# GarmageNet: A Multimodal Generative Framework for Sewing Pattern Design and Generic Garment Modeling

SIRAN LI*, Zhejiang Sci-Tech University and Style3D Research
CHEN LIU*, State Key Lab of CAD&CG, Zhejiang University and Style3D Research
RUIYANG LIU†, ZHENDONG WANG, and GAOFENG HE, Style3D Research
YONG-LU LI, Shanghai Jiao Tong University
XIAOGANG JIN, State Key Lab of CAD&CG, Zhejiang University
HUAMIN WANG, Style3D Research

Fig. 1. **GarmageNet** in Action: A diverse and sophisticated collection of garment assets automatically generated by our **GarmageNet** framework, along with their corresponding Garmages—our unified 2D–3D representation that encodes both sewing patterns and detailed geometry for seamless integration with existing garment modeling workflow. Altogether, GarmageNet generates garments across the spectrum of design complexity: from intricate multi-layered ensembles (3rd and 5th) and striking asymmetric styles (2nd and 4th) to form-fitting corsets requiring precise drape and structural fidelity (1st).

Realistic digital garment modeling remains a labor-intensive task due to the intricate process of translating 2D sewing patterns into high-fidelity, simulation-ready 3D garments. We introduce *GarmageNet*, a unified generative framework that automates the creation of 2D sewing patterns, the construction of sewing relationships, and the synthesis of 3D garment initializations compatible with physics-based simulation. Central to our approach is *Garmage*, a novel garment representation that encodes each panel as a structured geometry image, effectively bridging the semantic and geometric gap between 2D structural patterns and 3D garment shapes. *GarmageNet* employs a latent diffusion transformer to synthesize panel-wise geometry images and integrates *GarmageJigsaw*, a neural module for predicting point-to-point sewing connections along panel contours. To support training and evaluation, we build *GarmageSet*, a large-scale dataset comprising over 10,000 professionally designed garments with detailed structural and style annotations. Our method demonstrates versatility and efficacy across multiple application scenarios, including scalable garment generation from multi-modal design concepts (text prompts, sketches, photographs), automatic modeling from raw flat sewing patterns, pattern recovery from unstructured point clouds, and progressive garment editing using conventional instructions-laying the foundation for fully automated, production-ready pipelines in digital fashion. Our code and dataset will be publicly available.

CCS Concepts: • **Computing methodologies** → **Shape modeling**; **Reconstruction**; **Hierarchical representations**; **Shape representations**.

Additional Key Words and Phrases: Garment Modeling, Garment Dataset, Diffusion Generation

## 1 INTRODUCTION

Realistic digital clothing plays a vital role in entertainment and gaming by enhancing character immersion, and in fashion and e-commerce by accelerating product development and reducing costs. Despite this demand, 3D garment modeling—which spans

*Equal contribution and work conducted at Style3D Research.
†Corresponding author.

line-art creation, sewing pattern generation, and physics-based simulation—remains labor-intensive and technically complex. While learning-based methods have made notable strides in 2D design, automating the full pipeline is still challenging due to intricate geometry and expert-dependent tasks such as manual pattern drafting and garment initialization. These slow, skill-intensive processes are poorly suited to the fast fashion industry's need for speed and scalability. As deep neural networks (DNN) continue to advance, a central question emerges: can they truly automate sewing pattern generation and simulation-ready 3D garment modeling?

Garments are constructed from multiple flexible 2D panels joined through sewing patterns that define their final 3D shape, motion, and fit on the human body. The core challenge in digital garment modeling lies in capturing both 3D continuous geometry and 2D discrete structure. These patterns are not just templates. They encode vital semantic information that governs the transformation from 2D fabrics to complex 3D garments. Effective modeling must therefore preserve both 3D geometric integrity for realistic draping and appearance, and 2D structural correctness to maintain sewing relationships. Without a carefully designed representation, enforcing such structural constraints within neural networks can limit their flexibility and compromise geometric fidelity. To be fully effective and efficient, learning-based garment modeling must bridge the gap between 2D structural patterns and 3D garment geometry.

Recent learning-based approaches have made strides in garment modeling but remain constrained by trade-offs between 2D structure and 3D fidelity. *Forward garment modeling* operates in the sewing-pattern domain. They leverage sequential or diffusion-based frameworks to generate either vector-quantized sewing patterns with edge-wise sewing correspondence and rigid-transformation based 3D initialization [He et al. 2024; Li et al. 2025; Liu et al. 2024a; Nakayama et al. 2024], or to emit the parameters and programs of a parametric pattern-making DSL such as GarmentCode [Bian et al. 2024; Korosteleva and Sorkine-Hornung 2023; Zhou et al. 2024]. These methods then employ conventional cloth simulators to drape the generated patterns onto a target avatar. Although they preserve structural correctness by explicitly generating sewing patterns, they lack complete spatial context and therefore often fail to reproduce fine fold details and realistic drape geometry (Figure 2 (b)). In contrast, *Backward garment modeling* [Rong et al. 2024; Tochilkin et al. 2024; Xiang et al. 2024; Yu et al. 2025a; Zhang et al. 2024; Zhao et al. 2025] follows a geometry-first strategy, and mapping multi-modal design inputs directly into a draped 3D garment. These methods often employ continuous, optimizable implicit representations, such as distance or occupancy fields, as their underlying encoding to preserve geometric fidelity. However, because the structural information in UV or sewing pattern space is inherently discrete and discontinuous, making it difficult to integrate into such representations. As a result, these methods discard structural information at the representation level (Figure 2 (c)), rendering it extremely challenging, if not impossible, to recover sewing patterns after generation [Srinivasan et al. 2025; Yu et al. 2024]. These limitations highlight the need for a unified framework that combines the structural integrity of 2D sewing patterns, the geometric precision of 3D drapes, and seamless compatibility with physics-based cloth simulation workflows—precisely the objectives of GarmageNet.

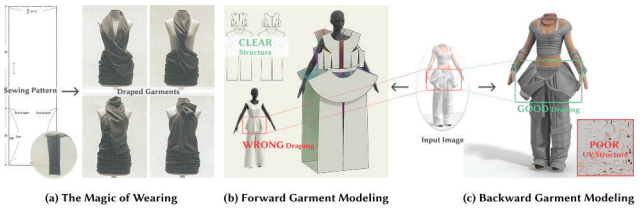



Fig. 2. Problems in forward modeling and backward modeling. The magic of wearing (a) indicates that the same sewing pattern could lead to various draping statuses, raising the problem of **Forward Modeling**, which focuses on sewing pattern structure but fails to ensure draping alignment with the input (b). On the other hand, **Backward Modeling** focuses on draping alignment but fails to preserve structure integrity in UV space (c).

In this paper, we introduce **GarmageNet**, the first unified framework, to our knowledge, that automatically generates 2D sewing patterns, infers sewing relationships, and produces simulation-ready 3D garment initializations. GarmageNet enables a variety of practical applications, including scalable garment generation from multimodal design concepts (text prompts, sketches, photographs), automatic modeling from raw flat sewing patterns, pattern recovery from unstructured point clouds, and progressive garment editing via conventional instructions.

At the core of GarmageNet is a novel garment representation called **Garmage**, designed to bridge the gap between 2D sewing patterns and 3D garment geometry by drawing inspiration from geometry images [Gu et al. 2002]. A Garmage consists of a structured set of per-panel geometry images, where sewing patterns define the UV space, the alpha channel encodes the 2D panel contour, the 2D bounding box captures the length–width ratio, and the 3D bounding box represents the spatial relationship of the panel to the human body. Serving as both a 2D image-based and 3D geometric representation, Garmage offers two key advantages: it retains the flexibility of standard image formats for easy integration with image-based algorithms, and it enables direct reconstruction of simulation-ready 3D garments.

Building upon Garmage, our **GarmageNet** formulates garment synthesis as a latent diffusion process. It first learns a compact manifold of admissible garment panel variations by encoding each cloth piece's quasi-static 3D geometry and corresponding 2D panel shape into a fixed-size latent token. Leveraging this latent space as a strong prior, we then introduce a diffusion transformer (DiT) that learning to produces valid assemblies of these latent tokens, ultimately yielding a complete Garmage capable of precisely delineating detailed 3D garment geometry while preserving panel-wise structural information.

To integrate Garmage into existing garment modeling pipelines, we propose **GarmageJigsaw**, a model for recovering sewing relationships between Garmage panels. Unlike traditional curve-based stitching definitions, GarmageJigsaw defines sewing as point-to-point connectivity along panel boundaries, extracted from Garmage's alpha channel. The model consists of two neural components: a *Point Classifier* to identify stitching points, and a *Stitch Predictor* to infer stitching pairs. Panel contours are then converted into Bézier curves using angle-detecting convolutional kernels to

identify corners, and predicted point-to-point stitches are reformatted into edge-to-edge connections to ensure compatibility with pattern design software.

GarmageNet requires a suitable dataset for training. However, the absence of efficient, flexible garment representations—coupled with a lack of large-scale, high-quality datasets that link 3D garments with 2D sewing patterns—has significantly hindered progress in this domain. To address this, we introduce **GarmageSet**, a large-scale dataset of over 10,000 professionally designed garments, each represented as a Garmage annotated with precise sewing patterns, structural details, and style attributes. GarmageSet not only supports the effective training and evaluation of GarmageNet but also demonstrates its ability to convert additional unstructured sewing patterns into well-organized Garmage representations. This establishes a self-reinforcing feedback loop: newly generated data expands the training corpus, progressively enhancing the quality, diversity, and generalization of GarmageNet's outputs. We summarize our main contributions as follows:

- We introduce *GarmageNet*, a novel uniform generation framework capable of producing complex and simulation-ready garments from various input modalities, including text prompts, sketch images, raw sewing patterns, or unstructured point clouds.
- We introduce *Garmage*, a novel and compact representation that seamlessly encodes a garment's discrete sewing pattern structure and continuous draping geometry into fixed-length latent tokens, facilitating efficient integration with diffusion-based generative models and multi-modal cross-attention conditioning.
- We introduce *GarmageJigsaw* to recover sewing relationships between Garmage panels by predicting point-to-point stitches along panel contours and convert Garmage into production-ready[1] garment assets, facilitating downstream editing of sewing patterns, material properties, and dynamic simulations.
- We introduce *GarmageSet*, an initial dataset of high-fidelity Garmages with detailed structural and style annotations. We further demonstrate that GarmageNet's generation capabilities can expand this dataset, establishing a scalable feedback loop for continuous improvement.
- Our framework surpasses existing forward and backward garment modeling approaches and unlocks a range of practical applications, including scalable multi-modal garment generation (from text, sketches, or photos), automatic 3D reconstruction from flat patterns, sewing-pattern recovery from point clouds, and interactive garment editing with intuitive commands.

## 2 RELATED WORK

In this section, we review advances in garment modeling (Section 2.1), datasets (Section 2.2), and structural object modeling (Section 2.3) that inspired the design of GarmageNet.

[1]We define "production-ready" as assets derived from real-world manufacturing data (in contrast to purely synthetic datasets such as GarmentCode), conforming to industry-grade quality standards, and fully compatible with existing garment production workflows.

### 2.1 Garment Modeling

*Traditional garment modeling* involves complex, labor-intensive steps such as pattern making, sewing identification, and cloth arrangement. Berthouzoz et al. [2013] introduced a machine learning-based sewing identification algorithm, but it requires manual garment initialization and carefully designed parsers for extracting panels and styling elements. Liu et al. [2024d] proposed an automatic initialization algorithm through panel classification and heuristic optimization, but it still relies on complete sewing patterns.

In *learning-based garment modeling*, early work like NeuralTailor [Korosteleva and Lee 2022] focused on reconstructing sewing patterns from unstructured point clouds. Later research evolved into two main approaches: Vector quantization-based methods, which transform sewing patterns into 1D sequences, such as DressCode [He et al. 2024] and SewFormer [Liu et al. 2023], which use GPT and Transformer architectures for text- and image-to-pattern generation. AIpparel [Nakayama et al. 2024] and SewingLDM [Liu et al. 2024a] further advanced tokenization for more complex patterns. Code-generation methods like Design2GarmentCode [Zhou et al. 2024] and ChatGarment [Bian et al. 2024] use large language models (LLMs) to generate parametric pattern-making DSLs (domain-specific language), such as GarmentCode [Korosteleva and Sorkine-Hornung 2023], supporting large-scale dataset generation.

*Implicit garment modeling methods* often rely on unsigned distance fields (UDF) [Yu et al. 2025a], manifold distance fields [Liu et al. 2024b], and Gaussian splatting [Liu et al. 2024c; Rong et al. 2024] to handle non-watertight garment geometry, and employ diffusion or GAN-based generative models to generate visually pleasant garment assets, with vivid dynamics [Rong et al. 2024; Xie et al. 2024]. However, how to transform those implicit representations into triangular or quadrilateral meshes relies on a solid iso-surface extraction algorithm, which remains quite a challenging problem.

With the rise of image generative models, recent approaches [Elizarov et al. 2024; Yan et al. 2024] have used the *geometry image* representation [Gu et al. 2002; Sander et al. 2003] for 3D geometry generation. Although constructing consistent UV spaces is challenging for general objects, these methods work well for garment modeling due to the inherent structure of its well-defined UV space (*i.e.*, sewing patterns) that adhere to industrial standards. For example, ISP [Li et al. 2024a,b] uses geometry images to capture garment deformations, while Yu et al. [Yu and Wang 2024] applied super-resolution to improve fine-grained simulation efficiency.

### 2.2 Garment Datasets

Learning-based 3D garment generation relies on high-quality datasets, which fall into three categories: scanning-based, simulation-based, and sewing pattern-based.

*Scanning-Based Datasets* [Antić et al. 2024; Bhatnagar et al. 2019; Ho et al. 2023; Lin et al. 2023; Ma et al. 2020; Pons-Moll et al. 2017; Tiwari et al. 2020; Wang et al. 2024b; Xu et al. 2023; Zhang et al. 2017] capture realistic garment appearances and shapes; however, isolating semantically meaningful parts from the raw scans remains a labor-intensive process, heavily reliant on manual efforts. As a result, these datasets typically lack sewing patterns that match the garment assets. Additionally, they are mostly derived from existing

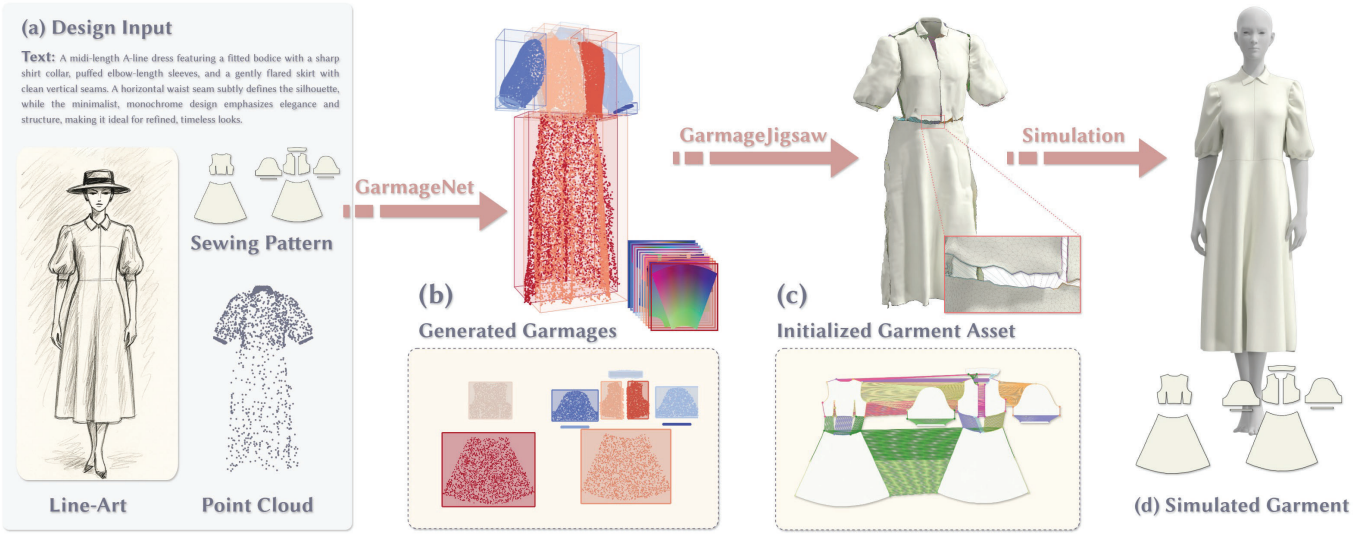



Fig. 3. Overview of our **GarmageNet** framework, which seamlessly converts multi-modal design inputs—including text descriptions, sewing patterns, line-art sketches, and point clouds (a)—into simulation-ready garment assets (d). Central to our framework is the novel **Garmage** representation (b), a unified 2D–3D structure encoding each garment as a structured set of per-panel geometry images. Leveraging **Garmage**, our approach efficiently recovers vertex-level sewing relationships and detailed 3D draping initializations (c), enabling direct and high-quality garment simulation.

commercial garment asset libraries, limiting their scale and design diversity.

*Simulation-based datasets* often include [Bertiche et al. 2020; Black et al. 2023; Gundogdu et al. 2019; Jiang et al. 2020; Narain et al. 2012; Patel et al. 2020; Santesteban et al. 2019; Xiang et al. 2020; Zou et al. 2023], which use physics engines to simulate and enhance the physical plausibility of synthetic 3D garments. While these datasets are more efficient to produce than 3D scanning datasets, they generally suffer from limited garment style diversity, poor garment deformation, and low-quality paired images, reducing their practical use for real-world image data tasks.

Additionally, *sewing pattern-based datasets* [Korosteleva et al. 2024; Korosteleva and Lee 2021] use parametric modeling to create garment models from sewing patterns, offering UV information but often focusing on simpler, single-layer styles. These datasets struggle with representing complex garments due to their lack of multi-layer structures and intricate sewing patterns, limiting their scalability and ability to model detailed, multi-layer garments.

## 2.3 Structural Object Modeling

Recent advancements in structural object modeling have enhanced the generation and reconstruction of Boundary Representation (B-rep) models, enabling more complex 3D shape synthesis for CAD applications. BRepGen [Xu et al. 2024] uses a diffusion-based approach to generate B-rep models hierarchically, capturing intricate geometries, while SolidGen [Jayaraman et al. 2022] employs autoregressive neural networks to predict B-rep components with indexed boundary representation, facilitating high-quality CAD model generation. ComplexGen [Guo et al. 2022] detects geometric primitives and their relationships to create structurally faithful CAD models.

StructureNet [Mo et al. 2019], DPA-Net [Yu et al. 2025b], and TreeSBA [Guo et al. 2025] focus on basic geometric shapes but struggle with non-rigid structures and sewing relationships in garments. StructEdit [Mo et al. 2020] targets local editing of geometric bodies, suitable for regular shapes but limited for flexible, multi-layer garments. 3D Neural Edge Reconstruction [Li et al. 2024c] reconstructs rigid object contours but does not handle flexible garment modeling.

Fracture assembly methods like PuzzleFusion++ [Wang et al. 2024a] and Jigsaw [Lu et al. 2024] infer matching relationships for rigid objects but fail with garments, where misaligned contours and segment-to-segment connections are common. In **Garmage**, we adapt this approach by treating panels as "fractures," predicting relationships between their contour points to establish sewing connections. Unlike rigid objects, garment contours may not align perfectly, and Garmage addresses this by incorporating garment-specific properties, such as curvature, edge smoothness, and sewing constraints in a learning-based framework.

# 3 OVERVIEW

Traditional digital garment modeling demands expert intervention to draft 2D sewing patterns and manually arrange panels around articulated avatars for physics-based simulation. Although learning-based approaches have begun to automate pattern creation, they typically lack explicit 3D geometric guidance, resulting in imprecise outputs and difficulty in handling complex draping behaviors.

Our approach introduces the first unified learning-based garment creation framework built upon a dual 2D/3D representation that embeds both sewing-pattern structure and quasi-static drape geometry in a unified image format. As illustrated in Figure 3, our method accepts multi-modal design inputs, including textual descriptions, design sketches, scanned point clouds, and raw sewing

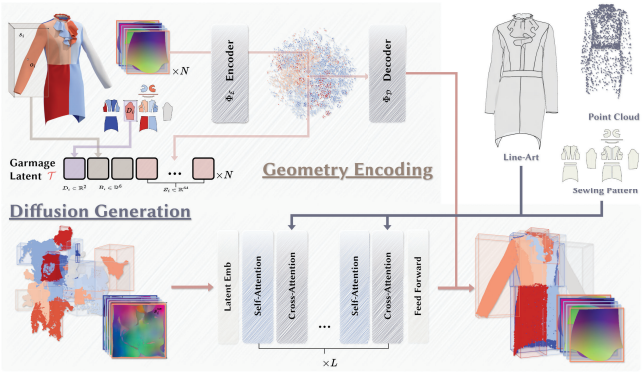



Fig. 4. Overview of our **GarmageNet** architecture. During the *geometry encoding* stage (top), each garment is encoded into a set of fixed-size (72-dimensional) latent vectors using a Variational Autoencoder (VAE). These compact latent representations serve as training targets for the subsequent *diffusion generation* stage (bottom). In the diffusion stage, we employ a diffusion transformer (DiT) denoiser, integrating multi-modal conditions, including line-art sketches, raw sewing patterns, and point clouds via cross-attention mechanisms to effectively guide and control the garment generation process.

patterns, and transfers those design inputs into Garmages with a specially-designed diffusion transformer (DiT).

After generation, we extract 2D sewing contours by thresholding the alpha channel of the generated Garmage and reconstruct each cloth panel's 3D shape by denormalizing the Garmage's RGB channels with the panel's bounding box. To recover stitching topology, our GarmageJigsaw module jointly leverages 2D silhouette features and 3D proximity of contour points, producing point correspondences that we fit with Bézier curves to yield manufacturable seam lines. As a result, our method extracts, from the generated Garmage, the garment's 2D sewing patterns, sewing relationships, and vertex-wise fine-grained initial 3D geometry positioned around a digital avatar—ready for physics-based simulation.

The effectiveness of our framework depends on a comprehensive, multimodal garment corpus, which we call GarmageSet. To support all input pathways, each entry in GarmageSet pairs a ground-truth Garmage (with per-panel geometry images, dimensions, and bounding boxes) with four aligned modalities: a natural-language description, a line-art sketch, a vectorized sewing pattern, and a uniformly sampled point cloud of the draped garment. GarmageSet spans $15k$ outfits across varied styles and categories, and provides the rich supervision necessary to train and evaluate GarmageNet.

## 4 GARMAGENET

While intertwining 3D geometry and 2D structure pose challenges for learning-based garment encoding and generation, it confers a unique advantage: garment assets inherently possess well-structured and semantically meaningful UV spaces (i.e., their sewing patterns). Each texel in a sewing-pattern panel simultaneously maps to a point on its corresponding 3D cloth piece, creating a natural bridge between structure and geometry. Garmage exploits this insight by converting each cloth piece into a panel-aligned geometry image, whose color channels encode the piece's quasi-static 3D geometry and whose alpha channel delineates the panel contour. By harnessing the complementary strengths of 2D and 3D representations, Garmage enables efficient, high-quality 3D garment creation without sacrificing structural fidelity.

### 4.1 The Garmage Representation

Traditionally, a garment asset is represented as a set of 3D cloth pieces $\mathcal{C} = \{C_i\}_{i=1}^N$ and their corresponding 2D sewing pattern panels $\mathcal{P} = \{P_i\}_{i=1}^N$, where each panel in the sewing pattern maps directly to a cloth piece in 3D space.

In Garmage, we model a garment as a set of per-panel geometry images [Gu et al. 2002; Sander et al. 2003], each of which simultaneously encodes a panel's 2D contour and its normalized 3D shape. Formally, we define

$$\mathcal{G} = \{(P_i, C_i)\}_{i=1}^N = \{(D_i, B_i, I_i)\}_{i=1}^N, \tag{1}$$

where $D_i = (h_i, w_i) \in \mathbb{R}^2$ gives the $i$–the panel's physical height and width dimension (in meters) aligned with the fabric's warp and weft; $B_i = (o_i, s_i) \in \mathbb{R}^6$ specifies the axis-aligned bounding box of cloth piece $C_i$, parametrized by its center $o_i \in \mathbb{R}^3$ and half-extents $s_i \in \mathbb{R}^3$; and $I_i \in \mathbb{R}^{H\times W\times 4}$ is a 4-channel image patch whose first three channels encode $C_i$'s geometry normalized by $B_i$ and whose alpha channel delineates the panel contour as an occupancy map.

To construct each image patch $I_i$, we rasterize the cloth piece $C_i$ under its panel $P_i$'s UV parameterization at a uniform resolution ($H = W = 256$). Before rasterization, we rotate the 2D panel $P_i$ so its warp direction aligns with the $v^+$ axis, then normalize its coordinates to $[-1, 1]$ by its physical dimension $D_i = (h_i, w_i)$. Simultaneously, we map every 3D vertex $v_j \in C_i$ into normalized space via $(v_j - o_i)/s_i \in [-1, 1]^3$ using its bounding box $B_i = (o_i, s_i)$. At each pixel center $u_p \in [-1, 1]^2$ (corresponding to a 3D point $p$), we test whether $p$ falls inside any triangle of the cloth piece; if so, we find the containing triangle's vertices $j$ and their barycentric weights $\beta_j(p) \in \mathbb{R}^3$ to $p$, then set

$$I_i(u_p) = \begin{cases} \left(\sum_j \beta_j(u_p) \frac{v_j - o_i}{s_i},\ 1\right), & p \in C_i, \\ (0, 0, 0),\ 0), & \text{otherwise.} \end{cases} \tag{2}$$

Rasterizing panels with sharp features (e.g., dart tips) requires careful handling of boundary aliasing. Following [Yan et al. 2024], we run the rasterization at an initial high $1024 \times 1024$ resolution and subsequently downsample to the target resolution $256 \times 256$ via sparse pooling.

### 4.2 Diffusion-based Garmage Generation

Garment panels serving similar functions often exhibit similar shape features and consistent spatial relationships relative to the human body. For instance, bodice panels typically feature characteristic structural elements such as necklines and armholes and are generally positioned over the chest region in 3D space. This regularity implies a strong correlation between a panel's silhouette (encoded in the alpha channel of $I_i$) and the geometry status of its corresponding

cloth piece (the remaining channels in $I_i$), motivating the compression of Garmages into a unified latent space that simultaneously captures both silhouette and geometry.

*4.2.1 Latent Encoding.* Consider the geometry-image component $I_i$ of the $i$–th panel in a Garamge $\mathcal{G}$, we leverage UNet-based variational autoencoder $\big(\Phi_{\mathcal{E}}(\cdot), \Phi_{\mathcal{D}}(\cdot)\big)$, to compress $I_i$ into a 64-dimensional latent vector

$$\begin{aligned} \Phi_{\mathcal{E}} &: \mathbb{R}^{256\times256\times4} \to \mathbb{R}^{64}, \quad \Phi_{\mathcal{E}}(I_i) = Z_i, \\ \Phi_{\mathcal{D}} &: \mathbb{R}^{64} \to \mathbb{R}^{256\times256\times4}, \quad \Phi_{\mathcal{D}}(Z_i) \approx I_i. \end{aligned} \tag{3}$$

To further reinforce 2D–3D correlation in the latent space, during training we randomly mask out the geometry channels of $I_i$ with probability 0.25 before passing it through the encoder $\Phi_{\mathcal{E}}(\cdot)$, and forcing the decoder $\Phi_{\mathcal{D}}(\cdot)$ to reconstruct the geometry part solely from the panel's silhouette during inference. This masking scheme also enables flexible Garmage generation from raw sewing patterns.

After latent compression, any garment can be represented as a set of fixed-length latent tokens,

$$\mathcal{G} = \mathcal{T} = \big\{T_i\big\}_{i=1}^{N} = \big\{(D_i \oplus B_i \oplus Z_i)\big\}_{i=1}^{N} \in \mathbb{R}^{N\times72} \tag{4}$$

where $N$ represents the number of panels in each garment, $D_i \in \mathbb{R}^2$ represents the 2D physical dimension of the panel, $B_i \in \mathbb{R}^6$ is the axis-aligned bounding box of its corresponding cloth piece and $Z_i \in \mathbb{R}^{64}$ is the geometry latent.

We train the autoencoder with MSE loss to minimize the reconstruction error, along with a low-weighted ($\lambda_{reg} = 1e-6$) KL divergence term between the encoder's approximate posterior $q_{\Phi_E}(z|I)$ and a standard normal distribution $p(z) = \mathcal{N}(0,1)$:

$$\mathcal{L}_{enc} = \frac{1}{N}\sum_{i=1}^{N} \|I_i - \Phi_D(z_i)\|_2^2 + \lambda_{reg} D_{KL}\big[q_{\Phi_E}(z|I_i) \,\|\, p(z)\big]. \tag{5}$$

*4.2.2 Diffusion Generation.* Based on the learned Garmage latent space, we train a diffusion transformer (DiT) to map random samples from the standard normal distribution $\epsilon \backsim \mathcal{N}(0,1)$ to valid Garmages based on various user input conditions $c$. Specifically, in the forward process, we gradually interpolate the input token $\mathcal{T}$ with random noise through $0 \le t \le 1000$ timesteps turning it into noisy states $\mathcal{T}_t = \sqrt{\alpha_t}\mathcal{T}_0 + \sqrt{1-\alpha_t}\epsilon_t$ at each timestep. In the backward process, we linearly embeds the noised latent $\{Z_{i,t}\}_{i=1}^{N}$ into patch tokens, embeds the 2D dimension and 3D bounding box $\{D_{i,t} \oplus B_{i,t}\}_{i=1}^{N}$ into position tokens, and add them together with the embedded timesteps to construct the noisy state, and train the diffusion transformer $\Psi_{\mathcal{G}}(\cdot)$ to recover the added noise from the previous timestep $t-1$ to $t$, conditioned on the input condition $c$:

$$\begin{aligned} &\Psi_{\mathcal{G}} : \mathbb{R}^{N\times72} \to \mathbb{R}^{N\times72}, \quad \Psi_{\mathcal{G}}(\mathcal{T}_t, t, c) \approx \epsilon_t, \\ &\Psi_{\mathcal{G}}(\mathcal{T}_t, t, c) = \mathbf{DiT}\Big(\mathrm{PosEmb}(\mathbf{D}_t \oplus \mathbf{B}_t) + \mathrm{MLP}(\mathbf{Z}_t),\ t, c\Big), \\ &\mathbf{D}_t \oplus \mathbf{B}_t = \{D_{i,t} \oplus B_{i,t}\}_{i=1}^{N} \in \mathbb{R}^{N\times8},\ \mathbf{Z}_t = \{Z_{i,t}\}_{i=1}^{N} \in \mathbb{R}^{N\times64}. \end{aligned} \tag{6}$$

We train $\Psi_{\mathcal{G}}(\cdot)$ using mean-squared error between the predicted noise $\Psi_{\mathcal{G}}(\mathcal{T}_0, c, t)$ and added noise $\epsilon_t$:

$$\mathcal{L}_{\Psi} = \mathbb{E}_{t,\mathcal{T}_0,\epsilon_t}\Big[\|\epsilon - \Psi_{\mathcal{G}}(\mathcal{T}_0, c, t)\|_2^2\Big], \tag{7}$$

where $\epsilon_t$ denotes the Gaussian noise added at timestep $t$. All panels in a Garmage are denoised in parallel while the self-attention mechanism of the Transformer backbone implicitly models the connections between panels, ensuring structural validity of the generated garment. For convenience, we zero-pad each garment to have a fixed number of panels $N = 32$ during training and discarding any panels whose bounding box volume $|B_i| < 0.075$ or 2D dimension $\|D_i\|^2 < 1e^{-4}$ at inference time to accommodate panel number variance.

*4.2.3 Dealing with Conditions.* During diffusion training, each design modality is first encoded into its own latent space by a pretrained encoder and then injected into the Garmage denoiser via cross-attention. *Text prompts* are mapped to a 1024-dimensional text latent using the CLIP text encoder; *Line-art sketches* are passed through a pretrained DINOv2 vision transformer, also yielding a 1024-dimensional image latent; and *unstructured point clouds* are processed by a PointTransformer v3 (PTv3) fine-tuned on GarmageSet for panel segmentation tasks, producing a 1024-dimensional point latent [Wu et al. 2024].

Unlike other modalities, raw sewing pattern conditioned generation is natively supported by GarmageNet via our VAE's masking scheme (Section 4.2.1), which allows for inferring the full 4-channel geometry from the silhouette alone. Consequently, when a *sewing pattern* is provided, its 2D dimensions $\mathbf{D}_0$ and geometry latents $\mathbf{Z}_0$ are known a priori, leaving only the 3D bounding-box $\mathbf{B}_0$ to be recovered via diffusion. In practice, at each diffusion step $t$, we corrupt $\mathbf{D}_0$ and $\mathbf{Z}_0$ to their noised version $\mathbf{D}_t, \mathbf{Z}_t$ at timestep $t$, concatenate them with $\mathbf{B}_t$, and allow the network to iteratively denoise $\mathbf{B}_t$ toward the desired $\mathbf{B}_0$.

It is worth noting that while Garmage inherits the geometry image representation, our panel-wise geometry image representation enables more efficient latent compression. Combined with the carefully designed GarmageNet architecture, which emphasizes the spatial and connectivity relationships between panels, our framework achieves significant improvements in both generation quality and efficiency compared to existing 3D generation methods based on geometry images, such as Omage[Yan et al. 2024] (Table2).

## 5 GARAMGE PROCESSING

With GarmageNet, we can synthesize complete Garmages from conventional design input, the next challenge is to integrate these rasterized panel images into traditional garment-modeling pipelines, which requires vectorizing panel contours and reestablishing sewing relationships. In the following section, we introduce GarmageJigsaw, a dedicated module that leverages Garmage's embedded 2D silhouettes and 3D spatial cues to robustly infer vertex-wise sewing correspondences, followed by post-processing routines that yield production-ready, vector-format sewing patterns.

### 5.1 Boundary Point Sampling

Conventional garment modeling systems define sewing relationships as continuous curve-to-curve correspondences. While straightforward, this edge-based scheme often introduces ambiguity due to ill-defined edge separation and complex many-to-many mappings

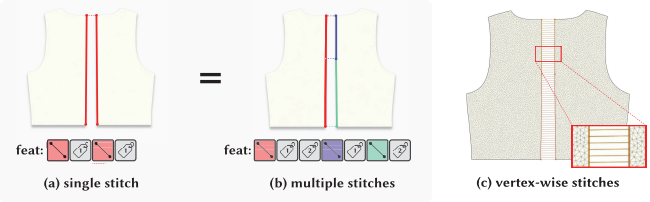



Fig. 5. Illustration of stitch representation ambiguity and our point-wise solution. Existing edge-based methods suffer from inconsistencies due to arbitrary edge splits: in (a) and (b), the red lines depict the same physical stitch, yet their extracted edge features (shown below) differ in both length and parameter encoding. In contrast, our point-wise stitching (c) directly anchors stitch correspondences to mesh vertices in physical space, producing consistent, robust sewing relationships independent of panel tessellation.

(Figure 5). In contrast, we represent sewing relationships as connectivity between boundary vertices of cloth pieces, which are not subject to further subdivision.

As noted above, each panel in Garmage is represented by a four-channel image patch $I_i \in \mathbb{R}^{256\times256\times4}$, whose alpha channel $[I_i]_4$ delineates the panel contour. We extract the set of 2D contour points as:

$$\begin{aligned} &\partial I_i \in \mathbb{R}^{k_i\times2}, \text{ and} \\ &\partial I_i = \big\{ u_p : [I_i]_4(u_p) > 0 \big\} \setminus \big\{ u : \big([I_i]_4 \ominus \Lambda\big)(u_p) > 0 \big\}, \end{aligned} \tag{8}$$

where $\ominus\Lambda$ denotes binary erosion with structuring element $\Lambda$, and $k_i$ refers to the number of contour points from image patch $I_i$. Denoting $[I_i]_{0:3}$ as the remaining geometric channels from $I_i$, we retrieve the corresponding 3D points for $\partial I_i$ from $[I_i]_{0:3}$ and denormalize them into world coordinate with the panel's corresponding bounding box $B_i$:

$$\rho I_i \in \mathbb{R}^{k_i\times3}, \ \rho I_i = \textbf{Denorm}\big( [I_i]_{0:3}(\partial I_i), \ B_i \big). \tag{9}$$

Note that panels may yield nonuniform point densities, we apply resampling under predefined particle distance to $\rho I_i$, ensuring that adjacent contour samples across all panels exhibit consistent 3D distances, producing normalized inputs for our **GarmageJigsaw** correspondence module.

## 5.2 Sewing Relation Recovery

With the resampled contour points, **GarmageJigsaw** recovers point-to-point sewing by jointly leveraging 2D silhouette and 3D geometric features. As shown in Figure 6, we first extract per-point features using two PointNet++ encoders, $\Phi_\rho(\cdot)$ on the 3D contour points $\rho\mathbf{I} \in \mathbb{R}^{K\times3}$ and $\Phi_\partial(\cdot)$ on the 2D pixels $\partial\mathbf{I} \in \mathbb{R}^{K\times2}$. These features are concatenated and fused through a series of point-transformer blocks $\Psi_p(\cdot)$ to yield a 128-dimensional per-point feature matrix

$$\begin{aligned} &f \in \mathbb{R}^{K\times128}, \quad f = \Psi_p\Big( \Phi_\rho(\rho\mathbf{I}) \ \oplus \ \Phi_\partial(\partial\mathbf{I}) \Big), \\ &\rho\mathbf{I} = \{\rho I_i\}_{i=1}^N \in \mathbb{R}^{K\times3} \text{ and } \partial\mathbf{I} = \{\partial I_i\}_{i=1}^N \in \mathbb{R}^{K\times2}. \end{aligned} \tag{10}$$

Here, $K = \sum_i k_i$ is the total number of contour points across all panels. A point classifier head $\Phi_{\text{cls}}(\cdot)$ then selects the subset $f^+ \in \mathbb{R}^{K^+\times128}$ of candidate sewing points by predicting sewing probability based on the point features $f$.

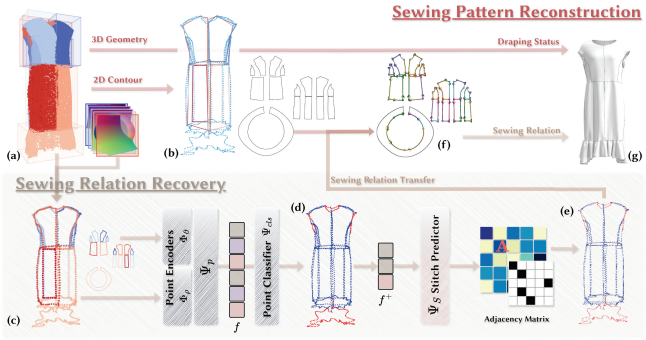



Fig. 6. Overview of sewing relationship recovery and simulation-ready sewing pattern reconstruction from the generated **Garmage** (a). Unlike previous edge-based methods, we predict vertex-level sewing relationships. Specifically, we first sample boundary points (c) from the generated Garmage representation. Our **GarmageJigsaw** takes the boundary points as input, and leverages a point classifier to identify sewing versus non-sewing points (d), followed by a stitch predictor that recovers point-to-point stitches (e), represented as an adjacency matrix. Concurrently, we extract vectorized sewing patterns (b) from the Garmage and transfer the predicted point stitches onto these vectorized patterns (f). We then reconstruct triangle meshes from the vectorized sewing pattern with a Delaunay triangulation constraint by the predicted stitches. Finally, we retrieve vertex-wise draping status from the generated Garmage, leading to a simulation-ready triangle mesh that can be directly integrated into any conventional cloth simulation engine to produce the physically plausible garment (g).

To predict pairwise correspondences, we apply two MLP heads $\Phi_{\text{prime}}(\cdot)$ and $\Phi_{\text{dual}}(\cdot)$ to disentangle primal and dual features:

$$\begin{aligned} &f^+_{\text{prime}} \in \mathbb{R}^{K^+\times128}, \ f^+_{\text{prime}} = \Phi_{\text{prime}}(f^+), \\ &f^+_{\text{dual}} \in \mathbb{R}^{K^+\times128}, \ f^+_{\text{dual}} = \Phi_{\text{dual}}(f^+), \end{aligned} \tag{11}$$

and combine them with a learnable symmetric weight matrix $\Lambda_\mathbf{A} \in \mathbb{R}^{128\times128}$, followed by a Sinkhorn normalization [Cuturi 2013] to produce the adjacency probability matrix:

$$\mathbf{A} \ = \ \textbf{Sinkhorn}\Big(\exp\Big(\frac{(f^+_{\text{prime}})^\top \, \Lambda_\mathbf{A} \, f^+_{\text{dual}}}{\tau}\Big)\Big) \ \in \ [0,1]^{K^+\times K^+}. \tag{12}$$

Here, $A_{ij} \approx A_{j,i}$ denotes the probability of a sewing exists between the $i$-th and $j$-th contour points, and $\tau$ is a temperature parameter according to [Lu et al. 2024]. The probability matrix $\mathbf{A}$ is processed with the Hungarian algorithm [Fischler and Bolles 1981], yielding the final point-to-point correspondences for seam reconstruction.

The entire GarmageJigsaw model is trained end-to-end with two complementary loss terms: a binary cross-entropy loss $\mathcal{L}_{\text{cls}}$ that supervises the predicted sewing-point probabilities against ground-truth labels $y_i \in \{0,1\}$, and a matching loss $\mathcal{L}_{\text{match}}$ that aligns the predicted adjacency matrix $\mathbf{A}$ with the ground-truth matrix $\mathbf{A}_{gt}$. Notably, to prevent the network from trivially minimizing $\mathcal{L}_{\text{mch}}$ by omitting sewing pairs, we pad $\mathbf{A}$ into a $K \times K$ matrix with zero columns and rows corresponding to non-sewing points, and compute the matching loss on the whole contour points set $(\rho\mathbf{I}, \partial\mathbf{I})$.

We train GarmageJigsaw on vertex-wise sewing data where each stitch is represented as a tuple of vertex IDs (Sec. **??**). In our ground-truth assets, sewn vertices are perfectly coincident with zero 3D Euclidean distance. However, the generated Garmages through diffusion often exhibit small seam gaps. To make the network robust to these artifacts, we apply the following data augmentations: First, we inwardly offset each panel's boundary facets toward its centroid by a random distance between 2 mm and 8 mm, transferring the original sewing relationships to these offset boundaries. Next, we introduce anisotropic noise parallel to seam directions at true sewing points, and isotropic noise to all other points along the offset boundary. Finally, we slightly perturb each panel's 3D bounding-box center and scale, as well as its 2D pattern dimensions to compensate for the generated positional noise.

### 5.3 Sewing Pattern Reconstruction

In conventional garment-modeling workflows, sewing patterns are represented as vectorized curves, with sewing relationships explicitly defined between these curve segments. To integrate Garmage-generated results seamlessly into existing pipelines, we must convert the predicted point-to-point sewings into curve-to-curve correspondences and vectorize the panel contours.

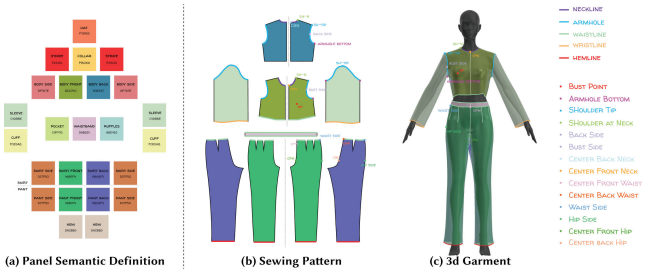
To vectorize the Garmage panel contours, we first detect corner points exhibiting sharp turning angles along the contour point set $\partial\mathbf{I}$ by employing a specially designed 1D convolutional filter. We then fit piecewise B-spline curves to contour points between adjacent corners, resulting in smooth, compact vector representations for each panel. This vectorization process effectively smooths slanted boundaries (e.g., the last panel of the 4−th garment in Figure 1) and fills small noisy holes (e.g., the 2−nd panel of the 5−th garment in Figure 1). Subsequently, we employ a heuristic algorithm to cluster point-to-point stitches predicted by GarmageJigsaw into curve-level sewing correspondences directly on these vectorized B-spline segments.

Finally, we triangulate each sewing-pattern panel into a mesh using constrained Delaunay triangulation [Rognant et al. 1999], guided by the vectorized panel contours and inferred sewing relationships. Specifically, the boundary facets of each cloth piece mesh consist of contour points uniformly resampled according to the sewing correspondences, ensuring smooth and well-aligned seams between adjacent panels.

Vertex positions for these triangulated meshes are determined by sampling the corresponding 3D coordinates from their associated Garmage geometry images using bilinear interpolation, resulting in a fine-grained initial draping state. In contrast to existing garment modeling frameworks that typically rely on coarse rigid transformations to position each panel, our Garmage-based approach provides vertex-level precision in the initial 3D placement. This capability allows us to accurately capture intricate folding behaviors and nuanced garment structures.

## 6 GARMAGESET

As noted, Garmage's vertex-level sewing and precise 3D initialization excel at modeling intricate drapes and folds, whereas existing datasets [Korosteleva et al. 2024; Luo et al. 2024; Zhu et al. 2020] are restricted to simple, flat garments and cannot fully evaluate our framework. To address this gap, we assembled a professionally curated, industrial-grade dataset *GarmageSet* showcasing complex folding behaviors and multi-layer structures, complete with manually validated structural and style annotations, as well as multimodal augmentations including line-art sketches and sampled point clouds.




(a) Panel Semantic Definition (b) Sewing Pattern (c) 3d Garment

Fig. 7. Garment structure definition and corresponding visualization on both sewing pattern space and 3D garments. (a) Color-coded definitions of eight structural (e.g., body front, sleeve) and seven decorative (e.g., pocket, ruffle) panel classes. (b) A sewing-pattern layout annotated by these semantic labels. (c) The corresponding 3D draped garment on the standard avatar, with each panel rendered according to its semantic class.

### 6.1 GarmageSet Construction

GarmageSet comprises $N$ = 14,801 unique garments spanning five major clothing categories like tops (Figure 11 (d,i)), pants, skirts, dresses, outerwears and several minor categories like bras (Figure 11 (d)), vests (Figure 11 (j)), pajamas *etc.* All garments are draped onto an A-posed standard avatar[2] to diminish the geometric variance brought by body sizes and poses.

*6.1.1 Data Acquisition.* Building **GarmageSet** entirely by hand would be prohibitively time-consuming. To scale the dataset construction efficiently, we adopt a component-centric strategy inspired by GarmentCode [Korosteleva and Sorkine-Hornung 2023]. As illustrated in Figure 9, we first construct a structured component library from in-the-wild sewing patterns and then task professional modelers with assembling garments by randomly selecting components, applying design modifications (e.g., adjusting width or length, or adding decorative features), and combining them into complete garments.

To build the component library, we collect a diverse set of raw sewing patterns and engage professional pattern makers to annotate them following the hierarchical garment structure definitions detailed in Sec. 6.1.2. This process yields a well-organized collection of reusable garment parts, categorized by role (e.g., bodice, sleeve, collar) and tagged with stylistic attributes curated by experienced fashion designers and pattern makers[3], as shown in Figure 8.

We then randomly sample valid combinations of components from the library and use QWen3 to propose 1–3 modification instructions for each combination, such as altering silhouette proportions or adding style-specific elements. These modified configurations

[2]Size S mannequin with Asian size 84.
[3]Some style tags and illustrations are adapted from Fashionpedia[Fashionary 2016].

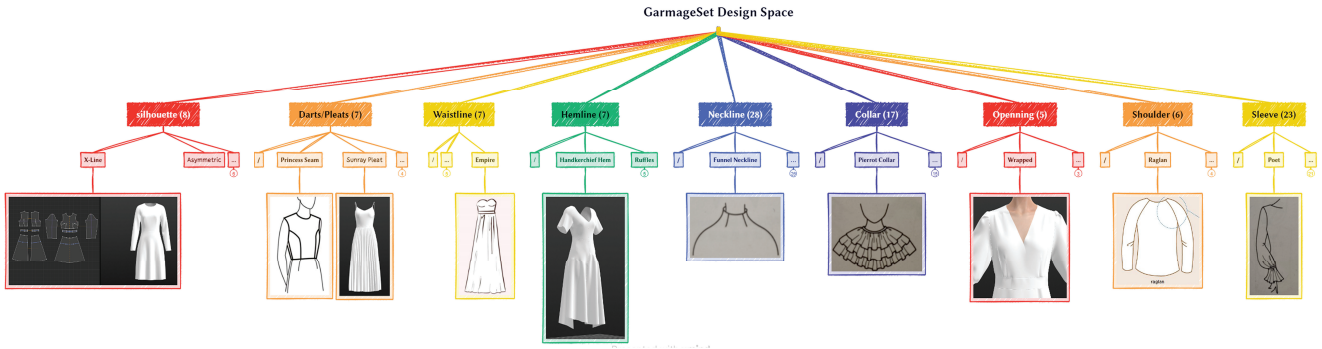



Fig. 8. Design space for GarmentSet. Each garment in GarmageSet is annotated along nine professionally defined design dimensions, including *silhouette* (8 options), *darts/pleats* (7), *waistline* (7), *hemline* (7), *neckline* (28), *collar* (17), *opening* (5), *shoulder* (6), and *sleeve* (23). Except for silhouettes, most of those design dimensions have a "/" option indicating that a particular dimension does not apply to the given garment (e.g., sleeve types are irrelevant for skirts or pants).

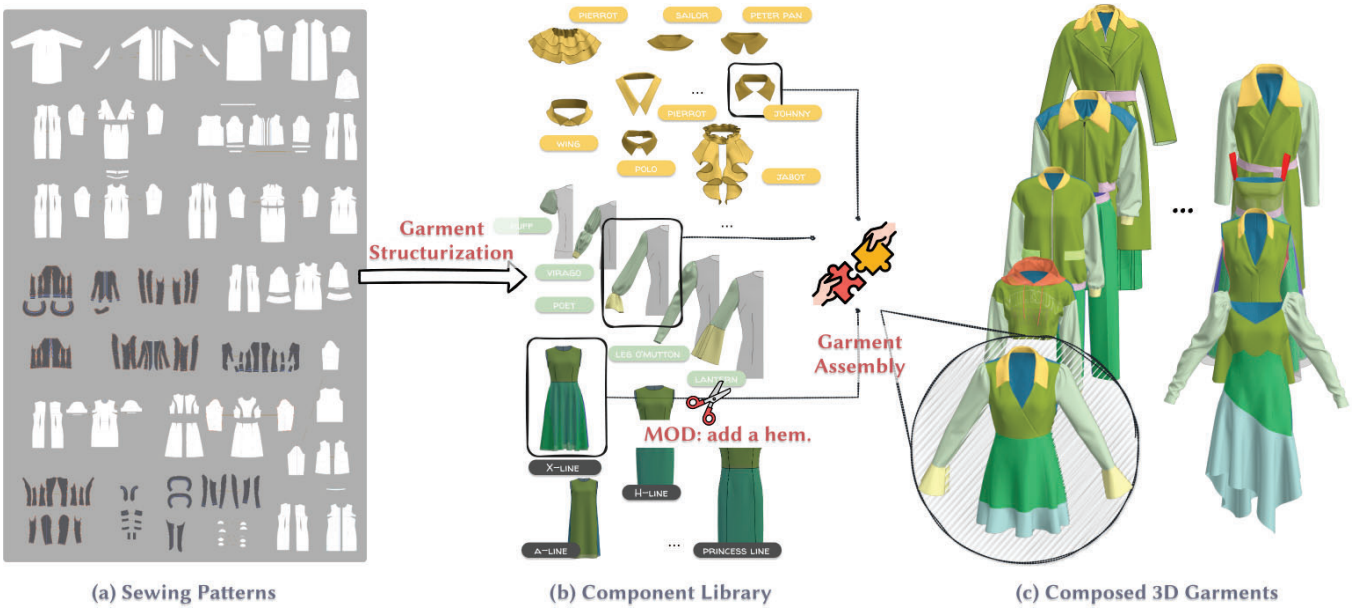



Fig. 9. Overview of our **GarmageSet** construction process. We first build a component library (b) by structuring sewing patterns collected in the wild (a). Professional modelers then randomly select several components from this library, apply design modifications such as adjusting width, length, or adding decorative details, and assemble them to create diverse, composed 3D garments (c). This approach enables efficient construction of a high-quality dataset capturing extensive design variability and structural complexity.

are assigned to professional garment modelers, who manually implement the design changes and assemble the components into finalized 3D garments.

This scalable and structured data acquisition process, carried out over eight months by a team of more than ten expert pattern makers and modelers, resulted in an industrial-scale dataset comprising 2,881 tops, 2,293 outerwear pieces, 857 pants, 1,523 skirts, 6,454 dresses, and 786 garments from other categories such as sportswear, bras, pajamas and cheongsam.

*6.1.2 Garment Structure Definition.* As illustrated in Figure 7, garments exhibit a hierarchical structure comprising panels, edges, and landmarks, each capturing distinct semantic and geometric characteristics essential for garment design and construction. To accurately represent and leverage these hierarchical details, we introduce a structured annotation scheme that clearly defines panel-level semantics, structural lines, and fashion landmarks, as described below.

Panel-level semantics are established by professional pattern makers based on panel shape, functional role, and placement relative to the human body. As shown in Figure 7, we identify eight structural classes—collar, sleeve, body front, body back, body side, skirt/pant front, skirt/pant back, and skirt/pant side—as well as seven decorative classes—hat, stripe, cuff, waist, hem, pocket, and ruffles. Annotators assign these semantic labels directly on the 2D sewing

patterns using a customized LabelStudio [Tkachenko et al. 2025] annotation tool. We leverage this panel-level semantics to finetune Point Transformer v3 for point cloud embedding during conditional Garmage generation.

Utilizing per-panel semantic annotations, we extract five types of **structural lines** that define interfaces between semantic panel groups. The neckline delineates boundaries between front/back bodice and collar panels; armholes separate bodice panels from sleeve panels; waistline defines the interface between waist panels and adjacent bodice or skirt/pant panels (or directly between bodice and skirt/pant panels if waist panels are absent); wristline marks the junction between sleeves and cuffs or the lower edge of sleeves if cuffs are absent; and hemline represents the boundary between bodice/skirt/pant panels and hem panels, or the lower edge of these panels when hem panels do not exist. During training, we augment our dataset by perturbing these structural lines, simulating realistic variations in sleeve length, garment length, waist height, and other key design parameters.

**Fashion landmarks** serve as critical reference points for precise garment construction and fitting. Examples include the shoulder tip (SH), bust point (BP), center front neck (CFN), and center front waist (CFW). These landmarks are annotated on both the 2D sewing patterns and their corresponding 3D models using consistent vertex IDs on the mesh. Such dual annotations help align sewing patterns from different garments into a standardized 2D space, eliminating positional ambiguity and facilitating more effective learning. Additionally, these landmarks are consistently projected onto multi-view 2D images, significantly enriching existing fashion landmark datasets and improving the accuracy of fashion landmark estimation and retrieval models, ultimately offering comprehensive support for diverse fashion AI applications.

*6.1.3 Data Formation And Multi-modal Augmentation.* For each garment, we partition its raw 2D patterns into individual panels $P_i$ and compute their physical dimensions $\mathbf{d}_i$ and axis-aligned bounding boxes $\mathbf{B}_i$. We then rasterize each cloth piece's normalized 3D mesh $C_i$ into a $256\times256\times4$ geometry image $I_i$, and construct the Garmage representation for the garment (Section 4.1).

The original **sewing information** is stored as vertex–vertex pairs $(v_z, v_b)$ in the cloth piece meshes. During Garmege rasterization, each vertex $v$ is projected to a 2D pixel coordinate $u \in \mathbb{R}^2$ in its panel image $I_i$. We record the tuple $(i, u)$, where $i$ is the panel index, and reformat each sewing pair into a paired panel-pixel representation:

$$s_k = \big( (i, u_k^{(a)}),\ (j, u_k^{(b)}) \big), \quad \mathcal{S} = \{s_k\}_{k=1}^{M}, \tag{13}$$

where $s_k$ denotes the $k$-th sewing connecting pixel $u_k^{(a)}$ (rasterized from vertex $v_a$) on the $i-$th panel and $u_k^{(b)}$ (rasterized from vertex $v_a$) on the $j-$th panel.

Furthermore, to train GarmageNet under diverse conditions, we align each Garmage $\mathcal{G}$ with four modalities:

- A manually annotated **short sentence** captures each garment's category, silhouette, and design details according to a set of professionally defined dimensions (Figure 8). During modeling, we ask the designers to label all applicable dimensions for a given garment asset and leverage Qwen3 [Yang et al. 2025] to reformat the annotation as a CLIP-compatible, comma-separated string, with the first segment always denoting the garment category. During training, we randomly delete at most 4 design detail descriptions.
- A set of **line-art sketches and clay renderings** to capture each garment's visual characteristics. These images are rendered from 24 uniformly sampled camera viewpoints arranged on a circle centered on the garment. The circle's radius is automatically adjusted so that, in the frontal view, the garment could nearly fill the frame. All sketches and clay renderings are output at $3840\times2048$ resolution, and we record each camera's transformation matrix in the standard NeRF format.
- A **point cloud** sampled from the garment mesh using Poisson-disk sampling (Open3D) to capture its geometric detail. To closely mimic real-world scans or multi-view reconstructions, which emphasize the exterior surface, we adapt sampling density by occlusion: outer panels are sampled at a high density, while inner panels use a sparser density. We randomly downsample these point clouds at varying rates to improve model robustness and performance.

Table 1. Panel counts (#Panels) and mean average precision (AP) for semantic segmentation by our fine-tuned PointTransformer v3, used to derive point-cloud embeddings for conditional Garmage synthesis. The uniformly high AP values across all categories confirm the model's robustness in extracting panel-level semantics from unstructured point clouds, thereby providing a reliable conditioning signal.

| Category | collar | sleeve | body front | body back | body side |
|---|---|---|---|---|---|
| #Panels | 9807 | 22576 | **34138** | 22608 | 2857 |
| AP | 0.95 | 0.97 | 0.94 | 0.94 | 0.40 |
| **Category** | **skirt/pant front** | **skirt/pant back** | **skirt/pant side** | **hat** | **stripe** |
| #Panels | 28782 | 25242 | 3491 | 2965 | 1142 |
| AP | 0.93 | 0.95 | 0.40 | 0.98 | 0.69 |
| **Category** | **cuff** | **waist** | **hem** | **pocket** | **ruffles** |
| #Panels | 9509 | 16880 | 3088 | 15571 | 2498 |
| AP | **0.98** | 0.96 | 0.79 | 0.93 | 0.42 |

## 6.2 Dataset Statistics

As summarized above, GarmageSet contains 14,801 professionally modeled garments spanning five major categories—tops (2,888), coats and outerwear (2,293), pants (857), skirts (1,523), and dresses (6,454)—plus 786 items in various minor categories. Each garment is annotated along nine professionally-defined design dimensions with over a hundred part-wise variations (Figure 8), yielding a combinatorial design space of more than $2.9454 \times 10^{11}$ topologically distinct configurations. Although smaller in size, GarmageSet covers substantially richer variation than GarmentCodeData [Korosteleva et al. 2024], which is limited to basic modifications (e.g., a single dart type for *FittedShirt* and one lapel style defined in *SimpleLapel*).

To quantify structural complexity, we randomly sample 10,000 garments (and 10,000 panels) from each dataset and compare statistics in Figure 10. GarmageSet garments average $13.59_{\pm7.89}$ panels and $46.01_{\pm26.45}$ per garment, with $8.62_{\pm5.01}$ edges per panel. By contrast, GarmentCodeData provides only $10.82_{\pm6.29}$ panels and


Fig. 10. Dataset statistics comparing GarmageSet and GarmentCodeData [Korosteleva et al. 2024]. Histograms illustrate (a) panels per garment, (b) edges per panel, (c) per garment, (d) mesh vertices per garment, and (e) mesh faces per garment distribution among the 10,000 sampled garments (or 10,000 panels) from both datasets. Dashed lines indicate the mean and standard deviation for each distribution. GarmageSet exhibits higher average values and broader variance across all metrics, indicating enhanced structural complexity and superior drape fidelity.

$30.26_{\pm 17.59}$ per garment, with $6.75_{\pm 4.20}$ edges per panel, indicating significantly lower structural richness. We present the per-category panel count and average segmentation precision in Table 1.

In terms of 3D drape fidelity, by setting the particle distance to 6mm during simulation, GarmageSet features $72{,}560.5_{\pm 27{,}886.7}$ vertices and $141{,}064.1_{\pm 55{,}899.8}$ faces per asset; while GarmentCodeData only has $13{,}998.86_{\pm 8{,}658.42}$ vertices and $26{,}400.82_{\pm 16{,}751.82}$ faces, demonstrating that GarmageSet delivers over fivefold higher mesh resolution and substantially richer structural detail. Figure 11 presents representative samples from GarmageSet, visually demonstrating its high geometric fidelity and intricate structural detail. For example, complex garment foldings and shirrings (c,f); multi-layered design (d), irregular splits (a,i,e,h) that hard to achieve with GarmentCode.

## 7 EXPERIMENTS

In this section, we first detail the implementation and training protocols for *GarmageNet* and *GarmageJigsaw* then quantify our frameworks' performance by evaluating sewing-pattern recovery quality, 3D geometry fidelity, and sewing accuracy.

### 7.1 Implementation Details

We randomly reserved 1,024 garment assets from *GarmageSet* for validation, using the remaining 13,777 assets for training.

GarmageNet was trained on a single NVIDIA A100 GPU over 1–2 days using a two-stage protocol. In the latent-encoding stage, we trained the VAE for 200 epochs with a batch size of 256, using the AdamW optimizer at a learning rate of $5\times10^{-4}$. This stage completes in approximately 2 hours. In the diffusion-generation stage, we employ a standard DDPM scheduler and train the denoiser for 20,000 epochs with a batch size of 4,096, which takes approximately 12 hours. In conditional generation with text prompts or point clouds, we need to incorporate augmentations such as random word dropout in prompts, variable point-cloud sampling densities, and on-the-fly embedding computation. Thus, extends total training time to roughly 24 hours.

We trained GarmageJigsaw using two NVIDIA RTX 4090 GPUs with a batch size of 28. The training was initialized with a learning rate of $1 \times 10^{-3}$, which was gradually decreased using cosine learning rate decay, ultimately reaching $2 \times 10^{-5}$ at the end of the training process. We train our *GarmageJigsaw* for 100 epochs, taking approximately 27 hours in total.

Table 2. Comparison of generation quality, diversity, and efficiency between *GarmageNet*, *Omage*[Yan et al. 2024], and *Surf-D*[Yu et al. 2025a]. Quality metrics include Minimum Matching Distance (MMD, $\times 10^{-3}$), Jensen–Shannon Divergence (JSD), point-cloud FID (p-FID), and point-cloud KID (p-KID), where lower values indicate better fidelity. Diversity is measured by Coverage (COV, %), and efficiency is assessed based on inference GPU memory usage (Mem.) and inference speed (measured in seconds).

| Method | Quality | | | | Diversity | Efficiency | |
|---|---|---|---|---|---|---|---|
| | MMD (↓) | JSD (↓) | p-FID (↓) | p-KID (↓) | COV (↑) | Mem. (↓) | Duration (↓) |
| Surf-D | 21.57 | 0.7907 | 46.61 | 0.1718 | 16.02% | 7 GB | 25.7s |
| Omages | 9.3 | 0.1185 | 29.38 | 0.1271 | 28.16% | 3.3 GB | 120s |
| Ours | **1.1264** | **0.0337** | **15.34** | **0.029** | **41.02%** | 4 GB | 8s |

### 7.2 Evaluation And Comparison

As previously demonstrated, GarmageNet can synthesize complete garment assets, encompassing 2D sewing patterns, sewing correspondences, and high-resolution 3D initializations. Accordingly, we evaluate its generation quality across these core dimensions.

*7.2.1 3D Garment Asset Quality.* We compare GarmageNet's garment generation quality against two representative non-watertight asset synthesis paradigms. The first is Omage [Yan et al. 2024], which typifies geometry-image-based 3D generation pipelines akin to our approach. The second is Surf-D [Yu et al. 2025a], an implicit-field method that generates surfaces via unsigned distance functions. Table 2 presents the comparison results according to five metrics:

- **Minimum Matching Distance (MMD)** measures the average closest-distance between each real sample and its generated counterpart (units of $10^{-3}$). A lower MMD indicates that, on average, every real garment has a very similar counterpart among the generated set.
- **Jensen–Shannon Divergence (JSD)** quantifies the overall distributional discrepancy. A lower JSD means that the probability distributions of real and generated samples are more similar.
- **Point-cloud FID (p-FID)** and **KID (p-KID)** assess generation fidelity using learned feature embeddings, with lower values indicating the generated feature distribution are closer to those of the real data.
- **Coverage (COV)** is the fraction of real samples matched by at least one generated sample (in percentage %). A higher COV indicates broader exploration of the real data manifold, i.e., greater diversity.


Fig. 11. Representative examples from our **GarmageSet**, demonstrating the dataset's rich diversity in garment categories, styles, and intricate folding patterns. Each asset includes detailed 3D garment meshes, corresponding point clouds, multi-view sketches, and Garmage representations, highlighting the dataset's capability to support complex garment modeling tasks, from layered structures and asymmetric silhouettes to precise fitting and sophisticated draping behaviors.


Fig. 12. Unconditional garment generation comparison between **GarmageNet**, **Omage** [Yan et al. 2024], and **Surf-D** [Yu et al. 2025a]. GarmageNet (left block) produces simulation-ready assets complete with vectorized sewing patterns, vertex-wise stitch relationships, and fine-grained 3D draping initializations (a,b,c,d). In contrast, Omage's outputs (top right) exhibit incomplete panels (g), grid-like tessellation artifacts (f), erroneous stitching between non-adjacent panels (h), and spurious triangles that connect a panel's boundary vertices back to the global origin (e). Surf-D's meshes (bottom right) suffer from unwanted holes (i, l) and frayed, irregular boundaries (j, k). These close-up comparisons highlight GarmageNet's superior geometric fidelity, coherent panel topology, and artifact-free mesh integrity.

For a fair comparison, all baseline methods were retrained on the full GarmageSet under the unconditional generation setting. Specifically, Omage [Yan et al. 2024] was trained at a resolution of $64 \times 64$, requiring approximately 50 hours for training and consuming 3.3MB of memory with an inference time of 120 seconds per sample. Surf-D [Yu et al. 2025a] was trained at a resolution of 512, where the VAE module took four days to train on two RTX 4090 GPUs, followed by 20 hours of diffusion model training. To compute point-cloud FID and KID scores, we adopt the pretrained PointNet++ feature extractor provided by Point-E [Nichol et al. 2022]. Each method generated 128 random samples using a single NVIDIA GeForce RTX 3060 for evaluation. As reported in Table 2, GarmageNet outperforms both Omage and Surf-D in terms of generation fidelity, diversity, and computational efficiency.

Figure 12 presents unconditional generation results from GarmageNet alongside those of Surf-D and Omage. Omage produces a single multi-chart geometry image for the entire garment, making its outputs vulnerable to irregular UV chart packing; addressing this requires a much larger network and longer training times. As shown, Omage's results appear coarse and often suffer from missing panels. Surf-D exemplifies a backward modeling approach, using an unsigned distance field (UDF) for generation and then extracting a triangle mesh. Consequently, it generates only a single, monolithic mesh without any explicit sewing-pattern structure, and the UDF-to-mesh conversion can introduce holes. In contrast, GarmageNet delivers panel-aware garments with complete and crisp per-panel structure, and fine-grained draping status.

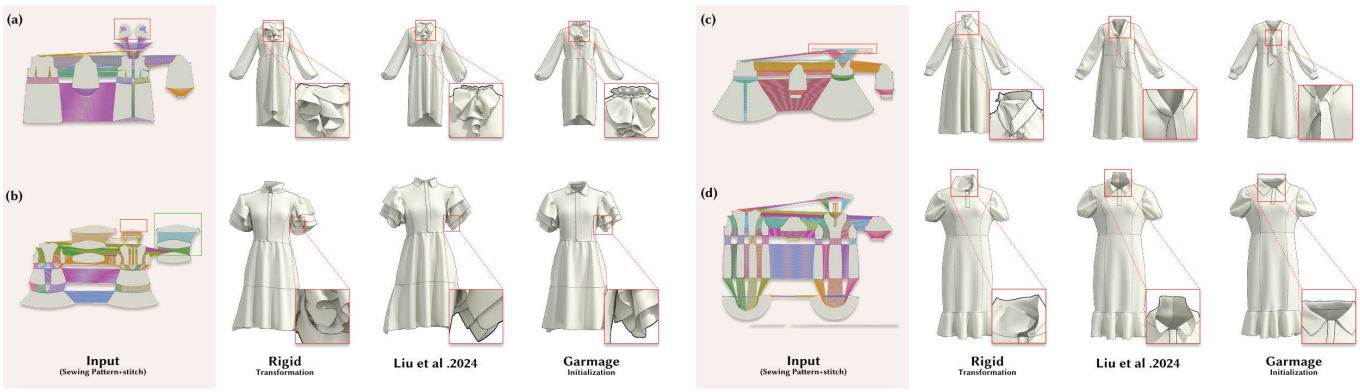



Fig. 13. Qualitative comparison of drape initialization methods on four representative garments: (a) complex overlapping panels, (b) multi-layered ruffles, (c) tie-closure details, and (d) flipped lapels. For each case, we show the input sewing pattern with ground-truth stitches (left) and the draped results under (from left to right) rigid panel transformations, the optimization-based method [Liu et al. 2024d], and our GarmageNet per-vertex initialization.

Table 3. Ablation study on sewing relationship recovery, comparing the performance of **GarmageJigsaw** trained with both 2D and 3D features versus models trained with only 2D or 3D features. The table reports key metrics including point classification precision (CP), recall (CR), average matching distance in millimeters (AMD), topological accuracy (tACC), and topological precision (tP).

| | CP (↑) | CR (↑) | AMD (↓) | tACC (↑) | tP (↑) |
|---|---|---|---|---|---|
| GarmageJigsaw | 99.16 | **97.13** | **6.610** | **96.79** | **98.68** |
| 3D-feat Only | **99.27** | 96.99 | 7.790 | 96.36 | 97.96 |
| 2D-feat Only | 99.21 | 97.12 | 10.59 | 96.28 | 97.70 |

## 7.3 Sewing Pattern Quality

We compared the sewing patterns generated by GarmageNet against those produced by the state-of-the-art forward modeling approach of Zhou et al.[Zhou et al. 2024], which arranges rigidly transformed panels around an avatar. As described in Section 4.2.3, we solicited garments from both methods—conditioned on the same text prompts or sketches—and asked professional pattern makers to evaluate them on two criteria: (1) agreement with the original description or sketch, and (2) the visual quality and smoothness of the panel outlines. The user study and CLIP score in Table 4 shows that GarmageNet's generated panels significantly outperform the baseline in both agreement and aesthetic quality.

*7.3.1 Sewing Accuracy Evaluation.* The *GarmageJigsaw* module, used for sewing recovery, is composed of two key components: the point classifier and the sewing predictor. To thoroughly assess the performance of *GarmageJigsaw*, we evaluate these two modules separately.

The point classifier operates as a binary classifier, and we evaluate its performance using precision and recall. The **classification precision (CP)** measures the proportion of correctly identified sewing points (i.e., true positives) among all predicted positives, while the **classification recall (CR)** indicates the proportion of true positive predictions among all sewing points in the ground truth. As shown in Table 3, the point classifier achieves a precision of 99.16% and a recall of 97.13%, indicating strong performance in identifying sewing points.

For the sewing predictor, we first evaluate the **panel-level** topological quality of the generated sewing patterns with:

- **Accuracy (tACC)**: The proportion of correctly predicted sewing connections (correct sewing pairs) out of all predicted connections. Higher values indicate better topological correctness.
- **Precision (tP)**: The proportion of correctly predicted sewing connections out of all predicted connections, where higher values reflect fewer false positives.

Additionally, we evaluate vertex-level sewing quality using **Average Matching Distance (AMD)**, which calculates the average Euclidean distance between predicted sewing correspondent and ground truth correspondent for all vertices (in millimeters). Lower AMD values indicate better alignment between predicted and actual sewing positions.

Table 3 summarizes the evaluation results with ablation studies on using only 2D or 3D features for sewing relationship recovery. These results confirm that combining both 3D and 2D features enables GarmageJigsaw to achieve more robust stitching recovery with lower AMD value and topological accuracy.

## 7.4 Fine-Grained 3D Initialization Evaluation

To quantify the benefits of GarmageNet's vertex-level initializations, we compare its simulation succession rate (SSR) against two baselines: (1) rigid transformations-based initialization as used in GarmentCodeData [Korosteleva and Sorkine-Hornung 2023]; and (2) optimization-based initialization from raw sewing patterns [Liu et al. 2024d].

We collect 150 sewing patterns with ground truth stitching relationships from GarmageSet, recover their initial drape status with GarmentNet and drape onto our standard Size S avatar using identical simulation settings as [Liu et al. 2024d] and compare the SSR as garments draped successfully onto the avatar without observable self-collision, body-collision, sliding errors *etc.* For the rigid baseline, we leverage the per-panel semantics in GarmageSet (Section 6.1.3)

to assign each panel a fixed pose, using standard rigid placement for body, skirt, and front/back panels, and cylindrical arrangement for tubular components such as sleeves and collars.

As a result, GarmageNet achieves an SSR of 91.41%, substantially higher than rigid initialization (59.38%) and on par with optimization-based initialization (93.75%).

Figure 13 presents representative cases, from which we can conclude that our fine-grained, per-vertex placements could provide robust draping initialization for complex designs (a), multi-layered garment (b), ties (c) and flipping lapels, while the other methods failed.

## 8 APPLICATIONS

We demonstrate the practical versatility and effectiveness of the proposed *GarmageNet* framework through four application scenarios that cover the full spectrum of digital garment modeling. These include interpreting abstract design concepts, automatically generating 3D garment assets from raw sewing patterns, reconstructing manufacturable sewing patterns from unstructured data, and performing conventional garment asset editing based on simple textual inputs. These scenarios showcase GarmageNet's ability to accurately translate diverse inputs into structurally sound and visually compelling garment assets, bridging the gap between creative ideation and real-world garment production.

### 8.1 Design Concept to Garment Generation

Generating garments directly from high-level design concepts, such as textual descriptions or minimalistic line-art sketches, significantly streamlines fashion design workflows, particularly in rapid prototyping and initial visualization stages. Unlike traditional methods that necessitate detailed technical specifications, GarmageNet interprets natural language prompts and simple sketches to automatically produce structurally correct and visually coherent 3D garments.

Qualitative evaluations supported by detailed X-ray renderings and UV-aligned normal maps reveal that GarmageNet effectively captures original design intents. The generated garments exhibit clearly defined seam structures, realistic draping, and well-articulated folds—key elements often compromised in outputs from existing frameworks such as Design2GarmentCode (forward generation) and Hunyuan3D v2.5 (backward generation).

Figure 14, 15 provide qualitative evaluations of garments generated from text prompts and line-art sketches, compared against two state-of-the-art baseline models: the forward generation approach, Design2GarmentCode[Zhou et al. 2024], trained on GarmentCodeData [Korosteleva et al. 2024], and the backward generation method, Hunyuan3D v2.5 [Zhao et al. 2025], trained on massive 3D assets. For each generated garment, we present X-ray renderings to reveal the underlying geometric structures and UV-aligned normal maps to intuitively assess the quality of the generated sewing patterns and the detailed fold structures. Our outputs demonstrate clear and accurate seam structures, precise garment draping, and refined folds which are inadequately represented by the baseline methods.

Leveraging the line-art sketch-conditioned GarmageNet as a baseline, our framework could further enable image-guided garment generation. Specifically, we employ a LoRA fine-tuned FLUX model (Appendix **??**) to translate photographic images into representative line-art sketches. These sketches subsequently guide the Garmage generation process, with results illustrated in Figure 16, underscoring the model's enhanced versatility and real-world applicability.

A comprehensive user study involving 20 professional fashion designers, pattern makers, and 3D apparel modelers validated our findings quantitatively. Each participant is asked to review 48 outputs (24 text-guided and 24 sketch-guided randomly sampled from 1000+ generated results) and select which method's result was best under three criteria:

- **Agreement** with the input prompt (i.e. how well the 3D garment matches the described or drawn design);
- **Garment Aesthetic** (overall visual and geometric quality of the 3D garment model);
- **Sewing Pattern Aesthetic** (quality and plausibility of the underlying pattern structure, as evident in the model and its UV seams).

The aggregated preference results (normalized percentages of selections for each model) in Table 4 indicate GarmageNet significantly outperformed the baselines across all metrics, being preferred in over 60% of cases for Agreement, 85% for Garment Aesthetic, and approximately 90% for Sewing Pattern Aesthetic in text-guided generation; and 77% for Agreement. 68.75%for Garment Aesthetic and 97.66% for Sewing Pattern Aesthetic. Further, GarmageNet achieved the highest normalized CLIPScore (0.3076), confirming superior semantic alignment with text descriptions.

### 8.2 Automatic Garment Modeling

Beyond its broad relevance in virtual reality and gaming, digital garment modeling also plays a critical role in apparel manufacturing by enabling manufacturers to visualize and validate sewing patterns before physical garment production. GarmageNet could naturally support this need by seamlessly converting raw 2D sewing patterns into accurate, fully draped 3D garment models without manual intervention. As discussed earlier, through the masked training scheme during latent encoding (Section 4.2.1), GarmageNet can seamlessly generate complete garment assets from raw sewing patterns, by providing fine-grained 3D initialization through the Garmage representation and establishing vertex-level stitching relationships using GarmageJigsaw.

Figure 17 presents garment generation results from raw sewing patterns. We highlight several of the original sewing pattern panels and their corresponding generated Garmage. The result indicate that GarmageNet effectively handles the task of automatic garment asset modeling based on sewing patterns, even for unconventional patterns like the hem panels in Figure 17 (a,f); Furthermore, despite the absence of explicit symmetry constraints during training, generated garments consistently display natural symmetry, illustrated by the highlighted panels in Figure 17 (a,c,f).

### 8.3 Sewing Pattern Recovery

Advancements in 3D scanning and multi-view reconstruction technologies have greatly facilitated capturing realistic garment shapes, typically represented as unstructured point clouds. However, such raw 3D data lacks the structured information essential for garment


Fig. 14. Text conditioned garment generation results and comparison with Design2GarmentCode [Zhou et al. 2024] and Hunyuan 3D 2.5 [Zhao et al. 2025]

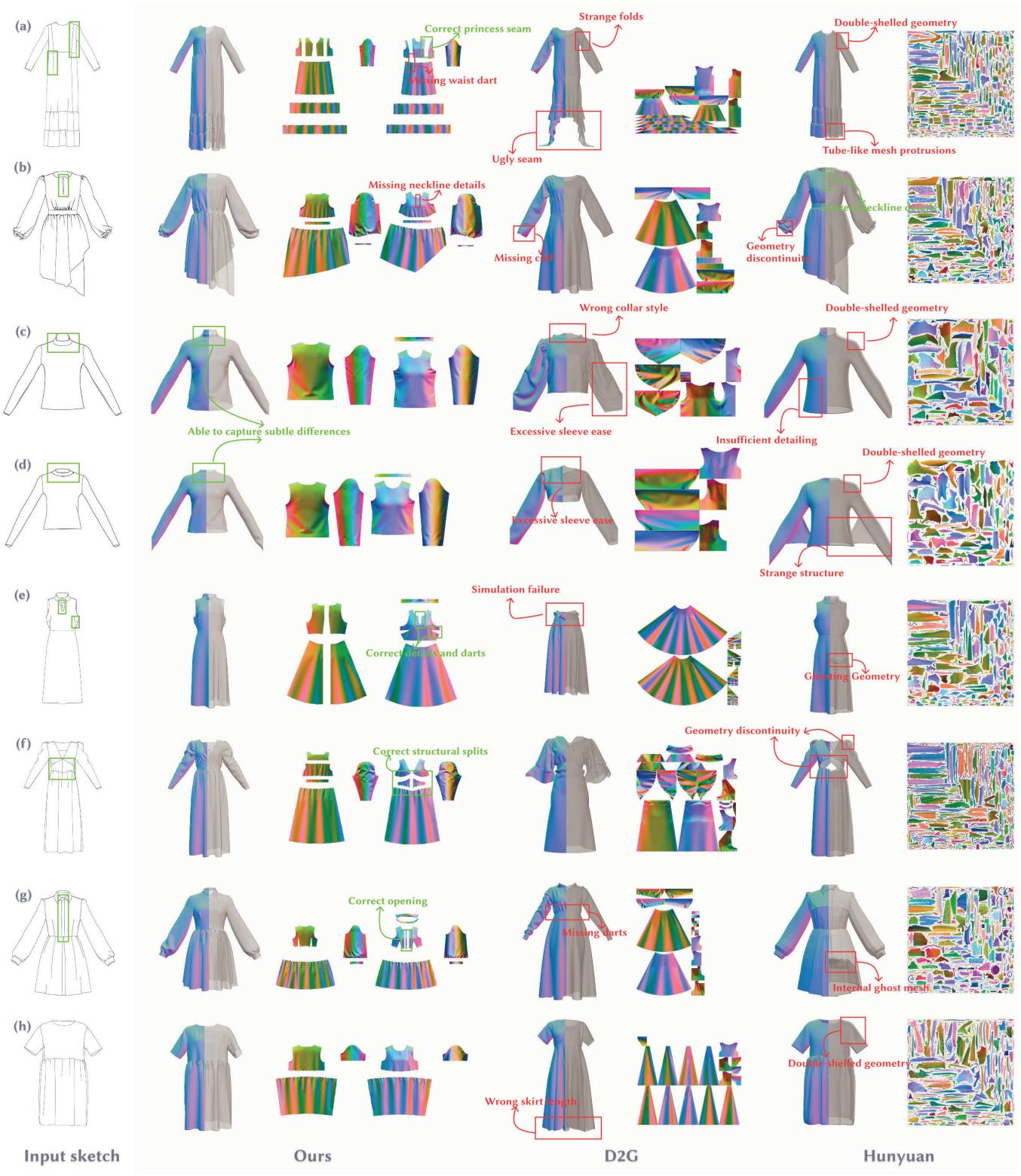



Fig. 15. Line-art guided garment generation results and comparison with Design2GarmentCode [Zhou et al. 2024] and Hunyuan 3D 2.5 [Zhao et al. 2025].

| Method | Text-Guided Generation | | | | Line-Art Guided Generation | | |
|---|---|---|---|---|---|---|---|
| | Agreement | Garment Aesthetic | Sewing Pattern Aesthetic | CLIPScore | Agreement | Garment Aesthetic | Sewing Pattern Aesthetic |
| **GarmageNet + GarmageJigsaw** | **62.50%** | **85.00%** | **90.42%** | **0.3076** | **77.34%** | **68.75%** | **97.66%** |
| Design2GarmentCode [Zhou et al. 2024] | 4.17% | 7.92% | 9.58% | 0.2955 | 0.0% | 10.16% | 2.34% |
| Hunyuan 3D v2.5 [Zhao et al. 2025] | 33.33% | 7.08% | 0.00% | 0.3016 | 22.66% | 21.09% | 0.0% |

Table 4. User study results for generation quality comparison of our method against state-of-the-art (SOTA) forward generation technique Design2GarmentCode (trained on GarmentCodeData), and backward generation technique Hunyuan3D 2.5. Here, *Agreement* evaluates the alignment between the generated garment and the design input (text or line-art sketch). *Garment Aesthetic* evaluates the geometric quality of the generated 3D garment asset, while *Sewing Pattern Aesthetic* evaluates the quality of the generated sewing pattern. We provide CLIPScore as an additional agreement evaluation on text-guided garment generation.


Fig. 16. Image-guided Garmage generation results. We transfer the image to line-art sketches (top right corner) with a LoRA finetuned FLUX model, then use the transferred sketch as input condition to control the generation of Garmage. The generated Garmages and simulated garment assets are demonstrated on the right side.

production, thus necessitating effective methods for recovering structured sewing patterns from unstructured 3D representations.

GarmageNet addresses this critical industry challenge by accurately transforming point-cloud data of draped garments into structured Garmages, successfully recovering detailed sewing patterns. Figure 18 showcases recovered sewing patterns, highlighting intricate folds and precise seam alignments. These recovered patterns closely match their original counterparts, demonstrating high accuracy in panel shapes, seam definitions, and adherence to industry production standards.

Qualitative analysis indicates that GarmageNet robustly identifies precise panel boundaries, seam connections, and garment folds from noisy input data, achieving reliable sewing pattern recovery even in complex garment configurations. This functionality positions GarmageNet uniquely within digital garment pipelines, effectively linking unstructured scan data to structured, production-ready garment assets, thereby significantly enhancing practical applicability in apparel manufacturing workflows.

## 8.4 Progressive Generation and Editing

Beyond generating garments directly from text prompts, our framework also supports advanced garment editing functionalities, such as adding, deleting, or replacing components of an existing garment. This capability significantly enhances the flexibility of the design process, allowing designers to iteratively refine garments based on new inputs while preserving key structural features.

Recall from Eq. 1 that a Garmage consists of a set of panels, each represented by a 2D dimension $D_i$, a 3D axis-aligned bounding box $B_i$, and a normalized geometry image patch $I_i$. After generating a garment using text prompts, users can modify the original prompts to reflect desired changes, such as removing or replacing specific garment components.

When a text prompt is updated, the garment is regenerated, and the newly generated panels are compared against the original panels based on their 2D dimensions and 3D bounding boxes. Panels with high similarity are marked for retention, while those with low similarity are flagged for modification. The editing process is akin to inpainting in image generation models, where only the panels requiring modification are regenerated. Retained panels are treated in a way similar to diffusion-based denoising, where the original features are preserved and augmented with noise according to the current timestep, guiding the model to retain the established characteristics of those panels. In this way, the modifications are localized to the relevant areas of the garment without disrupting the overall design, and new panels are generated in a manner that ensures smooth transitions at the interfaces between modified and retained panels (e.g., sleeve holes), maintaining coherence in both geometry and design.

Figure 19 illustrates the process where we first generate a *fitted, sleeveless dress* from text prompts (a), then add long sleeves to the dress (b), and modify the sleeves to puff sleeves (c). Next, we add

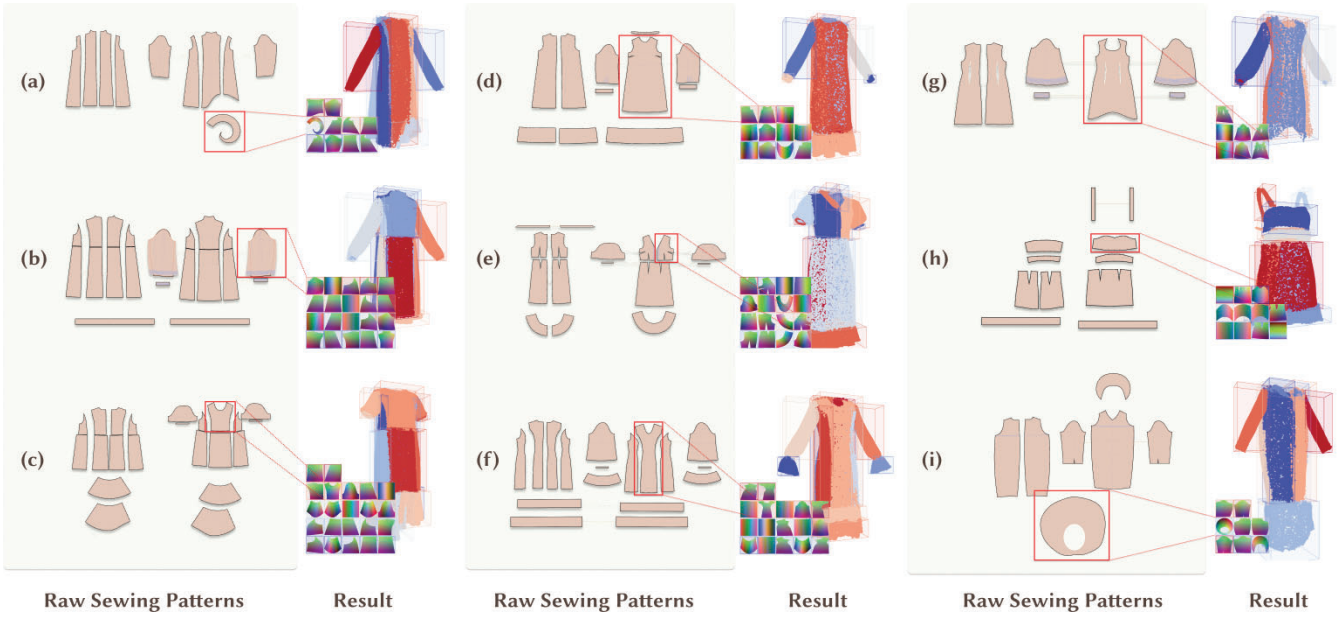



Fig. 17. Automatic garment modeling from raw sewing patterns. Given flat sewing patterns without sewing relationship, For clarity, we highlight specific panels in the sewing patterns to help readers identify the correspondence between the generated Garmage and the raw sewing pattern.

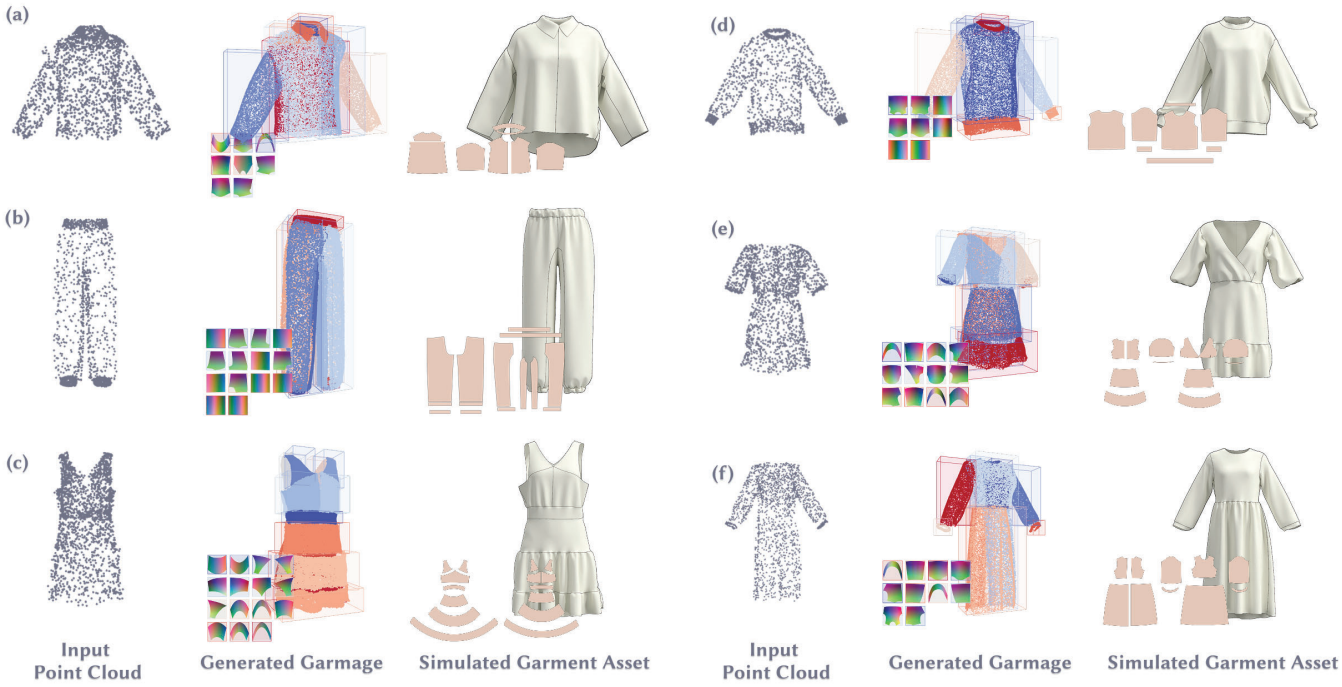



Fig. 18. Point-cloud-conditioned garment synthesis with **GarmageNet**. Each row (a–f) shows: (left) an unstructured, sparse point cloud captured from a draped garment; (center) the generated **Garmage** representation—consisting of per-panel geometry images (colored) and inferred panel contours (outlined); and (right) the final simulation-ready 3D garment asset, obtained by vectorizing the extracted sewing patterns, recovering vertex-wise stitches, and applying physics-based draping. These results demonstrate GarmageNet's ability to transform noisy, incomplete point clouds into fully structured sewing patterns and high-fidelity draped garments. We note, however, that the network may be leveraging the non-uniform sampling density of the input point clouds—implicitly revealing panel structure—to achieve these reconstructions.

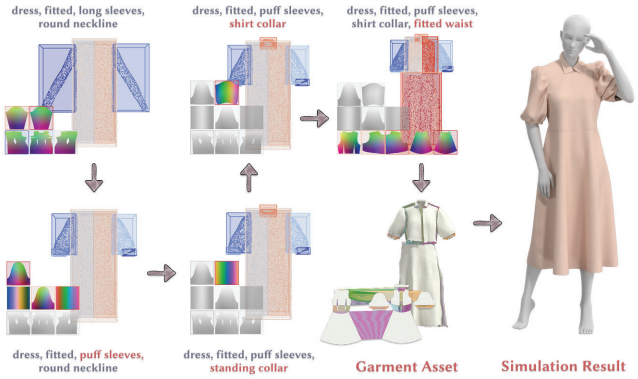



Fig. 19. Interactive garment editing using conventional design instructions. Starting from an initial Garmage (top-left), users issue sequential edits—e.g. replacing a round neckline with a shirt collar, adding a fitted waist, and switching to a standing collar—while all unchanged panels remain in grey and only the edited panels (in color) are updated in their geometry images. Each intermediate Garmage is decoded into a full garment asset and re-simulated, demonstrating how our framework seamlessly incorporates standard pattern-making edits into the generation and draping pipeline.

standing lapel collars to the dress (d) and modify them to shirt collars (e). Finally, if the user is dissatisfied with the generated result, we can even modify the entire initial bodice panels (f).

This progressive generation capability enables dynamic editing of garment designs, allowing for incremental refinement while preserving the structural integrity and stylistic consistency of the garment. The result is a flexible, iterative design process that leverages the power of AI to support real-time garment adjustments and refinements based on evolving design needs.

## 9 CONCLUSION

In this work, we introduced **GarmageNet**, the first end-to-end framework for unified 2D–3D garment synthesis. At its core lies *Garmage*, a novel panel-aligned geometry-image representation that encodes both discrete sewing-pattern structure and continuous draping geometry into a compact, image-based format. By training a latent-diffusion transformer on Garmage tokens, our approach supports unconditional and conditional generation from multiple design modalities—text, sketches, point clouds, and raw sewing patterns—while preserving fine-grained panel topology and delivering high-fidelity, simulation-ready initializations.

We further presented **GarmageJigsaw**, a dedicated module that leverages 2D silhouettes and 3D spatial cues to recover vertex-wise sewing relationships, enabling seamless conversion of generated Garmages into vectorized sewing patterns and triangulated meshes for physics-based simulation. Comprehensive evaluations on our industrial-grade **GarmageSet** demonstrate that GarmageNet outperforms state-of-the-art forward and backward generation methods in terms of quality, diversity, and robustness, and achieves a significantly higher simulation succession rate compared to rigid and optimization-based initializations.

## 10 LIMITATIONS AND FUTURE WORK

While GarmageNet demonstrates robust multimodal garment synthesis, several limitations remain. First, to contain dataset preparation costs, our framework is currently trained on an A-posed standard avatar with size S (or Asian size 84). Although the generated Garmages can be retargeted to other body shapes via existing auto-grading or draping algorithms, we plan to incorporate body-size conditioning and expand our dataset in future work. However, as Garmage could be seamlessly integrated into the existing garment modeling workflow, we will not incorporate body pose variance soon.

Second, our current stitching module operates only along panel boundary facets, limiting its ability to model components that attach along interior seams, such as a patch pocket. Extending the correspondence model to handle arbitrary vertex-to-vertex relationships is an important direction for future research.

Third, because GarmageNet is purely data-driven and does not yet incorporate physical feedback for pattern optimization, the generated panels can self-intersect or interpenetrate, producing simulation artifacts. In future work, we will integrate differentiable physical constraints and physics-based pattern refinement to eliminate these issues.

Fourth, while panel adjacency and symmetry often play a critical role in garment design, these structural priors are learned implicitly by our diffusion transformer. Explicit modeling of symmetry and hierarchical pattern relationships could further improve generation fidelity, which is also an interesting direction for future exploration.

Finally, all training data currently use a single fabric type, so material characteristics such as stiffness, weight, and weave are not yet reflected in the panel shapes or drape. However, the flexibility and scalability of our underlying diffusion-transformer backbone have been demonstrated by recent text-to-image (e.g., FLUX) and text-to-3D (*e.g.,* Tripo, Hunyuan, CLAY) models. As our dataset grows, we will incorporate more fabric types, enabling GarmageNet to model how different materials influence both panel geometry and overall garment drape.

Looking ahead, we plan to extend GarmageNet along several directions. Incorporating body-shape conditioning will allow garment personalization across diverse silhouettes. Integrating differentiable physics into the generation loop can further reduce simulation artifacts and enable material-aware draping. Finally, expanding GarmageSet to cover a wider range of fabrics and decorative techniques will enhance the model's ability to capture nuanced material behaviors and stylistic details. We believe GarmageNet paves the way for rapid, design-driven garment creation and holds promise for applications in virtual try-on, digital fashion design, and automated apparel manufacturing.

## REFERENCES

Dimitrije Antić, Garvita Tiwari, Batuhan Ozcomlekci, Riccardo Marin, and Gerard Pons-Moll. 2024. CloSe: A 3D Clothing Segmentation Dataset and Model. In *2024 International Conference on 3D Vision (3DV)*. IEEE, 591–601.

Floraine Berthouzoz, Akash Garg, Danny M Kaufman, Eitan Grinspun, and Maneesh Agrawala. 2013. Parsing sewing patterns into 3D garments. *Acm Transactions on Graphics (TOG)* 32, 4 (2013), 1–12.

Hugo Bertiche, Meysam Madadi, and Sergio Escalera. 2020. Cloth3d: clothed 3d humans. In *European Conference on Computer Vision*. Springer, 344–359.

Bharat Lal Bhatnagar, Garvita Tiwari, Christian Theobalt, and Gerard Pons-Moll. 2019. Multi-garment net: Learning to dress 3d people from images. In *Proceedings of the IEEE/CVF international conference on computer vision*. 5420–5430.

Siyuan Bian, Chenghao Xu, Yuliang Xiu, Artur Grigorev, Zhen Liu, Cewu Lu, Michael J Black, and Yao Feng. 2024. ChatGarment: Garment Estimation, Generation and Editing via Large Language Models. *arXiv preprint arXiv:2412.17811* (2024).

Michael J. Black, Priyanka Patel, Joachim Tesch, and Jinlong Yang. 2023. BEDLAM: A Synthetic Dataset of Bodies Exhibiting Detailed Lifelike Animated Motion. In *Proceedings IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*. 8726–8737.

Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems* 26 (2013).

Slava Elizarov, Ciara Rowles, and Simon Donné. 2024. Geometry Image Diffusion: Fast and Data-Efficient Text-to-3D with Image-Based Surface Representation. *arXiv preprint arXiv:2409.03718* (2024).

Fashionary. 2016. *Fashionpedia: The Visual Dictionary of Fashion Design*. Fashionary International Limited.

Martin A Fischler and Robert C Bolles. 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM* 24, 6 (1981), 381–395.

Xianfeng Gu, Steven J Gortler, and Hugues Hoppe. 2002. Geometry images. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*. 355–361.

Erhan Gundogdu, Victor Constantin, Amrollah Seifoddini, Minh Dang, Mathieu Salzmann, and Pascal Fua. 2019. Garnet: A two-stream network for fast and accurate 3d cloth draping. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 8739–8748.

Haoxiang Guo, Shilin Liu, Hao Pan, Yang Liu, Xin Tong, and Baining Guo. 2022. Complexgen: Cad reconstruction by b-rep chain complex generation. *ACM Transactions on Graphics (TOG)* 41, 4 (2022), 1–18.

Mengqi Guo, Chen Li, Yuyang Zhao, and Gim Hee Lee. 2025. TreeSBA: Tree-Transformer for Self-Supervised Sequential Brick Assembly. In *European Conference on Computer Vision*. Springer, 35–51.

Kai He, Kaixin Yao, Qixuan Zhang, Jingyi Yu, Lingjie Liu, and Lan Xu. 2024. DressCode: Autoregressively Sewing and Generating Garments from Text Guidance. *ACM Transactions on Graphics (TOG)* 43, 4 (2024), 1–13.

Hsuan-I Ho, Lixin Xue, Jie Song, and Otmar Hilliges. 2023. Learning locally editable virtual humans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 21024–21035.

Pradeep Kumar Jayaraman, Joseph G Lambourne, Nishkrit Desai, Karl DD Willis, Aditya Sanghi, and Nigel JW Morris. 2022. Solidgen: An autoregressive model for direct b-rep synthesis. *arXiv preprint arXiv:2203.13944* (2022).

Boyi Jiang, Juyong Zhang, Yang Hong, Jinhao Luo, Ligang Liu, and Hujun Bao. 2020. 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*. Springer, 18–35.

Maria Korosteleva, Timur Levent Kesdogan, Fabian Kemper, Stephan Wenninger, Jasmin Koller, Yuhan Zhang, Mario Botsch, and Olga Sorkine-Hornung. 2024. GarmentCodeData: A Dataset of 3D Made-to-Measure Garments With Sewing Patterns. In *Computer Vision – ECCV 2024*.

Maria Korosteleva and Sung-Hee Lee. 2021. Generating datasets of 3d garments with sewing patterns. *arXiv preprint arXiv:2109.05633* (2021).

Maria Korosteleva and Sung-Hee Lee. 2022. NeuralTailor: Reconstructing Sewing Pattern Structures from 3D Point Clouds of Garments. *ACM Trans. Graph.* 41, 4 (2022), 16 pages. https://doi.org/10.1145/3528223.3530179

Maria Korosteleva and Olga Sorkine-Hornung. 2023. GarmentCode: Programming Parametric Sewing Patterns. *ACM Transaction on Graphics* 42, 6 (2023), 16 pages. https://doi.org/10.1145/3618351 SIGGRAPH ASIA 2023 issue.

Lei Li, Songyou Peng, Zehao Yu, Shaohui Liu, Rémi Pautrat, Xiaochuan Yin, and Marc Pollefeys. 2024c. 3D Neural Edge Reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 21219–21229.

Ren Li, Corentin Dumery, Benoît Guillard, and Pascal Fua. 2024a. Garment Recovery with Shape and Deformation Priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1586–1595.

Ren Li, Benoît Guillard, and Pascal Fua. 2024b. Isp: Multi-layered garment draping with implicit sewing patterns. *Advances in Neural Information Processing Systems* 36 (2024).

Xinyu Li, Qi Yao, and Yuanda Wang. 2025. GarmentDiffusion: 3D Garment Sewing Pattern Generation with Multimodal Diffusion Transformers. arXiv:2504.21476 [cs.CV] https://arxiv.org/abs/2504.21476

Siyou Lin, Boyao Zhou, Zerong Zheng, Hongwen Zhang, and Yebin Liu. 2023. Leveraging intrinsic properties for non-rigid garment alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 14485–14496.

Chen Liu, Weiwei Xu, Yin Yang, and Huamin Wang. 2024d. Automatic Digital Garment Initialization from Sewing Patterns. *ACM Transactions on Graphics (TOG)* 43, 4 (2024), 1–12.

Lijuan Liu, Xiangyu Xu, Zhijie Lin, Jiabin Liang, and Shuicheng Yan. 2023. Towards Garment Sewing Pattern Reconstruction from a Single Image. *ACM Transactions on Graphics (SIGGRAPH Asia)* (2023).

Shengqi Liu, Yuhao Cheng, Zhuo Chen, Xingyu Ren, Wenhan Zhu, Lincheng Li, Mengxiao Bi, Xiaokang Yang, and Yichao Yan. 2024a. Multimodal Latent Diffusion Model for Complex Sewing Pattern Generation. arXiv:2412.14453 [cs.CV] https://arxiv.org/abs/2412.14453

Yufei Liu, Junshu Tang, Chu Zheng, Shijie Zhang, Jinkun Hao, Junwei Zhu, and Dongjin Huang. 2024c. ClotheDreamer: Text-Guided Garment Generation with 3D Gaussians. arXiv:2406.16815 [cs.CV]

Zhen Liu, Yao Feng, Yuliang Xiu, Weiyang Liu, Liam Paull, Michael J Black, and Bernhard Schölkopf. 2024b. Ghost on the Shell: An Expressive Representation of General 3D Shapes. In *ICLR*.

Jiaxin Lu, Yifan Sun, and Qixing Huang. 2024. Jigsaw: Learning to assemble multiple fractured objects. *Advances in Neural Information Processing Systems* 36 (2024).

Zhongjin Luo, Haolin Liu, Chenghong Li, Wanghao Du, Zirong Jin, Wanhu Sun, Yinyu Nie, Weikai Chen, and Xiaoguang Han. 2024. GarVerseLOD: High-Fidelity 3D Garment Reconstruction from a Single In-the-Wild Image using a Dataset with Levels of Details. *ACM Transactions on Graphics (TOG)* (2024).

Qianli Ma, Jinlong Yang, Anurag Ranjan, Sergi Pujades, Gerard Pons-Moll, Siyu Tang, and Michael J Black. 2020. Learning to dress 3d people in generative clothing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6469–6478.

Kaichun Mo, Paul Guerrero, Li Yi, Hao Su, Peter Wonka, Niloy Mitra, and Leonidas J Guibas. 2019. Structurenet: Hierarchical graph networks for 3d shape generation. *arXiv preprint arXiv:1908.00575* (2019).

Kaichun Mo, Paul Guerrero, Li Yi, Hao Su, Peter Wonka, Niloy J Mitra, and Leonidas J Guibas. 2020. StructEdit: Learning structural shape variations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8859–8868.

Kiyohiro Nakayama, Jan Ackermann, Timur Levent Kesdogan, Yang Zheng, Maria Korosteleva, Olga Sorkine-Hornung, Leonidas Guibas, Guandao Yang, and Gordon Wetzstein. 2024. AIpparel: A Large Multimodal Generative Model for Digital Garments. *Arxiv* (2024).

Rahul Narain, Armin Samii, and James F O'brien. 2012. Adaptive anisotropic remeshing for cloth simulation. *ACM transactions on graphics (TOG)* 31, 6 (2012), 1–10.

Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. 2022. Point-e: A system for generating 3d point clouds from complex prompts. *arXiv preprint arXiv:2212.08751* (2022).

Chaitanya Patel, Zhouyingcheng Liao, and Gerard Pons-Moll. 2020. Tailornet: Predicting clothing in 3d as a function of human pose, shape and garment style. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 7365–7375.

Gerard Pons-Moll, Sergi Pujades, Sonny Hu, and Michael J Black. 2017. ClothCap: Seamless 4D clothing capture and retargeting. *ACM Transactions on Graphics (ToG)* 36, 4 (2017), 1–15.

L Rognant, Jean-Marc Chassery, S Goze, and JG Planes. 1999. The Delaunay constrained triangulation: the Delaunay stable algorithms. In *1999 IEEE International Conference on Information Visualization (Cat. No. PR00210)*. IEEE, 147–152.

Boxiang Rong, Artur Grigorev, Wenbo Wang, Michael J. Black, Bernhard Thomaszewski, Christina Tsalicoglou, and Otmar Hilliges. 2024. Gaussian Garments: Reconstructing Simulation-Ready Clothing with Photorealistic Appearance from Multi-View Video. arXiv:2409.08189 [cs.CV]

Pedro V Sander, Zoë J Wood, Steven Gortler, John Snyder, and Hugues Hoppe. 2003. Multi-chart geometry images. (2003).

Igor Santesteban, Miguel A Otaduy, and Dan Casas. 2019. Learning-based animation of clothing for virtual try-on. In *Computer Graphics Forum*, Vol. 38. Wiley Online Library, 355–366.

Pratul P Srinivasan, Stephan J Garbin, Dor Verbin, Jonathan T Barron, and Ben Mildenhall. 2025. Nuvo: Neural uv mapping for unruly 3d representations. In *European Conference on Computer Vision*. Springer, 18–34.

Garvita Tiwari, Bharat Lal Bhatnagar, Tony Tung, and Gerard Pons-Moll. 2020. Sizer: A dataset and model for parsing 3d clothing and learning size sensitive 3d clothing. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*. Springer, 1–18.

Maxim Tkachenko, Mikhail Malyuk, Andrey Holmanyuk, and Nikolai Liubimov. 2020-2025. Label Studio: Data labeling software. https://github.com/HumanSignal/label-studio Open source software available from https://github.com/HumanSignal/label-studio.

Dmitry Tochilkin, David Pankratz, Zexiang Liu, Zixuan Huang, Adam Letts, Yangguang Li, Ding Liang, Christian Laforte, Varun Jampani, and Yan-Pei Cao. 2024. Triposr: Fast 3d object reconstruction from a single image. *arXiv preprint arXiv:2403.02151* (2024).

Wenbo Wang, Hsuan-I Ho, Chen Guo, Boxiang Rong, Artur Grigorev, Jie Song, Juan Jose Zarate, and Otmar Hilliges. 2024b. 4D-DRESS: A 4D Dataset of Real-World Human Clothing With Semantic Annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 550–560.

Zhengqing Wang, Jiacheng Chen, and Yasutaka Furukawa. 2024a. PuzzleFusion++: Auto-agglomerative 3D Fracture Assembly by Denoise and Verify. *arXiv preprint arXiv:2406.00259* (2024).

Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. 2024. Point Transformer V3: Simpler, Faster, Stronger. In *CVPR*.

Donglai Xiang, Fabian Prada, Chenglei Wu, and Jessica Hodgins. 2020. Monoclothcap: Towards temporally coherent clothing capture from monocular rgb video. In *2020 International Conference on 3D Vision (3DV)*. IEEE, 322–332.

Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. 2024. Structured 3d latents for scalable and versatile 3d generation. *arXiv preprint arXiv:2412.01506* (2024).

Tianyi Xie, Zeshun Zong, Yuxing Qiu, Xuan Li, Yutao Feng, Yin Yang, and Chenfanfu Jiang. 2024. Physgaussian: Physics-integrated 3d gaussians for generative dynamics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4389–4398.

Wenqiang Xu, Wenxin Du, Han Xue, Yutong Li, Ruolin Ye, Yan-Feng Wang, and Cewu Lu. 2023. Clothpose: A real-world benchmark for visual analysis of garment pose via an indirect recording solution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 58–68.

Xiang Xu, Joseph Lambourne, Pradeep Jayaraman, Zhengqing Wang, Karl Willis, and Yasutaka Furukawa. 2024. Brepgen: A b-rep generative diffusion model with structured latent geometry. *ACM Transactions on Graphics (TOG)* 43, 4 (2024), 1–14.

Xingguang Yan, Han-Hung Lee, Ziyu Wan, and Angel X Chang. 2024. An object is worth 64x64 pixels: Generating 3d object via image diffusion. *arXiv preprint arXiv:2408.03178* (2024).

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. 2025. Qwen3 Technical Report. *arXiv preprint arXiv:2505.09388* (2025).

Fenggen Yu, Yiming Qian, Xu Zhang, Francisca Gil-Ureta, Brian Jackson, Eric Bennett, and Hao Zhang. 2025b. Dpa-net: Structured 3d abstraction from sparse views via differentiable primitive assembly. In *European Conference on Computer Vision*. Springer, 454–471.

Jiawang Yu and Zhendong Wang. 2024. Super-Resolution Cloth Animation with Spatial and Temporal Coherence. *ACM Transactions on Graphics (TOG)* 43, 4 (2024), 1–14.

Xin Yu, Ze Yuan, Yuan-Chen Guo, Ying-Tian Liu, Jianhui Liu, Yangguang Li, Yan-Pei Cao, Ding Liang, and Xiaojuan Qi. 2024. TEXGen: a Generative Diffusion Model for Mesh Textures. *ACM Trans. Graph.* 43, 6, Article 213 (2024), 14 pages. https://doi.org/10.1145/3687909

Zhengming Yu, Zhiyang Dou, Xiaoxiao Long, Cheng Lin, Zekun Li, Yuan Liu, Norman Müller, Taku Komura, Marc Habermann, Christian Theobalt, et al. 2025a. Surf-D: Generating High-Quality Surfaces of Arbitrary Topologies Using Diffusion Models. In *European Conference on Computer Vision*. Springer, 419–438.

Chao Zhang, Sergi Pujades, Michael J Black, and Gerard Pons-Moll. 2017. Detailed, accurate, human shape estimation from clothed 3D scan sequences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4191–4200.

Longwen Zhang, Ziyu Wang, Qixuan Zhang, Qiwei Qiu, Anqi Pang, Haoran Jiang, Wei Yang, Lan Xu, and Jingyi Yu. 2024. CLAY: A Controllable Large-scale Generative Model for Creating High-quality 3D Assets. *ACM Trans. Graph.* 43, 4, Article 120 (July 2024), 20 pages. https://doi.org/10.1145/3658146

Zibo Zhao, Zeqiang Lai, Qingxiang Lin, Yunfei Zhao, Haolin Liu, Shuhui Yang, Yifei Feng, Mingxin Yang, Sheng Zhang, Xianghui Yang, et al. 2025. Hunyuan3d 2.0: Scaling diffusion models for high resolution textured 3d assets generation. *arXiv preprint arXiv:2501.12202* (2025).

Feng Zhou, Ruiyang Liu, Chen Liu, Gaofeng He, Yong-Lu Li, Xiaogang Jin, and Huamin Wang. 2024. Design2GarmentCode: Turning Design Concepts to Tangible Garments Through Program Synthesis. *arXiv preprint arXiv:2412.08603* (2024).

Heming Zhu, Yu Cao, Hang Jin, Weikai Chen, Dong Du, Zhangye Wang, Shuguang Cui, and Xiaoguang Han. 2020. Deep fashion3d: A dataset and benchmark for 3d garment reconstruction from single images. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*. Springer, 512–530.

Xingxing Zou, Xintong Han, and Waikeung Wong. 2023. CLOTH4D: A Dataset for Clothed Human Reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12847–12857.