WordRobe: Text-Guided Generation of Textured 3D Garments

Astitva Srivastava[1], Pranav Manu[1],
Amit Raj[2], Varun Jampani[3], and Avinash Sharma[1,4]

[1]IIIT Hyderabad [2]Google Research [3]Stability AI [4]IIT Jodhpur
{astitva.srivastava, pranav.m}@research.iiit.ac.in
amitrajs@google.com, varunjampani@gmail.com, avinashsharma@iitj.ac.in

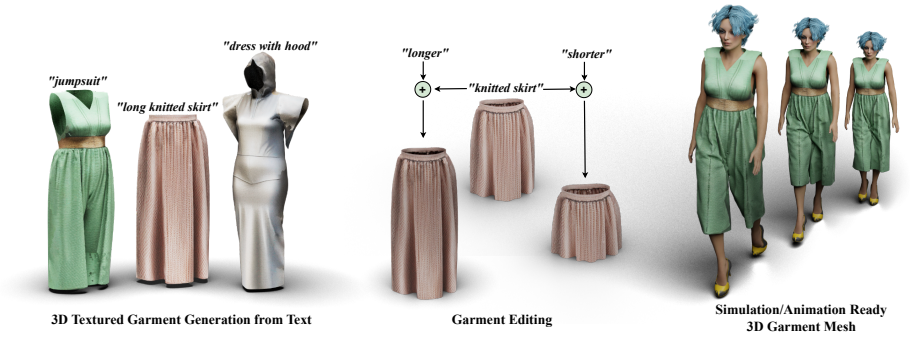

Fig. 1: Text-guided generation and editing of 3D textured garments using ***WordRobe***.

Abstract. In this paper, we tackle a new and challenging problem of text-driven generation of 3D garments with high-quality textures. We propose, ***WordRobe***, a novel framework for the generation of *unposed* & textured 3D garment meshes from *user-friendly* text prompts. We achieve this by first learning a latent representation of 3D garments using a novel coarse-to-fine training strategy and a loss for latent disentanglement, promoting better latent interpolation. Subsequently, we align the garment latent space to the CLIP embedding space in a weakly supervised manner, enabling text-driven 3D garment generation and editing. For appearance modeling, we leverage the zero-shot generation capability of ControlNet to synthesize view-consistent texture maps in a single feed-forward inference step, thereby drastically decreasing the generation time as compared to existing methods. We demonstrate superior performance over current SOTAs for learning 3D garment latent space, garment interpolation, and text-driven texture synthesis, supported by quantitative evaluation and qualitative user study. The unposed 3D garment meshes generated using *WordRobe* can be directly fed to standard cloth simulation & animation pipelines without any post-processing.

Keywords: 3D garment generation · text-to-3D · texture synthesis

1 Introduction

Recent advances in 3D content creation from textual description has large implications for modelling the virtual world. This includes a wide variety of assets such as objects [1], scenes [2], as well as humans [3]. Automated content creation has also fueled interest in generating 3D garments for applications in 3D virtual try-on, clothing for human avatars, gaming & animated characters, and AR/VR experiences. The 3D garments are typically represented as textured meshes to model the underlying surface geometry & appearance. However, creating large-scale 3D garments is prohibitively expensive, primarily due to the huge diversity in the shape, style, and appearance of the garments. Noise-free, *unposed* (i.e. in canonical/T-pose) 3D garment modelling is important for direct integration into simulation and animation pipelines. To achieve this, traditional approaches either employ design tools for manual garment creation (e.g., CLO [4]) or capture digital replicas of real garments via high-end scanners (e.g., Artec [5]). However, such approaches require significant design effort, and are expensive and difficult to scale up. Thus, there is an acute need to develop a scalable learning-based solution for automated 3D garment creation that effectively models shape, style & appearance of various garments. A variety of deep learning-based methods aim to digitize/reconstruct 3D garments from images, which can be broadly divided into two categories based on the garment representation, namely parametric or non-parametric. Parametric garment reconstruction methods [6,7] are restricted to tight-fitting and limited clothing designs due to reliance on garment templates derived from an underlying parametric human model (e.g. SMPL [8]). Nevertheless, their parametric nature supports high-quality texture maps and editing of the pose, size & shape [9]. On the other hand, non-parametric learning-based methods [10,11] can model garments of various styles and appearances (within training distribution). However, they yield *posed* geometry & low-quality textures, making the output garment ill-suited for direct integration into standard graphics pipelines. Furthermore, these methods offer no control over the shape and pose editing of the 3D garment. An alternate approach for non-parametric 3D garment modeling is effectively demonstrated in DrapeNet [12], highlighting the capability of MLPs to learn the shape distribution of 3D garments by encoding them in a latent space. This enables shape editing via latent interpolation. However, there is no support for texture which is crucial for modelling high-quality appearance details. Additionally, the generation from the garment latent space is uncontrolled and the latent manipulation is also defined by explicit per-component labels, which require significant annotation effort in case of a large variety of garments made of different components. Therefore, a controllable way to generate and edit 3D garments via intuitive inputs (e.g. images or text prompts) is desirable.

Recent text-to-3D methods [1,13–15] allow generation of generic 3D objects via *user-friendly* text prompts, eschewing the need for 3D modelling and artistic expertise. However, when employed for generating 3D garments, the surface quality of the generated garment mesh is subpar as compared to the methods trained specifically to model garments. While the overall quality of text-to-3D

methods is expected to improve as research progresses, the inherent 3D representations used by these methods have certain limitations in terms of representing 3D garments with open (non-watertight) surfaces and also lack support for editing or manipulation. Additionally, the majority of these methods rely on a multiview optimization process, which is computationally expensive and slow. To this end, we propose ***WordRobe***, a text-driven textured 3D garment generation framework. As shown in Figure 1, *WordRobe* generates high-quality *unposed* 3D garment meshes with photorealistic textures from *user-friendly* text prompts. We achieve this by first learning a latent space of 3D garments using a novel two-stage encoder-decoder framework in a coarse-to-fine manner, representing the 3D garments as unsigned distance fields (UDFs). We also introduce an additional loss function to further disentangle the latent space, promoting better interpolation. We devise a new metric to quantitatively study the effect of the proposed loss function on garment interpolation. Once the garment latent space is learned, we train a mapping network to predict garment latent codes from CLIP embeddings. This allows CLIP-guided exploration of the latent space, enabling text-driven 3D garment generation and editing. For training the aforementioned mapping network, we develop a novel *weakly-supervised* training scheme that eliminates the need for explicit manual text annotations. For text-guided texture synthesis, we leverage the capabilities of pretrained T2I models for generating diverse textures. Unlike existing multiview optimization-based methods [16–18], which are slow and expensive, we render the front & back depth maps of the 3D garments side-by-side in a single image and pass this image to ControlNet [19] for a depth-conditioned image generation. This novel approach enables text-driven texture synthesis in a single feed-forward step, saving time while outperforming existing SOTA [18] in maintaining view consistency. To the best of our knowledge, our method is the first one to enable the text-driven generation of high-fidelity 3D garments with diverse textures. In summary, our major contributions are as follows:

- A novel framework and training strategy for text-driven 3D garment generation via a garment latent space.
- A new disentanglement loss for promoting better separation of concepts in the latent space and a new metric to assess its performance.
- An optimization-free (single feed-forward) text-guided texture synthesis method that is both superior and efficient as compared to existing SOTA.

We also extend the 3D garments dataset proposed in [20] with diverse high-quality textures and corresponding text prompts, using the proposed approach. We plan to publicly release the dataset and code to further accelerate research in this space.

2 Related Work

3D Garment Digitization: Researchers have proposed several deep-learning methods [6,10,11,21,22] that attempt to digitize /reconstruct 3D garments from

images. Methods such as [23–26] use neural implicit representations (e.g. occupancy field, SDF, etc.) to model 3D clothed humans from monocular or sparse multi-view images, in a supervised learning setup. But, they fail to model garments separately from the body. ReEF [10] learns explicit boundary curves and segmentation field to model garments separately, while xCloth [11] proposed to use an alternate and efficient representation to achieve the same, while also obtaining texture maps. However, all these methods rely on high-quality real-world clothed human datasets [27] which have a limited diversity in terms of style & appearance since capturing such datasets at a large scale is expensive. Moreover, the reconstructed garments are *posed* according to the underlying body and have a sub-optimal surface quality. Another line of works takes inspiration from real-world garment creation and proposes both analytical [28] & neural methods [20] for procedurally generating production-ready *unposed* 3D garments. However, such approaches rely on sophisticated sewing patterns which are not intuitive to design. Some of the recent approaches [7, 9] avoid panel-based generation by building upon parametric human body templates (e.g., SMPL [8]) to generate parametric garments, however, they usually model tight-fitted garments with limited texture support.

Text-to-3D Generation: Recently, various text-to-3D methods have been proposed [1, 13–15, 29–31] which leverage powerful imaginative capabilities of text-to-image (T2I) diffusion models [32], combined with popular 3D representations (NeRF [33], DMTet [31], etc.) to generate 3D objects from the text prompts. However, most of these methods rely on a multiview optimization process which is computationally expensive. Moreover, NeRF-like representations are not suitable for modelling complex open garment surfaces, hence the output geometry quality is not sufficient to be directly integrated into a standard graphics pipeline. Additionally, these methods lack support for controllable manipulation or editing of the generated 3D mesh.

Text-Guided Texture Generation: Recently, Text-to-Image(T2I) Diffusion Models [34] have garnered significant interest which has led to works like [18, 35–37] for synthesizing 2D UV texture map for a given 3D mesh. The majority of these methods optimize the CLIP objective between the input text prompt and multiview images generated from a pretrained denoising diffusion model [32]. However, the output resolution is low, resulting in poor texture quality. Current SOTA method Text2Tex [18] utilizes a depth-aware image inpainting diffusion model to progressively fill in a high-resolution texture on a mesh conditioned on a text prompt. However, the progressive nature of texture filling makes this method relatively time-consuming. The resultant texture map also suffers from view inconsistencies because the denoising process across different views generates different images. On the contrary, we propose a texture generation method that generates view-consistent textures by generating all the views at once in a single feed-forward step. Note that our approach differs significantly from [37–39] since these methods explicitly train the Stable Diffusion [32] with orthogonal views

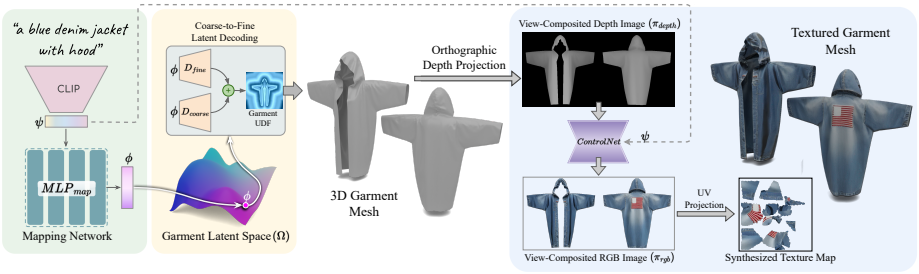

Fig. 2: Overview of the proposed method for text-guided 3D garment generation.

on 3D datasets with limited textural diversity (e.g. Objaverse [40]), whereas we leverage the zero-shot generation capabilities and the newly identified property of the pretrained ControlNet [19], enabling arbitrarily diverse textures.

3 Method

We propose *WordRobe*, a method to generate different types of 3D garments with openings (armholes, necklines etc.) and diverse textures via *user-friendly* text prompts. To achieve this, we incorporate three novel components in *WordRobe* – **3D garment latent space** (Ω) which encodes *unposed* 3D garments as latent codes, (Sec.3.1); **Mapping Network** (MLP_{map}) which predicts garment latent code from input text prompt (Sec.3.2); and **Text-guided texture synthesis** to generate high-quality diverse texture maps for the 3D garments (Sec.3.3). We provide an overview of the proposed method in Figure 2. At inference time, given an input text prompt, we first obtain its CLIP embedding ψ, which is subsequently passed to MLP_{map} to obtain the latent code $\phi \in \Omega$. We further perform two-step latent decoding of ϕ to generate the 3D garment as UDF, and extract the UV parametrized mesh representation for the same. Finally, we perform text-guided texture synthesis in a single feed-forward step by leveraging ControlNet [19] to obtain the textured 3D garment mesh.

3.1 3D Garment Latent Space

We propose to learn a latent space for the *unposed* 3D garments using a novel two-stage encoder-decoder framework. Inspired by DrapeNet [12], we adopt the Unsigned Distance Function (UDF) to represent the open garment surfaces. We employ DGCNN [41] as the garment encoder (ξ) to embed a variety of 3D garments into a latent representation by aggregating the multi-scale point features into a unified global embedding. As shown in Figure 3, given a 3D garment mesh $\mathbb{G}$, we sample the points on the surface of the mesh and pass them to the encoder ξ. The output of the encoder is a $k = 32$ dimensional garment latent code $\phi \in \Omega$, where Ω is the garment latent space, i.e. $\phi = \xi(\mathbb{G}_{points})$. We use coordinate-based MLP [42] as the decoder, which takes ϕ as input and

decodes it into the UDF representation of the corresponding garment. However, we observe that a single decoder is not suited for learning both a regularized latent space and at the same time, modelling high-frequency details, such as wrinkles and pleats (see Figure 9, Sec.4.4). Therefore, we propose to use two MLP decoders, D_{coarse} and D_{fine} to focus on two distinct objectives. Given a set of m query points, $\chi \in \mathbb{R}^{m\times 3}$, defined over a 3D grid, D_{coarse} predicts smooth (coarse) unsigned distance value for every query point according to the underlying geometry, conditioned on latent code ϕ. While D_{fine} predicts residual change in the output of D_{coarse} to capture finer details, i.e.,

$$\sigma_{fine} = D_{coarse}(\phi) + D_{fine}(\phi) = \sigma_{coarse} + \sigma_{delta} \tag{1}$$

We use Mesh-UDF [43] to convert the 3D garment meshes into UDFs, which acts as ground truth for training D_{fine}. Similarly, for training D_{coarse} we first decimate the 3D meshes, apply Laplacian Smoothing [**?**] and pass it to Mesh-UDF to obtain ground truth coarse UDFs.

As shown in Figure 3, we train the proposed framework in two stages, where we first jointly train encoder ξ and decoder D_{coarse} to learn a rich garment latent space while decoding the latent codes into coarse UDF representations. We adopt distance loss ($\mathcal{L}_{dist}$) and gradient loss ($\mathcal{L}_{grad}$) from [12], where $\mathcal{L}_{dist}$ is formulated as BCE loss between the predicted and ground truth UDF values (normalized and clipped in the range $[0, 1]$) and $\mathcal{L}_{grad}$ is the $L2$ distance between the gradients of predicted and ground truth UDFs. During training, we minimize $\mathcal{L}_{dist}$ and $\mathcal{L}_{grad}$ for each 3D query point $\mathbf{x} \in \chi$. In order to have a more organized and disentangled latent space, we introduce a disentanglement loss $\mathcal{L}_{latent}$, which encourages the batch covariance $\mathbf{\Sigma}_b$ of the individual dimensions of latent vectors to be an identity matrix and is defined for a batch b as follows:

$$\mathcal{L}_{latent} = \mathbf{\Sigma}_b - \mathbf{I}_k = \begin{bmatrix} var(\mathbf{l}_1) & covar(\mathbf{l}_1, \mathbf{l}_2) & \cdots & covar(\mathbf{l}_1, \mathbf{l}_k) \\ covar(\mathbf{l}_2, \mathbf{l}_1) & var(\mathbf{l}_2) & \cdots & covar(\mathbf{l}_2, \mathbf{l}_k) \\ \vdots & \vdots & \ddots & \vdots \\ covar(\mathbf{l}_k, \mathbf{l}_1) & covar(\mathbf{l}_k, \mathbf{l}_2) & \cdots & var(\mathbf{l}_k) \end{bmatrix} - \mathbf{I}_k \tag{2}$$

where, $\mathbf{l}_i = \{\phi_i^1, \phi_i^2, ..., \phi_i^q; 1 \leq i \leq k\}$ (q is the batch size), ϕ_i is the i^{th} dimension of the latent vector ϕ, and $\mathbf{I}_k$ in Equation 2 is $k \times k$ identity matrix. In other words, $\mathcal{L}_{latent}$ enforces dimensions of latent vector ϕ to be as independent of each other as possible, allowing ξ to encode the most prominent shape characteristics of the garments across different categories in the latent space Ω. This results in a more organized latent space, where manipulation of the latent vector along a single (or very few) dimension(s) might be sufficient to have a desirable shape change in the 3D garment. The respective loss functions for coarse and fine training stages are:

$$\begin{aligned} \mathcal{L}_{coarse} &= \lambda_{dist}\mathcal{L}_{dist} + \lambda_{grad}\mathcal{L}_{grad} + \lambda_{latent}\mathcal{L}_{latent} \\ \mathcal{L}_{fine} &= \lambda_{dist}\mathcal{L}_{dist} + \lambda_{grad}\mathcal{L}_{grad} \end{aligned} \tag{3}$$

Minimizing the above losses results in a latent space Ω where we can randomly sample a latent vector ϕ and perform a two-step decoding to generate a 3D

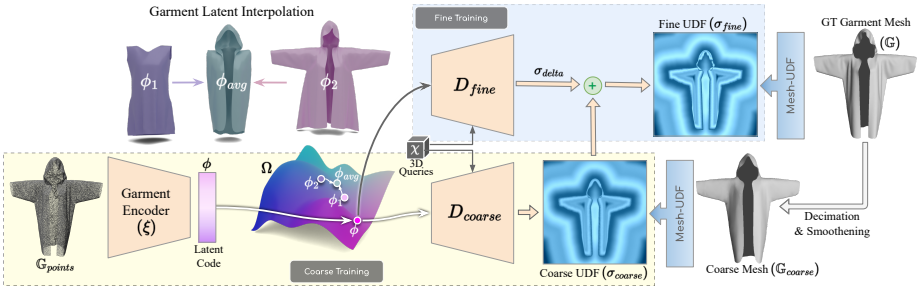

Fig. 3: The proposed coarse-to-fine training strategy for learning garment latent space.

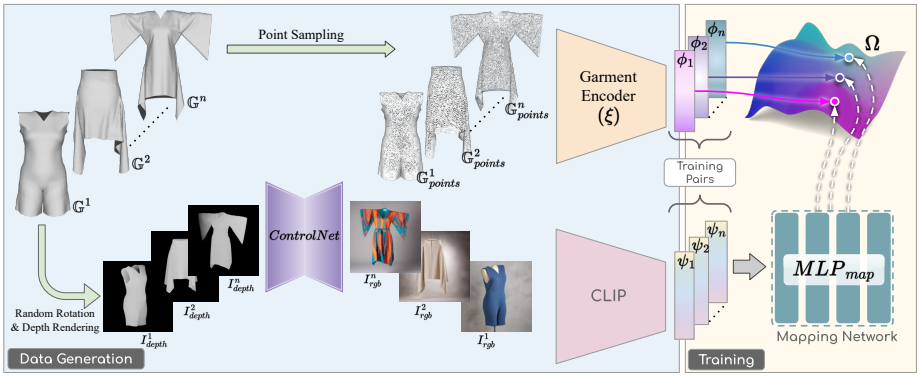

Fig. 4: Automated training data generation & weakly supervised training of MLP_{map}.

garment UDF associated with ϕ. 3D garment mesh is extracted by running a modified version of Marching Cubes proposed in [12].

3.2 CLIP-Guided 3D Garment Generation

We propose a novel weakly-supervised training scheme to align CLIP's latent space to the garment latent space Ω. Given a text prompt, we first pass it through CLIP's text encoder to get an embedding ψ. We employ a mapping network MLP_{map} that takes ψ as input and predicts a garment latent code ϕ.

In order to train MLP_{map}, a large set of garment latent codes and corresponding text prompts (to get corresponding CLIP embeddings) are required. To avoid explicit text annotations, we propose an automated way of generating training pairs (latent codes and CLIP embeddings). As illustrated in Figure 4, given a set of 3D garments $\mathbb{G}_{train} = \{\mathbb{G}^i | 1 \leq i \leq N\}$, we randomly rotate each garment mesh $\mathbb{G}^i$, render a depth map I^i_{depth}, and pass it to a depth-conditioned ControlNet [19] to generate a garment image I^i_{rgb}, guided by a garment agnostic *template prompt* – ***"a garment made of* {MATERIAL}, *with* {COLOR} *colors"***. We pick predefined values for MATERIAL={silk, cotton, wool, leather}

& COLOR={vibrant, dull, bright, shiny, matte} at random to construct the *template prompt*. We add additional prompts (e.g. high-quality, realistic, photoreal, etc.) to ensure that ControlNet produces high-quality images which are then manually verified. The generated image I^i_{rgb} is then passed to the CLIP's image encoder to generate clip embedding ψ_i. Concurrently, we sample points on the surface of every garment mesh $\mathbb{G}^i$ and feed them to the garment encoder ξ to get corresponding latent code ϕ_i. This technique eliminates the need for explicit manual text annotations for training the mapping network which is a huge benefit due to the lack of any such dataset.

Once all the corresponding pairs of ψ_i and ϕ_i are generated, we train MLP_{map} by minimizing the $L1$ loss between the MLP_{map}'s prediction and corresponding ϕ_i. During inference, the mapping network MLP_{map} takes the CLIP embedding ψ of a text prompt and predicts a latent vector ϕ, on which two-step decoding is performed to generate the 3D garment (as shown in Figure 2). This novel strategy enables taming the garment's latent space via text prompts.

3.3 Texture Synthesis

Once we have the extracted 3D garment mesh, our next aim is to generate high-quality appearance and store it in the form of a UV texture map guided by the same input text prompt. Though UV parametrization is suitable for storage and fast rendering of the mesh, it is not trivial to generate textures directly in the UV space, as UV parametrization introduces seams, disturbing the spatial arrangement of the mesh primitives, which are organized differently in UV space for different meshes and do not carry any semantic meaning to help learning. Thus, we propose a novel strategy to synthesize textures from text prompts.

Existing SOTA methods [36, 37] adopt Text-to-Image diffusion models in a multiview optimization framework, with the aim of having similar generations (in terms of colors, lighting, etc.) across different views while minimizing the CLIP objective. However, this approach is time-consuming and does not always guarantee view consistency as shown in Figure 8, as a small change in control (here, viewpoint) can drastically change the generated image. We, on the other hand, identify a highly useful property of ControlNet [19], which allows us to maintain consistency across different viewpoints of 3D garments in a single generation. More specifically, we empirically observed that if we composite multiview depth maps of a 3D object in a single image and pass it to ControlNet, the generated RGB image (guided by an input text prompt) tends to have consistent color values and lighting information across different views of that object in the image.

Leveraging the aforementioned property of ControlNet, we devise our text-driven texture synthesis methodology as follows. We first perform depth rendering of 3D garment mesh in two views − front and back, and combine them together to obtain a view-composited depth image π_{depth} as shown in Figure 2. Though any number of views can be used, We use front-back as a natural choice for partitioning a 3D garment to reduce the number of visible seams on the mesh (*please refer to supplementary for more details regarding this issue*), and also to

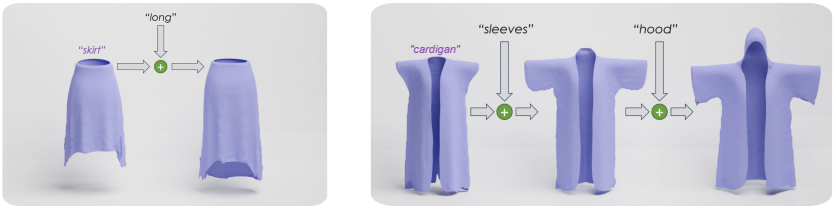

Fig. 5: Garments generated using *WordRobe* can be edited using simple text prompts.

maintain resolution. We use orthographic projection for rendering, as perspective projection leads to more information loss across tangential regions. π_{depth} is then passed to ControlNet which generates a 1024×1024 view-composited RGB image π_{rgb}, conditioned on CLIP embedding ψ of input text prompt. Finally, we UV parametrize the garment mesh and project texture information from π_{rgb} onto the UV texture map to obtain high-quality textured garment mesh.

3.4 Garment Editing via Latent Manipulation

WordRobe's encoding enables editing generated garment's attributes by manipulating its corresponding latent code ϕ. The learned garment space Ω is continuous and allows meaningful interpolation between different garment latent codes. As illustrated in Figure 7 (right), a meaningful garment can be obtained by taking a weighted average of two garment latent codes ϕ_1 & ϕ_2. For ***text-guided editing***, we introduce an intuitive approach that uses CLIP arithmetic [**?**] to automatically identify which dimensions of latent code to manipulate in order to achieve the desired change. As shown in Figure 5, we modify the length of the *"skirt"*, and add *sleeves* & *hood* to the original *"cardigan"* mesh using text prompts. ***Please refer to the supplementary for more details.***

4 Experiments & Results

We design several experiments and perform thorough qualitative & quantitative evaluations of our method, including ablative analysis. Regarding comparison with SOTA methods, since there is no existing method for direct text-driven *unposed* & *textured* garment generation, we individually compare our two-stage encoder-decoder framework with DrapeNet [12] and text-guided texture synthesis method with Text2Tex [18], both being SOTA in their respective tasks. Please refer to the supplementary for implementation and training details.

All the experiments are done on the 3D garment dataset proposed in [20], which has around $20,000$ *unposed* (canonicalized) garments spanning over 19 categories. We train our garment encoder-decoder networks on 12 categories and perform evaluations on the remaining unseen 7 categories, following the official train-test split provided by the authors of [20]. We are the first one to demonstrate generalization in learning 3D garment latent space on such a

relatively large dataset, about **30 times larger** than the datasets used by the current SOTA (DrapeNet authors show learning only on 600 training samples across 7 categories from CLOTH3D [44]).

4.1 Qualitative Results for Text-Driven Garment Generation

We visualize 3D textured garment meshes generated using *WordRobe* in Figure 6. As shown in the figure, *WordRobe* generalizes to a large variety of garment styles and textual appearance using *user-friendly* text prompts (e.g. *"blue denim cargo shorts"*, *"spiderman jacket with hood" etc.*). The text prompts in the figure are shortened for readability. ***Please refer to the supplementary for details reagarding the text prompts used for evaluation, additional qualitative results of our method, and a qualitative user study with*** 67 ***participants.***

4.2 Evaluation of Garment Latent Space

Qualitative Evaluation: In Figure 7 (left), we provide a qualitative comparison with DrapeNet [12] on random test samples from unseen garment categories along with the ground truth. It can be observed from the figure that DrapeNet tends to learn underlying shape, but fails to model garment details. On the other hand, our coarse-to-fine training strategy outperforms DrapeNet in modeling complex garments. We also demonstrate the interpolation capability of the latent space learned using the coarse-to-fine strategy in Figure 7 (right). For each row (a), (b) & (c), we first predict two latent codes ϕ_1 & ϕ_2 using appropriate text prompts and then generate interpolated garments by taking a weighted average of ϕ_1 & ϕ_2. As shown in the figure, the coarse & fine decoders decode the interpolated latent code into a meaningful garment geometry.

Quantitative Evaluation: We perform a quantitative comparison of our garment generation method with DrapeNet [12] in Table 1, where we report standard metrics, Chamfer Distance (CD) and Point-to-Surface (P2S) distance, on the test set (please refer to supplementary for the definition of these metrics). We achieve approx. 40% lower value for CD and 42% lower value for P2S, outperforming DrapeNet by a significant margin. This indicates that the surface quality of the generated 3D garments using *WordRobe* is superior to that of DrapeNet.

4.3 Evaluation of Texture Synthesis

Qualitative Evaluation: We perform texture synthesis on meshes taken from the test set, governed by the input text prompt from the user and compare with Text2Tex [18], as shown in Figure 8. Text2Tex [18] uses a multiview-optimization strategy, which is slow (takes around 5 min for a prompt on a single RTX 4090 GPU) and sometimes converges sub-optimally and results in view inconsistency

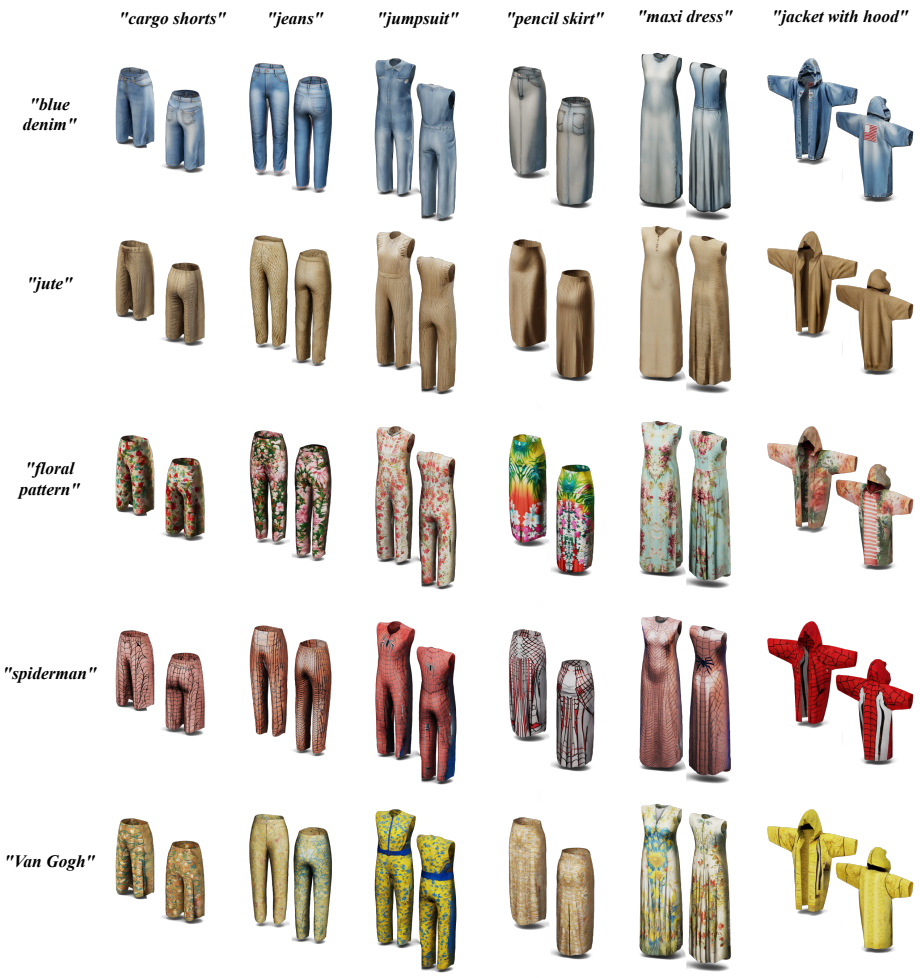

Fig. 6: High-quality *unposed* 3D garment meshes with diverse textures generated via *user-friendly* text prompts using *WordRobe*.

and patchy artifacts (highlighted in dotted red circles). On the other hand, our optimization-free view-composited method takes around 22 seconds under the same settings, while producing high-quality and view-consistent texture details. **Quantitative Evaluation:** For quantitative comparison with Text2Tex [18] for generating text-driven textures for a given garment mesh, we first use the texture maps obtained from both methods to render the input garment mesh in 4 random views. We then compute the average CLIP-Score (higher values are preferred) proposed in [45] between the input text prompt and the rendered images, and report it in Table 2, where we outperform Text2Tex [18] under three major variants of the CLIP encoder.

Table 1: Quantitative evaluation of garment encoding-decoding framework.

Method	CD↓	P2S↓
DrapeNet [12]	1.796	0.573
Ours* (Single Stage)	1.631	0.494
Ours (w/o $\mathcal{L}_{grad}$)	1.886	0.612
Ours (w/o $\mathcal{L}_{latent}$)	1.094	0.331
Ours	**1.078**	**0.329**

Table 2: Comparison of text-guided texture synthesis.

Method	CLIP Score ↑		
	ViT-H/14	ViT-L/14	ViT-B/16
Text2Tex [18]	0.263 ± 0.047	0.243 ± 0.041	0.232 ± 0.036
Ours	**0.304 ± 0.043**	**0.265 ± 0.037**	**0.241 ± 0.034**

4.4 Ablation Study

Single-stage vs Two-stage Decoding: We study the importance of two-stage (coarse-to-fine) decoding of the latent code to produce noise-free garment geometry. As shown in Figure 9, **Ours*** (proposed framework but with only a single decoder), results in holes & isolated noise (highlighted in cyan boxes). However, employing both coarse and fine decoders significantly suppresses the noise. This improvement in the surface quality is also evident from Table 1, where the two-stage framework (Ours) achieves lower values for CD and P2S as compared to single decoder variant **(Ours*)**.

Choice of Loss Functions: We show a qualitative ablative study on the choice of loss functions used in learning garment latent space in Figure 9 and report the quantitative numbers (CD & P2S) in Table 1. We observe that $\mathcal{L}_{grad}$ plays a significant role in reducing high-frequency surface noise by acting as a regularizer, resulting in lower values of CD and P2S. The use of $\mathcal{L}_{latent}$ provides an

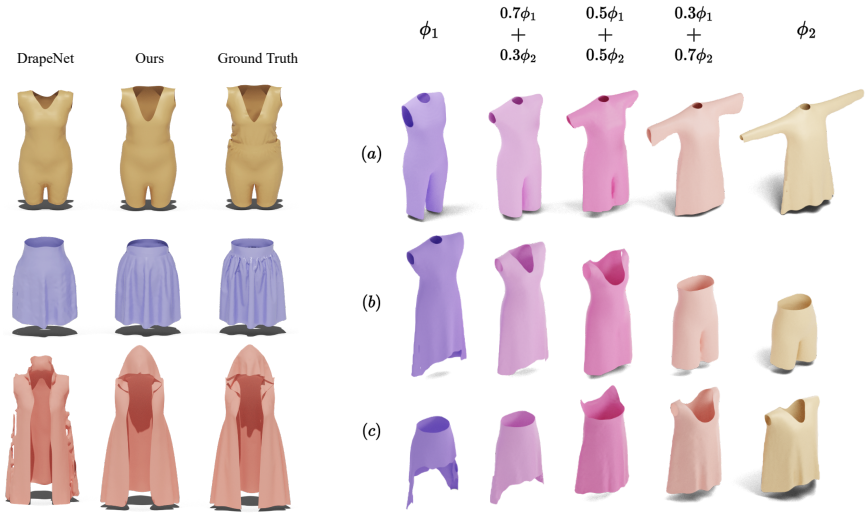

Fig. 7: Qualitative comparison with DrapeNet [12] **(left)**; and Interpolation of garment latent codes **(right)**.

improvement in the quality of the garment (especially around the boundaries), however, the drop in CD and P2S values is not very significant. At last, we also perform an ablative study over the choice of losses while training MLP_{map}, re-

Fig. 8: Qualitative comparison of Texture Synthesis. Our method provides better view consistency as compared to Text2Tex (red dotted circle) while being 13 times faster.

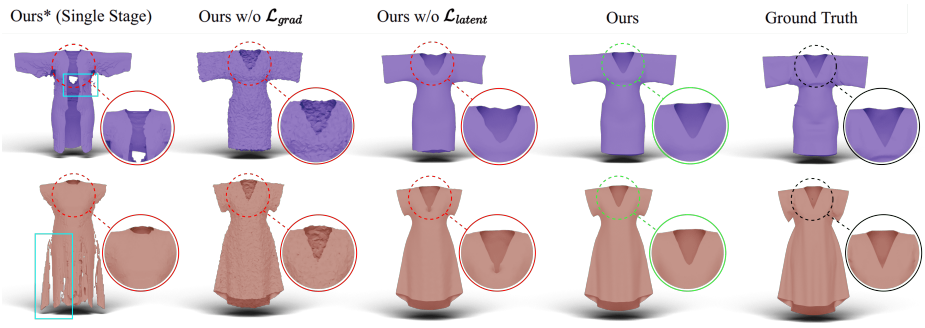

Fig. 9: Qualitative ablation of the proposed encoder-decoder framework.

Table 3: Quantitative evaluation of training losses for MLP_{map}.

Loss	$\mathbf{MSE_{test}} \downarrow$
$L2$	0.7433
$L1 + L_{cosine}$	0.3773
$L1$	**0.3481**

Table 4: Quantitative evaluation of latent interpolation.

	$\mathbf{\Delta_{area}} \downarrow$	$\mathbf{\Delta_{vol}} \downarrow$
w/o $\mathcal{L}_{latent}$	0.028	1.275
with $\mathcal{L}_{latent}$	**0.022**	**1.206**

port Mean Square Error (MSE) over the test set in Table 3, and conclude that $L1$ loss alone is a more suitable choice for learning a mapping from CLIP space to the garment latent space.

Interpolation Study: We conduct a quantitative ablation study in Table 4 to understand the effect of $\mathcal{L}_{latent}$ in achieving better interpolation in the garment latent space. Generally, interpolation is assessed qualitatively. Therefore, we formulate two novel metrics for evaluating the interpolation quantitatively, based on the assumption that while interpolating between two shapes, the surface area and volume of the resulting interpolated shape should also get interpolated accordingly [46]. Given two 3D garments $\mathbb{G}^1$ & $\mathbb{G}^2$ and their respective latent codes ϕ_1 & ϕ_2, the interpolated latent code is obtained as $\phi_{avg} = \alpha\phi_1 + (1-\alpha)\phi_2$, which is then decoded to obtain garment $\mathbb{G}^{avg}$. Here, α is the interpolation weight ranges between 0 and 1. We define interpolated surface area difference $\Delta_{area} = ||\mathcal{A}(\mathbb{G}^{avg}) - \{\alpha\mathcal{A}(\mathbb{G}^1) + (1-\alpha)\mathcal{A}(\mathbb{G}^2)\}||$ and interpolated volume difference $\Delta_{vol} = ||\mathcal{V}(\mathbb{G}^{avg}) - \{\alpha\mathcal{V}(\mathbb{G}^1) + (1-\alpha)\mathcal{V}(\mathbb{G}^2)\}||$, where $\mathcal{A}(\mathbb{G}_i)$ is the surface area and $\mathcal{V}(\mathbb{G}_i)$ is the volume (after hole-filling) of the garment mesh $\mathbb{G}_i$. We randomly create pairs from test garment meshes and use random values of α for each pair to compute Δ_{area} & Δ_{vol}, and report in Table 4. As evident from the table, the usage of $\mathcal{L}_{latent}$ during training results in lower values of Δ_{area} & Δ_{vol}, promoting better interpolation by providing a more organized latent space.

We propose ***WordRobe***, a novel method for text-driven generation and editing of textured 3D garments. *WordRobe* achieves SOTA performance in learning a 3D garment latent space and in generating view-consistent high-fidelity texture maps. The carefully designed two-stage decoding strategy helps in generating high-quality garment geometry, and the new disentanglement loss promotes better interpolation. Our weakly supervised CLIP-to-latent mapping technique enables text-driven garment generation without requiring any annotated dataset. We report superior qualitative & quantitative performance compared to existing methods and explain our design choices with appropriate ablative analysis. We believe our work paves the way towards production-ready *unposed* garment generation from text prompts.

References

1. Amit Raj, Srinivas Kaza, Ben Poole, Michael Niemeyer, Ben Mildenhall, Nataniel Ruiz, Shiran Zada, Kfir Aberman, Michael Rubenstein, Jonathan Barron, Yuanzhen Li, and Varun Jampani. Dreambooth3d: Subject-driven text-to-3d generation. *ICCV*, 2023. 2, 4
2. Jingbo Zhang, Xiaoyu Li, Ziyu Wan, Can Wang, and Jing Liao. Text2nerf: Text-driven 3d scene generation with neural radiance fields. *arXiv preprint arXiv:2305.11588*, 2023. 2
3. Nikos Kolotouros, Thiemo Alldieck, Andrei Zanfir, Eduard Gabriel Bazavan, Mihai Fieraru, and Cristian Sminchisescu. Dreamhuman: Animatable 3d avatars from text. 2023. 2
4. CLO. https://www.clo3d.com/en/. 2
5. Artec3D. https://www.artec3d.com/portable-3d-scanners. 2
6. Boyi Jiang, Juyong Zhang, Yang Hong, Jinhao Luo, Ligang Liu, and Hujun Bao. Bcnet: Learning body and cloth shape from a single image. In *European Conference on Computer Vision*. Springer, 2020. 2, 3
7. Enric Corona, Albert Pumarola, Guillem Alenyà, Gerard Pons-Moll, and Francesc Moreno-Noguer. Smplicit: Topology-aware generative model for clothed people, 2021. 2, 4
8. Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, October 2015. 2, 4
9. Zhaoqi Su, Tao Yu, Yangang Wang, and Yebin Liu. Deepcloth: Neural garment representation for shape and style editing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):1581–1593, 2023. 2, 4
10. Heming Zhu, Lingteng Qiu, Yuda Qiu, and Xiaoguang Han. Registering explicit to implicit: Towards high-fidelity garment mesh reconstruction from single images, 2022. 2, 3, 4
11. Astitva Srivastava, Chandradeep Pokhariya, Sai Sagar Jinka, and Avinash Sharma. xcloth: Extracting template-free textured 3d clothes from a monocular image. *Proceedings of the 30th ACM International Conference on Multimedia*, 2022. 2, 3, 4
12. Luca De Luigi, Ren Li, Benoit 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. 2, 5, 6, 7, 9, 10, 12
13. Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *arXiv preprint arXiv:2305.16213*, 2023. 2, 4
14. Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2023. 2, 4
15. Jingxiang Sun, Bo Zhang, Ruizhi Shao, Lizhen Wang, Wen Liu, Zhenda Xie, and Yebin Liu. Dreamcraft3d: Hierarchical 3d generation with bootstrapped diffusion prior, 2023. 2, 4
16. Xin Yu, Peng Dai, Wenbo Li, Lan Ma, Zhengzhe Liu, and Xiaojuan Qi. Texture generation on 3d meshes with point-uv diffusion, 2023. 3
17. Elad Richardson, Gal Metzer, Yuval Alaluf, Raja Giryes, and Daniel Cohen-Or. Texture: Text-guided texturing of 3d shapes, 2023. 3

18. Dave Zhenyu Chen, Yawar Siddiqui, Hsin-Ying Lee, Sergey Tulyakov, and Matthias Nießner. Text2tex: Text-driven texture synthesis via diffusion models. *arXiv preprint arXiv:2303.11396*, 2023. 3, 4, 9, 10, 11, 12
19. Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023. 3, 5, 7, 8
20. Maria Korosteleva and Sung-Hee Lee. Generating datasets of 3d garments with sewing patterns. In J. Vanschoren and S. Yeung, editors, *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021. 3, 4, 9
21. Bharat Lal Bhatnagar, Garvita Tiwari, Christian Theobalt, and Gerard Pons-Moll. Multi-Garment Net: Learning to dress 3D people from images. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019. 3
22. Heming Zhu, Yu Cao, Hang Jin, Weikai Chen, Dong Du, Zhangye Wang, Shuguang Cui, and Xiaoguang Han. Deep fashion3d: A dataset and benchmark for 3d garment reconstruction from single images. *ArXiv*, abs/2003.12753, 2020. 3
23. Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. PIFu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019. 4
24. 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 Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 4
25. Zerong Zheng, Tao Yu, Yebin Liu, and Qionghai Dai. Pamir: Parametric model-conditioned implicit representation for image-based human reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 4
26. Yuliang Xiu, Jinlong Yang, Xu Cao, Dimitrios Tzionas, and Michael J. Black. ECON: Explicit Clothed humans Optimized via Normal integration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023. 4
27. CVIT. 3dhumans: A rich 3d dataset of scanned humans, 2021. 4
28. Nico Pietroni, Corentin Dumery, Raphael Falque, Mark Liu, Teresa Vidal-Calleja, and Olga Sorkine-Hornung. Computational pattern making from 3d garment models. *ACM Trans. Graph.*, 41(4), jul 2022. 4
29. Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion, 2022. 4
30. William Gao, Noam Aigerman, Thibault Groueix, Vladimir G. Kim, and Rana Hanocka. Textdeformer: Geometry manipulation using text guidance, 2023. 4
31. Tianchang Shen, Jun Gao, Kangxue Yin, Ming-Yu Liu, and Sanja Fidler. Deep marching tetrahedra: a hybrid representation for high-resolution 3d shape synthesis, 2021. 4
32. Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. 4
33. Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 4
34. Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. 4
35. Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. Latent-nerf for shape-guided generation of 3d shapes and textures. *arXiv preprint arXiv:2211.07600*, 2022. 4

36. 4, 8
37. Yash Kant, Aliaksandr Siarohin, Michael Vasilkovsky, Riza Alp Guler, Jian Ren, Sergey Tulyakov, and Igor Gilitschenski. invs: Repurposing diffusion inpainters for novel view synthesis, 2023. 4, 8
38. Minda Zhao, Chaoyi Zhao, Xinyue Liang, Lincheng Li, Zeng Zhao, Zhipeng Hu, Changjie Fan, and Xin Yu. Efficientdreamer: High-fidelity and robust 3d creation via orthogonal-view diffusion prior, 2023. 4
39. Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation, 2023. 4
40. Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects, 2022. 5
41. Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics (TOG)*, 2019. 5
42. Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space, 2019. 5
43. Benoit Guillard, Federico Stella, and Pascal Fua. Meshudf: Fast and differentiable meshing of unsigned distance field networks. In *European Conference on Computer Vision*, 2022. 6
44. Hugo Bertiche, Meysam Madadi, and Sergio Escalera. Cloth3d: clothed 3d humans. In *European Conference on Computer Vision*, pages 344–359. Springer, 2020. 10
45. Shivangi Aneja, Justus Thies, Angela Dai, and Matthias Nießner. ClipFace: Text-guided Editing of Textured 3D Morphable Models. In *ArXiv preprint arXiv:2212.01406*, 2022. 11
46. Thor V. Christiansen, Jakob Andreas Bærentzen, Rasmus R. Paulsen, and Morten R. Hannemose. Neural Representation of Open Surfaces. *Computer Graphics Forum*, 2023. 14