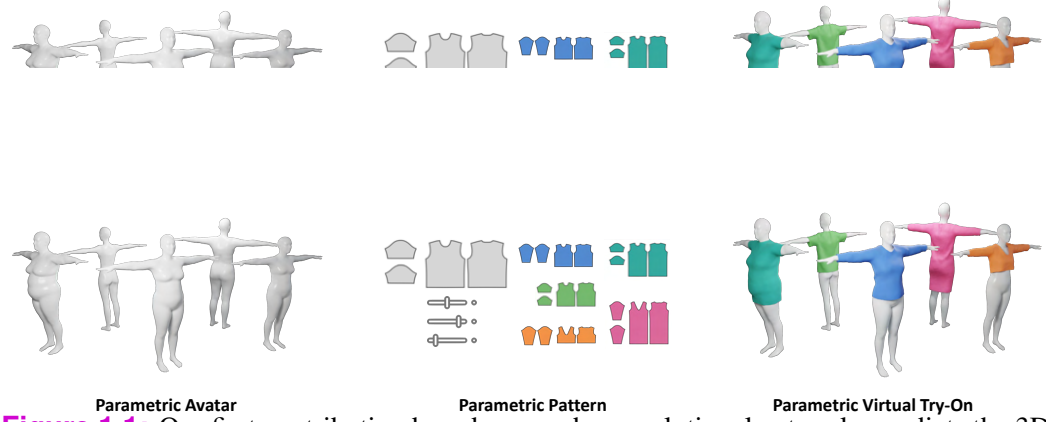

Figure 1.1: Our first contribution based on graph convolutional networks predicts the 3D draping for an arbitrary body shape and garment parameters at interactive rates. From left to right, a variety of body shapes obtained from a parametric avatar model, different 2D panel configurations of our parameterized garment types, and corresponding dressed 3D bodies generated with our fully convolutional approach.

This contribution led to the following publication:

- Raquel Vidaurre, Igor Santesteban, Elena Garcés, Dan Casas. "Fully Convolutional Graph Neural Networks for Parametric Virtual Try-On". *Computer Graphics Forum* (2020) [Vid*20]

Despite its novelty and power, the method has some limitations. Mainly, it requires long training times and high-frequency details are lost. This was especially noticeable when we started working with pose-dependent deformations. Meanwhile, in the image domain, generative models were becoming really popular. Denoising Diffusion Probabilistic Models (DDPMs) are a class of generative models that can produce high-quality images. They involve a forward diffusion process (where noise is added to a clean image) and a reverse denoising process (where the model is trained to remove the noise). Once trained, the network can synthesize high-quality images by iteratively denoising Gaussian noise. These models outperform previous generative models (such as the popular Generative Adversarial Networks, also known as GANs), and, unlike GANs, they don't suffer from training instability issues. Given their amazing results, we decided to encode wrinkles in the image space so that we could use this powerful tool for wrinkle synthesis.

In the second contribution of the thesis, we propose a bijective mapping between 3D deformations and displacement maps, that encode wrinkles as 3D offsets in RGB values. This representation allows us to employ a generative model based on DDPMs to generate images of such displacement maps conditioned by the design and pose parameters. These maps can be used to generate animated 3D garments that present fine-scale wrinkles (see Figure 1.2) indeed. Notably, the

Figure 1.2: Our second contribution based on garment deformations encoded as displacement maps enables the use of 2D diffusion models to generate 3D deformations. Here we can see three garment designs animated with our method

method remains agnostic to discretization and we offer a solution to condition the model on the previous garment state, enabling the generation of temporally-coherent sequences.

This contribution led to the writing of the following publication that is currently under review:

- Raquel Vidaurre, Elena Garcés, Dan Casas. "DiffusedWrinkles: A Diffusion-Based Model for Data-Driven Garment Animation".

1.2 Outline

This thesis is organized as follows:

- **Related Work.** Chapter 2 gives an overview of the research that has been done around virtual cloth representation and has inspired this thesis. The methods are grouped in physics-based simulation, 3D reconstruction and data-driven methods. We highlight the contributions, but also the limitations that motivate this research.

- **Background.** Chapter 3 covers some basic information that is important for the understanding of the thesis. We also define some notation and concepts that will be used throughout the technical chapters.

- **Fully Convolutional Graph Neural Networks for Parametric Virtual Try-On.** Chapter 4 describes our first technical contribution, which consists of a fully-convolutional framework to model cloth deformations in a parametric design space. We introduce our full approach, define our dataset, and evaluate its performance.

- **DiffusedWrinkles: A Diffusion-Based Model for Data-Driven Garment Animation.** Chapter 5 presents our second contribution. We define our 2D representation of deformations and the diffusion-based model. Finally, we evaluate the resulting deformations.

- **Conclusions.** Chapter 6 offers a discussion about the methods developed in the thesis. We summarize the key findings of the thesis and reflect about how they contribute to the field, as well as their limitations, potential applications, and future work.

2

Related work

Cloth digitization remains a key challenge for the Computer Graphics community, due to its wide range of applications, where the ability to accurately reproduce the movement of fabrics can really enhance realism in computer-generated environments. This chapter aims to discuss the diverse existing ways to simulate cloth, as well as the technical advancements that push the evolution of cloth simulation.

These advances are strongly pushed by the requirements of industry. The film industry requires high-quality, realistic, high-resolution, and visually pleasing simulations, while video games and VR applications seek interactivity, scalability, and efficiency. Alternatively, design-oriented applications require accurate simulations, that reproduce real-world behavior as closely as possible. Differentiable applications may also be useful for inverse design applications.

We focus especially on virtual try-on applications, that should provide a reliable preview of clothes. Cloth simulation needs to be **accurate** to ensure that virtual garments behave as they would in the real world. Besides, the simulation needs to be **efficient**, as interactivity can really enhance the virtual experience, especially in virtual reality settings. It also should be **scalable** as multiple customers may want to fit clothes simultaneously.

Existing methods can be roughly split into the following categories, which we briefly define below and extensively discuss in the rest of this chapter:

Physically-based simulations (PBS) have been the classical approach to tackle the virtual modeling of cloth movement. This family of methods uses mathematical models and computational techniques to simulate how cloth drapes, folds, and interacts with the body and other objects. PBS can produce highly realistic and dynamic visual effects, but their accuracy comes at the cost of increasing computational resources and time.

Cloth reconstruction involves capturing and reproducing the appearance and behavior of real garments. These methods usually use 3D scanning technology and imaging techniques to create digital models, and they are especially useful for applications where the goal is to replicate existing garments accurately but struggle with generalizing to new settings or body shapes.

Data-driven models leverage machine learning algorithms and large datasets to generate and predict cloth behavior. By training models, these methods can learn to produce plausible cloth dynamics in real-time. Data-driven approaches are particularly suitable for applications where efficient and scalable solutions are required, such as virtual try-on, as they offer a balance between realism and computational cost.

In the following chapter, we will overview the current solutions for cloth modeling. We will discuss the strengths of physically-based simulations, cloth reconstruction methods, and data-driven models, as well as their limitations.

2.1 Traditional Cloth Simulation

The traditional approach to model the behavior of cloth in computer graphics is physically-based simulation. It involves using mathematical models and physical laws, such as Newton's laws of motion and Hooke's law to simulate how cloth moves, folds, stretches and interacts with other objects [Nea*06]. The seminal paper on deforming objects using physics by Terzopoulos *et al.* [Ter*87] established the mathematical framework for physically-based modeling of elastical surfaces. They layed the foundation for believable cloth simulation, that responds to forces like gravity, wind and collisions in a natural way.

Mass-spring networks are one of the most popular ways to model cloth. The main idea is to discretize cloth into a set of particles connected with different types of springs to model the structure and elasticity of the cloth, as well as resistance to bending and shearing [Pro*95]. The simulation process consists of computing the forces on the particles, updating their positions and velocities based on the calculated forces, and detecting collisions to adjust the resulting forces. Mass-spring

Figure 2.1: Cirio *et al.* propose an efficient yarn-level simulation. **Source:** [CLO15]

systems are intuitive, easy to model, and efficient, but they lack accuracy and the behavior depends heavily on the mesh resolution and topology. Multiple advances have been made to enhance mass-spring models, like introducing deformation constraints to avoid unrealistic deformation for rigid cloth simulation [Pro*95], robust contact handling [BFA02], techniques to improve the wrinkling behavior of cloth [BMF05] and numerical stability and robustness [CK05]. Despite the advances made, the efficient computation and the great realistic looking results, mass-spring networks still fail to accurately reproduce real world cloth deformations, making them unsuitable for virtual try-on applications.

In contrast, some approaches attempt to accurately reproduce cloth behavior by representing cloth as a continuous surface, which is then discretized to solve numerically the differential equations that represent the mechanics of the given surface. The continuum fundation of Finite Elements Methods (FEM) enables the simulation of anisotropy and irregularities, while being resolution-independent [Mül*02; Gri*03; EKS03]. Others get as far as to simulating the cloth at yarn-level as rods interacting with each other [KJM08; KJM10]. Despite efforts made to accelerate this simulation by assuming that the rods are in persistent contact [Cir*14; CLO15] these simulations remain computationally expensive.

Performance is one of the main limitations of accurate cloth simulation, leading to numerous efforts to improve its efficiency. Baraf and Witkin [BW98] proposed implicit integration methods to allow for larger time steps. A very popular approach to create fast, stable and controllable simulations is Position Based Dynamics (PBD) [Mül*07; KCM12; Mül*14]. PBD methods offer robustness and speed, which makes them suitable for interactive environments like video games or virtual reality. However, they lack the accuracy needed for virtual try-on and they require expert tuning of parameters [Ben*14]. Similarly, Projective Dynamics (PD) offers a

Figure 2.2: The method proposed by Narain *et al.* dynamically refines and coarsens the simulated mesh to conform to the details of the cloth **Source:** [NSO12]

robust framework where deformations are treated as a sequence of local projections [Bou*14; Ly*20]. Additionally, some works have developed model reduction techniques [De *10; SB12] such as subspace simulation with the help of machine learning [Hol*19; Ful*19]. Another approach is to add details to enrich coarse simulations. These wrinkles can be learned by example from other high-resolution simulations [Wan*10; Kav*11; ZBO12] or computed at runtime on top of the coarse simulation [MC10; Roh*10; Gil*15]. Moreover, adaptative models attempt to provide the best compromise between speed and accuracy by self-adapting at space and time based on the state of the simulation [Man*17]. Concretely, some works provide solutions to dynamically refine and coarse areas of the mesh depending on how smooth or complex the deformations are [Lee*10; NSO12]. Others combine triangle-based with yarn-level simulation to enhance detail in some areas. GPU-based solvers have also been proposed to optimize simulations [Tan*16; FTP16; Tan*18b; Wan21; WWW22].

Beyond the primary goal of improving the accuracy of cloth simulations, there are several other significant applications in this field. One important area is the design and creation of digital garments. Some works explore the transformation from 2D patterns to 3D garments [Ber*13] and include pattern adjustment features [Ume*11; Bar*16], even automatic adjustment to fit a certain body [Wol*21]. Other approaches manage to generate a garment directly from sketches [Li*17a; Wan*18].

Another crucial application is the estimation of cloth parameters. While most simulations require expert parameter tuning, they don't necessarily reflect the physical characteristics of real-world fabrics. Accurately estimating these parameters is crucial for virtual try-on and design applications. To solve this problem, some works propose methods to estimate the parameters from video [Bha*03; Bou*13] or

from multi-view video [Sto*10]. Recently, multiple approaches have tackled the challenge of capturing parameters from video using data-driven models [Bou*13; Wu*16; YLL17; Ras*20; Run*20]. Alternatively, some methods use differentiable simulation to obtain the parameters that fit a target 3D shape [LLK19; Hu*20; Um*20]. Recent approaches go as far as optimizing body, design, and material parameters to fit the multi-view capture of a garment[Li*23].

Despite the significant advances in cloth simulation, traditional approaches face several limitations, especially in the field of virtual try-on applications. A notorious challenge is the accurate replication of diverse fabric behaviors, which is hindered by a need for extensive parameter tuning. Besides, physically-based simulation methods often struggle to meet real-time requirements, as high-fidelity simulations come at the cost of prohibitive computational costs. Furthermore, traditional simulations lack the ability to adapt dynamically, so usually changes in fabric or design require reconfiguration. It becomes especially challenging in the case of complex designs where we want to simulate the effect of additional design features like seams, zippers, pockets, elastic bands, etc. These limitations highlight the need for further research and for the integration of innovative techniques.Instead of solving a complex system of non-linear equations, data-driven methods are models with a big amount of parameters, that are optimized at training to solve a problem but only need to be evaluated once at runtime, which makes them very suitable for virtual try-on applications. Data-driven methods need data, and garment capture can potentially help us to get real-world fabric behavior, which (as we saw) is quite challenging for cloth simulation.

2.2 Cloth reconstruction

We refer to cloth capture as the process of capturing detailed data about real-world cloth, like texture, shape, and movement of fabrics. This information is then used to create a 3D model that reproduces the cloth's behavior. The precise reconstruction of the surface and properties of the garment can potentially be used later to dress new subjects or train data-driven models. Consequently, cloth capture and reconstruction techniques can be crucial for virtual try-on applications.

Early attempts managed to capture the surfaces of moving garments using color-coded patterns that can be aligned with a multi-view setting [Sch*05; WCF07]. For the first time, Bradley *et al.* [Bra*08] succeeded at reconstructing sequences of markerless garments, with relatively low wrinkle-level detail, with a multiview stereo algorithm. Notably, their method even achieves temporally coherent geometry with isometric cross-parameterization. To circumvent the challenge of capturing high-frequency details, Popa *et al.* [Pop*09] propose a method to add wrinkles to low-frequency captures, similar to the ones menti oned for simulation. The work presented by Zhou *et al.* [Zho*13] further reduces the input requirement and manages to reconstruct garments from a single image. By estimating the outlines of the garment, they create a smooth 3D model, that is further refined with shape-from-shading techniques.

While these methods effectively capture garment motion and surface details, they face a significant limitation in translating the captured data to new, unseen scenarios. Adjusting the fit of the garment to a wide variety of body shapes, while maintaining the fabric's behavior, remains an unsolved challenge.

Alternatively, some methods aim at recovering the complete body and sequence of motions of a dressed actor. Early methods for performance capture require an initial template and a multi-camera setting [De *08; Vla*08]. The mesh is optimized to fit the input images. Later, approaches appeared that capture performance from a single depth camera [Zha*14; Bog*15], a monocular video [Yan*18b], and even in outdoor settings [Rob*16; Xu*18]. Although re-animation of captured performance has proven to be feasible [Cas*14; Pra*16], all of these methods represent both body and garment with a single mesh. This approach clearly limits the application and extension of these garments to fit different body shapes.

More interestingly, some approaches attempt to reconstruct the body and cloth separately as distinct layers. Neophytou *et al.* [Nea*06] propose a three-layered model, by fitting the pose and shape parameters of a parametric human model to a sequence of meshes of a dressed human. Then, they estimate cloth as a residual of the body. This approach allows for parameter manipulation, so a new subject can be dressed with the obtained cloth and animated. Similarly, Pons-Moll *et al.* [Pon*17] present a remarkable work to reconstruct the underlying body shape and multiple garment meshes with fine detail from 4D scans, which can be transferred to new body shapes. Despite their notorious advances in terms of capturing deformations,

Figure 2.3: PERGAMO offers a data-driven framework to reconstruct garments from monocular RGB videos and to train a network using the obtained data to animate an avatar. **Source:** [CCC22]

the retargeted sequences may look unrealistic as they are just a copy of the captures, and they don't account for garment size and fit to completely different body shapes. Yang *et al.* [Yan*18a] take it a little bit further and analyze the motions of the obtained cloth layers to regress semantic parameters, such as material properties and garment size, enabling the representation of a richer garment model.

Some models apply deep learning techniques to enhance garment capture, like the work by Daněřek *et al.* [Dan*17], which consists of using Convolutional Neural Networks (CNN) to recover the 3D displacements of a garment from one or more images. However, this method is limited to a fixed garment template, which restricts its applications. A different approach uses 3D video scans to learn a statistical model of low-resolution deformations, combined with a Generative Adversarial Network (GAN) to create a high-resolution normal map encoding wrinkles [LCT18]. This method allows retargeting the scanned garment to new body shapes. Casado-Elvira *et al.* [CCC22] propose a method to, first, reconstruct the deformations of a garment from monocular RGB videos, and then, train a network with the reconstructed garments to estimate garment deformations from pose sequences. Other works [All*19; Bha*19] reconstruct hair and clothing on top of the parametric human SMPL model from several images of a dressed person.

In conclusion, while existing methods for cloth capture and reconstruction have made significant advances in capturing and retargeting real-world garments, they often fall short in generalizing the captured properties to different garments and accurately adapting the fit of the captured garment to different body morphologies. This limitation is due to their reliance on fixed templates and specific input data. On the other hand, deep learning models, which can learn complex patterns, offer a potential solution for generalizing garment fit and behavior across various scenarios. We believe that future research should focus on developing sophisticated

and adaptable learning algorithms, leveraging captured cloth data to build models that can handle multiple garments and realistically fit any body shape. In our work, we use synthetic data, however, data-driven models can potentially be trained using captured garments and learn to reproduce real-world cloth behavior.

2.3 Data-driven cloth modeling

Data-driven models are computational models that rely on data to learn patterns and make predictions. Unlike traditional models, which are often based on explicit mathematical equations derived from theoretical principles, data-driven models use statistical and machine learning techniques to infer relationships directly from data. In cloth modeling, data-driven methods typically aim to estimate the function that computes a desired output (*e.g.* a deformed 3D cloth) given a determined input (*e.g.* pose parameters).

Deep Learning (DL) is a specific type of data-driven model, characterized by the use of artificial neural networks. These networks are proficient in processing data and consist of multiple layers of interconnected nodes. The key advantage of DL models is their ability to automatically extract complex patterns from large amounts of data, and they are truly revolutionizing not only the scientific community but the whole world.

DL applications have been successful in numerous fields, such as Computer Vision. They have been applied with groundbreaking results to many tasks (even surpassing human performance in some of them!), including image classification [He*16], object detection [Red*16], face recognition [SKP15], or image generation [Rad*21; AI23]. They also excel at Natural Language Processing, where they have managed to improve the understanding and generation of human language, enabling the development of applications for translation and chat bots [Dee17; Ope22a]. Despite their well-known impact in these fields, their range of applications is countless, from autonomous driving to healthcare or finances. Seeing the performance of these technologies in very complex tasks, that would be unthinkable without Deep Learning, we think that it offers promising solutions to the challenges of cloth modeling.

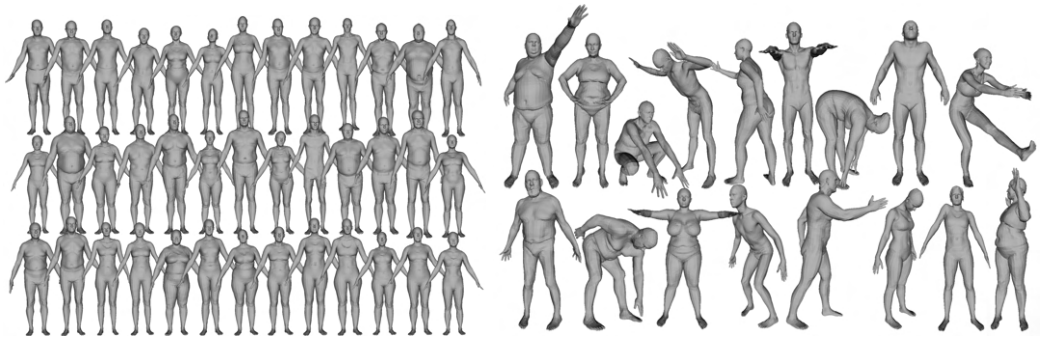

Figure 2.4: SMPL model is trained with a wide variety of body shapes and poses. **Source:** [Lop*15]

Data-driven models have been used to solve some problems related to deformation modeling. Quite popular are the statistical 3D human body models [Ang*05; FCS15; Lop*15; Che*18], which not only help in the creation of realistic virtual representations of human bodies but are also crucial for the reconstruction of complete body shapes from incomplete captured data. Particularly interesting to us is SMPL [Lop*15], which we use for our work. Their model provides a standardized parametrization of human body shapes and poses, essential for accurate 3D garment animation. On top of these models, even the addition of skin deformations has been reproduced with DL models [San*20b]. Some approaches use autoencoders on 3D meshes to model latent spaces to encode and generate deformations on meshes with fixed topology [Tan*18a; Ran*18].

A particularly interesting line of work is the use of data-driven models to accelerate or estimate physics-based simulations. Early works showed the potential of machine learning models to simulate fluids by using a regression forest to estimate the movement of particles [Lad*15] or by predicting the evolution of the system in a latent space [WBT19]. Sanchez-Gonzalez *et al.* [San*20a] propose a machine learning framework for particle simulation based on graph networks.

In the field of cloth modeling, data-driven strategies have been proposed to solve multiple tasks, such as clothed-human reconstruction [All*19; Sai*19; Sai*20], garment design [Wan*18; SLL20], animation [Wan*19; Hua*20; Ber*21] or virtual try-on [Gua*12; SOC19; Zha*21b].

Early data-driven models learned to deform garments as linear combinations of examples in a training set [Gua*12; Xu*14]. Later, deep-learning-based methods

explored the solution to estimate deformations as a function of pose [Gun*19], of shape and pose [SOC19; Gun*20], of shape, pose and style [PLP20], of pose and garment authoring [Wan*19], and of size [Tiw*20]. The design of garments has also been tackled with data-driven models. Particularly relevant for us is the work of Wang *et al.* [Wan*18], who learn a multi-modal subspace that enables the edition of a garment design using both the 2D pattern panels and a sketch of the desired drape. The method outputs the 3D draped garment according to different input modalities given a target body shape. Recent approaches use DL methods to reconstruct the sewing patterns of a garment given a point cloud [KL22] or a single RGB image [Liu*23].

Ma *et al.* [Ma*20] propose a probabilistic model for clothing that builds on top of SMPL. The different garments are represented as additive displacements that are applied to the full-body mesh of SMPL, and they employ a conditional VAE-GAN [Lar*16] to generate new garments. Despite their generative approach, which can reproduce global and local cloth deformations, they require a fully connected layer and the mesh has a fixed size and topology. Similarly, Bertiche *et al.* [BME20] learn a latent space for multiple garments and deformations. Their model is able to generate new garments for any pose and shape. However, it cannot cope with varying topology.

While most of these methods predict cloth deformation as 3D displacements at each vertex from a triangular mesh [Gun*19; SOC19; PLP20], this is not the only kind of virtual representation of garments. Some works represent cloth with point clouds [Ma*21a; Zak*21; Ma*21b], implicit representations [Tiw*21; Cor*21; San*22; De *23] or sketches [Wan*18]. Some methods approach the challenge of clothing humans from an image-based perspective. Their goal is to generate images of dressed humans without dealing with an underlying 3D representation of the garment [HSR13; Han*18; Yan*20; Zhu*23]. Despite the outstanding results of these methods, they don't account for the size of the garment, so the fit is not accurate, and they are usually trained with images of professional models, so their generalization to diverse body shapes is still a challenge.

We find another representation of deformations especially interesting. UV maps can be used to create a mapping between a 3D surface and an image. This representation allows us to store geometry information in a format that is suitable to be processed with standard convolutional networks. Given a garment, as long as UV maps are

aligned, the deformation reconstruction is agnostic to the mesh triangulation. Shen *et al.* [SLL20] introduce an image-based latent representation for sewing patterns. This representation enables them to generate deformations using a generative adversarial network (GAN) for the reconstruction of arbitrary garments. In the same line, Su *et al.* [Su*23] present a unified pipeline to deform garments with varying designs that can be parameterized by body, and shape. They represent these garments as distance maps of the SMPL vertices, employing the UV coordinates of SMPL. A common strategy for modeling wrinkles consists of adding detail to a coarse geometry. Lähner *et al.* [LCT18] combine a low-resolution statistical model that is then enhanced with high-resolution normal maps generated by a conditional GAN. Later works [Zha*21a] extend this approach to handle different materials.

Datasets are very important for data-driven models, as they can only be as good as the data you feed them. Some methods [LCT18; Tiw*20; Ma*20] use high-quality scans of dressed people. Capturing and reproducing the deformation of real clothes is a desirable capability of virtual try-on, but the acquisition process is challenging and expensive. Alternatively, some works [SOC19; Gun*19; PLP20; Gun*20; San*21] use synthetic data generated with physics-based simulators. This methodology enables the generation of custom data, correctly labeled in a controlled setting. Moreover, there is no need to have an expensive setup to get new data. For these reasons, we decided to use synthetic datasets to develop our frameworks.

Self-supervised methods use implicit metrics of the training data as supervision, instead of relying on labeled input data. Thus, they present an interesting alternative to avoid the challenges of creating a dataset. Although most learning-based method for cloth modeling are supervised, in the later years some works have explored the creation of self-supervised methods. The first approach is the one presented by Bertiche *et al.* [BME21], where the loss is computed as a sum of potential energies, instead of an error between predictions and ground truth. They formulate their network as the concatenation of a Multilayer Perceptron (MLP) to encode pose and non-linearities in an embedding space and a deformations matrix that returns the deformations of the unposed garment when multiplied with the pose embedding. Their method doesn't work with varying shape and a specific network needs to be trained for each garment and body, and their method is essentially static, so dynamic behaviours are impossible to model. Similarly, Santesteban *et al.* [SOC22] propose a method that builds upon the same idea. However, their regressor takes dynamics into account, as well as the shape body parameter of SMPL. Besides, a

more complex model of material results in more realistic deformations, but they need to train a regressor for each topology/garment. Recently, Grigorev *et al.* [Gri*23] take this idea even further and use Graph Neural Networks to create a garment-agnostic regressor, that estimates deformations for any garment depending on shape, pose, size, and material properties. Despite the great potential of self-supervised methods, the research conducted in this thesis is within the framework of supervised learning.

Dense networks (also known as fully-connected networks) have been widely used in machine learning applications due to their simplicity and flexibility. In the context of data-driven cloth modeling, these networks have been shown to learn to predict rich cloth deformations from synthetic datasets [Dan*17; SOC19; PLP20; San*21], real cloth captures [LCT18; CCC22], and self-supervised strategies [BME21; SOC22]. However, they present some intrinsic limitations. One major drawback is their inability to generalize to different discretizations of the cloth mesh.

In recent years, graph convolutions have emerged as a powerful tool in the field of machine learning. The main idea behind graph convolutions is to leverage the connectivity patterns of graphs to perform convolution operations. Graphs are represented as sets of nodes (entities) and edges (relationships) and graph convolution operators aggregate information from each node's neighbors to update its feature representation. Graph convolutions have become quite popular because data of multiple irregular domains can be represented with graph structures. Some well-known examples are social networks [HYL17], molecules [Gil*17], recommendation systems [Yin*18a], traffic data [Li*17c], and, of course, 3D meshes [Mon*17; VBV18]. Recently, some works have proposed graph-based networks to predict cloth dynamics [Pfa*20; Gri*23].

In the field of generative models, Denoising Diffusion Probabilistic Models (DDPMs) have been especially impactful, because of their ability to produce high-quality images and their stable training. Ho *et al.* [HJA20] set their formulation and, since then multiple works have improved their architecture, performance, and sampling efficiency [SME20; ND21; Ho*22a; Bla*23]. They are at the core of popular text-to-image synthesis models like DALL-E [Ope22b] or Stable Diffusion [AI23]. We will discuss DDPMs in detail in Chapter 5, as they are an important part of our model.

Our goal was to extend existing methods to work with arbitrary triangulations. The first contribution (Chapter 4) uses graph convolutions, that effectively handle irregular mesh structures by operating directly on the graph representation of the cloth. This ensures the underlying structure of the cloth is preserved regardless of its discretization. In the second contribution (Chapter 5) we explore the use of DDPMs to generate rich and detailed displacement maps in the UV space.

3 Background

This chapter covers some basic knowledge that can be useful for understanding the technical contributions presented in Chapters 4 and 5. First, I'm introducing parametric human models, which are crucial for our work, as they build on top of SMPL, a prevalent model.

3.1 Human model. SMPL

Human parametric models are statistical models designed to represent human body shapes and poses in a flexible and scalable manner. These models allow for the generation and manipulation of a wide range of bodies, by using a reduced set of parameters that define various aspects of the human body (usually, shape and pose parameters). This allows for easy manipulation of body shapes and animation of characters. Another key advantage is that they provide a reduced space of solutions for some tasks, like motion capture.

Early attempts interpolated between manually sculpted shapes [SRC01]. Allen *et al.* [ACP02] pioneered modeling bodies from scanned data, as a function of shape with an articulated template. However, it wasn't until SCAPE, the work by Anguelov *et al.* [Ang*05], that there was a full model of body deformations as a function of shape and pose. This work has several downsides (*e.g.* it has no skeletal structure), so many follow-up data-driven methods have been proposed. The most popular model is SMPL [Lop*15], which has become a standard for estimation, synthesis, and a great variety of applications. It is based on blend shapes and skinning and covers a wide range of body shapes. Later, it was extended to model hands [RTB17], faces [Li*17b], and hands and faces [Pav*19]. And then, they proposed a reduced and improved version of SMPL [OBB20].

In our work, we use SMPL [Lop*15] to represent human bodies. SMPL is a parametric model based on Principle Component Analysis (PCA) that represents the human body compactly and expressively with 2 sets of parameters. The shape of the human is parametrized by a low dimensional vector $\beta \in \mathbb{R}^{10}$, the principal components of PCA applied to the per-vertex deformation of a huge amount of 3D scans with multiple body shapes [Rob*02]. Note that they encode the deformations per-vertex, unlike its predecessor SCAPE [Ang*05], which is based on triangle deformations. The pose space is represented with the θ joint angles. Altogether, we assume that the body is defined as a mesh

$$\mathcal{M}(\beta, \theta) = W(T(\beta, \theta), \beta, \theta, \mathcal{W}), \tag{3.1}$$

where W is a skinning function that deforms the unposed mesh $T(\beta, \theta)$ depending on the pose parameters, θ, that correspond to the joint angles, the shape parameters, β, that determine the position of the joints, and the skinning matrix $\mathcal{W}$. Note that the unposed mesh $T(\beta, \theta)$ depends on both β and θ as offsets that depend on both sets of parameters are added to the original template (see Figure 3.1 for clarity.).

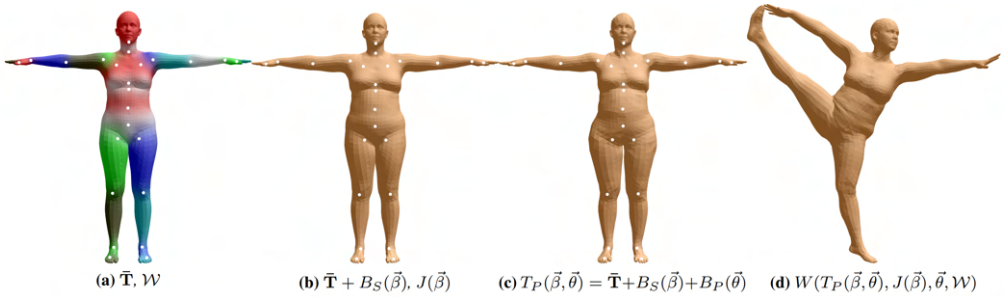

Figure 3.1: SMPL full model. a) Template with the color-coded skinning weights $\mathcal{W}$ and the original joints represented by white dots. b) Template with the shape-dependent deformations and joints corrected. c) Unposed mesh $T(\beta, \theta)$ with shape- and pose-dependent deformations. d) Deformed vertices $\mathcal{M}(\beta, \theta)$ reposed by the skinning function. **Source:** [Lop*15]

In the first contribution of this thesis (Chapter 4), we are interested in studying garment deformations that do not depend on pose (but on design, shape, and material). Thus there is no need to apply the skinning function, θ is set to $\mathbf{0}$, and the body mesh remains unposed. However, we will use the skinning weights as a semantic descriptor to enhance the capabilities of the network to reproduce local-specific deformations. For the second contribution (Chapter 5), we estimate deformations that depend on body pose and shape. Both β and θ parameters are used as input to condition the network. Similarly to SMPL, the deformations will

be added to an unposed template mesh, that can be transformed to pose space with a skinning function.

3.2 Baseline

The works presented in this thesis are heavily inspired by the work of Santesteban *et al.* [SOC19], who pioneered in investigating deep learning methods to model cloth deformation. Their model is built on top of SMPL and formulates the deformations of the garment in unposed space. Similarly to SMPL itself, they compute the deformed mesh as a sum of the garment template and the result of 2 regressors. The first regressor estimates per-vertex 3D offsets due to shape parameters, and the second computes the deformations caused by pose and shape parameters. The deformed cloth is then skinned and a post-process is applied to avoid penetrations with the body mesh.

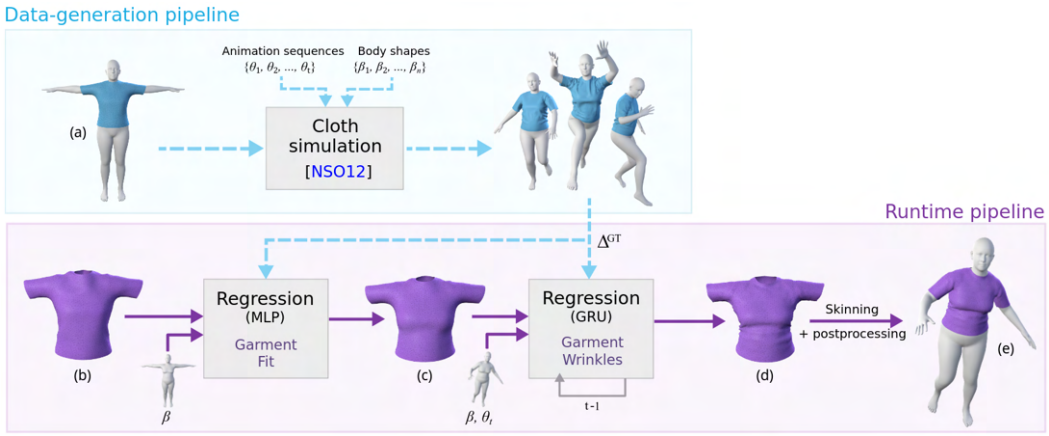

Figure 3.2: Santesteban *et al.* propose a data-driven method for cloth modeling that serves as baseline for our work. **Source:** [SOC19]

Similar to SMPL, they formulate the garment mesh as

$$\mathcal{M}_c(\beta, \theta) = W(T_c(\beta, \theta), J(\beta), \theta, \mathcal{W}_c), \tag{3.2}$$

where the unposed garment template is computed as

$$T_c(\beta, \theta) = \overline{\mathbf{T}} + R_G(\beta) + R_L(\beta, \theta) \tag{3.3}$$

by summing the offsets estimated by their regressors to the initial template.

The main limitation of their method is that it is garment-specific, meaning that any change in the design or discretization would require the training of a new network. In both of our works, we have a similar formulation, with the only difference being that we try to overcome their limitation by building regressors that work for several designs. Thus, in our methods, the garment is formulated as

$$\mathcal{M}_{\text{g}}(\beta, \theta, \mathbf{p}) = W(T_{\text{g}}(\beta, \theta, \mathbf{p}), J(\beta), \theta, \mathcal{W}), \tag{3.4}$$

where $\mathbf{p}$ is a vector containing the design parameters, and the unposed deformed garment mesh $T_{\text{g}}(\beta, \theta, \mathbf{p})$ is what we try to estimate in our proposed methods.

4

Fully Convolutional Graph Neural Networks for Parametric Virtual Try-On

The goal of this thesis is to explore data-driven methods for virtual try-on applications that generalize to different designs and topologies.

At the beginning of this thesis, most of the approaches that used learning-based models to predict cloth deformation were limited to a single garment. The reason for that is they were using networks with fully-connected layers. While these architectures are easy to work with, they carry some limitations. The first one is that they lose spatial information, as the triangular mesh representing the garment is flattened into an array, the network doesn't have any neighboring information. Secondly, they are densely connected, so they are big and have a large number of parameters, which increases the computational and storage cost, but also the chance of overfitting. Last but not least, they need to receive a vector of a fixed size as input.

Alternatively, convolutional layers learn filters that are applied across Euclidean domains. These filters can be used in tensors of arbitrary shape. By sliding the filters along the image, parameters are shared across the entire input, drastically reducing the total number of parameters and inherently accounting for neighboring information. This parameter sharing makes the filters translation-invariant and allows them to capture local dependencies while keeping the network smaller and increasing its generalization capabilities. Additionally, convolutional layers enable the creation of hierarchical structures, combining smaller features to detect more complex features, and fully-convolutional architectures (*i.e.* architectures

that don't have fully-connected layers) are not limited to a fixed input size. In the image domain, fully-convolutional networks are quite common (*e.g.* for image segmentation tasks) and they run with images of any size and shape.

However, while convolutions have proven to be very efficient and successful in grid-like domains, their extension to more complex structures (like 3D triangular meshes) is not trivial. In pursuit of this objective, we draw upon recent research that defines the necessary operators for graph-like structures [DBV16]. We propose a FCGNN (Fully Convolutional Graph Neural Network), which, when provided with a 3D parametric garment featuring arbitrary mesh topology and a desired body shape, yields precise 3D draped garment output.

Our geometric deep learning method operates by decoupling three distinct sources of deformations crucial for clothing fit: garment type, target body shape, and material. We start by building a parametric space for design. Using this design framework, we create a dataset of 3D garments and employ physics-based simulation [NSO12] to dress multiple body types. This dataset serves to train three different networks. First, we develop a regressor to predict the coarse 3D draping of a garment on an average body shape given its design parameters. Then, the surface of the garment is refined to produce a uniform triangular mesh, and the deformations due to the target body are estimated with a second regressor. Next, our final step adds the material-specific deformations, particularly fine-scale wrinkles. This regressor is further refined with a self-supervised strategy, that penalizes body-cloth collisions without the need for training data.

Altogether we built a full framework that copes with multiple parametric garments with arbitrary topology and fits them to a variety of body shapes.

4.1 Background

Before diving into the specifics of our method I would like to stop and explain some of the model's building blocks. We designed our FCGNN to have a U-Net architecture [RFB15], which was originally designed to segment medical images. This architecture consists of a contracting path - where convolutional and down-sampling

layers are concatenated - and an expanding path - that is almost symmetrical, and consists of a concatenation of convolutional and up-sampling layers. Intuitively, the contracting path provides context and the expanding path enables precise localization. To create this architecture there are two key ingredients: the convolutional operator and the pooling operator.

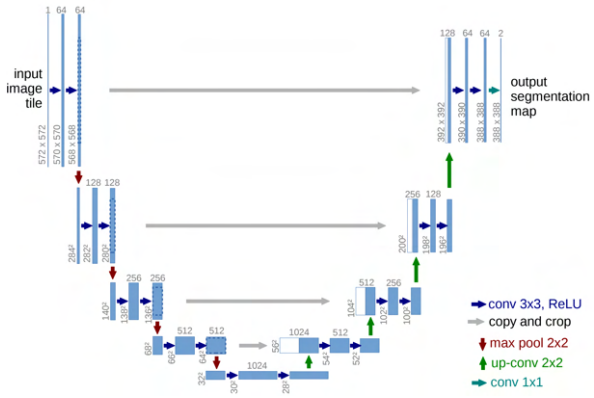

Figure 4.1: U-Net architecture was designed for biomedical image segmentation. Its main advantage is that it can capture information at different levels, while maintaining spatial information. **Source:** [RFB15]

4.1.1 Graph convolutions

Graph convolutions extend convolutions to graph-like structures, which makes sense because convolutions have some desirable properties. To do so, we treat the mesh as an undirected graph, where the vertices are the nodes and the edges are the links. Classical convolutions are clearly defined and easily parallelizable, but they only work for grid-like domains. The problem is that there isn't a standard way to extend their definition to irregular graph structures, see Figure 4.2 to get an intuition. We decided to use as our convolutional operator the truncated Chebyshev polynomial proposed by Deferrad *et al.* [DBV16], which is inspired by spectral graph filters. This operator shares some of the main qualities that we desire from classical convolutions, like looking at the local neighborhood, shared weight learning and generalization to new nodes, permutation invariancy, all of it while maintaining linear computational complexity.

Our definition: The normalized Laplacian is defined as $L = I_n - D^{-\frac{1}{2}}AD^{-\frac{1}{2}} \in \mathbb{R}^{n\times n}$ where A is the adjacency matrix of the graph ($A_{ij} = 1$ if there is an edge between vertices i and j, $A_{ij} = 0$ otherwise) and D is the diagonal degree matrix ($D_{ii} = \sum_j A_{ij}$ and $D_{ij} = 0$ if $i \neq j$).

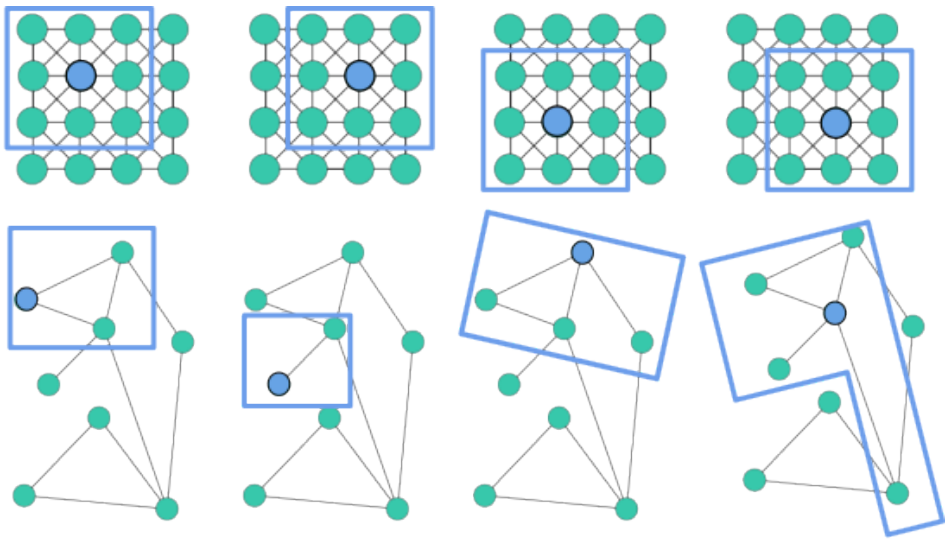

Figure 4.2: Intuition on convolutional operators. **Top**: Convolutions on Euclidean domains are well defined. The filter slides over the image, and the filter's weights are used to aggregate the neighbors' information for each pixel. **Bottom**: Convolutions in the graph domain are not so clear. Note how the neighborhoods have different sizes for each node.

The Chebyshev polynomial of order k, $T_k(x)$ can be recursively computed by

$$T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x) \tag{4.1}$$

with $T_0(x) = 1$ and $T_1(x) = x$. The polynomial parametrization for spectral localized filters that we use as our convolutional filters is then defined as

$$y = g_\theta(L)x = \sum_{k=0}^{K-1} \theta_k T_k(\tilde{L})x, \tag{4.2}$$

where $\tilde{L} = 2L/\lambda_{max} - I_n$ is the scaled Laplacian(λ_{max} is the principal eigenvalue of L) and $\theta_k \in \mathbb{R}^{F_{out} \times F_{in}}$ are the learnable parameters (F_{out} and F_{in} are the size of *out* and *in* feature vectors respectively).

Defining $\bar{x}_k = \theta_k T_k(\tilde{L})x$, we can make use of recurrence to compute $\bar{x}_k = 2\tilde{L}\bar{x}_{k-1} - \bar{x}_{k-2}$ with $\bar{x}_0 = x$ and $\bar{x}_1 = \tilde{L}x$. Then, $y = g_\theta(L)x = [\bar{x}_0, \bar{x}_1, ..., \bar{x}_{K-1}]\theta$, which can be efficiently computed.

Our early experiments showed that very big K didn't really improve the results while increasing the complexity and execution time of the operator, so we settled at

$K = 3$. Intuitively, this means that the output features corresponding to applying a convolutional filter on a vertex is influenced by its 2-ring neighborhood

4.1.2 Down- and Up-sampling

The pooling operation is again not trivial for such irregular domains. We decided to take an approach that makes sense from a geometrical point of view and was proposed by Ranjan *et al.* [Ran*18]. The key idea to their down-sampling operator is to pre-compute for each mesh the transform matrices $Q_d \in \{0,1\}^{(n \times m)}$ and $Q_u \in \mathbb{R}^{(m \times n)}$, where $m > n$ and m is the number of vertices in the original mesh and n is the number of vertices of the pooled mesh. To produce the down-sampling matrix, Q_d, we used the quadric error metrics algorithm [GH97]. Using quadric matrices, we estimate the surface error introduced by contracting each pair of vertices and iteratively decide which vertex to eliminate, until there are only n vertices left. The elements of the matrix are then set as $Q_d(i,j) = 0$ $\forall i = 0, 1, ..., n-1$ if the vertex v_j is discarded and $Q_d(i,j) = 1$ if v_j is the i-th kept vertex ($Q_d(k,j) = 0 \; \forall k \neq i$).

The up-sampling matrix Q_u is built at the same time as the down-sampling matrix Q_d. For each vertex v_j that is kept in down-sampling is reset with the up-sampling matrix ($Q_u(j,i) = 1 \iff Q_d(i,j) = 1$). Eliminated vertices are projected into the closest triangle of the down-sampled mesh and the barycentric coordinates of the projected vertex are used to build the up-sampling matrix. Mathematicaly, if a vertex v_j is discarded in the down-sampling preprocess, we search for its closest triangle in the coarse mesh, (v_a, v_b, v_c), and we obtain the projection of v_j onto the triangle, $\bar{v_j}$. The barycentric coordinates are found, such that $\bar{v_j} = \mathsf{w}_a v_a + \mathsf{w}_b v_b + \mathsf{w}_c v_c$ and $\mathsf{w}_a + \mathsf{w}_b + \mathsf{w}_c = 1$. Then, $Q_u(j,a) = \mathsf{w}_a$, $Q_u(j,b) = \mathsf{w}_b$, $Q_u(j,c) = \mathsf{w}_c$, and $Q_u(j,i) = 0 \; \forall i \notin \{a,b,c\}$

Additionally, the Laplacian of the down-sampled mesh is computed in the pre-process, as it is going to be required by the convolution operator in the deeper steps of the network. Note that all of these computations need to be done for each topology that is inputted to the Fully Convolutional Graph Neural Networks (FCGNN) used in our work. However, it only needs to be computed once, and the

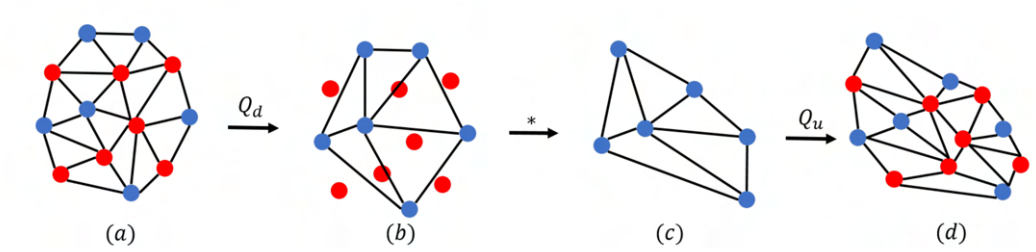

Figure 4.3: Intuition behind the pooling operator. (a) Given an initial template, (b) some nodes (red) are eliminated through matrix multiplication with Q_d, (c) the feature representations of the remaining nodes (blue) are modified by the network, (d) and the eliminated nodes are reconstructed as linear combinations of the remaining nodes, via matrix multiplication with Q_u. **Source:** [Ran*18]

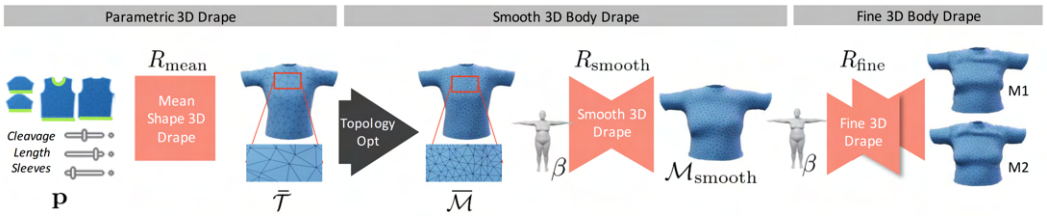

Figure 4.4: Overview of our full pipeline.

down and up-sampling operators are very efficient operations at runtime, as they are just multiplications by sparse matrices.

4.2 Method

Our objective is to accurately predict the 3D draping of garments, adaptable to any body shape, in the context of virtual try-on applications. We particularly focus on handling a diverse range of garment types, as most existing works overlook this feature due to the challenge of dealing with varying topology inputs.

To achieve this, we propose a three-stage approach outlined in Figure 4.4 that disentangles the different sources of deformations (due to garment type, body shape, and material) influencing clothing fit. Next, we will overview the different steps that build our pipeline.

Following the traditional garment design process, our initial step (Section 4.2.1) uses a set of 3 parameters $\mathbf{p}$ to define the 2D sewing patterns of a garment (sleeve length, chest circumference, and garment length), and learns a regressor $R_{\text{mean}}(\mathbf{p}) = \overline{\mathcal{T}}$ to estimate the corresponding 3D mesh $\overline{\mathcal{T}}$ of the garment draped into an average human shape. The first regressor returns a template with a fixed topology and a low dimensionality representing a wide range of designs (from tops to dresses, from short to long sleeves). To properly represent any garment, our second step (section 4.2.2) computes an optimized mesh topology $\overline{\mathcal{M}}$ for each garment, with uniform vertex distribution and triangle size. This new mesh with a new topology is then passed to our second regressor, R_{smooth}, to predict a smooth fit, given a target body shape β. The last step of the pipeline is a material-dependent regressor R_{fine} that produces a deformed mesh $\mathcal{M}_{\text{fine}}$ with a realistic draping of the garment on the target body shape with fine wrinkles. Both R_{smooth} and R_{fine} are implemented with (what we call) an FGCNN architecture, that copes with any garment design, shape, and (importantly) mesh topology. Furthermore, to avoid body-garment collisions in section 4.2.5 we introduce a novel self-supervised strategy to fine-tune the regressor R_{fine}. This approach incorporates a geometrically defined loss term that penalizes penetrations between the body and the garment, eliminating the need for costly ground truth data.

4.2.1 Parametric 3D Drape

To achieve accurate predictions of 3D draping for virtual try-on applications, it is essential to establish first the garment type. Inspired by the traditional clothing manufacturing processes, we propose a strategy based on 2D sewing patterns to characterize garment design properties. Our observation is that we can use a single 2D layout to effectively model a diverse range of garments by simply editing the length of some parameters. In fact, we can change from a short top to a dress, by changing the height of the front and back body panels. Similarly, adjusting the height of the sleeve panels can modify the length of the sleeves, and so does the height of the cleavage with a slight alteration of the front body panel.

Building upon this observation, we train a model to predict the coarse 3D drape of a garment based on a specific 2D sewing pattern. Here, we encode the parameters that define the 2D layout in a vector $\mathbf{p}$ that is fed into a non-linear regression

$R_{\text{mean}} : \mathbb{R}^{|P|} \rightarrow \mathbb{R}^{3\times|V(\overline{\mathcal{T}})|}$ that outputs the drape of the garment on a mean human shape $R_{\text{mean}}(\mathbf{p}) = \overline{\mathcal{T}}$ denoted by the overline symbol (we will consistently use the overline to signify mean-shape-related variables.

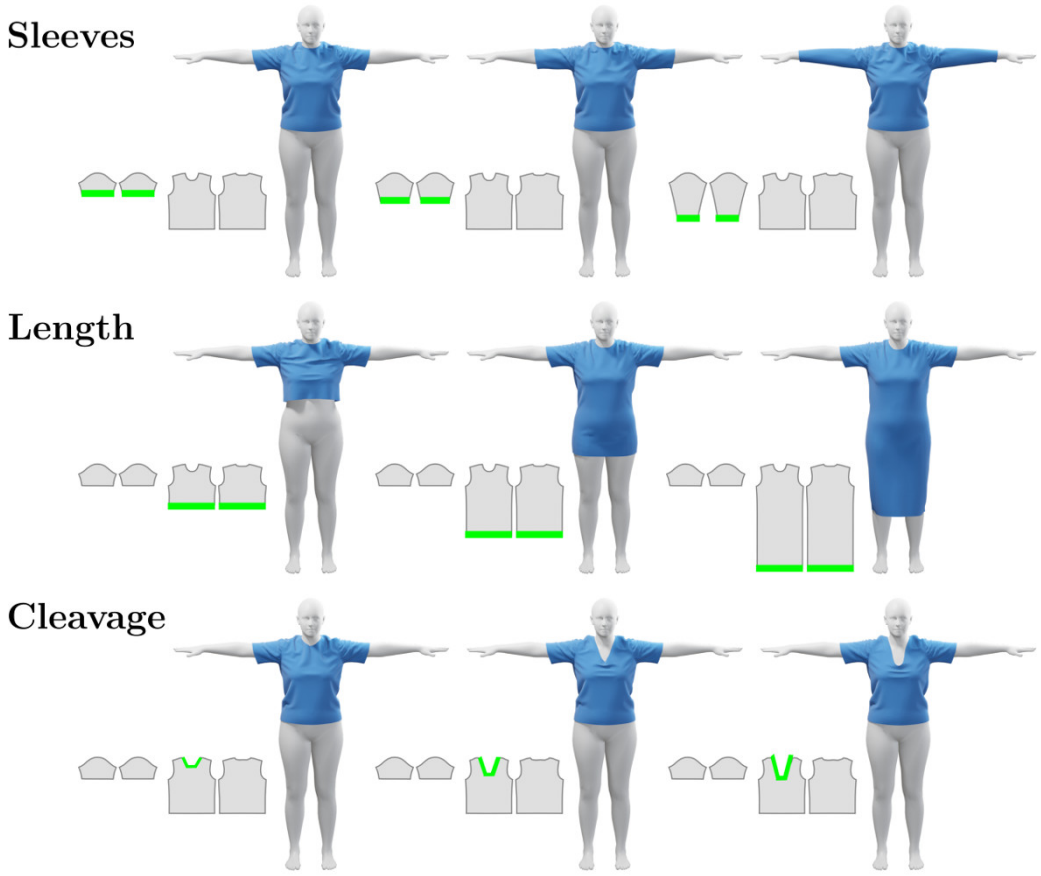

Figure 4.5: Sewing pattern parameters (rows) used to build our dataset of garments. Each column shows the effect of the minimum, mean, and maximum values for each parameter.

The motivation for this initial step is twofold: first, it roughly fits the garment on a generic human subject, which we use later in Section 4.2.2 to parameterize garment vertices using their closest body skinning weights; and second, it allows us to disentangle garment type-dependent deformations (*i.e.,* that depend on $\mathbf{p}$) from material-dependent and body shape-dependent deformations.

To train our regressor $R_{\text{mean}}(\mathbf{p})$ we build a dataset of 3D garments by manipulating a single 2D layout. Specifically, as shown in Figure 4.5, we manually edit parts of the 2D panels to design a family of garments including tops, t-shirts, sweaters, and short and long dresses. We then label each sample according to a set of measurements $\mathbf{p}$

in the corresponding 2D representation and simulate the sample worn by a mean human shape using a state-of-the-art physics-based cloth simulator [NSO12] (with remeshing option turned off) until it reaches equilibrium to obtain a 3D mesh $\overline{\mathcal{T}}$ of the draped garment. We implement the regressor $R_{\text{mean}}: \mathbb{R}^P \rightarrow \mathbb{R}^{3\times V^{\overline{\mathcal{T}}}}$ using a fully connected neural network that outputs the vertices positions of the mesh $\overline{\mathcal{T}}$ with a predefined topology.

4.2.2 Mesh Topology Optimization

To accurately represent the draping of 3D garments with fine-scale detail it is necessary to use a topology with sufficient resolution (*i.e.*, number of triangles) for each garment type. Since one of our goals is to build a model that can predict the deformations for a large family of garments, we need to adapt the topology of the mesh $\overline{\mathcal{T}}$ depending on the type of garment. To give a more practical example, we assume that the number of triangles required to represent high-quality draping of a t-shirt is smaller than those required for a long dress.

We model such garment type-dependent topology requirement by applying a remeshing operation to the coarse mean draped garment $\overline{\mathcal{T}}$. Specifically, we generate a new mesh

$$\overline{\mathcal{M}} = \phi(\overline{\mathcal{T}}, \mathbf{p}, T_{\text{dist}}, T_{\text{area}}), \tag{4.3}$$

where $\phi()$ is a remeshing operation that, given an input mesh $\overline{\mathcal{T}}$ and the 2D design parameters $\mathbf{p}$, aims at maintaining a (manually specified) average triangle distortion T_{dist} and surface area T_{area}. Notice that these parameters are constant for all garments, therefore we only need to set them once. We implement $\phi()$ based on the method proposed by Narain *et al.* [NSO12]. We write the optimized mesh as $\overline{\mathcal{M}} = \{\mathbf{V}^{\overline{\mathcal{M}}}, \mathbf{E}^{\overline{\mathcal{M}}}\}$, where $\mathbf{V}^{\overline{\mathcal{M}}} \in \mathbb{R}^{3\times V^{\overline{\mathcal{M}}}}$ are the vertices of the optimized surface, and $\mathbf{E}^{\overline{\mathcal{M}}}$ the edges of the mesh. Figure 4.6 shows an example of the template topology $\overline{\mathcal{T}}$ for a long dress design, which result in many degenerated triangles, and the optimized topology $\overline{\mathcal{M}}$. In practice, $\phi()$ works in the UV-space of the 2D panels, which are automatically sew together to obtain $\overline{\mathcal{M}}$. We have simplified the notation for the sake of clarity. Notice that the *surface* of $\overline{\mathcal{M}}$ and $\overline{\mathcal{T}}$ is analogous, but their topology is different.

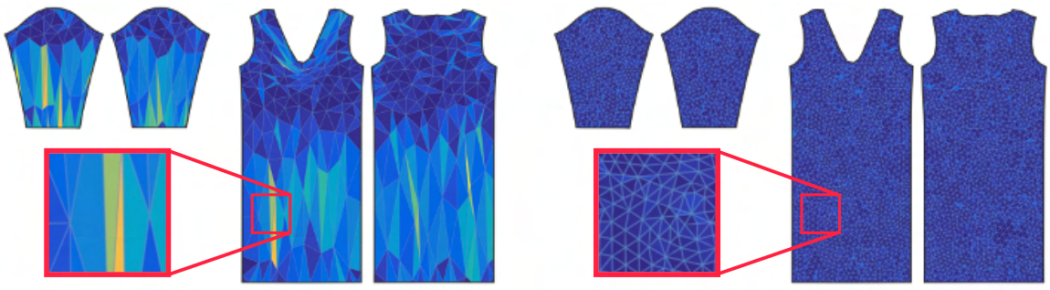

Figure 4.6: Garment type-dependent topology optimization, here shown in 2D panel space. Left: the 2D layout of a long dress design, with the template topology $\overline{\mathcal{T}}$. Right: the same design after the topology optimization step, resulting in the mesh $\overline{\mathcal{M}}$ with homogeneous triangle size and without degenerated geometry.

4.2.3 Smooth 3D Body Drape

Having the optimized mesh topology $\overline{\mathcal{M}}$ computed, in this second step we address the modeling of garment deformations caused by the target body shape. To represent parametric bodies, we use the popular model SMPL [Lop*15], which provides a PCA-based representation of human bodies in T-pose, parameterized by $\beta \in \mathbb{R}^{10}$. We use the first component throughout the paper since it encapsulates the largest variance in body shape. Importantly, SMPL also provides per-vertex rigging weights $\mathbf{w}_i$, which we use later in this section as a descriptor for garment vertices.

We therefore seek to learn a regressor R_{smooth} that deforms the mean shape garment $\overline{\mathcal{M}}$ and outputs a mesh that reproduces a smoothed drape of the garment onto the target body shape β. We design R_{smooth} such that it learns global and smooth deformations, which has two main advantages: first, it eases the learning task since it reduces the variance in data and second, it decouples target body-dependent deformations (*i.e.,* global stretching and draping effects) from material-dependent (*i.e.,* fine wrinkles) deformations, which we will learn on a subsequent step. However, formulating such regression task is not trivial: the topology of the input mesh $\overline{\mathcal{M}}$ is unknown at train time since we generate it at run time depending on the design parameters $\mathbf{p}$. Therefore, we cannot employ a fully connected network, where the input is a fix-size vector corresponding to the number of vertices of the mesh (a strategy commonly used in most of recent learning-based garment deformation methods [Wan*19; Wan*18; SOC19]) and, instead, we propose to use a graph-based fully convolutional architecture.

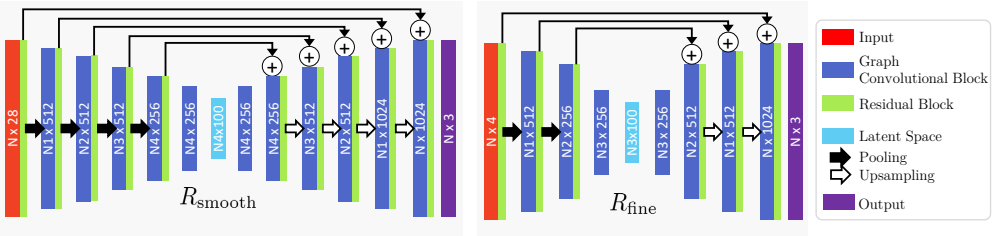

Figure 4.7: UNet-based architecture for regressors R_{smooth} (left) and R_{fine} (right). Each pooling or upsampling pass reduces or augments the number of nodes to half or double size. The input number of nodes is the same for both regressors, they differ in the number of intermediate layers, which is bigger for R_{smooth} as it has to learn a broader range of deformations.

Two key ingredients are required to design the regressor R_{smooth} as a graph-based fully convolutional neural network: first, a convolution operator that is able to deal with graph input and, second, an efficient graph pooling operator that is able to coarsen the mesh by clustering together similar vertices. Specifically for this work, for graph convolutions we use the operator based on truncated Chebyshev polynomial proposed by Defferrard *et al.* [DBV16], which has shown to be very efficient given its linear computational complexity and constant learning complexity, like classical convolutional neural networks (*e.g.,* for images or other Euclidean domains). For mesh coarsening we use the approach proposed by Ranjan *et al.* [Ran*18], which consists of precomputing down- and upsampling matrices using a traditional method for surface simplification by Garland and Heckbert [GH97]. Both operators were explained in detail in the background section (4.1).

Having the operators defined, we now explain how we design our fully convolutional regressor R_{smooth}. Starting from the mean shape 3D drape mesh $\overline{\mathcal{M}} = \{\mathbf{V}^{\overline{\mathcal{M}}}, \mathbf{E}^{\overline{\mathcal{M}}}\}$, we first build an analogous undirected graph $\overline{\mathcal{G}} = (\mathbf{N}, \mathbf{C})$, with as many nodes and edges, as vertices and edges in the mesh, $\mathbf{N} = \mathbf{V}^{\overline{\mathcal{M}}} \in \mathbb{R}^{3\times V^{\overline{\mathcal{M}}}}$ and $\mathbf{C} = \mathbf{E}^{\overline{\mathcal{M}}} \in \mathbb{R}^{3\times E^{\overline{\mathcal{M}}}}$, which we wish to use as input to the graph neural network. However, using vertices position as a descriptor for the graph nodes does not leverage all the information available in this context. Our key observation is that we can also append *semantic body part* information into the graph. To this end, for each garment vertex $\mathbf{v}_i^{\overline{\mathcal{M}}}$ we find the closest body vertex $\mathbf{v}_k^{\mathcal{B}}$, and append its associated rigging weights $\mathbf{w}_k$ into each graph node descriptor. Additionally, we also append the shape descriptor β to each node. Therefore, the i^{th} node of the graph $\overline{\mathcal{G}}$ is defined as $\mathbf{n}_i = \{\mathbf{v}_i^{\overline{\mathcal{M}}}, \mathbf{w}_k, \beta\} \in \mathbb{R}^{3+J+|\beta|}$, where J is the number of body joints (24 for SMPL

[Lop*15]), and $|\beta|$ the number of shape coefficients (1 for the results shown in this paper).

We then input the graph $\overline{\mathcal{G}}$ into our fully convolutional regressor

$$R_{\text{smooth}}(\overline{\mathcal{G}}) = \Delta_{\text{smooth}} \tag{4.4}$$

to predict a vector of 3D displacements $\Delta_{\text{smooth}} \in \mathbb{R}^{3\times V^{\overline{\mathcal{M}}}}$. The architecture of the network, inspired by the success of fully convolutional U-Net [RFB15] for image segmentation, is depicted in Figure 4.7. The final deformed mesh of this second stage is then computed by adding the predicted 3D offsets to the mean shape 3D drape

$$\mathcal{M}_{\text{smooth}} = \overline{\mathcal{M}} + \Delta_{\text{smooth}}. \tag{4.5}$$

To train the regressor R_{smooth} we create a dataset of ground-truth deformations of two different materials and a range of body shapes using the physics-based cloth simulation [NSO12]. We leverage the whole set of training data without introducing bias due to material-dependent deformations by first applying a Laplacian smoothing operator to each generated mesh, and then computing the average of each corresponding sample (*i.e.,* those with same topology, garment type, and target shape) before substracting it from the mean shape to obtain the displacements $\Delta^{\text{GT}}_{\text{smooth}}$. As a loss function we use the ℓ^2-norm of the error between ground truth displacements and predictions, in addition to the ℓ^2 regularization of the network weights

4.2.4 Fine 3D Body Drape

The garment mesh $\mathcal{M}_{\text{smooth}}$ successfully reproduces the global garment deformations due to target body shape, but lacks fine details that depend largely on the material. We address such source of deformations in this third and last step by further deforming the garment mesh. To this end, we learn to regress a new set of 3D displacements Δ_{fine} using a fully convolutional network that takes as input a graph $\mathcal{G}$ built from the vertices positions $\mathbf{v}_i^{\mathcal{M}_{\text{smooth}}}$ and its associated rigging weights, analogous to the graph $\overline{\mathcal{G}}$ described in Section 4.2.3

$$R_{\text{fine}}(\mathcal{G}) = \Delta_{\text{fine}}. \tag{4.6}$$

Our final predicted 3D drape $\mathcal{M}_{\text{fine}}$ is then computed by adding the fine displacements onto the mesh $\mathcal{M}_{\text{smooth}}$

$$\mathcal{M}_{\text{fine}} = \mathcal{M}_{\text{smooth}} + \Delta_{\text{fine}}. \tag{4.7}$$

To train the regressor R_{fine} we use the same simulated fits as in Section 4.2.3. However, in this case, we take advantage of the material-dependent deformations and train one regressor per material type. We generate the ground truth offsets $\Delta^{\text{GT}}_{\text{fine},\, m}$ per each material m by substracting the smoothed fits from the simulated fits. As loss function for R_{fine} we use the same loss as R_{smooth}, with the ground truth fine-scale displacements $\Delta^{\text{GT}}_{\text{fine},\, m}$ instead.

4.2.5 Self-Supervised Learning of Body-Garment Collisions

The objective losses used to train regressors R_{smooth} and R_{fine} minimize the reconstruction error but, due to expected residual errors in unseen shapes and topologies, this term alone does not guarantee predicted deformations to be free of body-garment collisions. This is a common issue in learning based solutions, which has been address with rendering tricks [De *10], postprocessing steps [SOC19], or explicit collision loss terms [Gun*19] using supervised training. Inspired by the later, we propose a collision loss term that we can train in a *self-supervised strategy*, and therefore does not require to generate expensive ground truth simulations. This is a major advantage over previous explicit collision losses.

Specifically, for each vertex of the garment $\mathbf{v}_i{}^{\mathcal{M}}$ we find the closest body vertex $\mathbf{v}_k^{\mathcal{B}}$ and compute the collision loss as

$$\mathcal{L}_{\text{collision}} = max(-\mathbf{n}_k^{\mathcal{B}}(\mathbf{v}_i{}^{\mathcal{M}} - \mathbf{v}_k^{\mathcal{B}}), 0), \tag{4.8}$$

where $\mathbf{n}_k^{\mathcal{B}}$ is the normal vector of the body vertex. The work of Gundogdu *et al.* [Gun*19] uses this loss to penalize collisions during training, but unless the train dataset is exhaustive enough, this approach does not guarantee collision-free results for unseen inputs. In our particular case this is particularly bad, since creating an

exhaustive dataset of cloth simulations is not feasible due to the arbitrary topology input of our method.

Therefore, starting from network weights trained for R_{smooth} and R_{fine}, we propose a novel strategy to fine-tune our network R_{fine} using Equation 4.8 to produce collision-free results for arbitrary inputs. The key insight of our approach is that evaluating the collision loss does *not* require ground-truth data. This allows us to feed the network with random inputs and train on the collision loss only until it converges to a value near zero. To this end, during the self-supervised step we sample random body shapes β and garment topologies $\overline{\mathcal{M}}$, feed them into our pipeline, and use the predicted mesh to fine-tune R_{fine} with Equation 4.8. Thanks to this strategy the number of collisions has been reduced by 70% during training, and 20% in validation.

4.3 Dataset and Implementation

Our ground truth dataset has been generated from 19 different garment pattern designs, two different topologies per design, and 201 values for the body shape β from the SMPL body model [Lop*15], uniformly sampled within the range -3 and 3 (from which 100 have been exclusively used for test). The resulting meshes have between 1,414 (for the simpler case) and 3,581 (for long dresses) vertices. The dataset was generated for two different materials. We split our 38 topologies into 31 for training and 7 for validation. We qualitatively validate our method in some completely new designs with unseen combinations of design parameters that are **not** in our dataset. To train R_{smooth} (Section 4.2.3) we apply Laplacian smoothing to all samples in our dataset and compute the average mesh with the different material samples.

$\overline{\mathcal{T}}$ has a fixed size of 403 vertices, value which dynamically changes for $\overline{\mathcal{M}}$ depending on the garment complexity after the topology optimization step. To generate our data for the first step described in Section 4.2.1, in order to avoid potential topology-related problems (*e.g.*, highly distorted triangles, irregular vertex positions, etc.) at simulation time, we first use a high-resolution mesh of 17,246 vertices, and then consistently downsample the simulated meshes to 403 vertices.

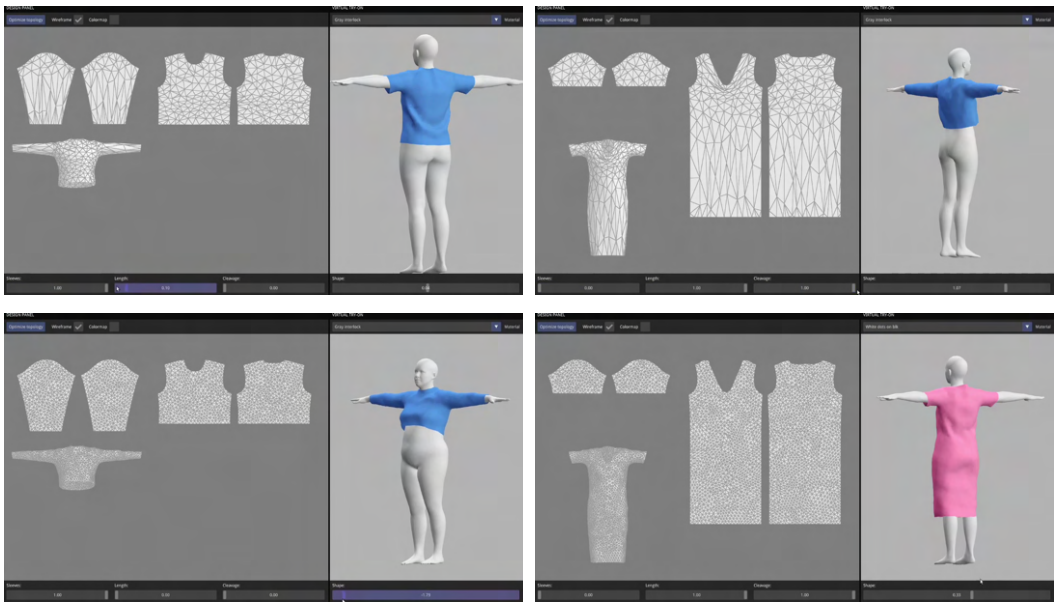

Figure 4.8: Demo implemented to test our method. The design panel allows for manipulation of the design parameters and deforms the low-resolution mesh $\overline{\mathcal{T}}$ interactively. Once the user has chosen the design, the "Optimize topology" button is pushed to activate the remeshing algorithm ($\overline{\mathcal{M}}$ is computed) and the preprocess (laplacians and down- and up-sampling matrices are computed). Then, the high-resolution garment is draped in the virtual try-on panel, where the shape of the human can be modified and the garment is draped interactively.

2D panel meshes are manually generated on a 3D modeling software, and the design parameters interpolate between these hand-made panels.

We have implemented our pipeline in TensorFlow for efficient GPU training and execution. The parametric 3D draping is a fully connected layer with 3 input neurons (one per design parameter) and a single hidden layer (of ten neurons) trained for less than a minute. Training the fully convolutional networks R_{smooth}, and R_{fine} took approximately 20, and 14 hours respectively, their architecture is depicted in Figure 4.7. Fine-tuning the self-supervised collisions took around one day. Everything was executed on a NVIDIA Titan X with 12GB.

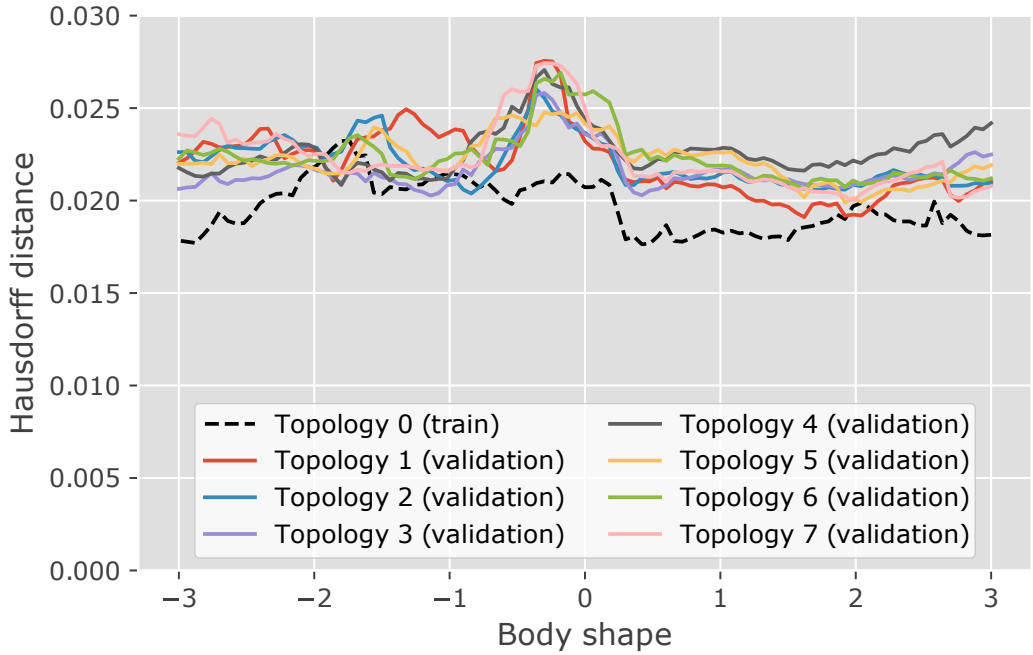

Figure 4.9: Generalization to new topologies. Hausdorff distance between the predicted and the ground truth meshes for a range of body shapes and 7 validation topologies. Errors in test topologies are consistent, demonstrating the generalization capabilities of our method, and on par to topologies used for training (dashed black).

4.4 Evaluation and Results

In this section, we quantitatively and qualitatively evaluate our results in different scenarios. Specifically, we demonstrate our generalization capabilities, compare with the state-of-the-art method of Santesteban *et al.* [SOC19], and with a newly proposed brute force baseline for parametric virtual try-on.

Evaluation of Generalization to New Topologies. In Figure 4.9 we quantitatively evaluate the generalization capabilities of the regressors R_{smooth} and R_{fine} to new topologies. Specifically, for a given garment parameters $\mathbf{p}$ and material for which we have ground truth simulated data, we randomized the topology (keeping the mean triangle area constant) of the mean shape predicted mesh $\overline{\mathcal{M}}$, and feed each topology to the regressors R_{smooth} and R_{fine} for a range of target shapes β. For each predicted mesh, we then compute the Hausdorff distance to the ground truth simulations. Results demonstrate that our method predictions are quantitatively

consistent, regardless the topology and target body shape. Importantly, we also show that the error of the topologies unseen at train time (*i.e.,* validation set) is on par with the error of topologies used to train (in dash black).

Comparison with Parametric Fully Connected Baseline. Despite the lack of methods than can cope with parametric garments due to the need for different topologies, an alternative brute-force approach could be to use a highly-dense topology in $\overline{\mathcal{M}}$ to represent *all* garments, followed by a fully-connected end-to-end network that predicts displacements over such mesh. This high dense topology would provide an over-discretized mesh which, although unnecessarily complex for small garments such as a t-shirt, would provide sufficient details for large garments such as dresses, technically enabling the use of fully-connected pipelines [SOC19]. We implemented such solution, which can be considered a baseline for data-driven parametric garments, and compared it with our fully convolutional approach.

In Figure 4.10 we present a quantitative evaluation of the precision accuracy of our method, and the fully connected baseline. Specifically, for a given garment design (unseen at training time) we compute the Hausdorff error for a range of target body shapes, and demonstrate that our predictions $\mathcal{M}_{\text{fine}}$ are consistently more accurate. Our hypothesis is that the fully-connected approach cannot generalize to garment types outside the training set due to the *global* nature of the densely connected neurons, that are unable to learn local features. In contrast, the convolutional nature of our approach is able to capture local features, and therefore correctly predicts deformations of garment types unseen at train time but locally present in train examples.

Furthermore, we also evaluate the memory footprint of each method, which also results favorable for us. The fully connected network size is 167 MB, while ours (R_{smooth} + R_{fine}) is 71MB. This is also expected, since the number of parameters for a fully connected network is significantly higher in comparison to the parameters used in the convolutional kernels. Note also that the fully connected approach needs to be fully trained for any new material while our approach enables easier generalization and transfer learning for new materials through fine-tuning R_{fine}.

Comparison with Santesteban et al. 2019. In Figure 4.11 we qualitatively compare our results with the state-of-the-art method of Santesteban *et al.* [SOC19], which is limited to a single garment. For a garment design analogous to the t-shirt used to train their method, we demonstrate that the predictions of both methods are on par (rows 1 and 2), while we are capable to predict the draping of a much larger number of garments (rows 3 and 4). This demonstrates the generalization capabilities of our method to arbitrary parametric garment design (and therefore, arbitrary topology).

Qualitative Results. In Figure 4.12 we show qualitative results of our method, for a variety of body shapes, garment types and topologies, all of them *unseen at train time*. Notice how the wrinkles predicted with our approach naturally match the expected behavior of the garment, and change for each shape-garment pair. This demonstrates that our method generalizes well to new garment types, topologies, and shapes. Check the supplementary video for more qualitative results.

In Figure 4.13 we show qualitative predictions of our method, for two different materials, but the same target body shape and garment type (both unseen at train time). We demonstrate how our final step R_{fine} is able to learn material-specific deformations, resulting in visually different folds and wrinkles. Specifically for this comparison, the blue t-shirt is train on `gray-interlock` (60% Cotton, 40% Polyester) material and the pink on `white-dots-on-black` (100% Polyester) from ARCSim materials [NSO12]. See [WOR11] for additional material details.

4.5 Conclusions

We have presented a method to predict the drape of a predefined parametric space of garments onto an arbitrary target body shape. To achieve this, we propose a novel fully convolutional graph neural network that, in contrast to existing methods, is not limited to a single garment or topology. Our novel pipeline, based on U-Net architecture and efficient graph convolutions, generalizes to unseen mesh topologies, garment parameters, and body shapes. To the best of our knowledge, ours is the first fully convolutional approach for virtual try-on purposes, which opens the door

to more general data-driven cloth animation methods based on geometric deep learning.

Despite our step forward in geometric learning-based solutions for cloth animation, our approach still suffers from the following weaknesses that could be addressed by follow-up works. Pose-dependent and material-dependent *input* parameters are not considered to our approach, and you need to retrain the model to consider these configurations. Multi-layer garments and contact with external forces are not considered either. Additionally, commercial garment design definitely requires more than 3 parameters. The analysis of the scalability of the proposed method to a larger garment space remains open for future research.

After the publication of this work, we addressed an important limitation with positive results. Initially we trained the network with two different triangulations for each design (see Section 4.3). To do so, we simulated the deformations with each of the meshes and used them to train the network, but the problem is that differences in the discretization lead to differences in the simulation, and the network was getting 2 slightly different deformations for each garment, which is confusing and led to smoothing. To overcome this issue, we generated new simulations for each design with high resolution meshes. Then, we generated 20 different topologies for each garment, and adjusted each sample to the high-resolution simulation. This augmented and normalized dataset helped the network and led to a significant improvement of the results.

Another significant limitation is the sampling preprocess. Although our sampling algorithm is very efficient once the matrices are precomputed, there is a computational cost at the pre-processing step. Besides, by construction, some features are lost in the pooled layers. We tried different approaches for learned pooling [GJ19; Yin*18b], but they didn't improve the quality of our method.

Even if our self-supervised strategy, significantly reduces the occurrence of penetrations, there are still some occurring in our results. For safety, we need to apply a post-processing step to handle them.

Last, but not least, the complexity of modeling deformations with graph convolutions was challenging. It involves very long training times and smoothing artifacts. These issues only went worse when we started dealing with bigger datasets and

more complex deformations. While the network did well at estimating the overall movement of pose-dependent deformations, it lost a lot of high-frequency detail. For that reason, we decided to try an image-based approach, that still is agnostic to discretization.

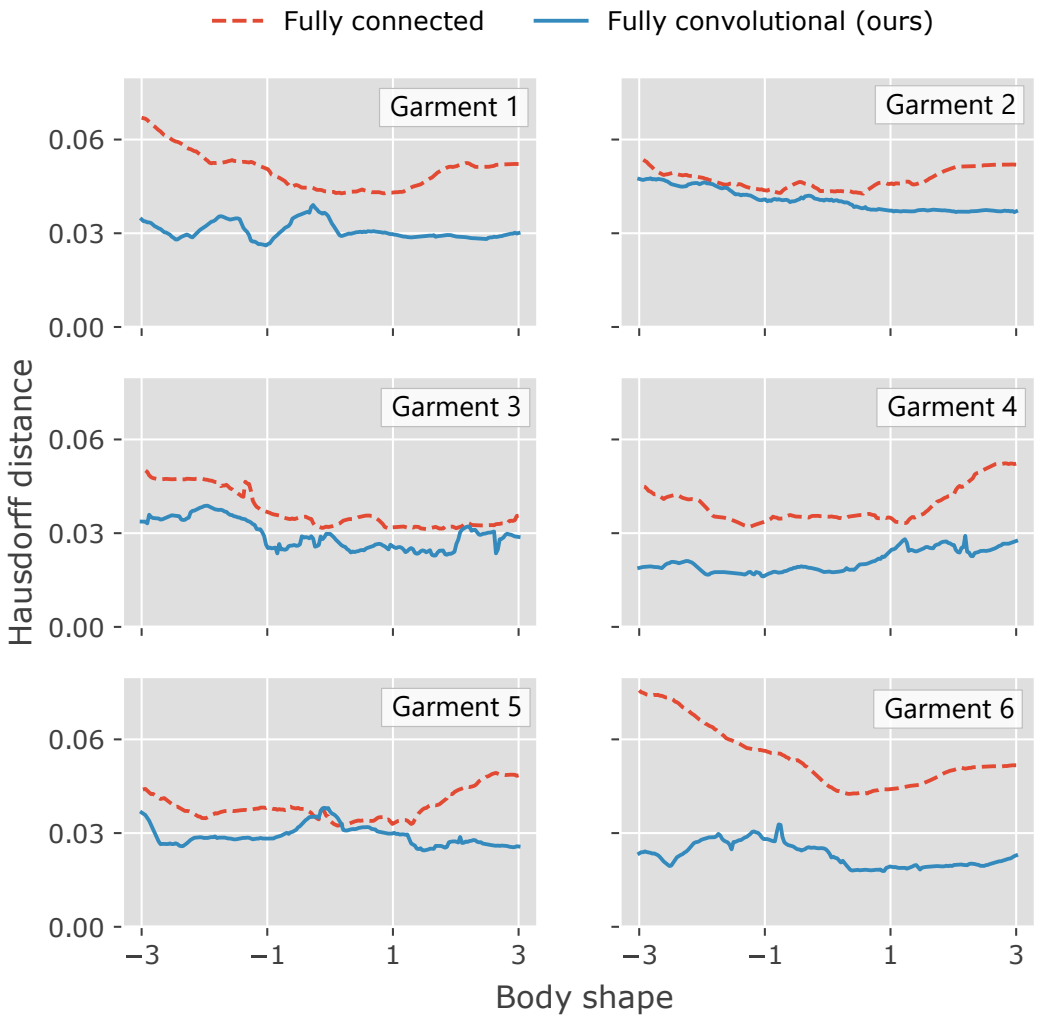

Figure 4.10: Quantitative evaluation of our fully convolutional (solid blue) approach and the fully-connected baseline (*i.e.*, using the same highly-dense topology for all garments and a fully-connected architecture, dashed red), for 6 garment designs not present in the training set. Our approach consistently outperforms the fully connected baseline since the latter cannot generalize well to unseen garment types.

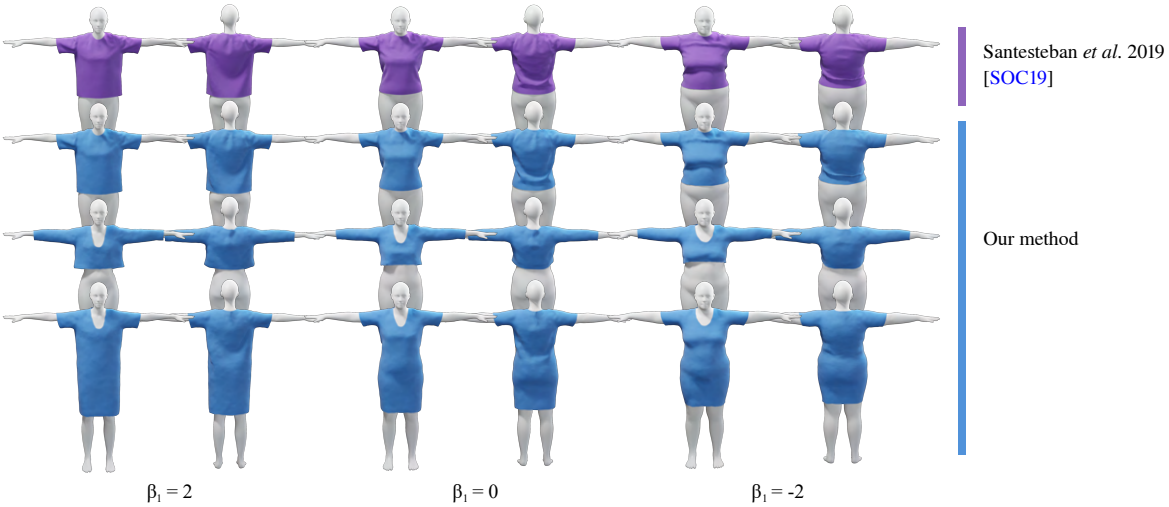

Figure 4.11: Qualitative comparison with the single-garment and fix topology method of Santesteban *et al.* [SOC19] and ours. When sampling the same garment type use to train their method, our results are on par with Santesteban's (rows 1, 2), while our approach allows for a much richer space of garment types and topologies (rows 3, 4).

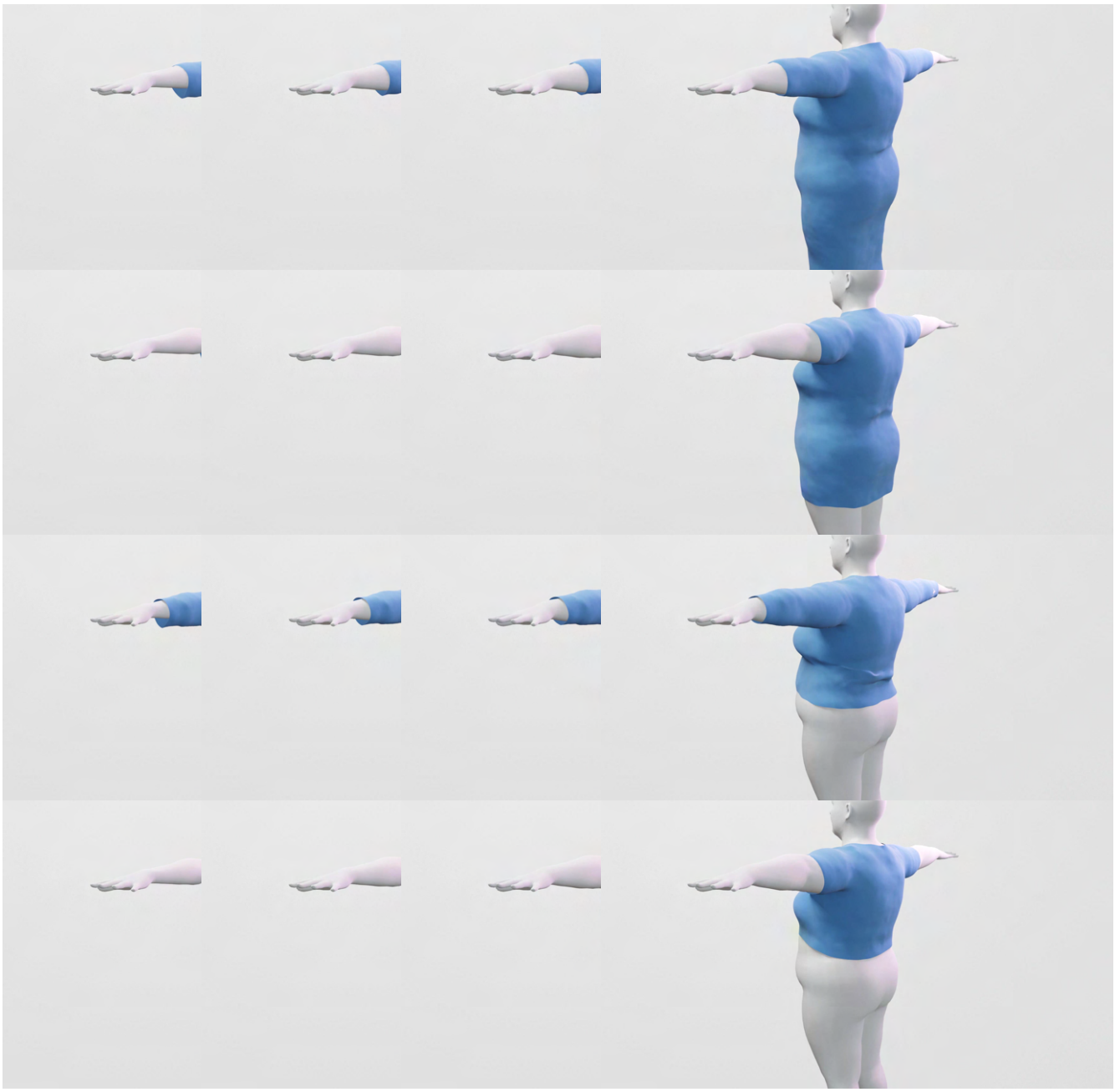

Figure 4.12: Virtual try-on results with our method, for a variety of garments (rows), fitted into a range of shapes (columns), both unseen at train time. Our method successfully predicts the drape of the garment, with natural folds and wrinkles at different scales that depend both on the input garment type and the target body shape.

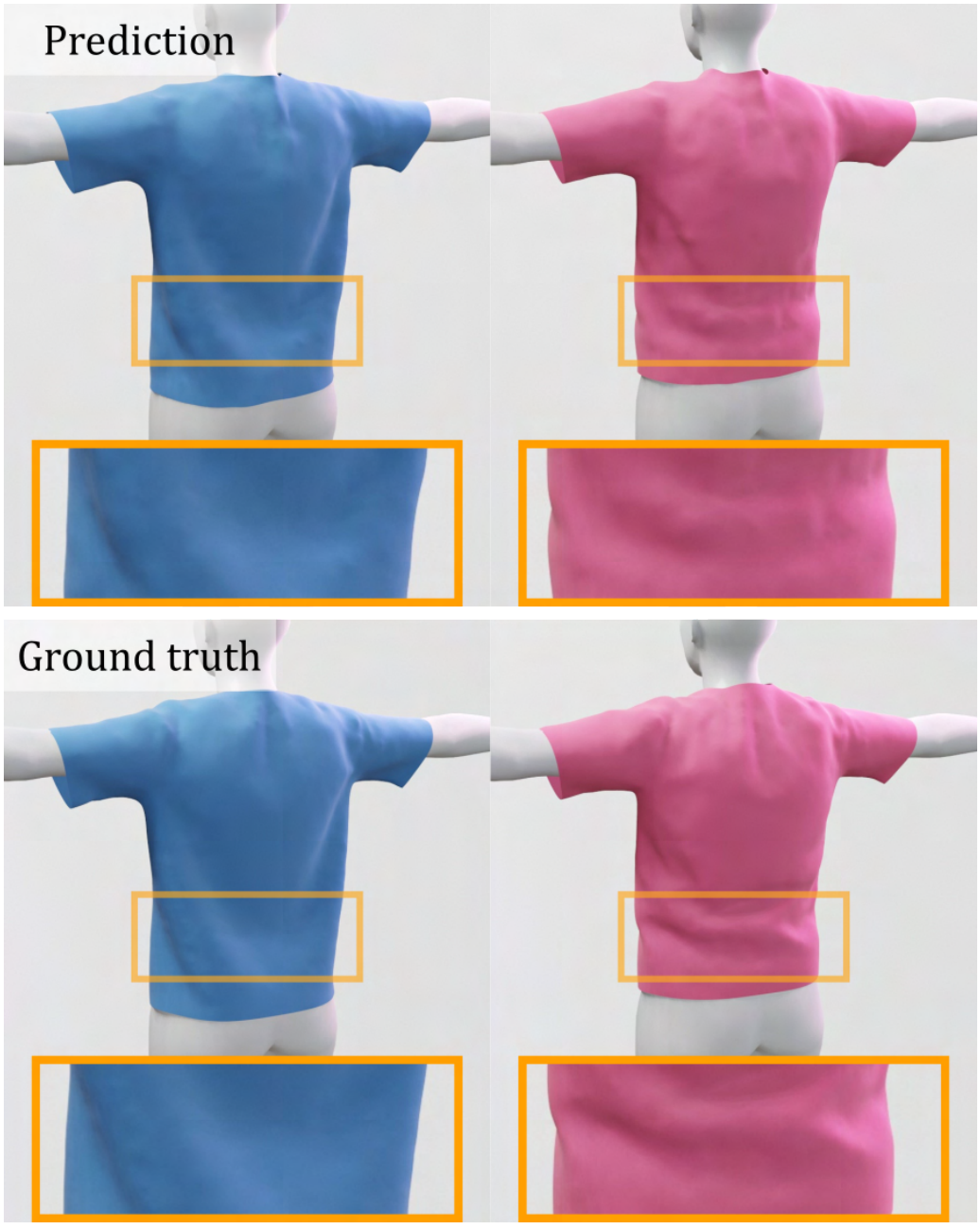

Figure 4.13: Deformations regressed by our method for two different materials, presented in blue and pink. We demonstrate that, given the same target shape and input garment type, our method (top) is able to learn material-specific details that produce distinctive folds and wrinkles, closely matching the ground truth deformations (down).

5

DiffusedWrinkles: A Diffusion-Based Model for Data-Driven Garment Animation

Learning-based methods provide an alternative to computationally expensive traditional physics-based approaches for the complex task of cloth modeling. However, these methods often face challenges in generalizing to unseen garment mesh discretizations and maintaining high-frequency detail. The main reason is that, while deep learning models are optimized to work on regular domains, their extension to complex and irregular domains, like 3D meshes is not trivial.

In our previous work (Chapter 4), we decided to tackle this problem by extending convolutional architectures to handle graph-like structures. Despite the novelty of the method and the good results, its extension to larger datasets was challenging, as training the networks took a long time and they struggled with reproducing fine wrinkles.

To circumvent this limitation, some works model 3D cloth with point clouds or implicit representations, but detailed and topologically consistent mesh outputs remain challenging. Alternatively, instead of working in the 3D domain, some works have explored the use of 2D image-based representations to encode 3D garments. The key idea underlying these approaches is to leverage the well-studied deep learning architectures for image processing to model garment details. A common approach is to use Generative Adversarial Networks to enrich or infer 2D representations. However, GANs are difficult to train (*e.g.*, they easily suffer from vanishing gradient or mode collapse issues [AB17]) and their expressiveness is limited.

Figure 5.1: Samples of a DDPM trained on ImageNet. **Source:** [DN21]

Recently, Denoising Diffusion Probabilistic Models [HJA20] have emerged as a successful alternative for image synthesis. These models are trained through a denoising diffusion process and are capable of generating high-quality images by denoising Gaussian noise. DDPMs have shown to outperform GANs in many tasks [DN21] while being faster and easier (as they don't suffer from collapse issues) to train (see Figure 5.1 to see some images generated by early DDPMs).

In this chapter, we show DiffWrinkles, our method to generate detailed deformations using DDPMs. The core idea of the method is to build a robust 2-dimensional representation of garment deformations and to train a DDPM to generate these deformation maps conditioned by pose, shape, and design parameters. Our model creates high-quality animations, is agnostic to mesh topology, and has the capability of synthesizing various plausible deformations for one pose-shape-design configuration.

Core to our model is the garment representation. We propose to encode the deformations of a dataset of animations as 2D layout-consistent displacement maps. With this representation, we can easily leverage the generative capabilities of DDPMs to synthesize new maps of deformations. We condition our diffusion model with a conditional embedding, containing pose, shape, and design information. Besides, due to the generative nature of the approach, we need to condition the model with the previous state of the garment, to create temporal-coherent sequences.

We qualitatively and quantitatively demonstrate that our approach is capable of generating high-quality 3D animations for a wide variety of garments, body shapes, and motions, outperforming the closest previous works for similar tasks that are based on graph neural networks or MLPs.

The remainder of the chapter is structured as follows. In Section 5.1 we describe DDPMs, which are crucial to our model. In Section 5.2 we introduce our novel garment representation which consists on a 3D mesh encoded with an MLP network to represent the global garment design, and an image-based representation to store folds and wrinkles produced by body pose and shape. We learn each of these terms in a data-driven strategy. To this end, in Section 5.3 we present our key contribution and introduce a diffusion-based model to learn predict our image-based wrinkles representation. Later, in Section 5.4 we demonstrate that our approach enables the animation of a large collection of designs, producing compelling folds and wrinkles in animated test sequences. Finally, in Section 5.5 we conclude with a summary of contributions and future work directions.

5.1 Background. Denoising Diffusion Probabilistic Models

Denoising Diffusion Probabilistic Models are a class of generative models that produce images by reversing a diffusion process. They have recently gained popularity, thanks to their stability, simplicity, and capability to produce high-quality images. The idea of using a diffusion process for generative learning was first introduced by Sohl-Dickstein *et al.* [Soh*15], but it wasn't until the seminal paper by Ho *et al.* [HJA20] that DDPMs were formalized and their effectiveness was proven. Since then, they have only gained prominence and multiple works have emerged improving their performance [DN21; ND21], scalability [PX23; Per*23] and their range of applications (video [Ho*22b; Bla*23], medical image reconstruction [Pen*22])

DDPMs are a class of deep generative models that produce images by reversing a diffusion process. They are based on two stages, the forward diffusion stage and the reverse diffusion stage. In the forward diffusion process, Gaussian noise is progressively added to the data over a fixed number of steps, until a normal

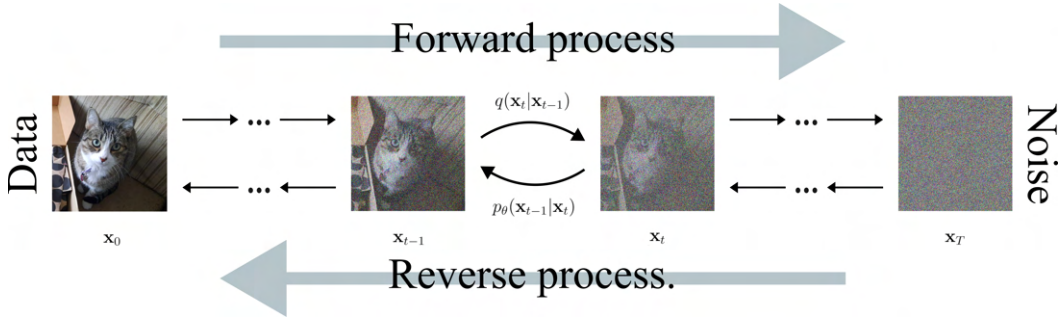

Figure 5.2: Intuition behind DDPMs. The forward process adds noise, and the reverse process learns to remove it.

distribution is obtained. In the reverse process, a model is trained to reverse the diffusion process (*i.e.* gradually remove the noise). This model can then recover images from the data distribution by iteratively denoising random samples of a normal distribution.

Let's describe the formulation of the different parts of DDPMs

Forward process. Let $\mathbf{x}_0$ be an original image of the dataset with data density $q(\mathbf{x}_0)$. The index 0 indicates that no noise has been added to the image. Then, the noised versions are obtained with a Markovian chain, as follows:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}), \forall t \in \{1, ..., T\}, \tag{5.1}$$

where $\mathcal{N}(\mathbf{x}; \mu, \theta)$ is the normal distribution producing x, with mean μ and covariance θ, T is the number of diffusion steps, and $\beta_1, ..., \beta_T$ are the parameters representing the noise schedule (later, we will discuss their properties further). Note that, if we define $\alpha_t = 1 - \beta_t$ and $\overline{\alpha}_t = \prod_{i=1}^{t} \alpha_i$, then the distribution can be rewritten as

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\overline{\alpha}_t}\mathbf{x}_0, (1-\overline{\alpha}_t)\mathbf{I}). \tag{5.2}$$

Sampling $\mathbf{x}_t$ from the distribution $q(\mathbf{x}_t|\mathbf{x}_0)$ is equivalent to computing

$$\mathbf{x}_t = \sqrt{\overline{\alpha}_t}\mathbf{x}_0 + \sqrt{(1-\overline{\alpha}_t)}\epsilon \tag{5.3}$$

where $\epsilon \sim \mathcal{N}(0, \mathbf{I})$.

Intuitively, this means that sampling the distorted image at a certain timestep t can be directly done by computing the weighted sum of the original image and a randomly sampled Gaussian noise image.

Noise schedule. $(\beta_t)_{t=1}^T \in [0, 1)$ represent the noise variance across the different diffusion steps. If they are chosen such as $\overline{\alpha}_T \approx 0$, then the distribution of $\mathbf{x}_T$ can be approximated by the standard Gaussian distribution. In the seminal paper by Ho *et al.* [HJA20] they set the schedule to be linearly distributed from $\beta_1 = 10^{-4}$ to $\beta_T = 0.02$, but later approaches demonstrated that cosine schedule yields to better results. The intuition is that this schedule makes $\overline{\alpha}_t$ change slower at the extremes, when t is close to 0 and T, avoiding abrupt changes in noise level.

Reverse process. In theory, new samples can be obtained from the distribution $q(\mathbf{x}_0)$ by starting from a sample $\mathbf{x}_T \sim p(\mathbf{x}_T) = \mathcal{N}(\mathbf{x}_T; 0, \mathbf{I})$ and following the reverse steps with the following distribution:

$$p(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu(\mathbf{x}_t, t), \Sigma(\mathbf{x}_t, t)), \tag{5.4}$$

where the mean $\mu_\theta(\mathbf{x}_t, t)$ and the covariance $\Sigma_\theta(\mathbf{x}_t, t)$ can be predicted with a neural network, given the noisy image $\mathbf{x}_t$ and the timestep t. Then,

$$p_\theta(\mathbf{x}_{0:T}) := p(\mathbf{x}_T) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t), \tag{5.5}$$

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \Sigma_\theta(\mathbf{x}_t, t)). \tag{5.6}$$

Training objective. As we want $p_\theta(\mathbf{x}_0)$ to fit the distribution of the data, the ideal objective would be to maximize the log likelihood of the distribution, but this is intractable. Instead, a variational lower bound on the negative log likelihood is used for optimization:

$$\mathbb{E}[-\log p_\theta(\mathbf{x}_0)] \leq \mathbb{E}_q[-\log \frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}] := \mathcal{L}_{vlb}(\theta), \tag{5.7}$$

so minimizing this function is equivalent to maximizing the likelihood. This idea was developed before by Kingma and Welling [KW14].

Ho *et al.* propose to fix $\Sigma_\theta(\mathbf{x}_t, t))$ and rewrite μ_θ as a function of noise ϵ_θ. After some mathematical derivations, they show that a simplified variant of the variational lower bound can be used for training:

$$\mathcal{L}_{simple}(\theta) := \mathbb{E}_{\epsilon,\mathbf{x}_t,t}[\|\epsilon - \epsilon_\theta(\mathbf{x}_t, t)\|_2^2] = \mathbb{E}_{\epsilon,\mathbf{x}_0,t}[\|\epsilon - \epsilon_\theta(\sqrt{\overline{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\overline{\alpha}_t}\epsilon, t)\|_2^2] \tag{5.8}$$

In practice, when training, t is randomly sampled from a uniform distribution and the noisy image $\mathbf{x}_t$ is computed with $\mathbf{x}_0$ and a random sample of Gaussian noise ϵ. Given t and $\mathbf{x}_t$, the network $\epsilon_\theta(\mathbf{x}_t, t)$ returns an estimation of ϵ, that is used to compute the loss to take gradient descent and optimize its parameters. Once trained, at sampling, starting from a random noise sample $\mathbf{x}_T$ we iteratively compute $\mathbf{x}_{t-1}$ as a function of $\mathbf{x}_t$ and t until $t = 1$.

5.2 Garment Representation

Our garment representation builds on top of the existing 3D parametric human models (*e.g.*, [Lop*15; JSS18]), borrowing their shape β and pose θ parameterization used to encode the identity and skeletal configuration of the subject. More specifically, and inspired by previous works [Vid*20; SOC19], we extend SMPL body model formulation [Lop*15] to represent a deformed garment as

$$M_g(\beta, \theta, \mathbf{p}) = W(T_g(\beta, \theta, \mathbf{p}), J(\beta), \theta, \mathcal{W}), \tag{5.9}$$

where W is a skinning function (*e.g.*, linear blend skinning, or dual quaternion), $J(\beta) \in \mathbb{R}^{3\times 24}$ the body joint positions, and $\mathcal{W}$ the skinning weights of a deformable garment $T_g(\cdot)$.

Our key difference with our previous work (described in Chapter 4) is the representation used to encode and learn the deformable garment $T_g(\cdot)$, which allows us to learn fine-wrinkle detail while being agnostic to both the template mesh topology and the surface discretization detail. To this end, we propose a deformable template

$$T_g(\beta, \theta, \mathbf{p}) = G_{\text{design}}(\mathbf{p}) + \phi(G_{\text{wrinkles}}(\beta, \theta, \mathbf{p})) \tag{5.10}$$

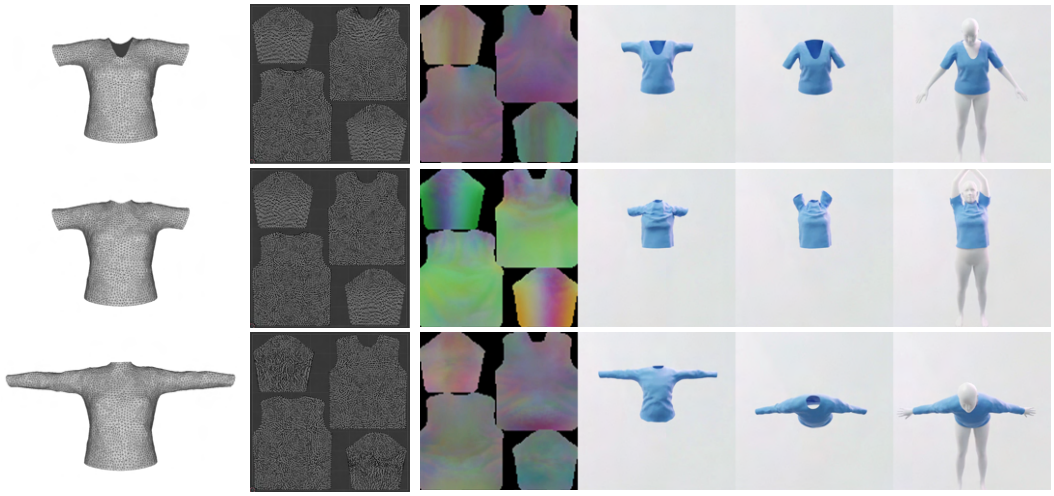

Figure 5.3: All our parametric garments have UV coordinates, such that their texture maps are aligned. The displacement maps are generated in the same layout and they are transformed to offsets in the 3D unposed space. Finally, the mesh is reposed and a post-process to remove penetrations is applied.

where the first term models the global deformation of garment due to the design parameter $\mathbf{p}$, and the second term models the local wrinkle details due to body pose θ, shape β, and design $\mathbf{p}$. In the rest of this section, we provide more details on how we model each of these terms.

The G_{design} term models the global design-dependent deformations in T-pose. In practice, we learn a function $G_{\text{design}} : |\mathbf{p}| \rightarrow N_{\text{g}} \times 3 + N_{\text{g}} \times 2$ using a shallow multilayer perceptron (MLP) network that outputs N_{g} 3D vertex positions and their corresponding 2D texture coordinates of a morphable T-shirt template parameterized by sleeve length, font-and-back pannel length, and cleavage (*i.e.*, the basic set of design parameters that enable the modelling of dresses, t-shirt, sweater, tops, and similar garments). Importantly, we design our garment model such that all designs share the same UV parametrization.

The G_{wrinkles} is our key contribution to the garment model, and addresses the goal of adding pose-dependent and/or shape-dependent deformations to the output of G_{design}. In contrast to previous works, which use displacements encoded in an MLP [SOC19; SOC22] or graph neural networks [Vid*20], we opt for encoding the deformations in a 2D displacement map stored as a RGB image (*i.e.*, a UV texturemap). The $\phi : 2 \rightarrow 3$ operator represents the projection operator from 2D pixel coordinates to 3D mesh coordinates which, in practice, we implement using

the known mesh surface parameterization implicit in the UV coordinates. Notice that, a key design choice of our garment representation is that all G_{design} outputs share a common mesh parametrization, which means that they all use *the same* 2D layout despite encoding different designs. This is a fundamental property of our representation that significantly simplifies the learning of garment wrinkles, since it spatially normalizes our ground truth data.

5.3 Data-Driven Diffusion-based Wrinkles

In this section, we describe how we learn the term G_{wrinkles} of our garment model defined in Equation 5.10 using a diffusion model.

Diffusion models can be conditioned on one or more input variables. To this end, these variables are typically encoded in a *conditional embedding*, as shown in Figure 5.4, through a Multilayer Perceptron (MLP), and introduced in the neural network in different layers. In the rest of this section, we first describe our conditional diffusion model for estimating wrinkles in a static scenario for a target pose, shape, and design (Section 5.3.1). Then, we describe how we can incorporate temporal constraints into our diffusion model to enable the generation of temporally coherent animations of 3D garments (Section 5.3.2). Implementation details are described in Section 5.4.2.

5.3.1 Pose-shape-and-design Conditional Wrinkles

Our goal is to learn a conditional diffusion model of the form $p(\mathbf{y}|\mathbf{c})$, where $\mathbf{y} \leftarrow G_{\text{wrinkles}}(\beta, \theta, \mathbf{p})$ is a UVs image representing the displacement vector and $\mathbf{c} = [\beta, \theta, \mathbf{p}]$ is the conditioning vector that includes shape β, pose θ, and design $\mathbf{p}$ parameters. Given G_{wrinkles}, the deformation is obtained through Equation 5.10.

Our diffusion model follows the formulation of Ho *et al.* [HJA20] that learns to predict the noise ϵ added at a certain step t of the markovian chain.

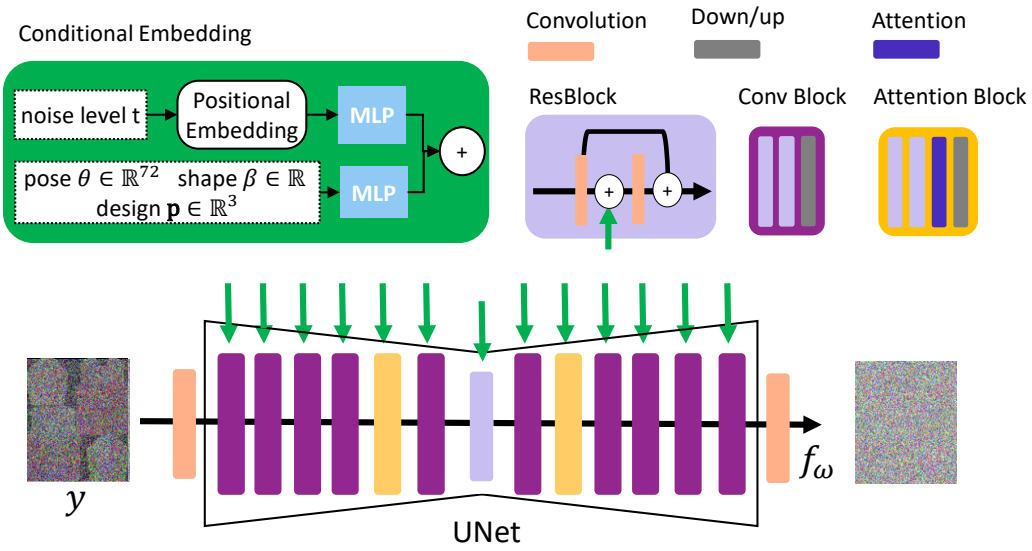

Figure 5.4: Our neural network architecture is a UNet with six Resnet blocks as shown in the diagram. The conditioning vector, aggregated in the ResNet blocks, contains an embedding of the pose, shape, and design parameters, as well as the noise level t.

For training, we iteratively add random noise to the ground truth data until convergence, according to the following loss function:

$$\mathcal{L}(\omega) = \mathop{\mathbb{E}}_{\epsilon,\mathbf{y}_0,t,\mathbf{c}} \left\| \epsilon - f_\omega \left(\mathbf{c}, \sqrt{\bar{\alpha}_t}\mathbf{y}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t \right) \right\|_2^2 , \tag{5.11}$$

where f_ω is the learned neural network, $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ is randomly generated Gaussian noise, $t \sim \mathcal{U}(\{1, ..., T\})$ is sampled from the Uniform distribution, and $\mathbf{y}_0$ the ground truth sample. Finally, $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$ is the aggregated noise variance that can be computed in closed form at any timestep t [HJA20].

For inference, we perform the *reverse* process iteratively computing the following equation:

$$\mathbf{y}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{y}_t - \sqrt{1-\alpha_t} f_\omega \left(\mathbf{c}, \mathbf{y}_t, t \right) \right) \tag{5.12}$$

At the beginning of the diffusion process ($t = \mathrm{T}$) the initial value for $\mathbf{y}_{\mathrm{t=T}}$ is virtually indistiguishable from Gaussian noise. Then, iteratively, from $t = \mathrm{T}$ until $t = 1$ this image is denoised by substracting the outputs predicted by the neural network f_ω until we obtain an approximation of $\mathbf{y}_0$.

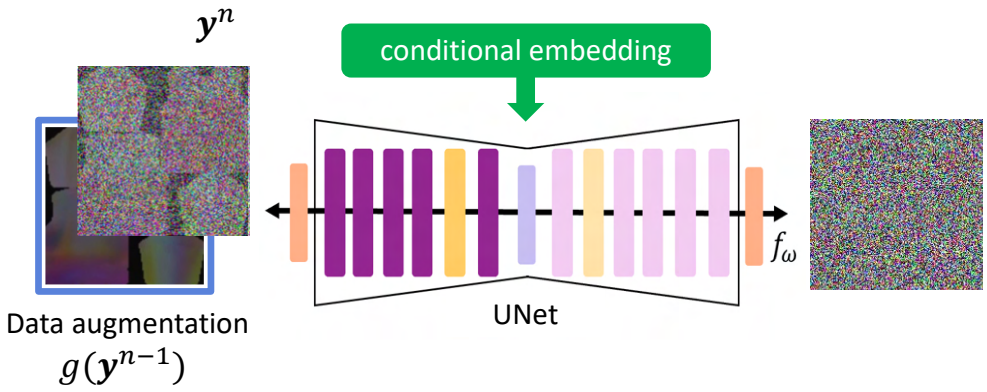

Figure 5.5: Temporal coherent diffusion model. To account for temporal consistency in the generated sequences while varying pose parameter, we concatenate the output of the previous frame in the sequence.

5.3.2 Temporally Coherent Garment Wrinkles

Using the diffusion model described in Section 5.3.1 we can generate plausible wrinkles conditioned on pose, shape, and design. However, if we sample the model for a sequence of poses, we will obtain a non-temporally coherent animation: consecutive frames will exhibit significantly different deformations. This is due to the generative nature of the model, since we sample it with random noise it can produce *different* results even for the same condition. This prevents the conditional model from Section 5.3.1 to generate temporally coherent animations of garments.

To tackle this issue, we take inspiration from cascade models for high-resolution image synthesis that condition a sample on a low-resolution version of the target image to drive the diffusion process towards a specific target [Ho*22a]. We propose to use a similar strategy to enforce temporal coherency. To this end, to synthesize the garment deformations at frame n, we further condition our diffusion model from Section 5.3.1 on the output image $\mathbf{y}^{n-1}$ of the previous frame $n-1$ of the sequence. In practice, we implement this by adding into our neural network f_ω an extra input $g(\mathbf{y}^{n-1})$ that is concatenated to $\mathbf{y}$. To avoid overfitting this new conditional signal to the training ground truth values of $\mathbf{y}^{n-1}$, we apply several perturbations $g()$ to the UV images will be described in the implementation section. Figure 5.5 illustrates this process.

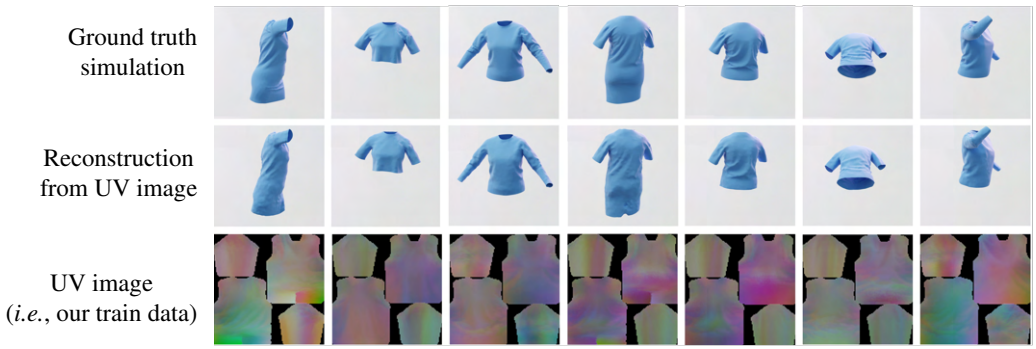

Figure 5.6: A few samples of our dataset. We simulate a large variety of garment designs on different body shapes and poses (top row), which we then convert into a UV image that encodes 3D garment deformations as per-vertex 3D displacements stored as RGB pixel values (bottom row). Using such image-based representation, we can faithfully reconstruct the original garment (middle row).

5.4 Results and Evaluation

In this section we first provide details about our dataset, discussing how we construct our UV image, and then provide implementation details. Finally, we qualitatively and quantitatively evaluate our results, and compare with competing state-of-the-art methods for 3D garments.

5.4.1 Dataset

To train our method, we first build a large dataset of UV-encoded deformations for a variety of garment designs worn by different body poses and shapes. To this end, we first manually create a deformable template of a 3D garment parameterized by length, sleeve, and cleavage. Importantly, all designs sampled by this parametric template share the same 3D-to-2D parameterization (*i.e.*, the same UV layout).

Using a state-of-the-art cloth simulator [NSO12], we statically simulate a wide variety of garment designs worn by different SMPL [Lop*15] body sequences from AMASS dataset [Mah*19]. For each simulated frame, similar to [SOC19], we project the deformed garment into a canonical state (*i.e.*, T-pose) by unposing the mesh using the inverse transform of the skinning weights of the underlying body pose. We then compute the per-vertex offset between the unposed mesh and the

template mesh and store it as an RGB value of a texture image using the known 3D-to-2D mapping. Following this strategy and using standard barycentric coordinates, we can assign a value to all pixels of the texture map. Generated texture maps effectively encode the 3D garment deformations in a convenient 2D image format that can be exploited with a diffusion model. Following the reverse process, we can reconstruct a deformed 3D garment by querying the UV texture value of each vertex, and then posing the garment using the skinning values of the target pose, as shown in Equation 5.9.

Figure 5.6 depicts a few samples of our dataset including ground truth simulations (top), the corresponding UV texture encoding deformations (bottom), and the reconstructed 3D garment from the UV images (middle). Notice that reconstructions closely match simulations, despite using an underlying very compact 2D representation (128×128 pixels for the results throughout the paper).

In practice, we simulate 17 designs of garments (7 different garment lengths, 6 different sleeves, and 4 types of cleavage) in 52 sequences, and generate a 128×128 pixels UV textures to encode the deformation of each frame. We train on 11 designs and leave out 6 designs and 5 sequences for validation. Once trained, our model generalizes to unseen combinations of garment parameters, producing plausible deformations for new garment designs.

5.4.2 Network Architecture and Implementation Details

Our neural network f_ω from Section 5.3.1 is implemented as a symmetrical UNet that consists of six downsampling residual layers. The fifth layer includes a spatial self-attention block, which has been proven successful in performing global reasoning [Vas*17]. Each ResNet layer has two layers, and the number of output channels for each UNet block is 128, 128, 256, 256, 512, 512. The conditional embedding is implemented as a 2-layer MLP with a 128 feature vector. We use a cosine noise scheduler of 100 timesteps. We train our model using a batch size of 8 for 100 diffusion steps using a single NVidia Titan X. On average, our model takes 3.5 seconds to generate an image.

Our temporally coherent diffusion model architecture described in Section 5.3.2 is analogous to the design of f_ω described above. The key difference is the input, which is expanded with the previous frame of the sequence. The architecture does not need to be updated as both images are concatenated, only changing the depth of the intermediate outputs. Because at this step the previous frame will already be converged, it will be a strong signal for the network and potential cause of overfitting. To avoid it, we apply a data augmentation process consisting of randomly applying Gaussian blur and color jitter effects.

5.4.3 Evaluation

We quantitatively and qualitatively evaluate our results, including comparisons to the closest state-of-the-art works on data-driven parametric garments.

Quantitative evaluation. Figure 5.7 presents a quantitative evaluation of our proposed diffusion model. The blue curve represents the model conditioned on pose-shape-and-design (Section 5.3.1), while the red curve represents our temporally-coherent model additionally conditioned on the previous state of the garment (Section 5.3.2). For each model, we plot the per-vertex position error (left) and the velocity error (right) compared to two ground truth simulations on two validation garments designs (top and bottom) unseen at train time.

Our temporally-coherent diffusion model consistently outperforms the static model only conditioned on pose-shape-and-design, delivering lower and much more stable per-frame vertex error. This is clearly observed at the vertex velocity error plots (Figure 5.7, right). Our temporal model (in red), conditioned on the previous garment state, closely matches the ground truth velocity, while a static per-frame deformation synthesis (in blue) significantly and incoherently differs from the ground truth. A qualitative visualization of this plot can be found in the supplementary video, showcasing smooth surface deformations over time when using our temporal model.

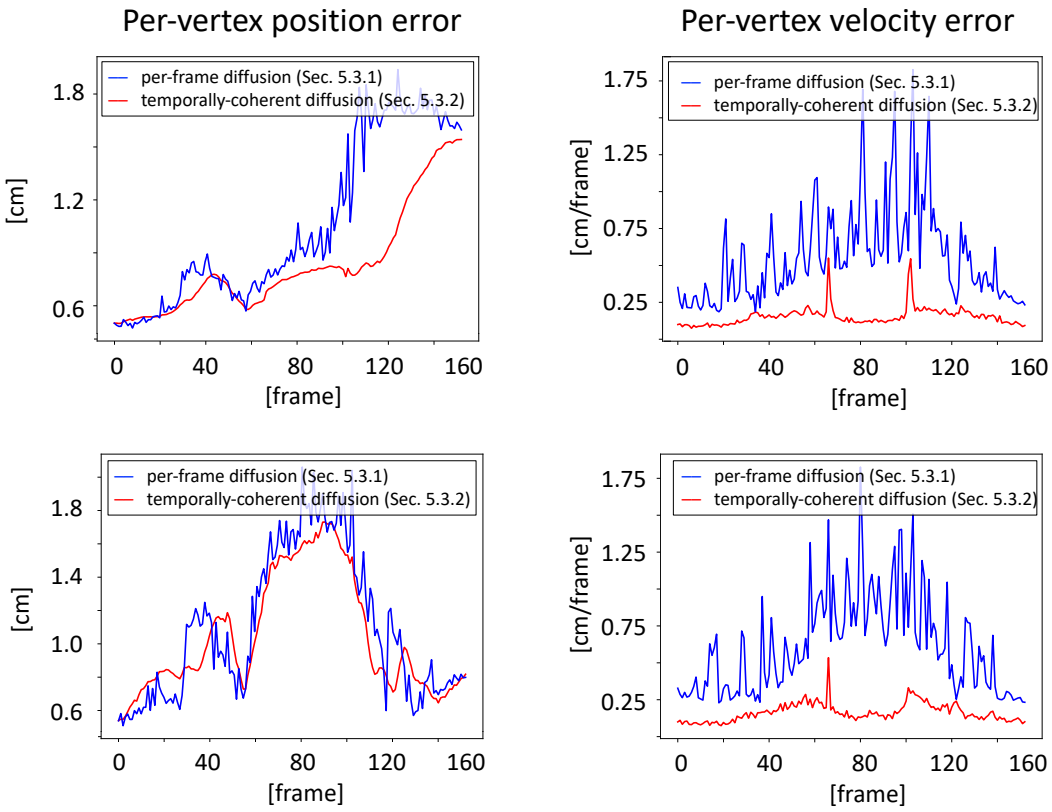

Figure 5.7: Quantitative evaluation of our temporally-coherent diffusion model (in red) and per-frame diffusion model (in blue). Since our temporal model is conditioned on the previous deformation state of the garment, the resulting animations are temporally smooth (see per-vertex velocity error, right) and closer to the ground truth surface (see per-vertex position error, left). Evaluated on two validation designs (top and bottom).

Qualitative evaluation. Figure 5.8 presents a qualitative comparison of the results obtained with our diffusion model (bottom) and ground truth simulations (top) on different garment designs under different body poses, all unseen at train time. Despite the challenging dynamic deformations, exhibiting a wide variety of folds and wrinkles in each frame, our model synthesizes fine deformations that closely match the ground truth. See the supplementary video for more animated results.

Figure 5.9 presents a large mosaic of five different validation garment designs (columns A-E) worn by differently posed bodies. Designs include long dresses with various neck and sleeve styles, t-shirts, tops, and shirts with different sleeve lengths. Notice that each frame exhibits unique nuances, showcasing rich, different, and dynamic folds and wrinkles that realistically match the underlying body pose.

This mosaic validates the large expressively of our proposed diffusion-based model. Similarly, Figure 1.2 shows three different designs worn during a hip-hop dancing motion from AMASS [Mah*19] dataset (sequence `50027`), exhibiting natural pose-shape-and-design 3D clothing deformations.

Qualitative comparison to state-of-the-art. Figure 5.10 presents a qualitative comparison with our previous method, presented in Chapter 4. We show garment deformations obtained by each method for a test design in various body shapes. Notice that our previous work does not model pose-dependent deformations, hence we limit our comparison to T-pose avatars. Our method obtains deformations that closely match the ground truth simulation, which demonstrates that our diffusion-based model is more expressive than the fully-convolutional graph model of our first contribution.

In the Figure 5.11 we also show some results to qualitatively compare our method with the self-supervised self-supervised methods SNUG [SOC22] and HOOD [Gri*23]. It is difficult to faithfully quantitatively compare these methods given the significant differences in representations, models, and goals. For example, SNUG is capable of modeling dynamics but it is limited to a single garment. Similarly, HOOD produces very compelling results and works also unseen garments, but it is not generative, does not explicitly incorporate design parameters, and uses a graph-based representation. Despite these differences, qualitative comparison demonstrates that our method is on par with the deformations showcased by state-of-the-art methods, while using a very compact image-based representation.

5.5 Conclusions

In this chapter we have presented DiffusedWrinkles, a generative method to synthesize 3D garment deformations conditioned on pose, shape, and design. Under the hood, our method uses a 2D diffusion-based model that encodes 3D garment deformations into texture maps. Leveraging a carefully designed 2D-to-3D surface parameterization, a wide family of 3D garment designs can be represented using a consistent 2D layout, which opens the door to image-based diffusion models to be

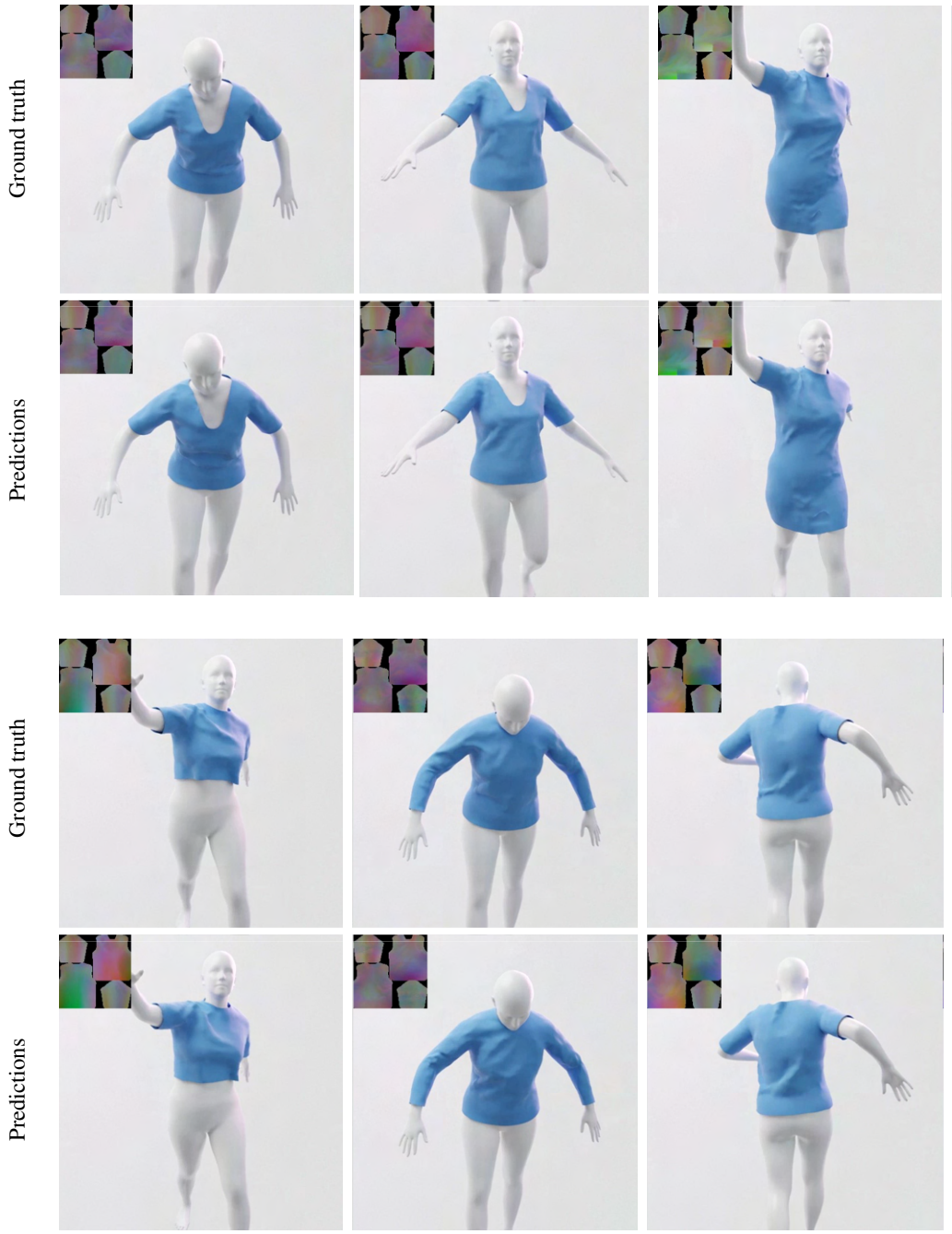

Figure 5.8: Ground truth vs our model on test sequences. Our predictions closely match the folds and wrinkles obtained with physics-based methods.

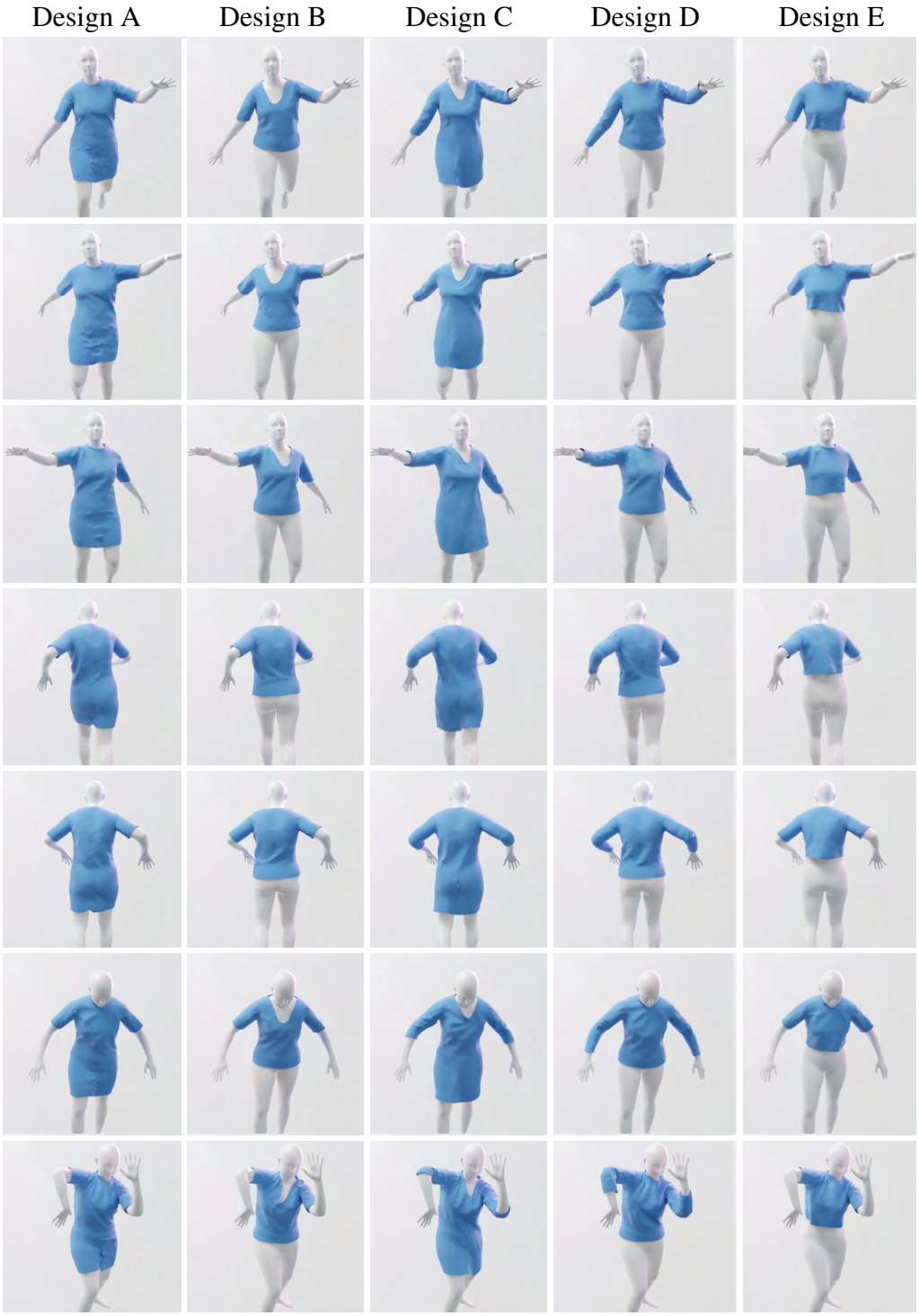

Figure 5.9: Qualitative results of five test garment designs (columns A-E) deformed using our diffusion-based model driven by a test motion from AMASS dataset. Notice how each sample exhibits unique garment folds and wrinkles that match the driving pose.

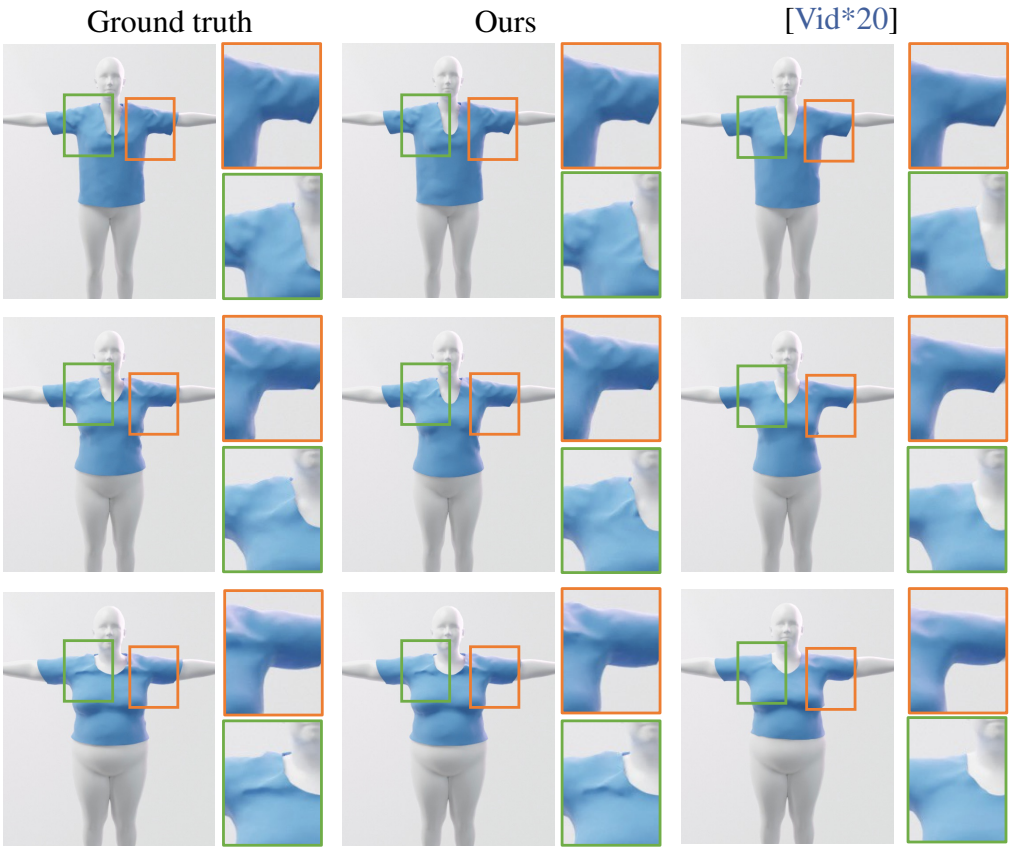

Figure 5.10: Qualitative comparison with [Vid*20] for a garment design unseen at train time. Our diffusion model predicts 3D deformations that closely match the ground truth, while the state-of-the-art method [Vid*20] produces over-smooth deformations.

used for parametric 3D garments. This approach manages to circumvent some of the limitations of the graph-based method presented in Chapter 4. For instance, the deformations modeled by DiffusedWrinkles are richer and more detailed, and the diffusion network is trained faster. To enable the synthesis of 3D animated results, we take inspiration from the cascade architectures for high-resolution diffusion models [Ho*22a] and propose a diffusion-based model conditioned on the current state of the garment that yields temporally-coherent 3D deformations. Our results show compelling 3D animations generated with a single model, capable of representing deformations as a function of body shape, pose, and designs.

Despite the step forward of our method in the field of data-driven garments, we suffer from a number of limitations. Body-garment collisions are a common issue in most of existing methods, and we also suffer from it. Similar to [SOC19; PLP20]

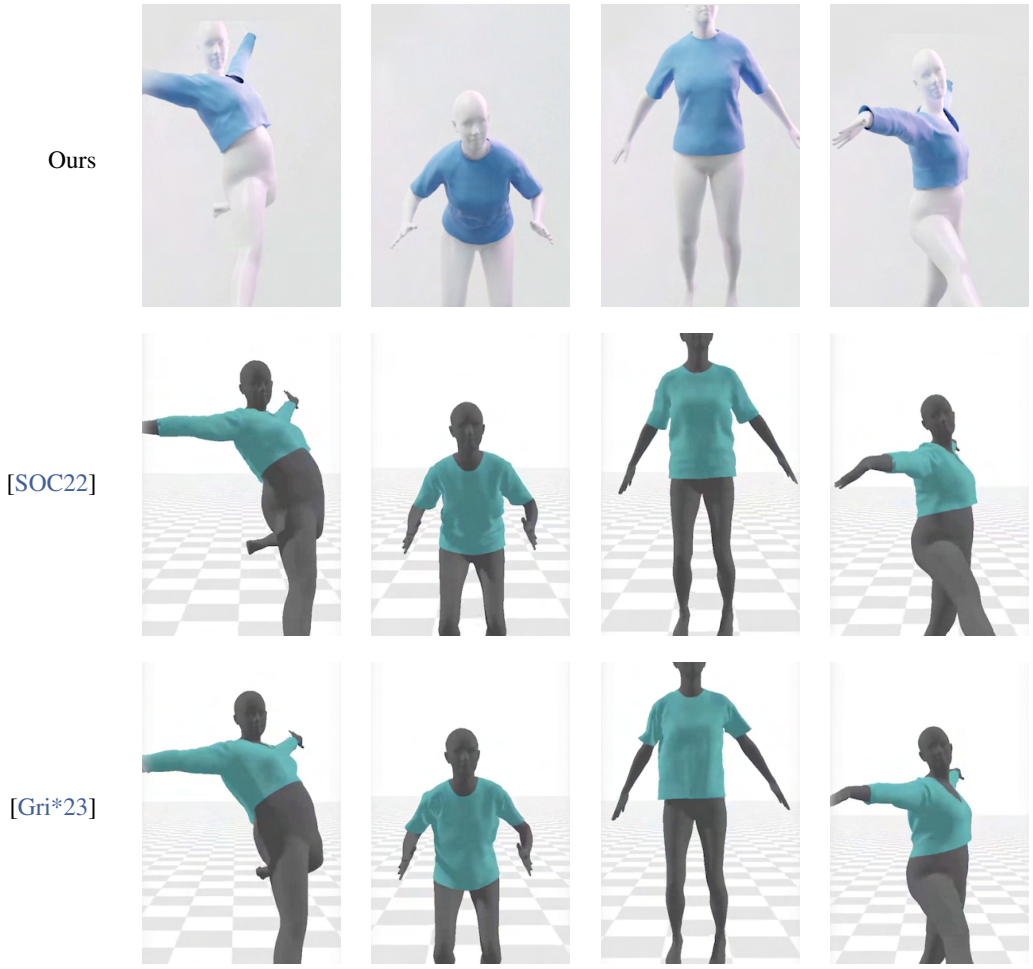

Figure 5.11: We emulate some sequences to qualitatively compare our method with current state-of-the-art methods SNUG [SOC22] and HOOD [Gri*23]

and follow-up works, at inference time we check for garment penetrations and push the problematic vertices outside. Our method is also limited by the expressivity of the underlying diffusion model. If the training samples increase significantly, the generalization capabilities can be reduced leading to over-smooth results. This could be addressed with the use of Latent Diffusion Model, which enables the use of more expressive subspaces for images. Finally, dynamic effects are currently not modeled. Our approach takes as input the current state of the garment which yields a temporally-coherent output, but a longer temporal window and a more complex architecture are needed to model time-dependent effects.

6 Conclusions

The main objective of this thesis is the development of data-driven frameworks for accurate 3D draping that are agnostic to surface discretization and garment design. To this end, we address the limitations of previous data-driven methods, that struggle with new garments and mesh topologies. The objective of our first approach (Chapter 4) was to create a fully convolutional network that estimates cloth deformation in the 3D space and can handle arbitrary mesh topologies and target body shapes. In the second approach (Chapter 5), we tried to take advantage of image generation models to generate cloth deformation.

In the course of my doctoral research we have made the following contributions:

- **Novel framework.** We proposed a geometric deep learning framework for parametric garments. Our model is based on graph convolutions and shows the ability to handle arbitrary mesh topologies.

- **Three-stage approach.** Our framework separates three sources of deformation into three different networks:

  - Parametric 3D Drape. An initial dense network estimates the rough shape of the garment draped on a mean human shape, given design parameters. The triangulation of this low-resolution mesh is then optimized, to avoid corrupt triangles and to increase the level of detail. The resulting mesh is the template of the garment that we want to deform.

  - Smooth 3D Body Drape. A fully convolutional regressor estimates the per-vertex offsets corresponding to the smooth deformations caused by the target body shape. The result of this step is a smooth mesh with the overall shape-dependent deformations.

  - Fine 3d Body Drape. The last fully convolutional regressor further refines the mesh, by regressing, for each vertex, the offsets that correspond to the material-dependent deformations. In practice, this network returns the wrinkles and details. As a result, we get the fully deformed mesh.

- **Self-supervised collision loss.** To avoid penetrations between the garment and the body, we propose a self-supervised strategy. Once the network is trained on the training dataset, we use *Parametric 3D Drape* to generate new designs (that aren't in our dataset and, therefore, don't have a ground truth). These new garments are used to refine the network *Fine 3D Drape* with a collision loss that penalizes inter-penetrations.

- **Generalization.** Our method leverages data-driven models to generalize to various garment types and body-shapes.

- **Novel 2D approach.** We represent 3D garment deformations as a 2D texture encoding 3D offsets with respect to a garment template in a consistent layout. The use of this representation enables the application of image-specific architectures to generate new displacement maps (and, equivalently, 3D deformations).

- **DiffWrinkles.** We trained a conditional diffusion model on a dataset of garments, taking profit of the generative capability of such models to synthesize new plausible deformations for a given pose, shape, and design.

- **Temporal coherence.** Proposed a solution to condition the model on an existing garment state, enabling the generation of temporally coherent sequences.

Our contributions show the potential of DL models to create versatile frameworks for cloth modeling. We believe that these kinds of models can be rich enough to reproduce the movement of cloth while significantly reducing the computational cost, compared to traditional methods. They open avenues of future work to further explore the application of data-driven models for applications like virtual try-on, or design tools.

In the process of designing, developing, implementing, and validating the contributions shown in this thesis, we have learned a few things. First, graph convolutions

can be applied to estimate deformations of arbitrary garments. They are great at predicting the rough deformation of the cloth and they have great generalization capabilities to new garments and discretizations. However, they are difficult to train and they struggle with the generation of fine wrinkles. Alternatively, our second contribution shows that DDPMs can generate rich displacement maps, that encode fine and detailed wrinkles, but this approach has some downfalls too. On one hand, right now our method requires garments with consistent UV-layout. This condition limits considerably the range of garments that can be deformed by the method, so a different, more general, representation of the displacement maps could really enhance the potential applicability of the model (we can draw inspiration from Su *et al.* [Su*23]). Besides, our displacement maps are small (128x128, limiting the quantity and quality of deformations that we can model), and the generation of images is quite slow (to generate each image we require 100 evaluations of the network). Luckily for us, DDPMs are evolving at an amazing pace, and using a new, more efficient, architecture based on diffusion can potentially accelerate this application (recent methods generate high-quality images with just one to four evaluations of the network [Sau*24]). Thus, denoising diffusion models have the potential to create deformations that are richer and more expressive than the ones estimated by graph-based networks.

Despite their promising results, our contributions face multiple limitations:

- Our models do not consider material-dependent input parameters. A richer material model would be an interesting consideration since material properties define fundamental characteristics of garments, such as type of wrinkles, overall folding, draping, etc. Some works have shown that such properties can be captured from data [Rod*23]. As they are, they need to be retrained to handle different configurations, and any material change would require the generation of a new dataset and the training of a new network. Instead, material parameters could be treated as inputs to condition the models.

- None of the models accounts for external forces or multi-layer garments, as cloth-to-cloth interactions are not even considered. Some works focus on solving this issue using neural fields [San*22]. Extending our frameworks to handle layers would be a very interesting line for future work.

- Commercial garment design requires more than three parameters. GarmentCode [KS23] leverages parametric garments to model a really wide variety of clothes. Further research needs to be conducted in order to study the scalability of the proposed frameworks to such high-dimensional parametric cloth spaces.

- The methods, being data-driven, might not fully capture the physical accuracy of real-world garments, especially under extreme conditions. Our approaches still have collisions.

- Supervised data-driven methods are only as good as the data we use for training. Improving our datasets and increasing the space of body shapes and designs would lead to better results.

There are several lines for potential future research stemming from the identified limitations. One particularly promising direction is the exploration of self-supervised methods. These methods have demonstrated significant potential [BME21; SOC22; Gri*23], especially considering that the creation of high-quality datasets remains a challenge for data-driven approaches. Integrating data-driven techniques with physics-based constraints could significantly reduce the time required for data creation and capture. Furthermore, due to the differentiable nature of neural networks, these methods could be capable of adjusting parameters based on real-world captured garment data. This combination of self-supervised learning and adjustment to captured data has the potential to advance virtual try-on and design applications, enabling them to operate in real-time.

Bibliography

[AI23] Stability AI. *Stable Diffusion XL 1.0 Model*. Accessed on June 20, 2024. 2023. URL: https://stablediffusionxl.com/ (cit. on pp. 16, 20).

[All*19] 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 (cit. on pp. 15, 17, 97).

[ACP02] Brett Allen, Brian Curless, and Zoran Popović. "Articulated body deformation from range scan data". In: *ACM Transactions on Graphics (TOG)* 21.3 (2002), pp. 612–619 (cit. on p. 23).

[Ang*05] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, et al. "SCAPE: Shape Completion and Animation for PEople". In: *Proc. of ACM SIGGRAPH*. 2005, pp. 408–416 (cit. on pp. 17, 23, 24).

[AB17] Martin Arjovsky and Leon Bottou. "Towards Principled Methods for Training Generative Adversarial Networks". In: *International Conference on Learning Representations*. 2017. URL: https://openreview.net/forum?id=Hk4_qw5xe (cit. on p. 51).

[BW98] David Baraff and Andrew Witkin. "Large Steps in Cloth Simulation". In: *Proc. of Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*. 1998, pp. 43–54 (cit. on pp. 11, 95).

[Bar*16] Aric Bartle, Alla Sheffer, Vladimir G Kim, et al. "Physics-driven pattern adjustment for direct 3D garment editing." In: *ACM Trans. Graph.* 35.4 (2016), pp. 50–1 (cit. on p. 12).

[Ben*14] Jan Bender, Matthias Müller, Miguel A Otaduy, Matthias Teschner, and Miles Macklin. "A Survey on Position–Based Simulation Methods in Computer Graphics". In: *Computer Graphics Forum* 33.6 (2014), pp. 228–251 (cit. on p. 11).

[Ber*13] Floraine Berthouzoz, Akash Garg, Danny M Kaufman, Eitan Grinspun, and Maneesh Agrawala. “Parsing Sewing Patterns into 3D Garments”. In: *ACM Transactions on Graphics (Proc. SIGGRAPH)* 32.4 (2013), pp. 1–12 (cit. on p. 12).

[BME20] Hugo Bertiche, Meysam Madadi, and Sergio Escalera. “CLOTH3D: Clothed 3D Humans”. In: *Proc. of European Conference on Computer Vision (ECCV)*. 2020 (cit. on p. 18).

[BME21] Hugo Bertiche, Meysam Madadi, and Sergio Escalera. “PBNS: Physically Based Neural Simulation for Unsupervised Garment Pose Space Deformation”. In: *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)* 40.6 (2021) (cit. on pp. 19, 20, 74, 97).

[Ber*21] 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 (cit. on p. 17).

[Bha*03] Kiran S. Bhat, Christopher D. Twigg, Jessica K. Hodgins, et al. “Estimating Cloth Simulation Parameters from Video”. In: *Proc. of ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. 2003, pp. 37–51 (cit. on p. 12).

[Bha*19] 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)*. 2019, pp. 5420–5430 (cit. on p. 15).

[Bla*23] Andreas Blattmann, Robin Rombach, Huan Ling, et al. “Align your latents: High-resolution video synthesis with latent diffusion models”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 22563–22575 (cit. on pp. 20, 53).

[Bog*15] Federica Bogo, Michael J Black, Matthew Loper, and Javier Romero. “Detailed Full-Body Reconstructions of Moving People From Monocular RGB-D Sequences”. In: *Proc. of Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 2300–2308 (cit. on p. 14).

[Bou*14] Sofien Bouaziz, Sebastian Martin, Tiantian Liu, Ladislav Kavan, and Mark Pauly. “ Projective Dynamics: Fusing Constraint Projections for Fast Simulation”. In: *ACM Transactions on Graphics (Proc. SIGGRAPH)* 33.4 (2014), pp. 1–11 (cit. on pp. 12, 95).

[Bou*13] Katherine L Bouman, Bei Xiao, Peter Battaglia, and William T Freeman. “Estimating the Material Properties of Fabric from Video”. In: *Proc. of Computer Vision and Pattern Recognition (CVPR)*. 2013, pp. 1984–1991 (cit. on pp. 12, 13).

[Bra*08] Derek Bradley, Tiberiu Popa, Alla Sheffer, Wolfgang Heidrich, and Tamy Boubekeur. “Markerless Garment Capture”. In: *ACM Transactions on Graphics (Proc. SIGGRAPH)* 27.3 (2008), p. 99 (cit. on pp. 14, 96).

[BFA02] Robert Bridson, Ronald Fedkiw, and John Anderson. “Robust Treatment of Collisions, Contact and Friction for Cloth Animation”. In: *ACM Trans. Graph.* 21.3 (2002), pp. 594–603 (cit. on pp. 11, 95).

[BMF05] Robert Bridson, Sebastian Marino, and Ronald Fedkiw. “Simulation of clothing with folds and wrinkles”. In: *ACM SIGGRAPH 2005 Courses*. 2005, 3–es (cit. on pp. 11, 95).

[CCC22] Andrés Casado-Elvira, Marc Comino Trinidad, and Dan Casas. “PERGAMO: Personalized 3D Garments from Monocular video”. In: *Computer Graphics Forum (Proc. of SCA), 2022* (2022) (cit. on pp. 15, 20, 96).

[Cas*14] Dan Casas, Marco Volino, John Collomosse, and Adrian Hilton. “4D Video Textures for Interactive Character Appearance”. In: *Computer Graphics Forum (Proc. Eurographics)* 33.2 (2014), pp. 371–380 (cit. on p. 14).

[Che*18] Zhi-Quan Cheng, Yin Chen, Ralph R Martin, Tong Wu, and Zhan Song. “Parametric modeling of 3D human body shape – A survey”. In: *Computers & Graphics* 71 (2018), pp. 88–100 (cit. on p. 17).

[CK05] Kwang-Jin Choi and Hyeong-Seok Ko. “Stable but responsive cloth”. In: *ACM SIGGRAPH 2005 Courses*. 2005, 1–es (cit. on p. 11).

[Cir*14] Gabriel Cirio, Jorge Lopez-Moreno, David Miraut, and Miguel A Otaduy. “Yarn-Level Simulation of Woven Cloth”. In: *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)* 33.6 (2014), pp. 1–11 (cit. on p. 11).

[CLO15] Gabriel Cirio, Jorge Lopez-Moreno, and Miguel A Otaduy. “Efficient simulation of knitted cloth using persistent contacts”. In: *Proc. of ACM SIGGRAPH / Eurographics Symposium on Computer Animation (SCA)*. 2015, pp. 55–61 (cit. on p. 11).

[Cor*21] 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 (cit. on p. 18).

[Dan*17] R Daněřek, Endri Dibra, Cengiz Öztireli, Remo Ziegler, and Markus Gross. "DeepGarment: 3D Garment Shape Estimation from a Single Image". In: *Computer Graphics Forum (Proc. Eurographics)* 36.2 (2017), pp. 269–280 (cit. on pp. 15, 20, 96).

[De *10] Edilson De Aguiar, Leonid Sigal, Adrien Treuille, and Jessica K Hodgins. "Stable Spaces for Real-time Clothing". In: *ACM Transactions on Graphics (Proc. SIGGRAPH)* 29.4 (2010) (cit. on pp. 12, 39, 95).

[De *08] Edilson De Aguiar, Carsten Stoll, Christian Theobalt, et al. "Performance capture from sparse multi-view video". In: *Proc. ACM SIGGRAPH*. 2008 (cit. on p. 14).

[De *23] Luca De Luigi, Ren Li, Benoît Guillard, Mathieu Salzmann, and Pascal Fua. "Drapenet: Garment generation and self-supervised draping". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 1451–1460 (cit. on p. 18).

[Dee17] DeepL. *DeepL Translate*. Accessed on June 20, 2024. 2017. URL: https://www.deepl.com/es/translator/ (cit. on p. 16).

[DBV16] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. "Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering". In: *Proc. of International Conference on Neural Information Processing Systems (NIPS)*. 2016, pp. 3844–3852 (cit. on pp. 28, 29, 37).

[DN21] Prafulla Dhariwal and Alexander Nichol. "Diffusion Models Beat GANs on Image Synthesis". In: *Advances in Neural Information Processing Systems (NeurIPS)* 34 (2021), pp. 8780–8794 (cit. on pp. 52, 53).

[EKS03] Olaf Etzmuß, Michael Keckeisen, and Wolfgang Straßer. "A fast finite element solution for cloth modelling". In: *11th Pacific Conference onComputer Graphics and Applications, 2003. Proceedings.* IEEE. 2003, pp. 244–251 (cit. on pp. 11, 95).

[FCS15] 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)*. 2015, pp. 57–64 (cit. on p. 17).

[FTP16] Marco Fratarcangeli, Valentina Tibaldo, and Fabio Pellacini. "Vivace: a Practical Gauss-Seidel Method for Stable Soft Body Dynamics". In: *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)* 35.6 (2016), pp. 1–9 (cit. on p. 12).

[Ful*19] Lawson Fulton, Vismay Modi, David Duvenaud, David IW Levin, and Alec Jacobson. “Latent-space Dynamics for Reduced Deformable Simulation”. In: *Computer Graphics Forum (Proc. Eurographics)* 38.2 (2019), pp. 379–391 (cit. on p. 12).

[GJ19] Hongyang Gao and Shuiwang Ji. “Graph u-nets”. In: *international conference on machine learning*. PMLR. 2019, pp. 2083–2092 (cit. on p. 45).

[GH97] Michael Garland and Paul S Heckbert. “Surface simplification using quadric error metrics”. In: *Proc. of the Annual conference on Computer Graphics and interactive techniques*. 1997, pp. 209–216 (cit. on pp. 31, 37).

[Gil*15] Russell Gillette, Craig Peters, Nicholas Vining, Essex Edwards, and Alla Sheffer. “Real-Time Dynamic Wrinkling of Coarse Animated Cloth”. In: *Proc. of ACM SIGGRAPH / Eurographics Symposium on Computer Animation (SCA)*. 2015 (cit. on p. 12).

[Gil*17] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. “Neural message passing for quantum chemistry”. In: *International conference on machine learning*. PMLR. 2017, pp. 1263–1272 (cit. on p. 20).

[Gri*23] Artur Grigorev, Bernhard Thomaszewski, Michael J Black, and Otmar Hilliges. “HOOD: Hierarchical Graphs for Generalized Modelling of Clothing Dynamics”. In: *Proc. of Computer Vision and Pattern Recognition (CVPR)*. 2023 (cit. on pp. 20, 65, 69, 74).

[Gri*03] Eitan Grinspun, Anil N. Hirani, Mathieu Desbrun, and Peter Schröder. “Discrete Shells”. In: *Proc. of ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA)*. 2003, pp. 62–67 (cit. on p. 11).

[Gua*12] Peng Guan, Loretta Reiss, David A Hirshberg, Alexander Weiss, and Michael J Black. “DRAPE: DRessing Any PErson”. In: *ACM Transactions on Graphics (Proc. SIGGRAPH)* 31.4 (2012) (cit. on p. 17).

[Gun*20] E. Gundogdu, V. Constantin, S. Parashar, et al. “GarNet++: Improving Fast and Accurate Static 3D Cloth Draping by Curvature Loss”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2020) (cit. on pp. 18, 19).

[Gun*19] Erhan Gundogdu, Victor Constantin, Amrollah Seifoddini, et al. “GarNet: A two-stream network for fast and accurate 3D cloth draping”. In: *Proc. of IEEE International Conference on Computer Vision (ICCV)*. 2019 (cit. on pp. 18, 19, 39, 97).

[HYL17] Will Hamilton, Zhitao Ying, and Jure Leskovec. “Inductive representation learning on large graphs”. In: *Advances in neural information processing systems* 30 (2017) (cit. on p. 20).

[Han*18] Xintong Han, Zuxuan Wu, Zhe Wu, Ruichi Yu, and Larry S Davis. “Viton: An Image-Based Virtual Try-On Network”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 7543–7552 (cit. on p. 18).

[HSR13] Stefan Hauswiesner, Matthias Straka, and Gerhard Reitmayr. “Virtual Try-On through Image-Based Rendering”. In: *IEEE transactions on visualization and computer graphics* 19.9 (2013), pp. 1552–1565 (cit. on p. 18).

[He*16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “Deep Residual Learning for Image Recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778 (cit. on p. 16).

[HJA20] Jonathan Ho, Ajay Jain, and Pieter Abbeel. “Denoising Diffusion Probabilistic Models”. In: *Advances in Neural Information Processing Systems (NeurIPS)* 33 (2020), pp. 6840–6851 (cit. on pp. 20, 52, 53, 55, 58, 59).

[Ho*22a] Jonathan Ho, Chitwan Saharia, William Chan, et al. “Cascaded Diffusion Models for High Fidelity Image Generation”. In: *Journal of Machine Learning Research* 23.47 (2022), pp. 1–33. URL: http://jmlr.org/papers/v23/21-0635.html (cit. on pp. 20, 60, 68).

[Ho*22b] Jonathan Ho, Tim Salimans, Alexey Gritsenko, et al. “Video diffusion models”. In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 8633–8646 (cit. on p. 53).

[Hol*19] Daniel Holden, Bang Chi Duong, Sayantan Datta, and Derek Nowrouzezahrai. “Subspace Neural Physics: Fast Data-Driven Interactive Simulation”. In: *Proc. of ACM SIGGRAPH / Eurographics Symposium on Computer Animation (SCA)*. 2019 (cit. on p. 12).

[Hu*20] Yuanming Hu, Luke Anderson, Tzu-Mao Li, et al. “DiffTaichi: Differentiable Programming for Physical Simulation”. In: *International Conference on Learning Representations*. 2020 (cit. on p. 13).

[Hua*20] 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*. 2020, pp. 3093–3102 (cit. on p. 17).

[JSS18] Hanbyul Joo, Tomas Simon, and Yaser Sheikh. “Total Capture: A 3D Deformation Model for Tracking Faces, Hands, and Bodies”. In: *Proc. of Computer Vision and Pattern Recognition (CVPR)*. 2018 (cit. on p. 56).

[KJM10] Jonathan M Kaldor, Doug L James, and Steve Marschner. “Efficient yarn-based cloth with adaptive contact linearization”. In: *Proc of ACM SIGGRAPH*. 2010 (cit. on p. 11).

[KJM08] Jonathan M. Kaldor, Doug L. James, and Steve Marschner. “Simulating Knitted Cloth at the Yarn Level”. In: *ACM Trans. Graph.* 27.3 (2008), pp. 1–9 (cit. on p. 11).

[Kav*11] Ladislav Kavan, Dan Gerszewski, Adam W. Bargteil, and Peter-Pike Sloan. “Physics-Inspired Upsampling for Cloth Simulation in Games”. In: *Proc. of ACM SIGGRAPH*. 2011 (cit. on p. 12).

[KCM12] Tae-Yong Kim, Nuttapong Chentanez, and Matthias Müller-Fischer. “Long range attachments – A method to simulate inextensible clothing in computer games”. In: *Proc. of ACM SIGGRAPH / Eurographics Symposium on Computer Animation (SCA)*. 2012, pp. 305–310 (cit. on p. 11).

[KW14] Diederik P. Kingma and Max Welling. “Auto-Encoding Variational Bayes”. In: *International Conference on Learning Representations (ICLR)*. 2014 (cit. on p. 55).

[KL22] Maria Korosteleva and Sung-Hee Lee. “NeuralTailor: Reconstructing Sewing Pattern Structures from 3D Point Clouds of Garments”. In: *ACM Trans. Graph.* 41.4 (2022) (cit. on p. 18).

[KS23] Maria Korosteleva and Olga Sorkine-Hornung. “GarmentCode: Programming Parametric Sewing Patterns”. In: *ACM Transaction on Graphics* 42.6 (2023). SIGGRAPH ASIA 2023 issue (cit. on p. 74).

[Lad*15] L’ubor Ladickỳ, SoHyeon Jeong, Barbara Solenthaler, Marc Pollefeys, and Markus Gross. “Data-driven fluid simulations using regression forests”. In: *ACM Transactions on Graphics (TOG)* 34.6 (2015), pp. 1–9 (cit. on p. 17).

[LCT18] Zorah Lahner, Daniel Cremers, and Tony Tung. “Deepwrinkles: Accurate and realistic clothing modeling”. In: *Proc. of European Conference on Computer Vision (ECCV)*. 2018 (cit. on pp. 3, 15, 19, 20, 96, 97).

[Lar*16] Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. “Autoencoding beyond Pixels Using a Learned Similarity Metric”. In: *Proc. of International Conference on International Conference on Machine Learning (ICML)*. 2016, pp. 1558–1566 (cit. on p. 18).

[Lee*10] Yongjoon Lee, Sung-eui Yoon, Seungwoo Oh, Duksu Kim, and Sunghee Choi. “Multi-Resolution Cloth Simulation”. In: 29.7 (2010), pp. 2225–2232 (cit. on p. 12).

[Li*17a] Changjian Li, Hao Pan, Yang Liu, et al. “Bendsketch: Modeling freeform surfaces through 2d sketching”. In: *ACM Transactions on Graphics (TOG)* 36.4 (2017), pp. 1–14 (cit. on p. 12).

[Li*17b] Tianye Li, Timo Bolkart, Michael J Black, Hao Li, and Javier Romero. "Learning a model of facial shape and expression from 4D scans." In: *ACM Trans. Graph.* 36.6 (2017), pp. 194–1 (cit. on p. 23).

[Li*17c] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting". In: *arXiv preprint arXiv:1707.01926* (2017) (cit. on p. 20).

[Li*23] Yifei Li, Hsiao-yu Chen, Egor Larionov, et al. "DiffAvatar: Simulation-Ready Garment Optimization with Differentiable Simulation". In: *arXiv preprint arXiv:2311.12194* (2023) (cit. on p. 13).

[LLK19] Junbang Liang, Ming Lin, and Vladlen Koltun. "Differentiable Cloth Simulation for Inverse Problems". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2019, pp. 771–780 (cit. on p. 13).

[Liu*23] Lijuan Liu, Xiangyu Xu, Zhijie Lin, Jiabin Liang, and Shuicheng Yan. "Towards garment sewing pattern reconstruction from a single image". In: *ACM Transactions on Graphics (TOG)* 42.6 (2023), pp. 1–15 (cit. on p. 18).

[Lop*15] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. "SMPL: A skinned multi-person linear model". In: *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)* 34.6 (2015), pp. 1–16 (cit. on pp. 17, 23, 24, 36, 38, 40, 56, 61, 97).

[Ly*20] Mickaël Ly, Jean Jouve, Laurence Boissieux, and Florence Bertails-Descoubes. "Projective Dynamics with Dry Frictional Contact". In: *ACM Transactions on Graphics (Proc. SIGGRAPH)* 39.4 (2020) (cit. on p. 12).

[Ma*21a] Qianli Ma, Shunsuke Saito, Jinlong Yang, Siyu Tang, and Michael J. Black. "SCALE: Modeling Clothed Humans with a Surface Codec of Articulated Local Elements". In: *Proc. of Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 16082–16093 (cit. on pp. 3, 18, 97).

[Ma*20] Qianli Ma, Jinlong Yang, Anurag Ranjan, et al. "Learning to Dress 3D People in Generative Clothing". In: *Proc. of Computer Vision and Pattern Recognition (CVPR)*. 2020 (cit. on pp. 18, 19, 97).

[Ma*21b] 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, pp. 10974–10984 (cit. on p. 18).

[Mah*19] 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)*. Oct. 2019, pp. 5442–5451 (cit. on pp. 61, 65).

[Man*17] P-L Manteaux, Christopher Wojtan, Rahul Narain, et al. “Adaptive physically based models in computer graphics”. In: *Computer Graphics Forum*. Vol. 36. 6. Wiley Online Library. 2017, pp. 312–337 (cit. on pp. 12, 95).

[Mon*17] Federico Monti, Davide Boscaini, Jonathan Masci, et al. “Geometric deep learning on graphs and manifolds using mixture model cnns”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 5115–5124 (cit. on p. 20).

[MC10] Matthias Müller and Nuttapong Chentanez. “Wrinkle Meshes”. In: *Proc. of ACM SIGGRAPH / Eurographics Symposium on Computer Animation (SCA)*. 2010, pp. 85–92 (cit. on p. 12).

[Mül*14] Matthias Müller, Nuttapong Chentanez, Tae-Yong Kim, and Miles Macklin. “Strain Based Dynamics”. In: *Proc. of ACM SIGGRAPH / Eurographics Symposium on Computer Animation (SCA)*. 2014, pp. 149–157 (cit. on p. 11).

[Mül*02] Matthias Müller, Julie Dorsey, Leonard McMillan, Robert Jagnow, and Barbara Cutler. “Stable real-time deformations”. In: *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation*. 2002, pp. 49–54 (cit. on pp. 11, 95).

[Mül*07] Matthias Müller, Bruno Heidelberger, Marcus Hennix, and John Ratcliff. “Position Based Dynamics”. In: *Journal of Visual Communication and Image Representation* 18.2 (2007) (cit. on pp. 11, 95).

[NSO12] Rahul Narain, Armin Samii, and James F O’brien. “Adaptive Anisotropic Remeshing for Cloth Simulation”. In: *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)* 31.6 (2012), pp. 1–10 (cit. on pp. 12, 28, 35, 38, 44, 61, 95).

[Nea*06] Andrew Nealen, Matthias Müller, Richard Keiser, Eddy Boxerman, and Mark Carlson. “Physically Based Deformable Models in Computer Graphics”. In: *Computer Graphics Forum* 25.4 (2006), pp. 809–836 (cit. on pp. 10, 14).

[NH14] Alexandros Neophytou and Adrian Hilton. “A layered model of human body and garment deformation”. In: *Proc. of International Conference on 3D Vision (3DV)*. 2014, pp. 171–178 (cit. on p. 96).

[ND21] Alexander Quinn Nichol and Prafulla Dhariwal. “Improved denoising diffusion probabilistic models”. In: *International conference on machine learning*. PMLR. 2021, pp. 8162–8171 (cit. on pp. 20, 53).

[Ope22a] OpenAI. *ChatGPT*. Accessed on June 20, 2024. 2022. URL: https://openai.com/chatgpt/ (cit. on p. 16).

[Ope22b] OpenAI. *Dall-E 2*. Accessed on June 20, 2024. 2022. URL: https://openai.com/index/dall-e-2 (cit. on p. 20).

[OBB20] Ahmed A A Osman, Timo Bolkart, and Michael J. Black. “STAR: A Sparse Trained Articulated Human Body Regressor”. In: *European Conference on Computer Vision (ECCV)*. 2020, pp. 598–613. URL: https://star.is.tue.mpg.de (cit. on p. 23).

[PLP20] Chaitanya Patel, Zhouyingcheng Liao, and Gerard Pons-Moll. “Tailornet: Predicting clothing in 3d as a function of human pose, shape and garment style”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 7365–7375 (cit. on pp. 18–20, 68, 97).

[Pav*19] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, et al. “Expressive Body Capture: 3D Hands, Face, and Body from a Single Image”. In: *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 10975–10985 (cit. on p. 23).

[PX23] William Peebles and Saining Xie. “Scalable diffusion models with transformers”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 4195–4205 (cit. on p. 53).

[Pen*22] Cheng Peng, Pengfei Guo, S Kevin Zhou, Vishal M Patel, and Rama Chellappa. “Towards performant and reliable undersampled MR reconstruction via diffusion model sampling”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2022, pp. 623–633 (cit. on p. 53).

[Per*23] Pablo Pernias, Dominic Rampas, Mats Leon Richter, Christopher Pal, and Marc Aubreville. “Würstchen: An Efficient Architecture for Large-Scale Text-to-Image Diffusion Models”. In: *The Twelfth International Conference on Learning Representations*. 2023 (cit. on p. 53).

[Pfa*20] Tobias Pfaff, Meire Fortunato, Alvaro Sanchez-Gonzalez, and Peter W Battaglia. “Learning mesh-based simulation with graph networks”. In: *arXiv preprint arXiv:2010.03409* (2020) (cit. on p. 20).

[Pon*17] Gerard Pons-Moll, Sergi Pujades, Sonny Hu, and. “ClothCap: Seamless 4D clothing capture and retargeting”. In: *ACM Transactions on Graphics (Proc. SIGGRAPH)* 36.4 (2017) (cit. on pp. 14, 96).

[Pop*09] Tiberiu Popa, Quan Zhou, Derek Bradley, et al. “Wrinkling Captured Garments Using Space-Time Data-Driven Deformation”. In: *Computer Graphics Forum (Proc. Eurographics)* 28.2 (2009), pp. 427–435 (cit. on p. 14).

[Pra*16] Fabián Prada, Misha Kazhdan, Ming Chuang, Alvaro Collet, and Hugues Hoppe. “Motion Graphs for Unstructured Textured Meshes”. In: *ACM Transactions on Graphics (Proc. SIGGRAPH)* 35.4 (2016) (cit. on p. 14).

[Pro*95] Xavier Provot et al. “Deformation constraints in a mass-spring model to describe rigid cloth behaviour”. In: *Graphics interface*. Canadian Information Processing Society. 1995, pp. 147–147 (cit. on pp. 10, 11, 95).

[Rad*21] Alec Radford, Jong Wook Kim, Chris Hallacy, et al. “Learning Transferable Visual Models from Natural Language Supervision”. In: *International Conference on Machine Learning (ICML)*. PMLR. 2021, pp. 8748–8763 (cit. on p. 16).

[Ran*18] Anurag Ranjan, Timo Bolkart, Soubhik Sanyal, and Michael J Black. “Generating 3D Faces Using Convolutional Mesh Autoencoders”. In: *Proc. of European Conference on Computer Vision (ECCV)*. 2018, pp. 725–741 (cit. on pp. 17, 31, 32, 37).

[Ras*20] Abdullah-Haroon Rasheed, Victor Romero, Florence Bertails-Descoubes, et al. “Learning to Measure the Static Friction Coefficient in Cloth Contact”. In: *Proc. of Computer Vision and Pattern Recognition (CVPR)*. 2020 (cit. on p. 13).

[Red*16] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. “You Only Look Once: Unified, Real-Time Object Detection”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 779–788 (cit. on p. 16).

[Rob*16] Nadia Robertini, Dan Casas, Helge Rhodin, Hans-Peter Seidel, and Christian Theobalt. “Model-Based Outdoor Performance Capture”. In: *Proc. of International Conference on 3D Vision (3DV)*. 2016, pp. 166–175 (cit. on p. 14).

[Rob*02] Kathleen M Robinette, Sherri Blackwell, Hein Daanen, et al. “Civilian American and European surface anthropometry resource (CAESAR), final report, volume I: Summary”. In: *Sytronics Inc Dayton Oh* (2002), p. 3 (cit. on p. 24).

[Rod*23] Carlos Rodriguez-Pardo, Melania Prieto-Martin, Dan Casas, and Elena Garces. “How will it drape like? capturing fabric mechanics from depth images”. In: *Computer Graphics Forum*. Vol. 42. 2. Wiley Online Library. 2023, pp. 149–160 (cit. on p. 73).

[Roh*10] Damien Rohmer, Tiberiu Popa, Marie-Paule Cani, Stefanie Hahmann, and Alla Sheffer. “Animation Wrinkling: Augmenting Coarse Cloth Simulations with Realistic-Looking Wrinkles”. In: *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)* 29.6 (2010), pp. 1–8 (cit. on p. 12).

[RTB17] Javier Romero, Dimitrios Tzionas, and Michael J. Black. “Embodied Hands: Modeling and Capturing Hands and Bodies Together”. In: *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*. 245:1–245:17 36.6 (Nov. 2017) (cit. on p. 23).

[RFB15] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *International Conference on Medical image computing and computer-assisted intervention (MICCAI)*. 2015, pp. 234–241 (cit. on pp. 28, 29, 38).

[Run*20] Tom F. H. Runia, Kirill Gavrilyuk, Cees G. M. Snoek, and Arnold W. M. Smeulders. “Cloth in the Wind: A Case Study of Physical Measurement Through Simulation”. In: *Proc. of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020 (cit. on p. 13).

[Sai*19] Shunsuke Saito, Zeng Huang, Ryota Natsume, et al. “PIFu: Pixel-Aligned Implicit Function for High-Resolution Clothed Human Digitization”. In: *Proc. of IEEE International Conference on Computer Vision (ICCV)*. 2019 (cit. on p. 17).

[Sai*20] Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. “Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 84–93 (cit. on p. 17).

[San*20a] Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, et al. “Learning to simulate complex physics with graph networks”. In: *International conference on machine learning*. PMLR. 2020, pp. 8459–8468 (cit. on p. 17).

[San*20b] Igor Santesteban, Elena Garces, Miguel A. Otaduy, and Dan Casas. “SoftSMPL: Data-driven Modeling of Nonlinear Soft-tissue Dynamics for Parametric Humans”. In: *Computer Graphics Forum (Proc. Eurographics)* 39.2 (2020) (cit. on p. 17).

[San*22] Igor Santesteban, Miguel Otaduy, Nils Thuerey, and Dan Casas. “ULNeF: untangled layered neural fields for mix-and-match virtual try-on”. In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 12110–12125 (cit. on pp. 3, 18, 73, 97, 102).

[SOC22] Igor Santesteban, Miguel A Otaduy, and Dan Casas. “SNUG: Self-supervised Neural Dynamic Garments”. In: *Proc. of Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 8140–8150 (cit. on pp. 19, 20, 57, 65, 69, 74, 97).

[SOC19] Igor Santesteban, Miguel A. Otaduy, and Dan Casas. “Learning-Based Animation of Clothing for Virtual Try-On”. In: *Computer Graphics Forum (Proc. Eurographics)* 38.2 (2019) (cit. on pp. 17–20, 25, 36, 39, 42–44, 48, 56, 57, 61, 68, 97).

[San*21] Igor Santesteban, Nils Thuerey, Miguel A Otaduy, and Dan Casas. “Self-Supervised Collision Handling via Generative 3D Garment Models for Virtual Try-On”. In: *Proc. of Computer Vision and Pattern Recognition (CVPR)*. 2021 (cit. on pp. 19, 20).

[Sau*24] Axel Sauer, Frederic Boesel, Tim Dockhorn, et al. “Fast high-resolution image synthesis with latent adversarial diffusion distillation”. In: *arXiv preprint arXiv:2403.12015* (2024) (cit. on pp. 73, 102).

[Sch*05] Volker Scholz, Timo Stich, Michael Keckeisen, Markus Wacker, and Marcus Magnor. “Garment Motion Capture Using Color-Coded Patterns”. In: *Computer Graphics Forum* 24.3 (2005), pp. 439–447 (cit. on pp. 14, 96).

[SKP15] Florian Schroff, Dmitry Kalenichenko, and James Philbin. “Facenet: A Unified Embedding for Face Recognition and Clustering”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 815–823 (cit. on p. 16).

[SLL20] Yu Shen, Junbang Liang, and Ming C Lin. “Gan-based garment generation using sewing pattern images”. In: *European Conference on Computer Vision*. Springer. 2020, pp. 225–247 (cit. on pp. 17, 19).

[SB12] 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*. ACM, 2012, pp. 1–50 (cit. on p. 12).

[SRC01] Peter-Pike J Sloan, Charles F Rose III, and Michael F Cohen. “Shape by example”. In: *Proceedings of the 2001 symposium on Interactive 3D graphics*. 2001, pp. 135–143 (cit. on p. 23).

[Soh*15] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. “Deep unsupervised learning using nonequilibrium thermodynamics”. In: *International conference on machine learning*. PMLR. 2015, pp. 2256–2265 (cit. on p. 53).

[SME20] Jiaming Song, Chenlin Meng, and Stefano Ermon. “Denoising diffusion implicit models”. In: *arXiv preprint arXiv:2010.02502* (2020) (cit. on p. 20).

[Sto*10] Carsten Stoll, Juergen Gall, Edilson De Aguiar, Sebastian Thrun, and Christian Theobalt. “Video-based reconstruction of animatable human characters”. In: *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)* 29.6 (2010) (cit. on p. 13).

[Su*23] Zhaoqi Su, Tao Yu, Yangang Wang, and Yebin Liu. “DeepCloth: Neural Garment Representation for Shape and Style Editing”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45.2 (2023), pp. 1581–1593 (cit. on pp. 19, 73).

[Tan*18a] Qingyang Tan, Lin Gao, Yu-Kun Lai, and Shihong Xia. "Variational Autoencoders for Deforming 3D Mesh Models". In: *Proc. of Computer Vision and Pattern Recognition (CVPR)*. 2018, pp. 5841–5850 (cit. on p. 17).

[Tan*16] Min Tang, Huamin Wang, Le Tang, Ruofeng Tong, and Dinesh Manocha. "CAMA: Contact-Aware Matrix Assembly with Unified Collision Handling for GPU-based Cloth Simulation". In: *Computer Graphics Forum* 35.2 (2016) (cit. on p. 12).

[Tan*18b] Min Tang, Tongtong Wang, Zhongyuan Liu, Ruofeng Tong, and Dinesh Manocha. "I-Cloth: Incremental Collision Handling for GPU-Based Interactive Cloth Simulation". In: *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)* 37.6 (2018) (cit. on p. 12).

[Ter*87] Demetri Terzopoulos, John Platt, Alan Barr, and Kurt Fleischer. "Elastically deformable models". In: *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*. 1987, pp. 205–214 (cit. on p. 10).

[Tiw*20] 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*. Springer. 2020, pp. 1–18 (cit. on pp. 18, 19, 97).

[Tiw*21] 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 (cit. on pp. 3, 18, 97).

[Um*20] Kiwon Um, Robert Brand, Yun Raymond Fei, Philipp Holl, and Nils Thuerey. "Solver-in-the-loop: Learning from differentiable physics to interact with iterative pde-solvers". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 6111–6122 (cit. on p. 13).

[Ume*11] Nobuyuki Umetani, Danny M. Kaufman, Takeo Igarashi, and Eitan Grinspun. "Sensitive Couture for Interactive Garment Modeling and Editing". In: *ACM Transactions on Graphics (Proc. SIGGRAPH)* 30.4 (2011) (cit. on p. 12).

[Vas*17] Ashish Vaswani, Noam Shazeer, Niki Parmar, et al. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017) (cit. on p. 62).

[VBV18] Nitika Verma, Edmond Boyer, and Jakob Verbeek. "Feastnet: Feature-steered graph convolutions for 3d shape analysis". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 2598–2606 (cit. on p. 20).

[Vid*20] Raquel Vidaurre, Igor Santesteban, Elena Garces, and Dan Casas. “Fully Convolutional Graph Neural Networks for Parametric Virtual Try-On”. In: *Computer Graphics Forum (Proc. SCA)* 39.8 (2020) (cit. on pp. 5, 56, 57, 68).

[Vla*08] Daniel Vlasic, Ilya Baran, Wojciech Matusik, and Jovan Popović. “Articulated Mesh Animation from Multi-View Silhouettes”. In: *Proc. of ACM SIGGRAPH*. 2008, pp. 1–9 (cit. on p. 14).

[Wan21] Huamin Wang. “GPU-based simulation of cloth wrinkles at submillimeter levels”. In: *ACM Transactions on Graphics (TOG)* 40.4 (2021), pp. 1–14 (cit. on p. 12).

[Wan*10] Huamin Wang, Florian Hecht, Ravi Ramamoorthi, and James F O’Brien. “Example-based wrinkle synthesis for clothing animation”. In: *ACM SIGGRAPH 2010 papers*. 2010, pp. 1–8 (cit. on pp. 12, 95).

[WOR11] Huamin Wang, James F O’Brien, and Ravi Ramamoorthi. “Data-Driven Elastic Models for Cloth: Modeling and Measurement”. In: *ACM Transactions on Graphics (Proc. SIGGRAPH)* 30.4 (2011), pp. 1–12 (cit. on p. 44).

[Wan*18] Tuanfeng Y Wang, Duygu Ceylan, Jovan Popović, and Niloy J Mitra. “Learning a Shared Shape Space for Multimodal Garment Design”. In: *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)* 37.6 (2018) (cit. on pp. 12, 17, 18, 36, 97).

[Wan*19] Tuanfeng Y Wang, Tianjia Shao, Kai Fu, and Niloy J Mitra. “Learning an Intrinsic Garment Space for Interactive Authoring of Garment Animation”. In: *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)* 38.6 (2019) (cit. on pp. 17, 18, 36).

[WCF07] Ryan White, Keenan Crane, and D. A. Forsyth. “Capturing and Animating Occluded Cloth”. In: *ACM Transactions on Graphics (Proc. SIGGRAPH)* 26 (3 2007) (cit. on p. 14).

[WBT19] Steffen Wiewel, Moritz Becher, and Nils Thuerey. “Latent space physics: Towards learning the temporal evolution of fluid flow”. In: *Computer graphics forum*. Vol. 38. 2. Wiley Online Library. 2019, pp. 71–82 (cit. on p. 17).

[Wol*21] Katja Wolff, Philipp Herholz, Verena Ziegler, et al. “3D Custom fit garment design with body movement”. In: *arXiv preprint arXiv:2102.05462* (2021), pp. 1–12 (cit. on p. 12).

[WWW22] Botao Wu, Zhendong Wang, and Huamin Wang. “A GPU-Based Multilevel Additive Schwarz Preconditioner for Cloth and Deformable Body Simulation”. In: *ACM Transactions on Graphics (TOG)* 41.4 (2022), pp. 1–14 (cit. on p. 12).

[Wu*16] Jiajun Wu, Joseph J Lim, Hongyi Zhang, Joshua B Tenenbaum, and William T Freeman. “Physics 101: Learning Physical Object Properties from Unlabeled Videos”. In: *The British Machine Vision Conference (BMVC)*. 2016 (cit. on p. 13).

[Xu*18] Weipeng Xu, Avishek Chatterjee, Michael Zollhöfer, et al. “MonoPerfCap: Human Performance Capture From Monocular Video”. In: *ACM Transactions on Graphics* 37.2 (2018) (cit. on p. 14).

[Xu*14] Weiwei Xu, Nobuyuki Umentani, Qianwen Chao, et al. “Sensitivity-optimized Rigging for Example-based Real-Time Clothing Synthesis”. In: *ACM Transactions on Graphics (Proc. SIGGRAPH)* 33.4 (2014) (cit. on p. 17).

[Yan*20] Han Yang, Ruimao Zhang, Xiaobao Guo, et al. “Towards Photo-Realistic Virtual Try-On by Adaptively Generating-Preserving Image Content”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 7850–7859 (cit. on p. 18).

[Yan*18a] Jinlong Yang, Jean-Sébastien Franco, Franck Hétroy-Wheeler, and Stefanie Wuhrer. “Analyzing Clothing Layer Deformation Statistics of 3D Human Motions”. In: *Proc. of European Conference on Computer Vision (ECCV)*. 2018 (cit. on pp. 15, 96).

[YLL17] Shan Yang, Junbang Liang, and Ming C. Lin. “Learning-Based Cloth Material Recovery From Video”. In: *Proc. of IEEE International Conference on Computer Vision (ICCV)*. 2017 (cit. on p. 13).

[Yan*18b] Shan Yang, Zherong Pan, Tanya Amert, et al. “Physics-Inspired Garment Recovery from a Single-View Image”. In: *ACM Transactions on Graphics* 37.5 (2018) (cit. on p. 14).

[Yin*18a] Rex Ying, Ruining He, Kaifeng Chen, et al. “Graph convolutional neural networks for web-scale recommender systems”. In: *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 2018, pp. 974–983 (cit. on p. 20).

[Yin*18b] Zhitao Ying, Jiaxuan You, Christopher Morris, et al. “Hierarchical graph representation learning with differentiable pooling”. In: *Advances in neural information processing systems* 31 (2018) (cit. on p. 45).

[Zak*21] Ilya Zakharkin, Kirill Mazur, Artur Grigorev, and Victor Lempitsky. “Point-Based Modeling of Human Clothing”. In: *Proc. of IEEE International Conference on Computer Vision (ICCV)*. 2021 (cit. on pp. 3, 18, 97).

[Zha*21a] Meng Zhang, Tuanfeng Wang, Duygu Ceylan, and Niloy J Mitra. “Deep Detail Enhancement for Any Garment”. In: *Computer Graphics Forum* 40.2 (2021), pp. 399–411 (cit. on pp. 19, 97).

[Zha*14] Qing Zhang, Bo Fu, Mao Ye, and Ruigang Yang. “Quality Dynamic Human Body Modeling Using a Single Low-cost Depth Camera”. In: *Proc. of Computer Vision and Pattern Recognition (CVPR)*. 2014 (cit. on p. 14).

[Zha*21b] Fuwei Zhao, Zhenyu Xie, Michael Kampffmeyer, et al. “M3d-vton: A monocular-to-3d virtual try-on network”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 13239–13249 (cit. on p. 17).

[Zho*13] Bin Zhou, Xiaowu Chen, Qiang Fu, Kan Guo, and Ping Tan. “Garment Modeling from a Single Image”. In: *Computer Graphics Forum* 32.7 (2013), pp. 85–91 (cit. on pp. 14, 96).

[Zhu*23] Luyang Zhu, Dawei Yang, Tyler Zhu, et al. “Tryondiffusion: A Tale of Two Unets”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 4606–4615 (cit. on pp. 18, 97).

[ZBO12] Javier S Zurdo, Juan P Brito, and Miguel A Otaduy. “Animating Wrinkles by Example on Non-Skinned Cloth”. In: *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 19.1 (2012), pp. 149–158 (cit. on p. 12).

A

Resumen

Las vestimentas han evolucionado de una necesidad básica de protección y comodidad hacia una forma de comunicación, que afecta profundamente a cómo los individuos somos percibidos por la sociedad, y en consecuencia a cómo interactuamos entre nosotros. La manera de vestir refleja múltiples aspectos de la identidad de las personas: clase social, intenciones, cultura, profesión... Consecuentemente, la fabricación, distribución y consumo de ropa impulsan el crecimiento económico y proporcionan empleo en todo el mundo, haciendo de la industria de la moda una potencia económica.

La representación digital de prendas de vestir está ganando popularidad, especialmente en los ámbitos del entretenimiento y de la moda. La moda digital permite el diseño de prendas para avatares, potenciando la expresión personal y la identidad en entornos virtuales. En películas y videojuegos, las prendas virtuales son cruciales para la creación de personajes y ambientaciones creíbles. El diseño eficaz y la simulación eficiente de prendas virtuales son clave para lograr realismo e interactividad, especialmente en experiencias de realidad virtual y aumentada.

Por otro lado, la simulación digital de ropa tiene el potencial de revolucionar la industria de la moda, al permitir a los diseñadores visualizar y crear prototipos de prendas en un espacio virtual, lo que podría reducir considerablemente el tiempo, los recursos y los residuos asociados a la creación física de prototipos. Adicionalmente, las aplicaciones de probador virtual, impulsadas por el auge de las compras en línea, ofrecen a los clientes la posibilidad de ver cómo les queda la ropa desde la comodidad de sus casas. Esta opción reduce la necesidad de comprar en tiendas físicas, así como la probabilidad de devoluciones en las compras en línea, mejorando la experiencia de compra y contribuyendo a la sostenibilidad al disminuir los residuos de prendas no vendidas y la cantidad de transportes. Este tipo de aplicaciones, sin embargo, requieren simulaciones rápidas, escalables y precisas.

La informática gráfica lleva años trabajando en la simulación de prendas virtuales. Los tradicionales métododos de simulación basada en físicas (PBS), aunque consiguen resultados realistas, suelen conllevar un alto coste computacional y requieren de expertos para su creación y ajustes. En cambio, los modelos basados en datos, que aprenden a reproducir los movimientos a partir de conjuntos de datos, ofrecen un rendimiento más rápido y el potencial de escalar mejor. Sin embargo, estos modelos presentan dificultades a la hora de generalizar a nuevas estructuras, especialmente con mallas irregulares que representan prendas tridimensionales.

Intentaremos abordar estas limitaciones mediante la ampliación de métodos de aprendizaje profundo para su aplicación en prendas tridimensionales con diversidad de diseños y triangulaciones. La investigación desarrollada parte de modelos pioneros en regresión de deformaciones y se centra en mejorar su capacidad de generalización, así como en superar los retos que surjan.

El objetivo principal de la tesis es desarrollar modelos basados en el aprendizaje automático de datos para el drapeado preciso de prendas 3D. Inicialmente, la tesis explora el desarrollo de redes neuronales totalmente convolucionales aplicadas sobre grafos, para modelar prendas 3D. Para ello, se crea un espacio de prendas paramétrico, mediante el cual se generan prendas con diversos diseños que se adaptan a diferentes cuerpos sin depender de una topología fija. Sin embargo, este método tiene limitaciones en el tiempo de entrenamiento y en la generación de detalles de alta frecuencia.

Para abordar estos problemas, se introduce un nuevo enfoque, que emplea modelos probabilísticos de difusión de eliminación de ruido (DDPM) para la síntesis de arrugas de alta calidad. Mediante la representación de deformaciones 3D como mapas de desplazamientos, un modelo generativo puede producir arrugas finas y animar prendas 3D con un modelo agnóstico a la discretización. Este enfoque predice secuencias de deformaciones de alta calidad con coherencia temporal.

A.1 Antecedentes

La digitilización de ropa presenta un reto significativo para la comunidad de la informática gráfica, debido a sus múltiples aplicaciones y a su complejidad. La simulación precisa de telas puede mejorar las experiencias en entornos virtuales, y es crucial para diversas industrias (animación, videojuegos, efectos especiales, realidad virtual, diseño de moda...). En particular, las aplicaciones de probador virtual, requieren que la simulación de ropa sea **precisa** (que su comportamiento replique con la máxima fidelidad el de la ropa real), **eficiente** (para permitir la interacción de los usuarios) y **escalable** (de manera que múltiples usuarios puedan usar la aplicación de manera simultánea). Este capítulo revisa los diferentes enfoques para el modelado de telas.

Simulaciones Basadas en Física

El enfoque tradicional para abordar este desafío son las simulaciones basadas en física(PBS), que consisten en aunar modelos matemáticos y leyes físicas para simular el movimiento de la ropa.

Uno de los métodos más empleados para la simulación física de telas son los sistemas conocidos como masa-muelle, en los que la superficie de la tela se discretiza como un sistema de partículas conectados por muelles [Pro*95; BFA02; BMF05]. Este sistema es popular debido a su simplicidad y efectividad, pero no destacan por su precisión. La otra opción más popular es la simulación mediante el método de elementos finitos (FEM) [Mül*02; EKS03], que consiste en aproximar la simulación considerando la superficie de la tela como una superficie continua, permitiendo mayor precisión y la reproducción de efectos más complejos, como la anisotropía.

Dado que el coste computacional es una limitación considerable en estos métodos, han surgido muchos intentos de acelerar las simulaciones mediante aceleración de la integración [BW98; Mül*07; Bou*14], reducción del modelo [De *10; Wan*10] o modelos adaptativos [NSO12; Man*17].

A pesar del realismo proporcionado por estos métodos, en general presentan un compromiso entre eficiencia y precisión. Además, reproducir las deformaciones de

prendas reales no es trivial, ya que los parámetros físicos de la simulación han de ser ajustados.

Reconstrucción

Entendemos por captura de prendas el proceso de obtener datos detallados de prendas reales, incluídos textura, forma y movimiento, con el fin de crear modelos 3D capaces de reproducir el comportamiento de dicha prenda. La reconstrucción precisa de la superficie y las propiedades de prendas reales es esencial para simular como ajusta esa prenda a nuevos sujetos o para entrenar modelos basados en datos, por lo que entendemos que las técnicas de captura y reconstrucción de prendas son cruciales a la hora de desarrollar aplicaciones de probadores virtuales.

Los primeros métodos empleaban múltiples cámaras y patrones codificados con colores [Sch*05], pero las investigaciones han ido evolucionando hacia la captura sin marcadores [Bra*08], y, más adelante, a reconstrucciones a partir de una sola imagen RGB [Zho*13]. Recientemente, se han desarrollado enfoques que reconstruyen el cuerpo y la ropa como capas separadas. Muchos métodos ajustan los parámetros de un modelo estadístico humano para recuperar el cuerpo y modelan la ropa como desplazamientos respecto al cuerpo [NH14; Pon*17], algunos incluso estiman parámetros del material o de talla[Yan*18a].

Las técnicas de aprendizaje profundo han ayudado a la captura de prendas mediante la estimación de desplazamientos 3D a partir de imágenes para prendas fijas [Dan*17]. Estos métodos consiguen incluso generar deformaciones para nuevas animaciones a partir de los datos obtenidos [LCT18; CCC22].

Estos métodos capturan eficazmente los detalles de la superficie y el movimiento de la prenda, pero adaptar las prendas capturadas a distintas formas del cuerpo sigue siendo un desafío.

Modelado de tejidos basado en datos

Los modelos basados en datos consisten en la utilización de técnicas estadísticas y de aprendizaje automático (ML) para aprender relaciones directamente de los datos (por ejemplo, usar muchas simulaciones para inferir deformaciones a partir de los parámetros de pose).

Esta familia de modelos ha sido fundamental en la creación de modelos estadísticos 3D del cuerpo humano, como SMPL [Lop*15] (modelo que usamos en la tesis), que permiten la representación de gran diversidad de cuerpos mediante un espacio paramétrico reducido. En el campo de las prendas de vestir, han surgido enfoques basados en datos para la reconstrucción [All*19], el diseño [Wan*18] y la animación de prendas virtuales [SOC19].

A pesar de que las prendas virtuales acostumbran a ser representadas como mallas tridimensionales, muchos modelos basados en datos usan representaciones alternativas, como nubes de puntos [Ma*21a; Zak*21], representaciones implícitas [Tiw*21; San*22], esbozos [Wan*18], imágenes [Zhu*23] o mapas UV [LCT18; Zha*21a]. El empleo de diferentes representaciones permite sacar provecho de las estudiadísimas arquitecturas de aprendizaje profundo que funcionan con datos estructurados (como imágenes o vectores de características de tamaño fijo).

Estos modelos se pueden entrenar mediante conjuntos de datos sintéticos [SOC19; Gun*19; PLP20] o capturados [LCT18; Tiw*20; Ma*20] (aprendizaje supervisado), o mediante técnicas auto-supervisadas, que consisten en el uso de métricas implícitas para supervisar la optimización [BME21; SOC22].

Los métodos desarrollados en esta tesis se encuentran en el marco de los modelos supervisados mediante datos sintéticos. El primero trabaja sobre las deformaciones de los vértices de las prendas en espacio 3D, usando redes convolucionales de grafos para evitar restricciones en la discretización de las mallas. En el segundo, representamos las deformaciones mediante mapas de desplazamientos. De este modo, podemos hacer uso de redes generativas para sintetizar nuevas y detalladas deformaciones, en forma de imágenes.

A.2 Objetivos

El objetivo principal de esta tesis es el desarrollo de modelos eficientes para el drapeado automático de prendas mediante técnicas de aprendizaje profundo (DL). A pesar del éxito de estas arquitecturas en otros campos, su aplicación a la simulación de ropa presenta ciertos desafíos. La adaptabilidad de este tipo de métodos a

dominios irregulares, así como su capacidad de generalización a nuevos escenarios son limitadas. Esta tesis aborda esta limitación mediante la creación de modelos que generalicen a nuevas prendas y que no dependan de una discretización fija.

Para ello, se proponen dos enfoques diferentes. Primero buscamos una arquitectura que extienda el potencial de las convoluciones en imágenes al irregular espacio de las mallas tridimensionales. El segundo enfoque, sin embargo, explora la generación de mapas 2D que representan deformaciones en 3D.

A.3 Metodología

Para la realización de la tesis se ha seguido la siguiente metodología:

Revisión bibliográfica

Una vez definidos los objetivos de la tesis, se procedió a realizar una revisión bibliográfica exhaustiva en el ámbito de los modelos de simulación de ropa. Dicha revisión se encuentra detallada en el Capítulo 2 y de ella nace nuestro interés por los métodos de aprendizaje para la animación de ropa, así como el descubrimiento de que todos los modelos que operaban en espacio tridimensional estaban limitados a una triangulación fija. El campo del aprendizaje profundo, así como el de su aplicación para el modelado de prendas, está en continuo crecimiento, por lo que la revisión bibliográfica ha sido un proceso iterativo que se ha ido actualizando a lo largo de todo el desarrollo de la tesis.

Diseño de un modelo convolucional para el drapeado de prendas

Tras analizar el estado del arte, observamos que los métodos que emplean técnicas de DL para la animación de ropa en 3D usan capas densas, por lo que su generalización a nuevas triangulaciones es inviable. Además, este tipo de modelos no tienen en cuenta la información espacial y tienden a sobreajustarse a los datos de entrenamiento, a costa de perder capacidades de generalización.

Para evitar estas limitaciones, proponemos un modelo basado en convoluciones de grafos. Nuestro método aprende de manera diferenciada las deformaciones

provocadas por tres causas distintas: diseño de la prenda, forma del cuerpo y material. Para ello, primero entrenamos una red que, dados unos parámetros de diseño, devuelve una aproximación de baja resolución de la prenda vestida por una persona media. La triangulación de esta malla es optimizada para que las deformaciones ofrezcan suficiente nivel de detalle. A continuación, una red convolucional con estructura de U-Net estima las deformaciones suavizadas, debidas a la forma del cuerpo y el diseño. Finalmente, otra red convolucional añade los detalles de baja frecuencia, provocados principalmente por el material de la prenda. Al contrario, que los métodos previos, este enfoque permite regresar el drapeado de toda una familia paramétrica de prendas, sin limitación en cuanto a sus discretizaciones.

En el Capítulo 4 explicamos los detalles de diseño e implementación de nuestro método, y demostramos su eficiencia y capacidad de generalización en comparación con métodos previos. Sin embargo, nos encontramos con algunas limitaciones. Principalmente, los tiempos de entrenamiento son muy largos y las redes que entrenamos mostraban dificultades a la hora de reproducir detalles de alta frecuencia.

Diseño de un modelo generativo de mapas de desplazamiento para el modelado de ropa

Tras el desarrollo de nuestra primera herramienta y a raíz de las dificultades que nos encotramos, decidimos cambiar el enfoque. Con el auge de los modelos generativos de imagen, se nos ocurrió aprovechar su potencial para generar mapas de desplazamientos.

En nuestra segunda contribución proponemos representar las deformaciones de las prendas a través de mapas de desplazamiento. A partir de los mapas obtenidos entrenamos una red de difusión, concretamente una DDPM (Denoising Diffusion Probabilistic Model) que genera nuevos mapas a partir de condiciones de diseño de la prenda y de forma y pose del cuerpo. Este enfoque permite la síntesis de varias deformaciones, dado un mismo conjunto de condiciones. Mediante un condicionamiento temporal, conseguimos deformaciones continuas para secuencias de movimiento. Los detalles de esta contribución se encuentran descritos en el Capítulo 5

A.4 Resultados

En el curso de la investigación de esta tesis, se han realizado las siguientes contribuciones:

- Hemos propuesto un novedoso marco de aprendizaje profundo geométrico para prendas parámtericas. Nuestro modelo está basado en convoluciones de grafos y ofrece la capacidad de manejar triangulaciones de malla arbitrarias.

- Nuestro enfoque separa las diferentes fuentes de deformaciones en tres etapas diferentes. Para cada una de ellas entrenamos una red.

  - *Parametric 3D Drape*. Una primera red densa genera una aproximación general de la forma de la prenda sobre un cuerpo medio dados los parámetros de diseño. Esta prenda pasará por un proceso de *retopo*, que optimizará la triangulación para evitar triangulos irregulares y para aumentar el nivel de detalle, será la plantilla para el siguiente paso.

  - *Smooth 3D Body Drape*. Una red convolucional calcula los desplazamientos por vértice originados por la forma del cuerpo, sobre la plantilla de la prenda que hemos obtenido en el paso anterior. Mediante este paso obtenemos la forma suavizada de la prenda deformada al ponerla sobre un cuerpo dado.

  - *Fine 3D Body Drape*. La última red convolucional regresa, para cada vértice, las deformaciones correspondientes al material. El resultado de este paso es una malla con más detalles y arrugas. Esta red es específica del material, así que no generaliza a nuevos materiales.

- Para evitar penetraciones entre la malla del tejido y del cuerpo, proponemos una estrategia auto-supervisada. Para ello, una vez entrenada la red, generamos automáticamente nuevos diseños que no están en nuestro conjunto de datos (mediante el regresor *Parametric 3D Drape*). Empleamos estos diseños para refinar los pesos de la red final *Fine 3D Drape* con una función de coste que penaliza las penetraciones.

- En el segundo bloque, proponemos una representación de deformaciones a través de mapas de texturas. Esta representación está alineada para todos los diseños y permite codificar las deformaciones como imágenes.

- Gracias a la representación con imágenes podemos entrenar un modelo generativo que, condicionado a diseño, pose y forma del cuerpo, sintetiza nuevos mapas, que se pueden traducir en nuevas deformaciones para triangulaciones arbitrarias. Este modelo permite la reproducción de detalles más finos, así como la generación de deformaciones plausibles y distintas para una misma configuración.

- Dada una secuencia de poses, la naturaleza generativa del modelo impide la continuidad en las deformaciones obtenidas. Para obtener secuencias con coherencia temporal, proponemos el condicionamiento de la red generativa con el estado previo de la prenda.

A.5 Conclusiones

El objetivo de esta tesis es el desarrollo de modelos de aprendizaje para el drapeado de prendas, que sean agnósticos a la discretización de la superficie. Para ello, en la primera contribución (Capítulo 4) se propone un modelo basado en convoluciones de grafos, que predice la deformación de una prenda en función de su diseño y la forma del cuerpo humano que la lleva. En la segunda contribución (Capítulo 5), se plantea el uso de modelos de síntesis de imagen para la generación de deformaciones en ropa.

El uso de modelos de aprendizaje en el dominio de las mallas 3D presenta un alto grado de complejidad, y el desarrollo de estos métodos, no ha estado falto de complicaciones. A pesar, de ello, proponemos dos representaciones diferentes de prendas para el aprendizaje de deformaciones con diversidad de diseños y discretizaciones. Ambas contribuciones son novedosas y muestran el potencial de los modelos de aprendizaje para la creación de aplicaciones eficientes para la simulación de ropa.

Durante el desarrollo de estas investigaciones hemos comprobado varias cosas. Por un lado, hemos visto que las convoluciones de grafos se pueden emplear para estimar deformaciones en mallas con topología arbitraria. Muestran buenos resultados en cuanto al aprendizaje de las deformaciones a grandes rasgos y generalizan bien a nuevas prendas con nuevas discretizaciones. Sin embargo, cuestan mucho de entrenar y encuentran dificultades a la hora de aprender a reproducir arrugas detalladas. Por otro lado, las DDPMs tienen la capacidad de generar mapas de desplazamiento detallados, que codifican deformaciones de alta frecuencia. Sin embargo, nuestro enfoque requiere que los mapas de UV estén alineados, condición que limita considerablemente el rango de prendas sobre el que se puede aplicar nuestro método. Quizás, una representación de los desplazamientos más general, podría ampliar la aplicabilidad del método. Además, los mapas que generamos son pequeños (128x128) y la generación de imágenes lenta. Por suerte, esta familia de modelos está de moda y contínuamente se publican nuevos métodos con eficiencia y calidad mejorada. Sin duda, probar arquitecturas novedosas ayudaría a reducir estas limitaciones (métodos recientes [Sau*24] generan imágenes de alta calidad con una sola evaluación del modelo). Por esto, creemos que las redes generativas (y en concreto, las DDPM) pueden tener la clave para generar deformaciones con mayor riqueza y expresividad.

A pesar del enfoque pionero y los prometedores resultados de nuestras contribuciones, encontramos múltiples limitaciones. Nuestros modelos no consideran ningún tipo de parámetro de material como entrada a la red. Las propiedades de los materiales afectan significativamente al comportamiento de la tela y estaría bien que condicionaran el resultado de la red, en lugar de aprenderse de manera implícita. Estos métodos no tienen en cuenta fuerzas externas ni interacciones entre ropa, de manera que no se podrían emplear para probadores multi-capa. El método ULNeF [San*22] se centra en solucionar estas interacciones mediante campos implícitos, y extender nuestros métodos en esta dirección sería interesante.