ChatGarment: Garment Estimation, Generation and Editing via Large Language Models

Siyuan Bian[1,2] Chenghao Xu[1,3] Yuliang Xiu[1,4] Artur Grigorev[1,5] Zhen Liu[1,6]
Cewu Lu[2] Michael J. Black[1] Yao Feng[7,8,†]

[1]Max Planck Institute for Intelligent Systems [2]Shanghai Jiao Tong University
[3]EPFL [4]Westlake University [5]ETH Zürich [6]Mila, University of Montreal
[7]Meshcapade [8]Stanford University

chatgarment.github.io

(a) Input image (b) Estimated sewing pattern, 3D garment, and its animation (c) Input garment (d) Edited sewing pattern (e) Edited garment

Figure 1. As a multimodal 3D garment creator, ChatGarment understands both images and language. It can estimate complex 3D garments represented as sewing patterns from a single image, which can be easily animated and simulated. It also supports garment editing based on text instructions.

Abstract

We introduce ChatGarment, a novel approach that leverages large vision-language models (VLMs) to automate the estimation, generation, and editing of 3D garments from images or text descriptions. Unlike previous methods that struggle in real-world scenarios or lack interactive editing capabilities, ChatGarment can estimate sewing patterns from in-the-wild images or sketches, generate them from text descriptions, and edit garments based on user instructions, all within an interactive dialogue. These sewing patterns can then be draped on a 3D body and animated. This is achieved by finetuning a VLM to directly generate a JSON file that includes both textual descriptions of garment types and styles, as well as continuous numerical attributes. This JSON file is then used to create sewing patterns through a programming parametric model. To support this, we refine the existing programming model, GarmentCode, by expanding its garment type coverage and simplifying its structure for efficient VLM fine-tuning. Additionally, we construct a large-scale dataset of image-to-sewing-pattern and text-to-sewing-pattern pairs through an automated data pipeline. Extensive evaluations demonstrate ChatGarment's ability to accurately reconstruct, generate, and edit garments from multimodal inputs, highlighting its potential to simplify workflows in fashion and gaming applications. Code and data are available at https://chatgarment.github.io/.

1. Introduction

Imagine seeing a photo of someone in a nice looking outfit that you would like for yourself or your 3D avatar. Our goal is to take that photo and convert it into a 2D sewing pattern and then enable the user to use language to edit the pattern, for example, making the sleeves longer or editing the neckline. We then drape the garment on a 3D body and animate it. This is illustrated in Fig. 1. With this approach, given just an image and/or text, we construct sewing patterns for 3D garments that can be immediately used in fashion or gaming applications. This lightweight process is in stark contrast to how sewing patterns and 3D garments are traditionally created by an artist or clothing designer. In contrast to the traditional labor-intensive process, our approach exploits

[†]This work was done while YF was at Meshcapade.

large vision-language models (VLMs) for the first time to democratize the capture and design of clothing.

Recent work explores the generation or estimation of 3D clothing using text and/or images as input [6, 8, 21, 42]. These methods use various 3D representations including 3D meshes, point clouds, or implicit representations like Unsigned Distance Functions (UDFs) and Signed Distance Functions (SDFs). We observe, however, that most clothing is designed and manufactured using 2D patterns, and posit that this is a more natural representation. Sewing patterns can be seamlessly integrated into existing garment design pipelines for animation and manufacturing. Recent approaches [16, 30, 33] use sewing patterns for garment estimation and generation, but mapping images or text to sewing patterns remains challenging due to limited training data, leading to poor generalization across diverse clothing types. These methods also does not support interactive editing, making post-processing labor-intensive for 3D artists. Fortunately, recent work, such as GarmentCode [24, 25], provides a programmatic framework for encoding garment information into a JSON file. This JSON file includes textual descriptions of garment types, styles and numerical attributes, offering a rich representation of sewing patterns. The JSON file is then processed by the GarmentCode framework to produce 2D sewing patterns, which can subsequently be draped on a 3D body. However, to date, there is no automated method to directly extract a GarmentCode representation (JSON file) from an image or text description.

Our observation is that a JSON file resembles natural language, making it suitable for interpretation by large language models (LLMs). At the same time, Vision-LLMs (VLMs) excel at understanding images and garment concepts, such as types of garments and specific features like long sleeves. If we enable the VLM to translate its garment understanding into the structured format of GarmentCode, then it becomes capable of generating sewing patterns. This allows the VLM to take an image as input and output a garment, while also supporting garment generation and editing through text descriptions or interactive dialogue.

Specifically, we introduce **ChatGarment**, a method that finetunes a VLM to take text queries, optionally with images, as input and outputs a JSON garment file. This JSON file is then used for GarmentCode generation. The JSON file includes textual descriptions of garment types (e.g., skirts, pants), styles (e.g., collar type), and continuous numerical attributes (e.g., pant length). Inspired by previous work [11, 14, 26], we introduce a new token (`<ENDS>`) to represent these numerical values and train an MLP projection layer to decode them from the token embedding. Those numbers, combined with the VLM's language output, are formatted into the final JSON file. In our approach, we keep the VLM's vision encoder fixed, fine-tune the language model using LoRA [18], and jointly train the MLP projection layer. To adapt GarmentCode [24] for more efficient VLM training, we create an improved version that supports a wider range of clothing types (e.g. open-jacket) and removes unnecessary settings to reduce token usage in training. We also develop an automated data construction pipeline that leverages existing tools for garment generation, draping, simulation, rendering, and automatic labeling with GPT-4o. This pipeline enables building a dataset with image-to-GarmentCode and text-to-GarmentCode pairs, including 20K new garments and 1M images with detailed descriptions, supporting both garment creation and editing.

We evaluate ChatGarment across a diverse set of tasks, including those specifically trained for, and novel applications that demonstrate the generalization capabilities of VLMs. For single-image garment reconstruction, ChatGarment outperforms prior sewing-pattern-specific and LLM-based models on the Dress4D [54] and CloSE [2] evaluation datasets. Additionally, we demonstrate garment editing, generation, and multi-turn dialogue, highlighting our approach's versatility. ChatGarment can flexibly use text and images to create and edit 3D garments that can be readily animated, introducing a new workflow for garment design.

We summarize our contributions as follows:

- A unified model capable of estimating, generating and editing sewing patterns from multimodal inputs.
- The first approach to leverage VLMs for directly generating JSON files for garment creation.
- A refined version of GarmentCode that supports more diverse garment types and is optimized for VLM training.
- An automatic data construction pipeline for generating image-to-sewing-pattern and text-to-sewing-pattern data, along with the release of a large-scale dataset.

2. Related Work

Garment Creation. Most 3D garments have predefined templates (*e.g.*, sewing patterns) and categories (*e.g.*, skirts, shirts, pants, dresses, *etc.*), but their non-rigid nature makes them highly deformable with dynamic motions. Efforts to create 3D garments can be grouped into "template-based" or "free-form" methods, based on their 3D representations.

A line of template-based approaches register predefined garment templates to 2D observations, which could be real captures, e.g. from monocular video [5, 21, 36, 42] and images [3, 6, 7, 62], generated data [49], or derived from the "proxy gradient" computed from 2D diffusion models [27, 47] via score distillation sampling (SDS) [41]. When the template is classified correctly, the predefined garment template, refined through a coarse-to-fine step, ensures topological stability and correctness. Unfortunately, template-based methods struggle to accurately model garments that deviate from the predefined templates, particularly those with diverse styles, such as long skirts, or garments that undergo topological changes, like opening a jacket.

To address these limitations, free-form representations like (un)signed distance functions [6, 8, 50, 59], occupancy fields [45, 46, 56, 57], tetrahedrons [20, 35, 48], and point clouds [37, 38] are leveraged. However, this flexibility can compromise the inherent structure of garments, potentially leading to non-garment shapes and floating artifacts. Without shape regularization, incomplete observations lead to the reconstructed unseen parts being overly smooth. Even worse, rigging or simulating these free-form garments becomes extremely challenging.

The sewing pattern is a well-balanced solution – with pre-defined yet flexible templates. Several methods exploit sewing patterns to reconstruct 3D garments. MulayCap [51] estimates the parameters of a sewing pattern from a monocular video. ISP [30] parameterizes sewing patterns as implicit functions in a 2D space. Using ISP as a garment representation, GarmentRecovery [29] trains a feedforward network to infer sewing patterns from in-the-wild photos. NeuralTailor [23] uses an RNN to produce more complex sewing patterns from point clouds. Wang *et al*. [53] learn a shared garment shape space that can be inferred from multimodal input such as sketches, SewFormer [33] uses a transformer to infer multi-panel patterns from monocular images, and DressCode [16] estimates quantized sewing patterns with a GPT-based model from text inputs. Despite promising results, these approaches are limited to basic sewing patterns, featuring only front and back panels, or constrained by the limited sewing patterns seen in the training data. Additionally, none of the above works have explored using programming parametric sewing patterns, such as GarmentCode [24, 25] to create more complex garments, either in generation or reconstruction.

VLMs for 3D tasks. With their broad understanding of the world, LLMs and VLMs are widely used to reason about 3D. There has been significant work on generating, editing, and analyzing 3D objects and scenes [13, 17, 19, 31, 44, 52, 58, 61]. While these methods show the power of VLMs to reason about objects, they do not address our problem of representing 2D patterns that correspond to 3D shapes – this is a special property of garments. Some attempts have been made to connect sewing patterns with language, such as DressCode [16]. This motivates us to present ChatGarment, which connects a large language model to GarmentCode, sharing a similar approach to recent strategies [11, 32] by finetuning LLMs [18, 34, 43]. Powered by VLMs, ChatGarment significantly improves generalization and robustness in reconstructing out-of-domain images and increases diversity in 3D clothing generation. It also supports sewing pattern editing and multi-turn dialogue, greatly simplifying the 3D garment workflow.

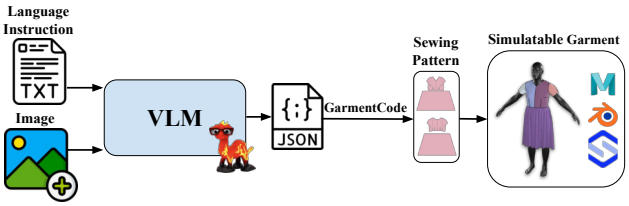


Figure 2. **Pipeline of ChatGarment.** ChatGarment takes text or an image as input and generates a JSON file. The JSON file is decoded into 2D sewing patterns using GarmentCode [24] and then draped onto the human body. The final 3D garments are compatible with simulation software (*e.g*., MAYA, Blender, Style3D, *etc*.).

3. Method

3.1. Pipeline

The goal of ChatGarment is to simplify the workflow of garment creation by automatically estimating, generating, and editing garments from multimodal image and text input. To achieve this, we finetune LLaVA [32], a multimodal large language model f_ϕ. It takes an input image X_v and text prompts X_q, and produces a textual response Y_t as $Y_t = f_\phi(X_q, X_v)$ or $Y_t = f_\phi(X_q)$ in the absence of an image. From this output, a structured JSON file J_t is extracted as a subset of Y_t, which will be decoded into sewing patterns using the GarmentCode [24] decoder D_g. This pipeline is illustrated in Fig. 2.

LLM-friendly Rich GarmentCode. We use GarmentCode to convert the JSON file into a 3D garment mesh in the canonical pose. GarmentCode [24] is a programming parametric model that uses a domain-specific language (DSL) for creating 2D sewing patterns, with a hierarchical JSON structure to define garment components and stitching rules. These patterns can be draped onto body models via a warp-based pattern stitching algorithm [24, 39], and further be simulated easily. GarmentCode can model diverse garments with intricate geometric details, including different cuts, frills, and pleats. To better align GarmentCode with our model's language-driven requirements, we develop a refined version, GarmentCodeRC (Richer & Cleaner) with two major improvements (see Fig. 3).

First, we modify stitching and panel positioning to better support diverse garment types, such as open-front jackets, high-waist skirts, and fitted pant legs. This allows us to model a broader range of real-world garments and aligns more closely with natural garment descriptions. Second, we simplify GarmentCode's JSON configuration to better suit LLM training. The original configuration is redundant, applying the same settings across all garment types. We optimize this by automatically removing irrelevant settings (e.g., omitting skirt-related parameters for upper-body garments). This adjustment reduces the average language token count from 900 to 350, decreasing ambiguity in LLM training and improving efficiency. Furthermore, to stabilize model train-

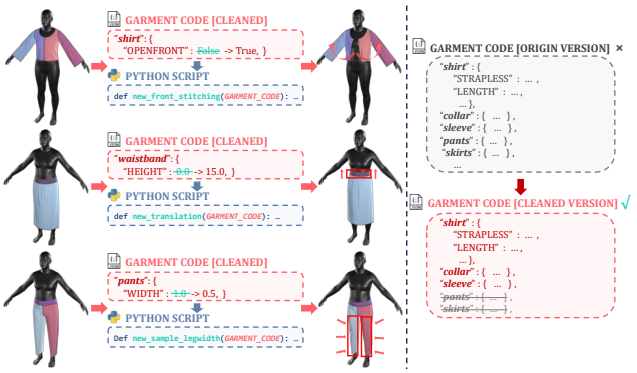


Figure 3. **GarmentCodeRC**. Left: new options to model open-front jackets, high-waist skirts, and tight pant legs. Right: simplified JSON configuration for more efficient LLM training.

ing, we normalize all floating-point values to a $[0, 1]$ range. Examples of the GarmentCodeRC JSON configuration are given in *Sup. Mat.* Throughout this paper, "GarmentCode" refers to our enriched and simplified version.

LLM for Textual and Numeric Outputs. The output JSON file J_t contains both textual and numeric descriptions. Previous research [11, 14, 26] has shown that LLMs struggle in predicting continuous and precise numerical values. To overcome this, following ChatPose [11, 26], we encode these continuous values into language token embeddings and train a projection layer to recover accurate numerical values, such as garment length. In this setup, the language embedding for Y_t is represented as H_t. From the textual output Y_t, upon encountering the start token <STARTS>, we extract the textual content of the JSON file, which spans from <STARTS> to <ENDS>. The language embedding at the end token <ENDS>, where numeric information is encoded, is represented as $H_t^{\mathbf{n}}$. A projection layer uses this embedding to predict the numerical values as $\mathbf{n} = g_\Theta(H_t^{\mathbf{n}})$. Finally, we merge these numerical values with the textual descriptions extracted earlier and format them into our final sewing pattern JSON configuration file J_t.

3.2. Automatic Data Construction

To train ChatGarment, we need image-to-JSON and text-to-JSON paired data. While some work like SewFormer [33] offers image-to-JSON pairs, they fall short in garment variety. For example, they lack tight-sleeve garments and only support a limited variety of skirts. Additionally, text descriptions for garment editing are missing in current datasets. We address these gaps by developing an automated data construction pipeline that integrates various resources.

Garment Generation, Simulation, and Rendering. Following GarmentCodeData [25], we sample 20,000 garments by randomizing values in sewing pattern configurations. To improve real-world garment relevance, we adjust the sampling ratios and assign lower weights for asymmetric garments and those with intricate cuffs. To create outfits, we pair top and bottom garments together or use single-piece items like dresses or jumpsuits. The garment set is draped on a SMPL-X [40] neutral model.

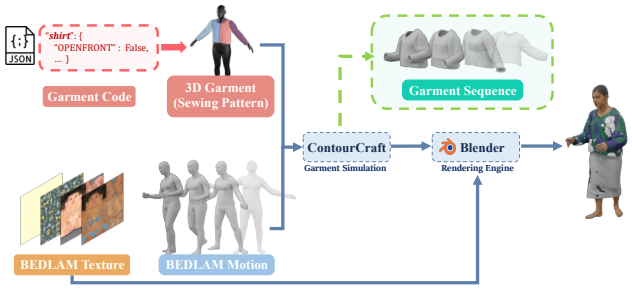


Figure 4. **Data Construction Pipeline.** We generate garments from JSON configurations, simulate them with ContourCraft [15] and render with Blender.

We automate garment simulation using ContourCraft [15], chosen for its stability and scalability, with SMPL-X motion sequences sampled from the BEDLAM dataset [4]. Specific garment vertices are attached to corresponding body vertices using predefined rules to prevent garments from slipping off. For example, the upper edges of pants and skirts are connected to the nearest body vertices, and similar attachments are made for upper-body garment vertices near the shoulders. We filter out simulations with significant interpenetration or improper positioning in two steps. First, GarmentCode performs a sewing-pattern validation check, ensuring the integrity of panel edges and stitching. Second, we verify simulation results by assessing inter-penetration, and identify garments at risk of slipping off by inspecting the proximity of each garment vertex to the nearest body part. Each garment is then rendered in Blender from four random perspectives at elevation angles between 0 and 30 degrees, with garment and body textures sourced from the BEDLAM dataset. These steps are shown in Fig. 4.

GPT-4o Labeling. The previous method [16] prompts GPT-4o to generate the general geometric descriptions for garments, such as the garment types, lengths, widths and sleeve lengths. However, these general descriptions are insufficient for garment detail understanding: the finetuned VLM struggles to interpret detailed features of garment components, such as pant cuffs, or execute specific garment edits. To address this, we construct a new dataset with low-level descriptions for individual garment parts. For image-based reconstruction, we instruct GPT-4o to generate descriptions for all visible garment parts in the image. For garment editing, we sample 5,000 garment parts (shirt main body, collar, sleeves, sleeve cuffs, waistband, pant legs, pant cuffs, and skirts), and use GPT-4o to generate descriptive text for each part. Assembling these components yields 20,000 new garments with detailed annotations.

For garment editing, we randomly select two garments that differ in certain parts, with the editing prompt being: "`Change the garment sewing pattern by modifying <PART-1> to <DESC-1>, <PART-2> to <DESC-2>, ⋯ while keeping other parts unchanged.`". `<PART-1>, <PART-2>, ⋯` represent part names that differ between the two garments, and `<DESC-1>, <DESC-2>` are target garment part descriptions.

We finally build a synthetic dataset of 40K garments and 1 million images with text annotations, supporting garment reconstruction and editing from multimodal inputs. Plus, we use GPT-4o to label part-level details of 38K SHHQ [12] images to make the VLM better understand in-the-wild garment images. Compared to previous garment sewing pattern datasets [33], our dataset offers: 1) **Diversity**: GarmentCode generates a wider range of real-world garment types; 2) **Detailed Descriptions**: Garments are labeled with detailed geometric features; 3) **Precise Control**: Allows for precise, low-level garment manipulation.

3.3. Training

We use LLaVA [32] as our base VLM and finetune it with LoRA [18], with its parameters denoted as ϕ_{lora}. Additionally, we optimize the LLM token embedding e_{token}, the LLM head layer $\texttt{lm\_head}_{\Theta'}$, and the projection layer g_{Θ}.

Loss Function. As described in Sec. 3.1, the numeric embedding is extracted from the token `[ENDS]` as $H_t^{\mathbf{n}}$. An MLP projection layer maps the embedding $H_t^{\mathbf{n}}$ to 76 float values $\mathbf{n}$ in the JSON configuration. Given the ground truth text output $\hat{Y}_t$ and sewing pattern values $\hat{\mathbf{n}}$, we optimize with:

$$\mathcal{L} = \text{CE}(\hat{Y}_t, Y_t) + \lambda_{\mathbf{n}} \left| (\hat{\mathbf{n}} - \mathbf{n}) \cdot \mathbf{m} \right|, \tag{1}$$

where the first term is a cross-entropy loss, and the second term is the L1 difference for numeric values. $\mathbf{m}$ is a mask that filters out irrelevant numeric values, and $\lambda_{\mathbf{n}} = 0.1$ is the weight for the L1 loss. See more training details in *Sup. Mat.*

3.4. Multi-turn Dialogue with Multimodal Inputs.

After training, our model can handle multimodal input reconstruction, garment generation, and editing, as reflected in the training data. Additionally, it exhibits *zero-shot* reasoning for garment sewing patterns within multi-turn dialogues, despite being trained only on single-turn dialogue datasets. This enables users to create a garment from an image or text description and then refine it within the same interaction. A typical workflow might start with the VLM estimating a sewing pattern from an image. After reviewing the result, the user can request specific adjustments, providing a flexible, user-centered design process (see Fig. 5 for an example).

4. Experiments

We conduct experiments on image-based reconstruction (Sec 4.1), garment editing (Sec 4.2) and text-based garment generation (Sec 4.3), along with ablation studies (Sec 4.4).

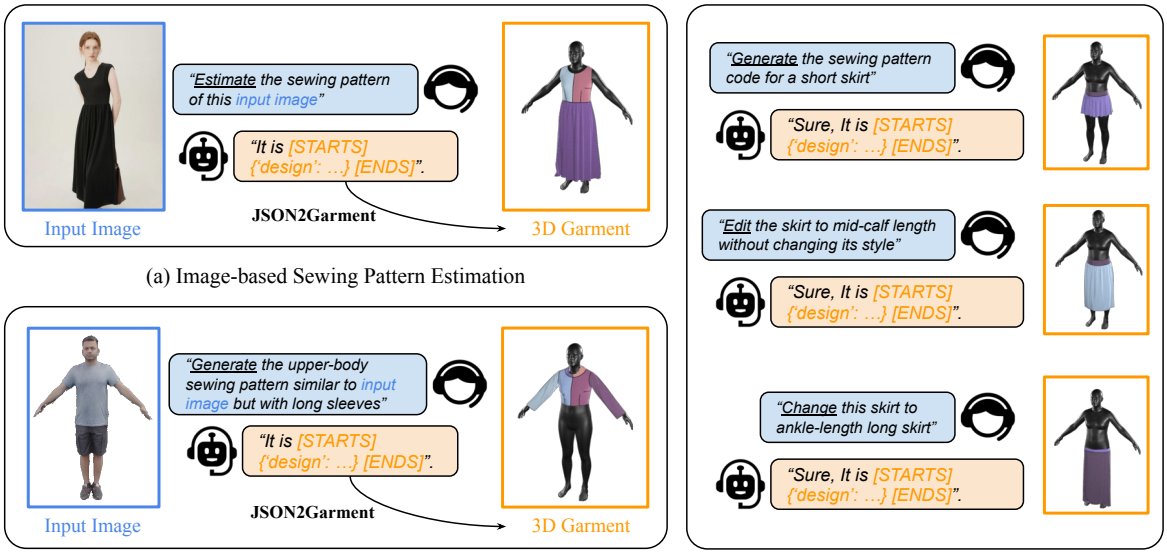
4.1. Image-based Reconstruction

Benchmarks. We conduct image-based reconstruction experiments on the CloSe [2] and Dress4D [54] datasets. CloSe is a large-scale scan collection including 3D clothing segmentations across 18 clothing categories. Following CloSe, we use 145 scans with accurate SMPL-X fits as our validation set. As CloSe offers limited instances of loose garments, such as skirts and dresses, we further add samples from Dress4D, which has real-world outfits recorded using a multi-view volumetric capture system. We include 4 loose-fitting outfits and 36 images rendered from them.

Implementation Details. Our approach applies the Chain-of-Thought (CoT) [55] method. We first prompt ChatGarment to generate a text description of the outfit in the image. The generated text descriptions are combined with the input image to estimate the final garment JSON configuration. More details are provided in *Sup. Mat.*

Baselines. We compare our method with two previous approaches, SewFormer [33] and GarmentRecovery [29]. We use the publicly available code and checkpoints for these methods in our evaluation. Additionally, we introduce several new baseline models for comparison:

- **DressCode**. We employ DressCode [16] to generate garments from image descriptions. Specifically, we utilize GPT-4o to generate garment descriptions from input images. These descriptions are processed by DressCode to create sewing patterns.
- **GPT-4o**. We design prompt templates and query GPT-4o using in-context learning (ICL) [10]. We use Garment Generator [22] as the sewing pattern configuration tool, as it has significantly fewer possible configurations and allows for more efficient template design and significantly fewer conversation rounds. We first prompt GPT-4o to choose the most suitable template from a predefined list, then query the parameter value by displaying three garment renderings with uniformly sampled parameter values.
- **LLaVA***. In this setup, we use the LLaVA model to directly output the numerical values for the sewing pattern instead of using the sewing pattern projection layer. The remaining components are identical to our method.
- **LLaVA-T**. We first employ GPT-4o to generate detailed textual descriptions of the garment, and ChatGarment is prompted to reconstruct the sewing pattern based solely on the text description.
- **LLaVA-I**. We input the image into the model and prompt it to directly output the sewing pattern without referring to text descriptions.



(a) Image-based Sewing Pattern Estimation

(b) Image-guided Sewing Pattern Generation

(c) Text-based Sewing Pattern Generation and Editing

Figure 5. **ChatGarment's dialog modes.** Images and texts are adaptively combined to guide garment generation and editing. Text output between `STARTS` and `ENDS` contains information about the JSON configuration, which is then converted to 3D garments (JSON2Garment) for visualization and simulation purposes.

Methods	Target-Pose		A-Pose		
	CD (↓)	F-Score (↑)	CD (↓)	F-Score (↑)	Failure Rate (↓)
Sewformer [33]	27.06	0.55	20.66	0.58	4.3%
DressCode [16]	20.16	0.60	12.12	0.61	11.4%
GarmentRecovery [29]	9.69	0.55	5.75	0.64	-
GPT-4o	11.04	0.62	10.44	0.65	0
LLaVA*	5.29	0.72	6.29	0.76	0
LLaVA-T	6.17	0.72	7.25	0.74	0
LLaVA-I	4.32	**0.75**	4.93	**0.78**	0
Ours	**3.12**	**0.75**	**3.06**	**0.78**	0

Table 1. **Quantitative comparisons of Image-based garment reconstruction on Dress4D.** ChatGarment achieves the best performance on all metrics.

Methods	CLoSE		
	CD (↓)	F-Score (↑)	Failure Rate (↓)
SewFormer [33]	9.70	0.708	8.33 %
DressCode [16]	15.77	0.616	7.0%
GarmentRecovery [29]	**2.39**	0.785	-
GPT-4o	3.52	0.755	0
LLaVA*	3.08	0.775	0
LLaVA-T	3.76	0.779	0
LLaVA-I	4.55	0.761	0
Ours	2.94	**0.790**	0

Table 2. **Quantitative comparisons of Image-based reconstruction on CloSe.** The garments are deformed to the target pose.

Since methods use different default human poses, we apply Linear Blend Skinning [1] to re-pose all garments to the ground-truth body shape and pose for fair comparison.

Results. We employ the mean chamfer distance, the mean F-Score at $\tau = 0.001$ and the stitching failure rate as quantitative metrics. As Tab. 1 and Tab. 2 show, ChatGarment significantly outperforms existing methods on loose garments in Dress4D. On the CloSE dataset, which mostly features tight garments, GarmentRecovery [29] does well by leveraging garment segmentation and normal maps as guidance in an optimization-based framework. However, these cues are insufficient for accurately reconstructing loose garments. Our method has similar performance in this case, but is more general. Also, our algorithm robustly reconstructs garments with no stitching failures. In contrast, DressCode and SewFormer often generate invalid garment panels leading to failed stitching. The primary difference lies in the output format: DressCode and SewFormer predict the panel edges and stitches of the sewing pattern, while our method and the GPT-4o-based approach estimate the JSON configuration. Although direct edge and stitch prediction offers greater flexibility in theory, our experiments show that it does not enhance reconstruction accuracy. Instead, it increases the likelihood of generating invalid sewing patterns when handling out-of-distribution (OOD) data.

Our model demonstrates superior performance compared to LLaVA*, highlighting the effectiveness of the proposed projection layer. Furthermore, our CoT-based two-step reconstruction algorithm outperforms both image-based (LLaVA-I) and text-based (LLaVA-T) reconstruction baselines. The text descriptions detail garment types and styles, offering valuable information for reconstruction, while images reveal intricate details that are hard to describe in words. Thus, combining both text and image inputs is the most effective strategy for image-based garment reconstruction.

Figure 6 shows our model accurately estimates garment type, style, and details like skirt hems and collar styles. In contrast, other methods often misinterpret or overlook these

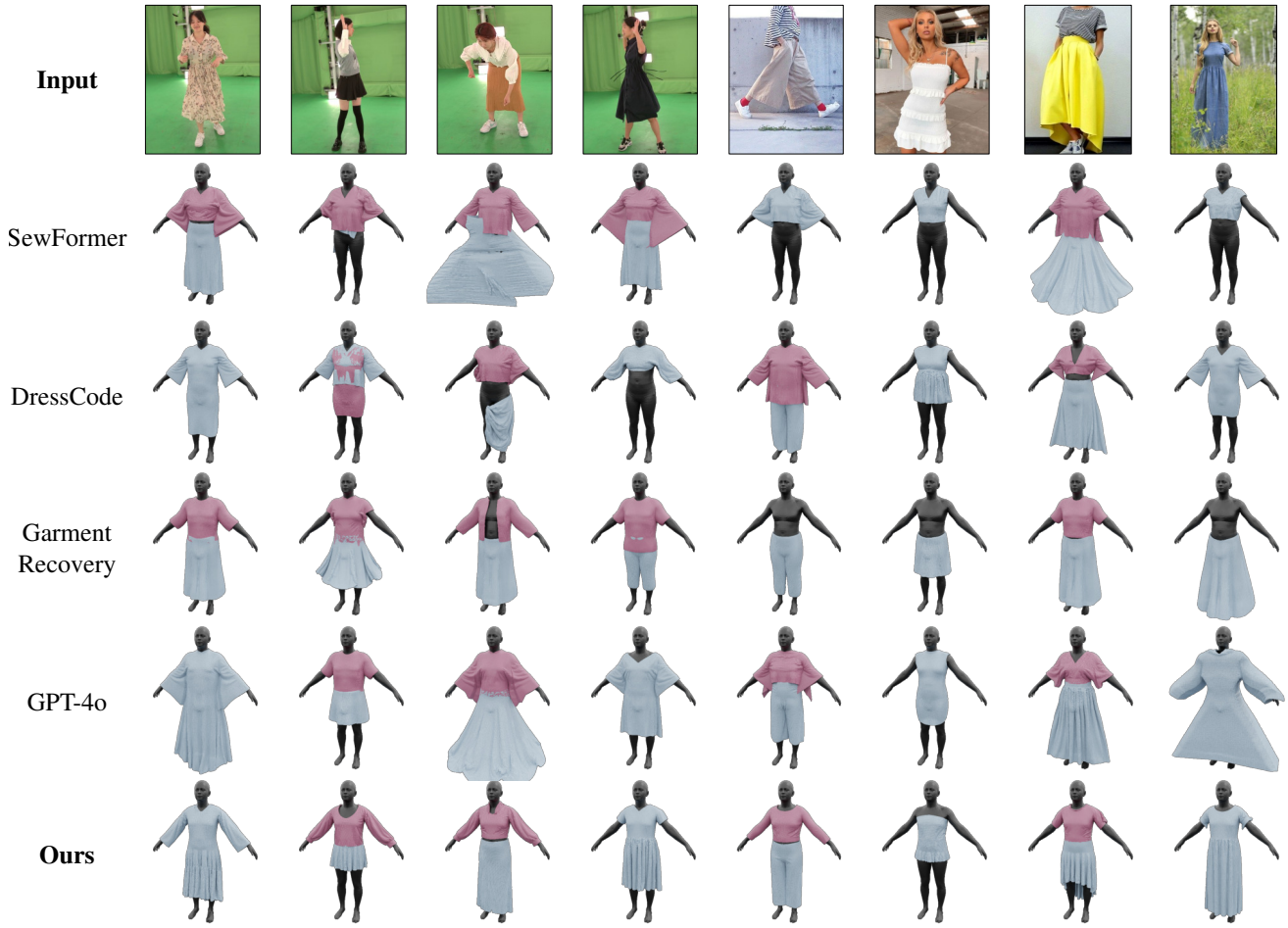


Figure 6. **Image-based Garment Reconstruction.** Unlike SewFormer [33], DressCode [16], GarmentRecovery [29], and the GPT-4o-based method, which often make mistakes or ignores garments. ChatGarment faithfully captures the shape, style and composition of the garments.

details. SewFormer occasionally fails to reconstruct garments due to detection or stitching failure, and GarmentRecovery lacks support for whole-body dress reconstruction. Additionally, inaccuracies in 2D garment segmentation [28] can lead to errors in GarmentRecovery reconstruction.

Garment designers often create image sketches before making sewing patterns. Surprisingly, despite not being trained on garment sketch images, ChatGarment demonstrates strong generalization, accurately predicting garment types and styles from sketches (Fig. 8).

4.2. Garment Editing

Benchmarks. To evaluate garment editing performance, we construct an extra evaluation dataset using methods outlined in Sec. 3.2. We manually select garment pairs with clear textual descriptions and substantial visual differences. This dataset consists of 135 garment pairs, each accompanied by corresponding images and text descriptions. The evaluation of garment editing involves three components: the source garment, the target garment, and the editing prompts. The goal is to derive the sewing pattern for the target garment using the source garment and the editing prompts.

Implementation Details. For garment editing, we concatenate the source garment JSON file and the editing prompt together and pass it into ChatGarment.

Baselines. We modify DressCode [16] and SewFormer [33] for comparison with ChatGarment:

- **DressCode with edited text.** We generate a text description of the edited garment and use DressCode to create the corresponding sewing patterns. The description is produced by GPT-4o, which takes as inputs: 1) garment editing instructions and 2) front and back views of the source garment. The generated description follows DressCode's original input format.
- **SewFormer with edited images.** We obtain the front view of the edited garment with TurboEdit [9] to which we pass 1) the front view of the source garment, 2) the text description of the source garment, and 3) the editing instructions, and then use SewFormer to turn the image of the edited garment into the corresponding sewing pattern.

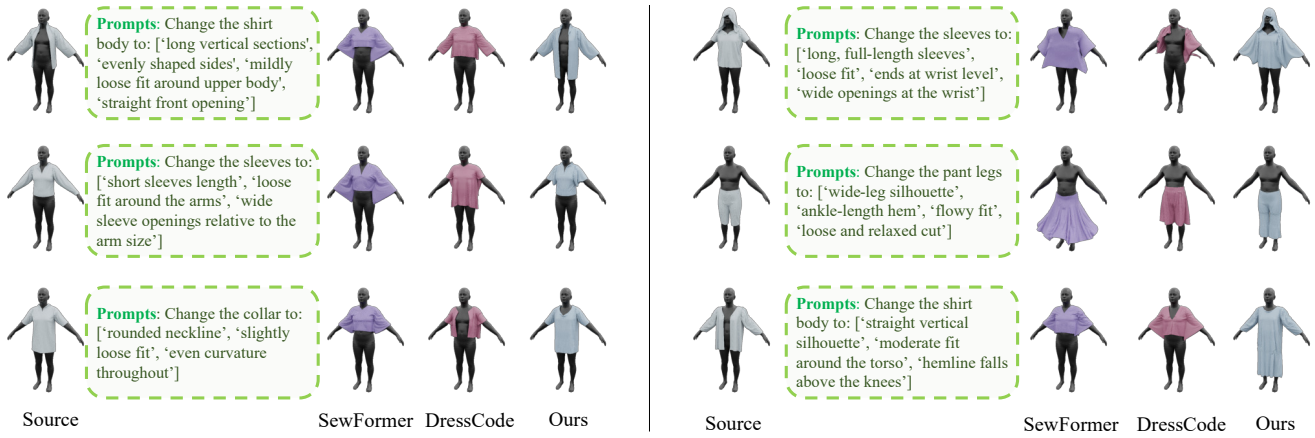


Figure 7. **Results on Garment Editing.** We present six examples where the models (SewFormer [33], DressCode [16], and ChatGarment) edit the source garment shown in the source image according to the given editing instructions in the prompt boxes.

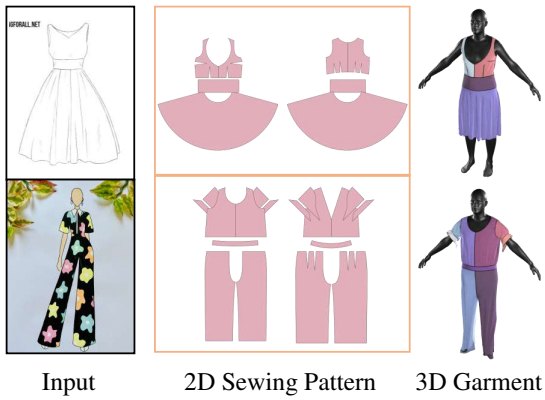


Figure 8. **Reconstruction results from garment sketches.** The generated garments accurately capture the shape and style of the input sketch, despite our model being trained without sketches.

Results. We compare our method with SewFormer and Dresscode on Chamfer Distance and F-score in Tab. 3, where our approach demonstrates superior performance. The qualitative results in Fig. 7 show that our method accurately modifies the targeted garment parts according to the input prompt. In contrast, SewFormer and DressCode often fail to precisely follow the prompt instructions and frequently alter other unintended areas of the garment.

4.3. Text-based Generation

Following DressCode [16], we use GPT-4o to generate 150 highly diverse text labels from 85 selected in-the-wild images featuring various tops, skirts, pants, and dresses. Next, we render the generated 3D garments with default textures and compute the CLIP score using the provided text labels. See quantitative results in Tab. 4. Our model achieves a higher CLIP score compared to DressCode [16], highlighting the superior prompt-following capabilities of our approach. The qualitative results are shown in *Sup. Mat.*

Methods	CD (↓)	F-Score (↑)
SewFormer [33]	28.77	0.548
DressCode [16]	10.97	0.750
Ours	**2.51**	**0.893**

Table 3. **Quantitative comparisons for garment editing.** ChatGarment outperforms other two methods by a large margin.

Methods	CLIP score (↑)
DressCode [16]	22.8
Ours	**23.7**

Table 4. **Quantitative comparisons for text-based garment generation.** ChatGarment achieves a higher CLIP score than DressCode, indicating better alignment with the text inputs.

4.4. Ablation

We evaluate the impact of different aspects of ChatGarment, including multi-modal LLM backbones, and various training datasets. Please refer to *Sup. Mat.*

5. Conclusion

We introduce ChatGarment, a VLM-based model that estimates garments from images and generates or edits garments based on text descriptions. Leveraging the capabilities of LLMs, ChatGarment supports multi-turn dialogue interactions for 3D garment design. It surpasses existing sewing pattern-specific and LLM-based methods in accurately estimating sewing patterns from a single image. While ChatGarment excels in many areas, it may struggle with precise garment editing. For instance, altering the skirt length might unintentionally affect other parts of the garment. This issue could be addressed with more detailed text instructions or a differential CSG optimizer [60] for finer adjustments. Nonetheless, ChatGarment offers the first VLM-based framework for unified garment estimation, generation, and editing. Future directions could include expanding into garment manufacturing design or improving clothing simulation with more fine-grained material attributes.

Acknowledgments and Disclosure

We thank *Maria Korosteleva* for help on GarmentCode, and *Giorgio Becherini* for advice on data acquisition and processing. *Yuliang Xiu* is funded by Max Planck Institute for Intelligent Systems, the Research Center for Industries of the Future (RCIF) at Westlake University, and the Westlake Education Foundation.

While MJB is a co-founder and Chief Scientist at Meshcapade, his research in this project was performed solely at, and funded solely by, the Max Planck Society.

References

[1] Rinat Abdrashitov, Kim Raichstat, Jared Monsen, and David Hill. Robust skin weights transfer via weight inpainting. In *SIGGRAPH Asia 2023 Technical Communications*, pages 1–4. 2023. 6

[2] Dimitrije Antić, Garvita Tiwari, Batuhan Ozcomlekci, Riccardo Marin, and Gerard Pons-Moll. CloSe: A 3D clothing segmentation dataset and model. In *International Conference on 3D Vision (3DV)*, 2024. 2, 5

[3] 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/CVF international conference on computer vision*, pages 5420–5430, 2019. 2

[4] Michael J Black, Priyanka Patel, Joachim Tesch, and Jinlong Yang. Bedlam: A synthetic dataset of bodies exhibiting detailed lifelike animated motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8726–8737, 2023. 4

[5] Andrés Casado-Elvira, Marc Comino Trinidad, and Dan Casas. Pergamo: Personalized 3d garments from monocular video. In *Computer Graphics Forum*, pages 293–304. Wiley Online Library, 2022. 2

[6] Enric Corona, Albert Pumarola, Guillem Alenya, Gerard Pons-Moll, and Francesc Moreno-Noguer. Smplicit: Topology-aware generative model for clothed people. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11875–11885, 2021. 2, 3

[7] R Daněřek, Endri Dibra, Cengiz Öztireli, Remo Ziegler, and Markus Gross. Deepgarment: 3d garment shape estimation from a single image. In *Computer Graphics Forum*, pages 269–280. Wiley Online Library, 2017. 2

[8] 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*, pages 1451–1460, 2023. 2, 3

[9] Gilad Deutch, Rinon Gal, Daniel Garibi, Or Patashnik, and Daniel Cohen-Or. Turboedit: Text-based image editing using few-step diffusion models. *arXiv preprint arXiv:2408.00735*, 2024. 7

[10] Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Tianyu Liu, et al. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*, 2022. 5

[11] Yao Feng, Jing Lin, Sai Kumar Dwivedi, Yu Sun, Priyanka Patel, and Michael J. Black. ChatPose: Chatting about 3d human pose. In *CVPR*, 2024. 2, 3, 4

[12] Jianglin Fu, Shikai Li, Yuming Jiang, Kwan-Yee Lin, Chen Qian, Chen-Change Loy, Wayne Wu, and Ziwei Liu. Stylegan-human: A data-centric odyssey of human generation. *arXiv preprint*, arXiv:2204.11823, 2022. 5

[13] Rao Fu, Xiao Zhan, Yiwen Chen, Daniel Ritchie, and Srinath Sridhar. Shapecrafter: A recursive text-conditioned 3d shape generation model. *Advances in Neural Information Processing Systems*, 35:8882–8895, 2022. 3

[14] Siavash Golkar, Mariel Pettee, Michael Eickenberg, Alberto Bietti, Miles Cranmer, Geraud Krawezik, Francois Lanusse, Michael McCabe, Ruben Ohana, Liam Parker, et al. xVal: A continuous number encoding for large language models. *arXiv preprint arXiv:2310.02989*, 2023. 2, 4

[15] Artur Grigorev, Giorgio Becherini, Michael Black, Otmar Hilliges, and Bernhard Thomaszewski. Contourcraft: Learning to resolve intersections in neural multi-garment simulations. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–10, 2024. 4

[16] Kai He, Kaixin Yao, Qixuan Zhang, Jingyi Yu, Lingjie Liu, and Lan Xu. Dresscode: Autoregressively sewing and generating garments from text guidance. *ACM Transactions on Graphics (TOG)*, 43(4):1–13, 2024. 2, 3, 4, 5, 6, 7, 8

[17] Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3d-llm: Injecting the 3d world into large language models. *Advances in Neural Information Processing Systems*, 36:20482–20494, 2023. 3

[18] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *ICLR*, 2022. 2, 3, 5

[19] Ziniu Hu, Ahmet Iscen, Aashi Jain, Thomas Kipf, Yisong Yue, David A Ross, Cordelia Schmid, and Alireza Fathi. Scenecraft: An llm agent for synthesizing 3d scenes as blender code. In *Forty-first International Conference on Machine Learning*, 2024. 3

[20] Yangyi Huang, Hongwei Yi, Yuliang Xiu, Tingting Liao, Jiaxiang Tang, Deng Cai, and Justus Thies. TeCH: Text-guided Reconstruction of Lifelike Clothed Humans. In *International Conference on 3D Vision (3DV)*, 2024. 3

[21] Boyi Jiang, Juyong Zhang, Yang Hong, Jinhao Luo, Ligang Liu, and Hujun Bao. Bcnet: Learning body and cloth shape from a single image. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16*, pages 18–35. Springer, 2020. 2

[22] Maria Korosteleva and Sung-Hee Lee. Generating datasets of 3d garments with sewing patterns. *arXiv preprint arXiv:2109.05633*, 2021. 5

[23] Maria Korosteleva and Sung-Hee Lee. Neuraltailor: Reconstructing sewing pattern structures from 3d point clouds of garments. *ACM Transactions on Graphics (TOG)*, 41(4):1–16, 2022. 3

[24] Maria Korosteleva and Olga Sorkine-Hornung. GarmentCode: Programming parametric sewing patterns. *ACM Transaction*

on Graphics, 42(6), 2023. SIGGRAPH ASIA 2023 issue. 2, 3

[25] Maria Korosteleva, Timur Levent Kesdogan, Fabian Kemper, Stephan Wenninger, Jasmin Koller, Yuhan Zhang, Mario Botsch, and Olga Sorkine-Hornung. GarmentCodeData: A dataset of 3D made-to-measure garments with sewing patterns. In *Computer Vision – ECCV 2024*, 2024. 2, 3, 4

[26] Peter Kulits, Haiwen Feng, Weiyang Liu, Victoria Abrevaya, and Michael J Black. Re-thinking inverse graphics with large language models. *Transactions on Machine Learning Research*, 2024. 2, 4

[27] Boqian Li, Xuan Li, Ying Jiang, Tianyi Xie, Feng Gao, Huamin Wang, Yin Yang, and Chenfanfu Jiang. Garmentdreamer: 3dgs guided garment synthesis with diverse geometry and texture details. In *3DV*, 2025. 2

[28] Peike Li, Yunqiu Xu, Yunchao Wei, and Yi Yang. Self-correction for human parsing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(6):3260–3271, 2020. 7

[29] Ren Li, Corentin Dumery, Benoît Guillard, and Pascal Fua. Garment recovery with shape and deformation priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1586–1595, 2024. 3, 5, 6, 7

[30] Ren Li, Benoît Guillard, and Pascal Fua. Isp: Multi-layered garment draping with implicit sewing patterns. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 3

[31] Dingning Liu, Xiaoshui Huang, Yuenan Hou, Zhihui Wang, Zhenfei Yin, Yongshun Gong, Peng Gao, and Wanli Ouyang. Uni3d-llm: Unifying point cloud perception, generation and editing with large language models. *arXiv preprint arXiv:2402.03327*, 2024. 3

[32] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning, 2023. 3, 5

[33] Lijuan Liu, Xiangyu Xu, Zhijie Lin, Jiabin Liang, and Shuicheng Yan. Towards garment sewing pattern reconstruction from a single image. *ACM Transactions on Graphics (TOG)*, 42(6):1–15, 2023. 2, 3, 4, 5, 6, 7, 8

[34] Weiyang Liu, Zeju Qiu, Yao Feng, Yuliang Xiu, Yuxuan Xue, Longhui Yu, Haiwen Feng, Zhen Liu, Juyeon Heo, Songyou Peng, Yandong Wen, Michael J. Black, Adrian Weller, and Bernhard Schölkopf. Parameter-efficient orthogonal finetuning via butterfly factorization. In *ICLR*, 2024. 3

[35] Zhen Liu, Yao Feng, Yuliang Xiu, Weiyang Liu, Liam Paull, Michael J. Black, and Bernhard Schölkopf. Ghost on the shell: An expressive representation of general 3d shapes. In *The Twelfth International Conference on Learning Representations*, 2024. 3

[36] Zhongjin Luo, Haolin Liu, Chenghong Li, Wanghao Du, Zirong Jin, Yinyu Nie, Weikai Chen, and Xiaoguang Han. Garverselod: High-fidelity 3d garment reconstruction from a single in-the-wild image using a dataset with levels of details. In *ACM Transactions on Graphics (TOG)*, 2024. 2

[37] Qianli Ma, Jinlong Yang, Siyu Tang, and Michael J. Black. The power of points for modeling humans in clothing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10974–10984, 2021. 3

[38] Qianli Ma, Jinlong Yang, Michael J. Black, and Siyu Tang. Neural point-based shape modeling of humans in challenging clothing. In *International Conference on 3D Vision (3DV)*, pages 679–689, 2022. 3

[39] Miles Macklin. Warp: A high-performance python framework for gpu simulation and graphics. https://github.com/nvidia/warp, 2022. NVIDIA GPU Technology Conference (GTC). 3

[40] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3D hands, face, and body from a single image. In *CVPR*, 2019. 4

[41] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *ICLR*, 2023. 2

[42] Lingteng Qiu, Guanying Chen, Jiapeng Zhou, Mutian Xu, Junle Wang, and Xiaoguang Han. Rec-mv: Reconstructing 3d dynamic cloth from monocular videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4637–4646, 2023. 2

[43] Zeju Qiu, Weiyang Liu, Haiwen Feng, Yuxuan Xue, Yao Feng, Zhen Liu, Dan Zhang, Adrian Weller, and Bernhard Schölkopf. Controlling text-to-image diffusion by orthogonal finetuning. In *NeurIPS*, 2023. 3

[44] Zeju Qiu, Weiyang Liu, Haiwen Feng, Zhen Liu, Tim Z Xiao, Katherine M Collins, Joshua B Tenenbaum, Adrian Weller, Michael J Black, and Bernhard Schölkopf. Can large language models understand symbolic graphics programs? *arXiv preprint arXiv:2408.08313*, 2024. 3

[45] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Hao Li, and Angjoo Kanazawa. PIFu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *ICCV*, pages 2304–2314, 2019. 3

[46] Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. PIFuHD: Multi-Level Pixel-Aligned Implicit Function for High-Resolution 3D Human Digitization. In *CVPR*, pages 81–90, 2020. 3

[47] Nikolaos Sarafianos, Tuur Stuyck, Xiaoyu Xiang, Yilei Li, Jovan Popovic, and Rakesh Ranjan. Garment3dgen: 3d garment stylization and texture generation. In *3DV*, 2025. 2

[48] Tianchang Shen, Jun Gao, Kangxue Yin, Ming-Yu Liu, and Sanja Fidler. Deep marching tetrahedra: a hybrid representation for high-resolution 3d shape synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 3

[49] Yu Shen, Junbang Liang, and Ming C. Lin. Gan-based garment generation using sewing pattern images. In *ECCV*, 2020. 2

[50] Astitva Srivastava, Pranav Manu, Amit Raj, Varun Jampani, and Avinash Sharma. Wordrobe: Text-guided generation of textured 3d garments. In *ECCV*, pages 458–475. Springer, 2025. 3

[51] Zhaoqi Su, Weilin Wan, Tao Yu, Lingjie Liu, Lu Fang, Wenping Wang, and Yebin Liu. Mulaycap: Multi-layer human performance capture using a monocular video camera. *IEEE Transactions on Visualization and Computer Graphics*, 28(4): 1862–1879, 2020. 3

[52] Chunyi Sun, Junlin Han, Weijian Deng, Xinlong Wang, Zishan Qin, and Stephen Gould. 3d-gpt: Procedural 3d modeling with large language models. *arXiv preprint arXiv:2310.12945*, 2023. 3

[53] Tuanfeng Y. Wang, Duygu Ceylan, Jovan Popović, and Niloy J. Mitra. Learning a shared shape space for multimodal garment design. *ACM Trans. Graph.*, 37(6), 2018. 3

[54] Wenbo Wang, Hsuan-I Ho, Chen Guo, Boxiang Rong, Artur Grigorev, Jie Song, Juan Jose Zarate, and Otmar Hilliges. 4D-DRESS: A 4D dataset of real-world human clothing with semantic annotations. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2, 5

[55] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022. 5

[56] Yuliang Xiu, Jinlong Yang, Dimitrios Tzionas, and Michael J. Black. ICON: Implicit Clothed humans Obtained from Normals. In *CVPR*, 2022. 3

[57] Yuliang Xiu, Jinlong Yang, Xu Cao, Dimitrios Tzionas, and Michael J. Black. ECON: Explicit Clothed humans Optimized via Normal integration. In *CVPR*, 2023. 3

[58] Runsen Xu, Xiaolong Wang, Tai Wang, Yilun Chen, Jiangmiao Pang, and Dahua Lin. Pointllm: Empowering large language models to understand point clouds. In *European Conference on Computer Vision*, pages 131–147. Springer, 2025. 3

[59] Zhengming Yu, Zhiyang Dou, Xiaoxiao Long, Cheng Lin, Zekun Li, Yuan Liu, Norman Müller, Taku Komura, Marc Habermann, Christian Theobalt, et al. Surf-d: Generating high-quality surfaces of arbitrary topologies using diffusion models. In *ECCV*, pages 419–438. Springer, 2025. 3

[60] Haocheng Yuan, Adrien Bousseau, Hao Pan, Chengquan Zhang, Niloy J Mitra, and Changjian Li. Diffcsg: Differentiable csg via rasterization. *arXiv preprint arXiv:2409.01421*, 2024. 8

[61] Longwen Zhang, Ziyu Wang, Qixuan Zhang, Qiwei Qiu, Anqi Pang, Haoran Jiang, Wei Yang, Lan Xu, and Jingyi Yu. Clay: A controllable large-scale generative model for creating high-quality 3d assets. *ACM Transactions on Graphics (TOG)*, 43 (4):1–20, 2024. 3

[62] Heming Zhu, Lingteng Qiu, Yuda Qiu, and Xiaoguang Han. Registering explicit to implicit: Towards high-fidelity garment mesh reconstruction from single images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3845–3854, 2022. 2