

Automatic Digital Garment Initialization from Sewing Patterns

CHEN LIU, State Key Lab of CAD and CG, Zhejiang University, China and Style3D Research, China

WEIWEI XU, State Key Lab of CAD and CG, Zhejiang University, China

YIN YANG, The University of Utah, USA and Style3D Research, USA

HUAMIN WANG, Style3D Research, China

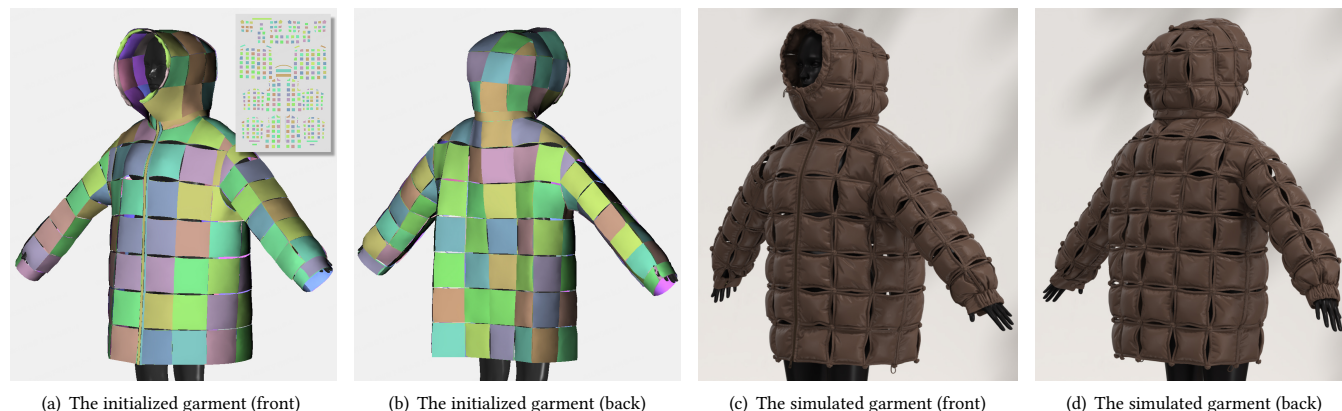


Fig. 1. A quilted coat example with 358 pattern pieces. By providing the sewing pattern and the sewing relationships for this garment, as depicted in the inset of (a), our system reliably and efficiently calculates an initialization with no folding or intersection in 17 seconds, as shown in (a) and (b). This initial setup forms the basis for a followup physics-based simulator, effortlessly producing visually stunning simulations in (c) and (d). Without our system, users would encounter substantial challenges in preparing a garment of this complexity for simulation.

The rapid advancement of digital fashion and generative AI technology calls for an automated approach to transform digital sewing patterns into well-fitted garments on human avatars. When given a sewing pattern with its associated sewing relationships, the primary challenge is to establish an initial arrangement of sewing pieces that is free from folding and intersections. This setup enables a physics-based simulator to seamlessly stitch them into a digital garment, avoiding undesirable local minima. To achieve this, we harness AI classification, heuristics, and numerical optimization. This has led to the development of an innovative hybrid system that minimizes the need for user intervention in the initialization of garment pieces. The seeding process of our system involves the training of a classification network for selecting seed pieces, followed by solving an optimization problem to determine their positions and shapes. Subsequently, an iterative selection-arrangement procedure automates the selection of pattern pieces and employs a phased initialization approach to mitigate local minima associated with numerical

optimization. Our experiments confirm the reliability, efficiency, and scalability of our system when handling intricate garments with multiple layers and numerous pieces. According to our findings, 68 percent of garments can be initialized with zero user intervention, while the remaining garments can be easily corrected through user operations.

CCS Concepts: • **Computing methodologies** → **Physical simulation**.

Additional Key Words and Phrases: Physics-based cloth simulation, numerical optimization, digital fashion, local minima, sewing pattern

ACM Reference Format:

Chen Liu, Weiwei Xu, Yin Yang, and Huamin Wang. 2024. Automatic Digital Garment Initialization from Sewing Patterns. *ACM Trans. Graph.* 43, 4, Article 74 (July 2024), 12 pages. <https://doi.org/10.1145/3658128>

1 INTRODUCTION

With the rise of digital fashion businesses and the progress in generative AI models, digital sewing patterns have become more accessible and affordable. This advancement raises an intriguing question: how can we effortlessly convert digital sewing patterns into well-fitted digital garments on human avatars, all through a fully automated process? This capability is in high demand for a range of digital fashion and entertainment applications, as it forms a key component in the automated creation of 3D garments and characters.

Unfortunately, when presented with a sewing pattern and its sewing relationships, cloth simulation often grapples with non-uniqueness, primarily attributable to the local minima of the simulation objective. A didactic example is the buckling of cloth. As a cloth stripe undergoes compression from both ends, it can bend

Authors' addresses: Chen Liu, State Key Lab of CAD and CG, Zhejiang University, Hangzhou, China and Style3D Research, Hangzhou, China, eric.liu@lincetex.com; Weiwei Xu, State Key Lab of CAD and CG, Zhejiang University, Hangzhou, China, xww@cad.zju.edu.cn; Yin Yang, The University of Utah, Salt Lake City, USA and Style3D Research, Salt Lake City, USA, yangzzy@gmail.com; Huamin Wang, Style3D Research, Hangzhou, China, wanghmin@gmail.com.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM. 0730-0301/2024/7-ART74 \$15.00 <https://doi.org/10.1145/3658128>

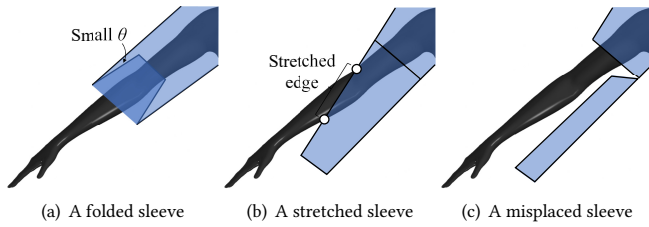
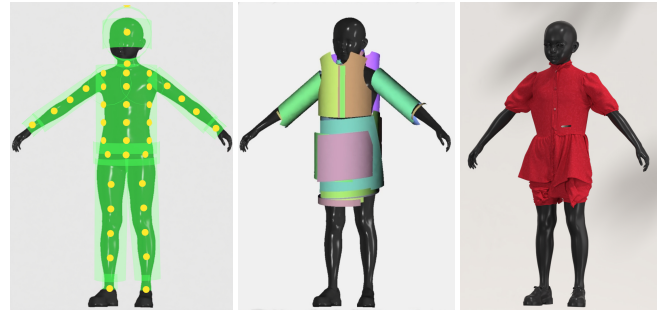


Fig. 2. Improper simulation outcomes of a sleeve on the body. Each outcome signifies a local minimum in the simulation problem. Ideally, clothing should be worn properly by the body with no folding or intersection.

either forward or backward, resulting in different folds. This non-uniqueness is exacerbated when the simulation objective includes collision potentials, giving rise to various visual artifacts. These may include self-folding in Fig. 2a, cloth-body intersection in Fig. 2b, or pieces erroneously trapped outside of the body in Fig. 2c. To mitigate the local minima issue, a natural solution is to employ a suitable initialization. In essence, the goal of an initialization is to position the sewing pieces around the human body without folding or intersection, thus enabling the generation of visually acceptable digital garments through simulation. An intersection-free initialization is also important to simulators that rely on continuous collision detection and interior point methods [Li et al. 2020].

While researchers have long acknowledged the challenge of addressing local minima in simulation, their primary focus has traditionally revolved around mitigating numerical instability [Volino and Thalmann 2000; Wang et al. 2023; Wu and Kim 2023]. In this pursuit, the complexities of establishing a suitable initialization have often been overshadowed. A relatively straightforward aspect of garment initialization involves the assignments of sewing pieces to body parts. Such assignments can be automatically extracted from pattern shapes and labels [Berthouzoz et al. 2013], or manually determined by users. However, once sewing pieces have been assigned to body parts, a more complex challenge arises: determining the appropriate initial shape for each sewing piece. This task is notably more intricate than arranging decomposed solid components [Lu et al. 2023; Sellán et al. 2019; Zhang et al. 2020], given that sewing pieces, due to their deformable nature, are prone to excessive folding and intersection.

Consider that each sewing piece has already been assigned to a body part, with each part associated with a predetermined (yellow) location and a predefined (green) surface, either cylindrical or planar, as depicted in Fig. 3a. To establish the initial configuration of the pieces for a digital garment, previous research and systems typically place the sewing pieces at these specified locations on the body part. The initial shapes are then determined by their projections onto the corresponding surfaces, as shown in Fig. 3b. While sufficient for simple garments, this method proves challenging and requires considerable manual intervention for complex garments characterized by small pieces, multiple layers, or asymmetric sewing boundaries such as gathering, as depicted in Fig. 3c. Sketch-based systems, like the one introduced in [Igarashi and Hughes 2002], enhance flexibility in the manual initialization of sewing pieces. However, they too become unmanageable with increasing garment



(a) Body part locations (b) Manual initialization (c) The simulated garment

Fig. 3. Manual initialization and its simulated result. Under the assumption that every piece is assigned to a body part, previous systems typically initialize a sewing piece by aligning it with the designated (yellow) location of the body part and then projecting it onto a predefined (green) surface, as depicted in (a). However, despite investing five minutes in manually fine-tuning the positions of these pieces and eliminating overlaps as shown in (b), the resulting simulated garment in (c) remains unsatisfactory, in comparison with our result in Fig. 13f.

complexity. Overall, our goal is to eliminate manual intervention, maintaining a seamless and automated process for creating digital garments unsupervised.

Assuming a digital garment is provided with the sewing pattern and all sewing relationships, we present an automatic initialization system to tackle the local minima issue in subsequent simulations. Our system is founded on a phased approach, gradually introducing pattern pieces and objective potentials into an optimization-based initialization procedure. This phased approach not only addresses the local minima but also offers unique advantages.

- It achieves high efficiency by optimizing only a small set of pattern pieces at a time.
- It conveniently resolves inter-piece intersections based on the order of piece arrangement.
- It is user-friendly and allows user intervention whenever necessary, especially if the pattern input is imperfect.

Based on this approach, we make the following contributions:

- **Seeding.** We train a classification network to automatically select the initial piece(s) for arrangement, referred to as the *seed*. Our key observation is that neck or waist pieces often serve as more effective seeds, compared to other pieces. Subsequently, we formulate an optimization problem to initialize the seed shape on the human body.
- **Selection.** We propose a heuristic function for selecting the next piece to be arranged. We also illustrate the essential criteria for simultaneously arranging multiple pieces and introduce a novel algorithm for automated detection and merging of these pieces.
- **Arrangement.** We formulate the arrangement of the selected piece(s) as an optimization problem. To resolve the local minima issue, we adopt a phased approach by progressively adding potentials into the objective. Furthermore, we provide

a set of criteria for assessing the arrangement quality. These criteria enable our system to reattempt the arrangement if it does not meet the standards.

We have implemented the proposed automatic initialization system on the CPU and tested it alongside our in-house simulation engine. Our experiments confirm that the system can handle the initialization of a diverse range of garments on various human body avatars, often necessitating minimal to no user intervention. Most garments can be initialized within a matter of seconds, and the system exhibits scalability in handling complex, multi-layered garments, including the square down coat with 358 pieces, as depicted in Fig. 1. Code and data for this paper are at <https://www.kaggle.com/style3d>.

2 RELATED WORK

2.1 Physics-based cloth simulation

Physics-based cloth simulation has been a significant area of research in computer graphics since the seminal work by Baraff and Witkin [1998]. Depending on the representation of cloth, cloth simulation techniques fall into three categories: spring-based [Bridson et al. 2003; Choi and Ko 2002; Liu et al. 2013], continuum-based [Narain et al. 2012; Volino et al. 2009], and yarn-based [Cirio et al. 2014; Kaldor et al. 2008, 2010]. In recent years, cloth simulation research has predominantly focused on three critical directions: modeling and characterizing the mechanical properties of cloth [Miguel et al. 2012, 2013; Sperl et al. 2022; Wang et al. 2011], effective handling of self-frictional contacts [Bridson et al. 2002; Brochu et al. 2012; Chen et al. 2023; Li et al. 2018, 2020; Ly et al. 2020; Tang et al. 2018], and enhancing simulation performance through numerical algorithms [Narain et al. 2016; Tamstorf et al. 2015; Wang and Yang 2016; Wu et al. 2020]. Like other simulation challenges, cloth simulation is notable for its issue with local minima, particularly when dealing with cloth-body collisions using repulsion potentials. While researchers have explored the impact of local minima on the stability of numerical solvers [Volino and Thalmann 2000; Wang et al. 2023; Wu and Kim 2023], there remains an uncharted domain – the existence of multiple solutions, each mathematically plausible but visually unsatisfactory for properly draped garments.

2.2 Pattern design and optimization

In the past, researchers predominantly considered sewing pattern modeling as a tool to assist fashion manufacturing, often entailing extensive user interaction. However, in recent years, there has been a growing acknowledgment of the importance of automatic sewing pattern generation within the digital fashion and entertainment industries. Research in this field can be broadly classified into two main directions: optimizing existing sewing patterns to meet specific user-defined objectives [Bartle et al. 2016; Ly et al. 2018; Umetani et al. 2011; Wang 2018], and reconstructing sewing patterns from various sources, such as parametric templates [Korosteleva and Lee 2021; Korosteleva and Sorkine-Hornung 2023], 3D meshes [Goto and Umetani 2021; Pietroni et al. 2022], user sketches [Wang et al. 2018], 3D scans [Bang et al. 2021; Korosteleva and Lee 2022], and images [Liu et al. 2023; Yang et al. 2018].

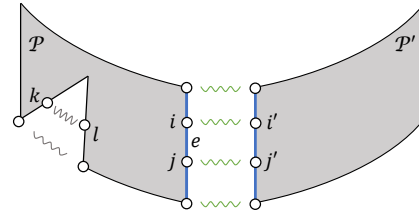


Fig. 4. Two distinct sewing pieces within the 2D pattern space. On the final garment in the 3D space, they are meant to be joined using (green) sewing strings with zero reference lengths. Each string, along with its ending vertices, establishes a sewing relationship between the pieces.

Our system seamlessly integrates with the majority of existing automatic sewing pattern optimization and generation techniques, as they inherently provide patterns with sewing relationships that are essential for our system. Our system also benefits from emerging research in pattern parsing techniques [Berthouzoz et al. 2013], offering alternative approaches for establishing sewing relationships and identifying piece assignments.

3 BACKGROUND

We start by introducing the concept of a sewing pattern along with the associated sewing relationships. Let \mathcal{P} and \mathcal{P}' represent two distinct pieces within the 2D space where the pattern is defined, known as the *pattern space*, as depicted in Fig. 4. A *sewing relationship* is defined by a pair of vertices $\{i, i' | i \in \mathcal{P}, i' \in \mathcal{P}'\}$, which are designated to be joined by a (green) *sewing string* in the 3D garment space. By default, we assume that the sewing string is of zero reference length, suggesting that, without external influences, these vertices will align at the same spatial point in the final 3D garment.

Consider an edge $e = \{i, j\}$ within \mathcal{P} . If both vertices i and j are connected to \mathcal{P}' , we label e as a *sewing boundary edge*. The collection of all such edges forms the (blue) *sewing boundary* of \mathcal{P} , denoted as $\partial\mathcal{P}$. It's important to note that although a sewing boundary edge is typically located on the boundary of a piece, it can also appear within the interior.

A sewing pattern may include dart features, indicated by vertices from the same piece – such as k and l in Fig. 4 – joined by (gray) internal strings. We choose not to identify these connections as sewing strings or relationships; instead, we consider them as internal elastic springs that contribute to the garment's deformation potential, a topic we will delve into more deeply in Subsection 5.2.

4 SYSTEM PIPELINE

The core concept of our system hinges on a phased approach, where sewing pieces are initialized sequentially – one piece at a time – with the introduction of objective potentials gradually applied during each phase. This method raises an essential question: how do we determine the order of arrangement? This includes choosing the initial piece, the *seed*, and deciding which piece follows after several have been arranged. To address these considerations, we rely on three key insights:

- **Top-to-bottom arrangement.** It's advantageous to arrange pieces from top to bottom. This allows the partially initialized garment to counteract gravity effectively.
- **Prioritizing large pieces.** Starting with larger pieces is beneficial since they significantly influence the arrangement of subsequent smaller pieces.
- **Early arrangement around the torso.** Pieces surrounding the torso, such as those near the neck or waist, should be arranged early. This strategy facilitates the growth of other pieces around the body, preventing them from being incorrectly positioned, as shown in Fig. 11a.

By adhering to these principles, we propose an automatic garment initialization system with three key processes as Fig. 5 shows. The first process is seeding, which uses a classification network to find the first piece(s) to be arranged and applies geometric optimization to calculate its initial shape. If the automatically selected seed is ineffective, users can manually choose their seed. Once the system initializes the seed, it runs the selection process to find the next piece(s) and employs the arrangement process to initialize its position and shape. Compared with the other processes, the arrangement process is more computation-intensive. It contains three iterative steps and terminates only when the arranged shape is satisfactory enough. The system keeps running the selection-arrangement procedure until all pieces have been arranged.

The system operates based on the following assumptions.

- The body should be rigged, with predefined locations assigned to its parts, as shown in Fig. 3a.
- The body should be in an A-pose with both arms extended 50 degrees away from the torso.
- The pattern should include all of its sewing relationships, defined by sewing pairs and strings.
- The pattern mesh should be downsampled to a uniformly low resolution, enhancing system performance while preserving piece shapes and sewing relationships. In our system, the resolution is set to an average vertex distance of 20mm.
- Each piece should face away from the body when incorporated into a garment worn on the body.
- Each piece should be oriented upright¹ within the pattern space, matching their orientation on the garment. Specifically, for a sleeve piece, the upright direction is determined by aligning from the wrist up to the shoulder.

Currently, we rely on preprocessing to meet these assumptions, and we anticipate that sewing patterns generated by AI models will naturally conform to these assumptions in the future.

Our system also offers postprocessing to enhance the generated garment. During this procedure, a quasistatic simulation of the entire garment is conducted using a small step size and greater repulsion strength parameters. This simulation settles the garment under the influence of gravity, eliminates remaining intersections, and adjusts the body to a desired pose if it differs from the A-pose.

¹This assumption is reasonable in both academia and the fashion industry. It is because upright pieces are not only visually recognizable, but also provide consistency in constructing garments from textured, oriented fabrics.

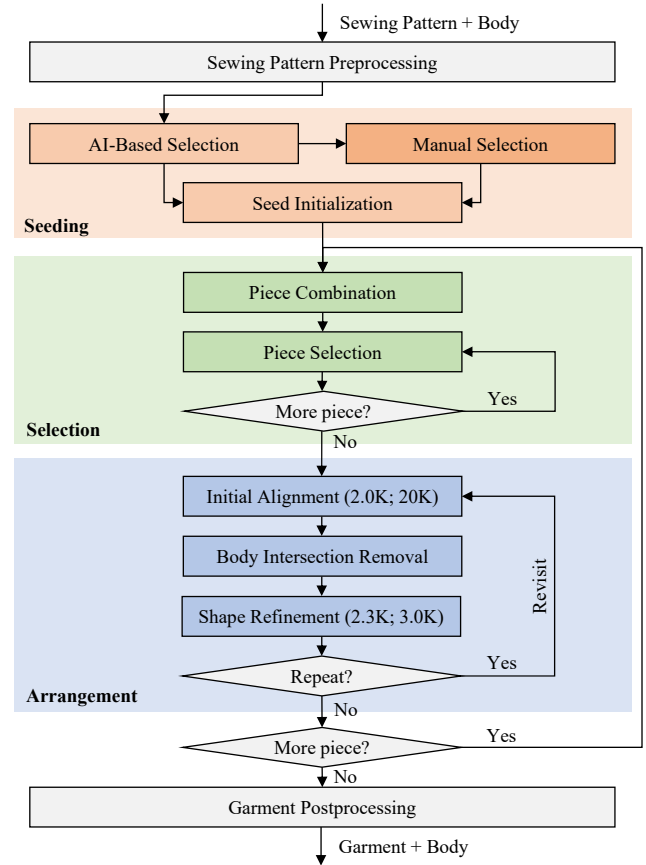


Fig. 5. The system pipeline. Our system consists of three key processes, with its core being a selection-arrangement procedure that sequentially arranges pattern pieces until none remains. During this procedure, the system may visit its steps multiple times to achieve higher parallelization and improved result quality. The numbers in the brackets are the numbers of solver iterations allocated to each step in the first and subsequent visits.

5 SEEDING

The seeding procedure in our system serves a dual purpose: it determines the seed piece(s) and the location while also initializing the seed shape. As shown in Subsection 8.2 and 8.3, the choice of the seed significantly influences system performance, with neck, waist, or chest pieces often proving effective seeds in our experiments. Based on this observation, we introduce both automatic and manual seeding methods in Subsection 5.1, and we discuss geometric optimization for initializing the seed in Subsection 5.2.

5.1 Seed Selection Methods

In the seed selection step, our objective is to identify the piece(s) that covers the neck, waist, or chest if the other two are not found. This step can be omitted if the sewing pattern already designates pieces for these specific body parts. Otherwise, we approach it as a pattern piece classification problem.

A minor complication is the possibility of multiple pieces covering the waist. To mitigate this, the system identifies all rectangular

pieces with width-to-height aspect ratios exceeding three. These pieces are then merged if they are horizontally connected by sewing strings. This practice allows us to consolidate multiple waist pieces into a single one.

5.1.1 AI-based seed selection. Given a pattern piece defined in the 2D reference pattern space, we first need to construct the encoding for determining the category it belongs to. We choose radial sampling [Liu et al. 2023] to define the shape feature of the piece. However, the shape feature alone is insufficient for determining the category, as many pieces share similar shapes but correspond to different parts on a garment. Therefore, we consider not only the piece's own shape, but also its surroundings, including both the shapes of the neighboring pieces and its sewing relationships. Since the pieces can be located anywhere within the pattern space, the length or direction of a sewing string in the pattern space is unimportant. Instead we extract the features of the sewing relationships between piece \mathcal{P} and its neighbor \mathcal{P}' based on where their sewing boundary is located with respect to \mathcal{P}' :

$$S_{\mathcal{P} \rightarrow \mathcal{P}'} = \sum_{e' \in \partial \mathcal{P}'} C(e', \mathcal{P}) U(e'), \quad (1)$$

where $\partial \mathcal{P}'$ is the sewing boundary of piece \mathcal{P}' , $U(e')$ is the column area of \mathcal{P}' beneath edge e' as Fig. 6a shows, and $C(\mathcal{X}, \mathcal{Y})$ is a function testing whether every vertex in set \mathcal{X} is connected with set \mathcal{Y} by any sewing string \mathcal{S} :

$$C(\mathcal{X}, \mathcal{Y}) = \begin{cases} 1, & \forall x \in \mathcal{X}, \exists y \in \mathcal{Y} \text{ s.t. } \{x, y\} \in \mathcal{S}, \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

While there exist many other ways to encode the sewing relationships, we choose Eq. 1 because it can be reused later for the piece selection heuristic function (in Section 6) and it well signifies neck and waist pieces. Intuitively, if $S_{\mathcal{P} \rightarrow \mathcal{P}'}$ is large, \mathcal{P} is positioned above a bulky piece \mathcal{P}' on the garment², suggesting that \mathcal{P} could be a neck or waist piece while \mathcal{P}' could be a chest or leg piece underneath. If a pattern includes both neck and waist pieces, Eq. 1 tends to favor one over the other, depending on the garment piece's coverage of the body's upper or lower parts. Specifically, for tops, it gives priority to the neck piece.

Our AI-based seed selection method is built upon a two-layer graph attention network [Veličković et al. 2018]. The initial layer aggregates the features of a piece and its surroundings, including both piece shape features and sewing relationship features, using an attention mechanism. The subsequent layer operates as a classifier, using softmax activation to forecast the likelihood of the piece falling into one of seven categories: neck, waist, front chest, back chest, front pelvis, back pelvis, and others.

5.1.2 Manual seed selection. If AI-based seed selection fails to find any seed or if the seed it finds is notably incorrect, the system provides users with an option to manually select the seed. This process is analogous to the arrangement procedure in existing systems: users must select the piece(s) and assign them to pre-defined neck, waist, or chest locations. Our system offers 15 such locations.

²As mentioned previously in Section 4, we assume that all of the pieces are oriented upright in the 2D reference pattern space.

5.2 Seed Initialization

After selecting the seed piece(s), we would like to initialize its shape $\mathbf{x} \in \mathbb{R}^{3N}$ around the body, where N is the number of seed vertices. To achieve this, we propose to solve an optimization problem $\mathbf{x} = \arg \min F^{\text{init}}(\mathbf{x})$ with the following objective:

$$F^{\text{init}}(\mathbf{x}) = F^{\text{cent}}(\mathbf{x}) + F^{\text{up}}(\mathbf{x}) + F^{\text{dist}}(\mathbf{x}) + F^{\text{def}}(\mathbf{x}). \quad (3)$$

The centering potential, $F^{\text{cent}}(\mathbf{x}) = \frac{1}{2} k^{\text{ctr}} \|\mathbf{x}_c - \mathbf{x}_c^0\|^2$, pulls the geometric center \mathbf{x}_c of the seed piece(s) toward the assigned body part location \mathbf{x}_c^0 with a strength parameter k^{ctr} . Meanwhile, the upward potential aims to maintain the seed's orientation upward:

$$F^{\text{up}}(\mathbf{x}) = -\frac{1}{2} k^{\text{up}} \sum_i \text{Sign} \left((\mathbf{r}_i - \mathbf{r}_c)^\top \begin{bmatrix} 0 \\ 1 \end{bmatrix} \right) (\mathbf{x}_i - \mathbf{x}_c)^\top \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad (4)$$

where \mathbf{r}_i and \mathbf{r}_c are the reference vertex positions in the 2D pattern space, and k^{up} is the upward strength parameter. The body distance potential repels the seed from the body while maintaining a designated cloth-body distance D :

$$F^{\text{dist}} = \frac{1}{2} k^{\text{dist}} \sum_i \max(\text{Sign}(\mathbf{n}(\mathbf{x}_i) \cdot \nabla \phi(\mathbf{x}_i)), 0) (\phi(\mathbf{x}_i) - D)^2, \quad (5)$$

where $\mathbf{n}(\mathbf{x}_i)$ is the normal of vertex i , $\phi(\mathbf{x}_i)$ is the signed distance function of the body constructed using the fast marching method [Osher and Fedkiw 2002], and k^{dist} is the strength parameter. Eq. 5 disables the body distance potential of vertex i if the vertex is facing toward the body. This is to prevent the seed from being stuck in a self-folded local minimum state.

Finally, we need the deformation potential $F^{\text{def}}(\mathbf{x})$ to constrain the deformation of the entire seed shape, including both planar and bending deformations. For simplicity, we choose the spring model to limit planar deformation for each mesh edge, and the quadratic model [Bergou et al. 2006] to limit bending deformation for each dihedral edge. Other deformation models can also be effective in this context. Since the primary aim of the deformation potential is to limit shape deformation, rather than to generate realistic wrinkles akin to actual fabric, we deliberately set a relatively high default value for the bending stiffness parameter. This helps prevent self-intersections among the fine details of the wrinkles.

We use an iterative solver to solve this optimization problem. Please refer to Section 8 for solver and parameter details.

6 PIECE SELECTION

Given a set of pattern pieces already being arranged on the body, we are now faced with the task of determining the next piece(s) for arrangement. This process is pivotal, as it not only influences the quality of the final result, but also dictates how the system handles inter-piece intersections in Subsection 7.3. In this section, we will first introduce the heuristic score function, which helps us evaluate and select an individual piece. Following that, we will explore the importance of arranging multiple pieces simultaneously and our approach to selecting them.

6.1 Single Piece Selection

Let's first discuss the selection of a single piece. We propose the following heuristic function to choose a piece \mathcal{P} with the highest

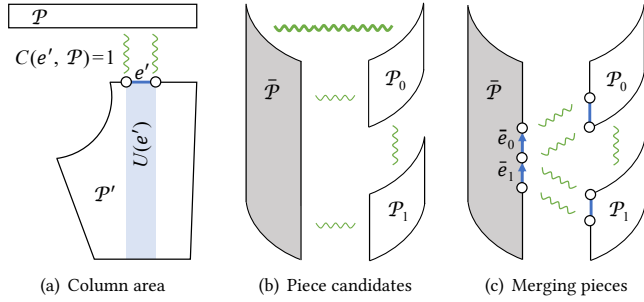


Fig. 6. The scenarios involved in the piece selection process. Our system determines the next piece(s) to be arranged by a heuristic function, which is based on the (green) sewing relationships among the pieces.

heuristic score:

$$H(\mathcal{P}) = \frac{\sum_{e \in \partial \mathcal{P}} C(e, \bar{\mathcal{P}}) L_e^0}{\sum_{e \in \partial \mathcal{P}} L_e^0} + s_a A(\mathcal{P}) + s_b S_{\bar{\mathcal{P}} \rightarrow \mathcal{P}}, \quad (6)$$

in which $\bar{\mathcal{P}}$ is the set of already arranged pieces, L_e^0 is the reference length of edge e , $A(\mathcal{P})$ is the reference area of piece \mathcal{P} , $S_{\bar{\mathcal{P}} \rightarrow \mathcal{P}}$ is the sewing relationship score in Eq. 1, and s_a and s_b are two weight variables. In Eq. 6, the first term gives precedence to the pieces whose sewing boundaries are mostly determined by $\bar{\mathcal{P}}$ already. The second term prioritizes the arrangement of large pieces. The third term underscores the importance of selecting pieces that can be well supported from the top, such as \mathcal{P}_0 with thickened sewing strings on the top in Fig. 6b. Without this term, the system may opt for \mathcal{P}_1 , which could sag excessively under gravity during shape refinement (in Subsection 7.3), due to missing support from the top.

6.2 Multiple Piece Selection

In practice, it is necessary to arrange multiple pieces simultaneously for two compelling reasons.

The first reason arises when a single piece must be intentionally divided into multiple components, often due to the use of different fabrics, such as the trench coat in Fig 13n. Arranging these pieces consecutively could result in an improper fit to the corresponding body part. To tackle this issue, we introduce a piece combination step as shown in Fig. 5, aimed at assessing the feasibility of selecting and arranging multiple pieces together as a cohesive unit. Specifically, this step evaluates the continuity of the sewing boundary of $\bar{\mathcal{P}}$, exploring a seamless transition between its connection to one piece candidate and the next. For example, let \bar{e}_0 and \bar{e}_1 be two adjacent sewing boundary edges of $\bar{\mathcal{P}}$ in Fig. 6c, connected with two piece candidates \mathcal{P}_0 and \mathcal{P}_1 , respectively. If the transition between \bar{e}_0 and \bar{e}_1 is smooth, we treat \mathcal{P}_0 and \mathcal{P}_1 as part of a larger, combined piece. Subsequently, we calculate the heuristic score for this amalgamated piece, similar to other individual pieces, and decide whether it should be selected next, as described in Subsection 6.1.

The second reason is related to parallelization. According to Section 8, on average, each piece comprises just 197 vertices, with over half of the pieces containing fewer than 100 vertices. Sequencing such small pieces one after another would not fully leverage the power of parallel processing. To address this challenge, we employ

the following approach: we iteratively select additional pieces using the aforementioned process as Fig. 5 shows, until the total number of vertices in the selection set reaches a specified cap. In our system, this cap is set at 1,024. Our experiments demonstrate that this practice reduces the computational time by 60 to 80 percent.

7 PIECE ARRANGEMENT

We formulate the arrangement of the newly selected piece(s) as an optimization problem with the following objective function:

$$F(\mathbf{x}) = F^{\text{def}}(\mathbf{x}) + F^{\text{sew}}(\mathbf{x}) + F^{\text{ext}}(\mathbf{x}) + F^{\text{body}}(\mathbf{x}) + F^{\text{self}}(\mathbf{x}). \quad (7)$$

This objective shares the same deformation potential, as described in Eq. 3. However, it operates within a distinct problem domain and incorporates additional potentials. Before discussing these differences, it is important to note that this optimization is prone to local minima issues if solved immediately, as illustrated in Fig. 8a. To overcome this challenge, we employ a phased approach, gradually introducing new potentials into the system over three steps.

7.1 Initial Alignment

Let $\bar{\mathcal{P}}$ be the set of arranged pieces and \mathcal{P} be the newly selected piece(s). We want to make an initial alignment of \mathcal{P} to $\bar{\mathcal{P}}$, without considering body or self intersection. To do so, we first initialize \mathcal{P} by applying an affine transformation $\{\mathbf{t}, \mathbf{A} \mid \mathbf{t} \in \mathbb{R}^3, \mathbf{A} \in \mathbb{R}^{3 \times 2}\}$, which minimizes the sewing gap between \mathcal{P} and $\bar{\mathcal{P}}$ as:

$$\{\mathbf{t}, \mathbf{A}\} = \arg \min \sum_{\{i,j\} \in \mathcal{S}, i \in \mathcal{P}, j \in \bar{\mathcal{P}}} \|\mathbf{t} + \mathbf{A}\mathbf{r}_i - \mathbf{x}_j\|^2, \quad (8)$$

where $\{i, j\}$ is a sewing pair between \mathcal{P} and $\bar{\mathcal{P}}$, \mathbf{r}_i is the 2D pattern position of vertex i , and \mathbf{x}_j is the 3D garment position of vertex j . To solve Eq. 8, we use its closed-form solution [Müller et al. 2005], which requires at least two distinct sewing pairs between \mathcal{P} and $\bar{\mathcal{P}}$. If that is not true, we simply set $\mathbf{A} = [\mathbf{I} \ 0]^T$ and then find \mathbf{t} for translational alignment only. Once we transform \mathcal{P} , we optimize its shape by minimizing a truncated objective function: $F(\mathbf{x}) = F^{\text{def}}(\mathbf{x}) + F^{\text{sew}}(\mathbf{x}) + F^{\text{ext}}(\mathbf{x})$. Here we define the sewing potential by quadratic energies:

$$F^{\text{sew}}(\mathbf{x}) = \frac{1}{2} k^{\text{sew}} \sum_{\{i,j\} \in \mathcal{S}, i \in \mathcal{P}, j \in \bar{\mathcal{P}}} \|\mathbf{x}_i - \mathbf{x}_j\|^2, \quad (9)$$

where k^{sew} is the sewing strength parameter. We define the external potential as:

$$F^{\text{ext}}(\mathbf{x}) = \frac{1}{2} k^{\text{fix}} \sum_{j \in \mathcal{P}} \|\mathbf{x}_j - \mathbf{x}_j^0\|^2, \quad (10)$$

where k^{fix} is its strength parameter. The goal of the external potential here is to prevent each vertex j of the already arranged piece $\bar{\mathcal{P}}$ from leaving its arranged position \mathbf{x}_j^0 .

To smooth the boundary between \mathcal{P} and $\bar{\mathcal{P}}$, we define the problem domain \mathbf{x} as the union of \mathcal{P} and the two-ring sewing boundary neighborhood on $\bar{\mathcal{P}}$, as Fig. 7a shows. We also integrate sewing edge pairs into the dihedral edge set for bending, if the two sewing edges (with blue arrows) are topologically consistent. We incorporate each potential component into the total objective, if all of the relevant vertices exist in \mathbf{x} . To help reduce the local minima issue related to bending deformation, we further increase the magnitude of the

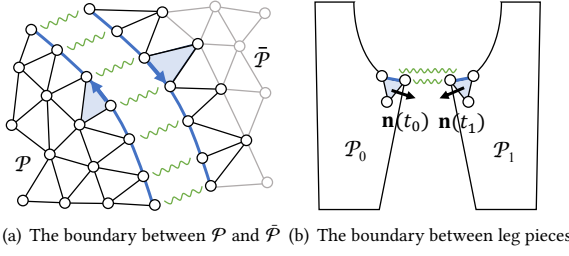


Fig. 7. The sewing boundaries. To smooth the sewing boundary, we add the vertices within the two-ring neighborhood on $\bar{\mathcal{P}}$ into the problem domain, as depicted in (a). However, we cannot improve the smoothness near the crotch, as the leg pieces are supposed to face against each other as (b) shows.

bending stiffness parameter. The results of a selected sleeve after affine transformation and initial alignment are shown in Fig. 8b and 8c, respectively.

7.2 Body Intersection Removal

The body intersection removal step resembles the initial alignment step, but with additional repulsion potentials:

$$F^{\text{body}}(\mathbf{x}) = \frac{1}{2}k^{\text{body}} \sum_i (\min(\phi(\mathbf{x}_i) - \epsilon, 0))^2, \quad (11)$$

where $\phi(\mathbf{x}_i)$ is the signed distance function of the body, k^{body} is the body repulsion strength parameter, and ϵ is the repulsion buffer distance. In our system, $\epsilon = 4\text{mm}$.

If we activate the body repulsion potentials for all of the vertices immediately, the simulation could easily be trapped in local minima with body intersections, as evident in Fig. 8d, since the initial alignment of \mathcal{P} does not account for body collisions. To address this issue, we recognize that the sewing boundary vertices connected to $\bar{\mathcal{P}}$ do not experience the intersection issue. Therefore, we incrementally activate the potentials of the piece vertices through breadth-first search initiated from the sewing boundary. This method mimics the real-world process of donning a garment: as the body extends, the clothing untangles and covers the body, as Fig. 8e shows.

7.3 Shape Refinement

Our last aim is to resolve self-intersections of cloth and make further refinements to the garment shape.

There are two types of self-intersections: intra-piece intersections and inter-piece intersections. Since existing untangling algorithms [Baraff et al. 2003; Volino and Magnenat-Thalmann 2006] are expensive and cannot guarantee the removal of inter-piece intersections, we focus on addressing them exclusively. But instead of developing new untangling algorithms as in [Buffet et al. 2019], we take a simple approach. Our assumption is that piece \mathcal{P} forms an inside-outside relationship with the already arranged pieces $\bar{\mathcal{P}}$. More precisely, \mathcal{P} must be situated outside of $\bar{\mathcal{P}}$, according to $\bar{\mathcal{P}}$'s normal direction. Based on this assumption, we apply a self-repulsion potential to every vertex $i \in \mathcal{P}$ and its closest triangle $t = \{j, k, l\}$ within $\bar{\mathcal{P}}$:

$$F^{\text{self}}(\mathbf{x}) = \frac{1}{2}k^{\text{self}} \sum_{i \in \mathcal{P}} (\min((\mathbf{x}_i - \mathbf{x}_j) \cdot \mathbf{n}(t) - \epsilon, 0))^2, \quad (12)$$

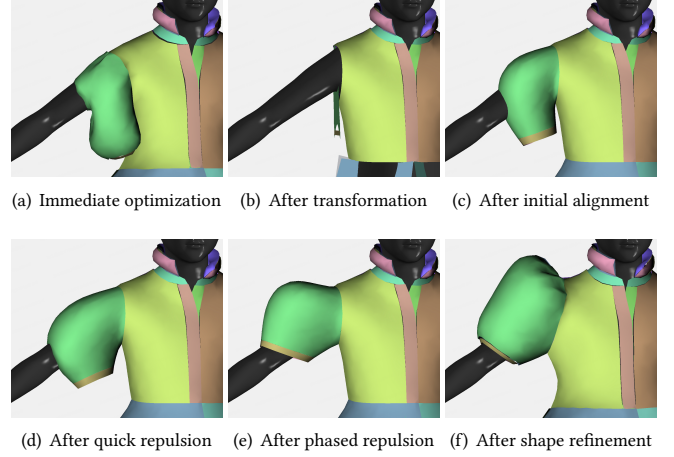


Fig. 8. The arrangement results of a sleeve piece in the puff-sleeve dress example, after other pieces have been arranged. To arrange the selected sleeve, we adopt a phased approach that gradually adds new potentials into the objective, as depicted in (b), (c), (e), and (f). Without this approach, the optimization can lead to local minima, as shown in (a) and (d).

in which $\mathbf{n}(t)$ is the constant normal of triangle t , k^{self} is the self-repulsion strength parameter, and ϵ is the same buffer distance used in Eq. 11. One advantage of our approach is that we do not require the complete elimination of intersections during the arrangement process, which would necessitate a large k^{self} and significant computational time. Instead, we can address any remaining intersection in post-processing, as discussed in Section 4. This is possible since the overall inside-outside relationship among pieces remains unchanged once it is determined by the piece arrangement order.

During shape refinement, our optimization occurs in two phases. In the first phase, we introduce self-repulsion potentials into the objective and reduce the bending stiffness parameter to its default value, allowing cloth to bend more readily. In the second phase, we add a mild gravitational term into the external potential F^{ext} . This modification enables cloth to sag in a natural manner, while preventing excessive drooping. Furthermore, we expand the problem domain beyond the two-ring sewing boundary neighborhood of $\bar{\mathcal{P}}$ and fix the vertices on $\bar{\mathcal{P}}$ only if they are away from the boundary. This expansion helps refine the garment shape near the sewing boundary, especially for gathering as Fig. 8f shows. In our system, we define the expanded domain as the 16-ring neighborhood from the sewing boundary.

7.4 Termination Conditions

After the completion of all three steps, we assess the quality of the arrangement based on the following criteria:

$$\begin{cases} \theta_{\text{in}} = \min_{\{t_0, t_1\} \in \mathcal{N}} \mathbf{n}(t_0) \cdot \mathbf{n}(t_1) < \theta_{\text{in}}^0, \\ \theta_{\text{out}} = \min_{\{t_0, t_1\} \in \mathcal{B}} \mathbf{n}(t_0) \cdot \mathbf{n}(t_1) < \theta_{\text{out}}^0, \\ s_{\text{max}} = \max_{\{i, j\} \in \mathcal{E}} (\|\mathbf{x}_i - \mathbf{x}_j\| - L_{ij}^0) > s_{\text{max}}^0, \end{cases} \quad (13)$$

in which $\mathbf{n}(t)$ is the normal of triangle t , \mathcal{N} is the set of neighboring triangles, and \mathcal{B} is the set of topologically consistent triangles adjacent to each other after sewing, such as the two blue ones in Fig. 7a.

The first criterion, $\theta_{in} < \theta_{in}^0$, evaluates whether a piece has undergone excessive folding or self-intersection. Ideally, such issues should be prevented by setting a relatively large bending stiffness parameter, as detailed in Subsection 5.2. When this criterion is met, we contend that the current number of iterations is inadequate. Consequently, we repeat the arrangement process until θ_{in} increases adequately, ensuring that the piece is appropriately flattened.

The second criterion, $\theta_{out} < \theta_{out}^0$, assesses whether two adjacent pieces have been folded excessively along their sewing boundary. Although this could indicate an undesirable configuration in the garment, it might also be an intentional aspect of the design, as exemplified by the crotch area created by the joining of leg pieces, as shown in Fig. 7b. Consequently, if $\theta_{out} < \theta_{out}^0$, we proceed to revisit the arrangement process only once, accepting the ensuing result regardless of its nature.

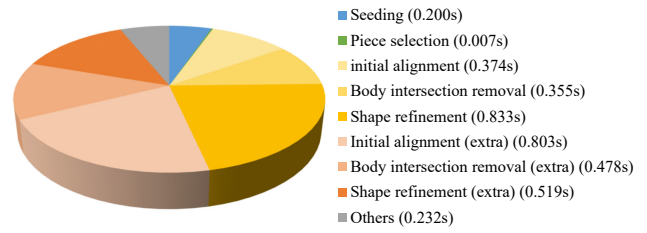
Finally, $s_{max} > s_{max}^0$ identifies more than necessary stretching in any spring edge, typically resulting from cloth-body intersections, as depicted in Fig. 2b. This stretching could stem from inadequate piece selection or from the arrangement process itself. Since the exact cause is not immediately clear, we expand the selection set once by including the neighboring pieces of the currently selected piece(s), and then revisit the arrangement process.

The arrangement process concludes when no additional revisits are needed by the criteria. Thus, the computational expense incurred during this process is determined by two factors: the number of iterations the solver requires at each step and the total number of revisits. In our experience, the initial alignment step is crucial to the quality of the final arrangement; typically, most pieces can be adequately arranged after just a few initial alignment iterations without needing any revisits. This observation has led us to deliberately limit the number of initial alignment iterations in the first visit to reduce computational demands. The specific iteration counts for each step of the process are detailed in Fig. 5.

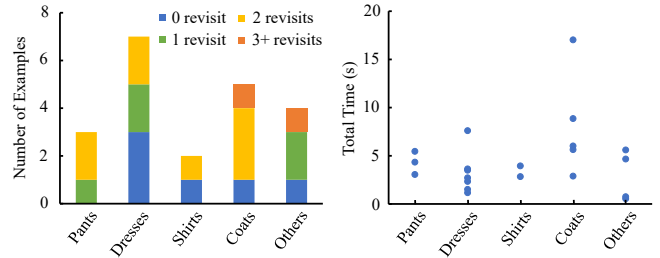
8 RESULTS AND DISCUSSIONS

We implement our system exclusively on the CPU to ensure compatibility across multiple hardware platforms. The implementation of our system is solver-independent for optimization and simulation tasks. In practice, we employ a solver based on the parallelized gradient descent method with Jacobi preconditioning and Chebyshev acceleration [Wang and Yang 2016], utilizing a fixed step size (with $\alpha = 0.4$) and a fixed spectral radius (with $\rho = 0.9994$). In our experiments, we maintain consistent parameters across all examples. These include the strength and stiffness associated with potential terms, the weight parameters used by the heuristic function in Eq. 6, and the thresholds specified for termination conditions in Eq. 13. While projective dynamics [Bouaziz et al. 2014] could also be used for initial alignment, we have found it less capable of handling phased intersection removal, as discussed in Section 7.2.

We assess the performance of our system using 21 sewing patterns designed for six body types. These patterns encompass various



(a) A computational breakdown of the puff-sleeve dress example



(b) The examples with revisits

(c) The total computational times

Fig. 9. The charts depicting the number of revisits and the time spent for each example. These visuals establish a clear correlation between the number of revisits and the computational time. It also highlights the system's efficiency, showing high variability when processing dresses and coats.

garment types. Depending on the design, the number of vertices (after resampling) ranges from 1.5K to 23K, and the number of pieces varies from 7 to 358. On average, each piece contains 197 vertices, and about 52 percent of the pieces contains 100 vertices or fewer.

8.1 Network Training and Dataset

The training of our pattern piece classification network uses the AdamW optimizer, with the initial learning rate set to 5×10^{-5} , the backbone set to 5×10^{-6} and the weight decay set to 10^{-4} . The whole training process takes 90,000 iterations and lasts 11 hours on two NVIDIA® GeForce RTX™ 4090 GPUs with the batch size equal to 8. To train this network, we have collected a dataset of 23,276 sewing patterns, covering a broad array of garments such as pants, dresses, shirts, and coats. Each pattern is associated with its sewing relationships, and all of the pieces are labeled according to corresponding body parts. We use 80 percent of the data for training and the rest for testing.

8.2 Efficiency Evaluation

We evaluate the efficiency of our system on a workstation with an Intel® Core™ i9-13900K 3.00 GHz CPU. Fig. 9a provides a breakdown of the computational time dedicated to the puff-sleeve dress example. Among the three processes involved, the arrangement process is the most expensive one. Within the arrangement process, initial alignment and shape refinement are the most expensive steps.

According to Subsection 7.4, if the first pass fails to meet the criteria, we revisit the piece arrangement process and increase the number of iterations. Fig. 9a illustrates that this practice nearly doubles the computational cost when the system revisits the process

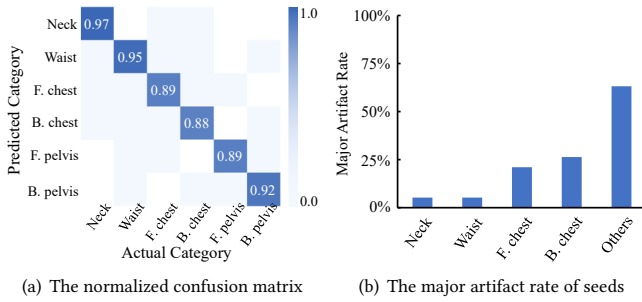


Fig. 10. Failure analysis of ineffective seeding. The effectiveness of our seeding process can be compromised in two ways: firstly, by the system’s inability to identify neck, waist, or chest pieces as depicted in (a); and secondly, by the failure of the identified seeds to prevent major artifacts, as shown in (b). The data in (b) further suggests that neck and waist pieces are more effective seeds compared to chest pieces and others.

twice in this example. Fortunately, our experiments, summarized in Fig. 9b, show that 52 percent of the examples require zero or one revisit, and only 10 percent of the examples need more than two revisits. Consequently, our system can process the majority of the examples within 10 seconds as Fig. 9c shows.

Seed selection is pivotal in determining the number of revisits. In general, using better and more seeds can considerably reduce the need for revisits, associated computational costs, and even the occurrence of artifacts, as further discussed in Subsection 8.3. However, evaluating seed quality can be challenging without running the whole system at the first place. To ensure a fair assessment that mirrors real-world usage cases, we report performance using the very first effective seed in our experiments.

8.3 Failure Evaluation

The system successfully initializes a garment, if the whole process involves zero user intervention and the result contains no obvious artifacts. In this regard, the system can fail for two reasons.

8.3.1 Ineffective seeding. We consider a seed to be ineffective if it leads to major artifacts, such as those in Fig. 11a, which cannot be rectified through simple manual intervention. According to our system, this ineffectiveness can stem from the system’s inability to identify neck, waist, or chest pieces, or from its failure to prevent major artifacts even when these pieces are identified.

Our dataset indicates that neck or waist pieces are present in 86 percent of the patterns. Furthermore, as illustrated in Fig. 10a, our system boasts a 96 percent success rate in detecting either a neck or waist piece and an 89 percent success rate for chest piece detection. This results in an overall approximate success rate of 95 percent for identifying any of these pieces.

However, identifying these pieces does not guarantee effectiveness. As summarized in Fig. 10b, major artifacts may still emerge when the system relies on these pieces as seeds. In this experiment, we assume that both left and right chest pieces are selected as seeds, when they are both available, otherwise the system is more likely to fail as shown in Fig. 11a. Additionally, we assume that the initial shapes of other pieces are predetermined, given the challenges in

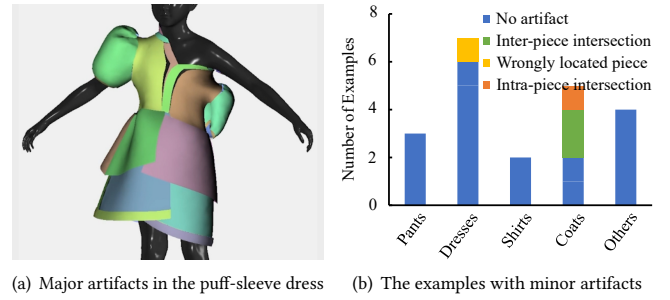


Fig. 11. Major and minor artifacts reported in our examples. While major artifacts cannot be predicted without running the whole system and they require manual seed re-selection, minor artifacts can be addressed by straightforward manual operations during the post-processing stage.

assigning their body part locations according to Fig. 3a, especially for smaller pieces. In practice, two out of 21 garments in our experiments, the quilted coat in Fig. 1 and the slip dress in Fig. 13i, cannot be initialized by using automatically selected seeds, necessitating manual intervention to re-select additional pieces as seeds.

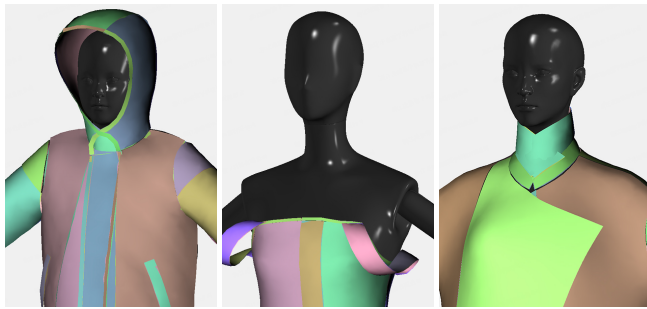
8.3.2 Minor artifacts. While the use of an effective seed prevents major artifacts, it may still cause minor artifacts, including inter-piece intersection, wrongly located piece, and intra-piece intersection shown in Fig. 12. According to our experiments in Fig. 11b, 81 percent of the garments can be initialized with no minor artifacts. Among the minor artifacts, inter-piece intersection is probably the simplest one as it is due to an incorrect arrangement order. For instance, suppose that \mathcal{P}_a and \mathcal{P}_c are two connected pieces and \mathcal{P}_b overlaps with them. When the order is $\mathcal{P}_a \leftarrow \mathcal{P}_b \leftarrow \mathcal{P}_c$, \mathcal{P}_b would inevitably intersect with \mathcal{P}_a or \mathcal{P}_c . To resolve this artifact, users can rearrange the order and redo simulation. The other artifacts, although slightly more complex, can also be fixed through simple manual intervention, which is beyond the scope of this study.

In summary, using automatically selected seeds, our system initializes 15 out of 21 garments without encountering any major or minor artifacts. With a seed detection rate of 95 percent, we estimate our system’s overall success rate at about 68 percent. Moreover, the instances where the system does not succeed can still be addressed through straightforward user operations.

8.4 Limitations

According to Subsection 4, our system has specific requirements for body and pattern inputs. When these requirements are not met, the system may experience inefficiency or failures. In particular, the requirement for pieces to face away from the body can contradict the design of certain garment features, such as linings and collars. Failure to address this issue can lead to inter-piece intersections caused by incorrect repulsion directions. Currently, we mitigate this challenge by segmenting these pieces and flipping their orientations back and forth during pre-processing and post-processing, thereby undermining the automated flow of our system.

Even with the requirements met, ineffective seeding, especially if the pattern misses neck, waist or chest pieces, can cause the system



(a) Inter-piece intersection (b) Wrongly located pieces (c) Intra-piece intersection



(d) Inter-piece intersection fixed by user intervention (e) Wrongly located pieces fixed by user intervention (f) Intra-piece intersection fixed by user intervention

Fig. 12. Typical minor artifacts occurred to the garments initialized by our system. While all of them can be fixed by user intervention as shown in (d), (e), and (f), our implementation provides an option for users to fix inter-piece intersection only.

to fail to work automatically. After the system arranges the garment, the initialized shape may still contain minor artifacts as shown in Subsection 8.3, some of which can be resolved by simple user intervention. Our system, using downsampled mesh resolution and signed distance function, does not account for cloth intersections with detailed body parts such as finger tips and hair strands. Finally, it is intriguing to know if the system can handle garments designed for humanoid and animal bodies, as we have not carried out such tests yet.

9 CONCLUSIONS AND FUTURE WORK

We present an automatic garment initialization system, which serves as the basis for automatic digital garment creation. The effectiveness of our system hinges on the assumption that the sewing relationships provide sufficient guidance for phased initialization of each pattern piece. But this assumption also implies that the system’s automation relies on the choice of initial seeds and the arrangement order, which is not guaranteed to be optimal in all cases.

In the future, we plan to validate our system with additional garment cases and diverse human body avatars, with a primary focus on improving its reliability and efficiency. We have a strong interest in leveraging AI models to reduce the system’s dependency on seed selection and arrangement order. Additionally, we aim to refine the system to eliminate the requirement for sewing patterns to

include complete sewing relationships. This change is particularly important as it eases the burden on digital pattern makers and pattern generation algorithms.

ACKNOWLEDGMENTS

We thank Vittie Zhou and the digital product creation team at Style3D for technical support and troubleshooting assistance. This work is partially supported by NSF 2301040, 2008915, 2244651, 200856, and the key R&D program (No. 2024C01069) of Zhejiang Province, China.

REFERENCES

- Seungbae Bang, Maria Korosteleva, and Sung-Hee Lee. 2021. Estimating Garment Patterns from Static Scan Data. *Comput. Graph. Forum* 40, 6 (Sept. 2021), 273–287.
- David Baraff and Andrew Witkin. 1998. Large Steps in Cloth Simulation. In *Proceedings of SIGGRAPH 98 (Computer Graphics Proceedings, Annual Conference Series)*, Eugene Fiume (Ed.). ACM, 43–54.
- David Baraff, Andrew Witkin, and Michael Kass. 2003. Untangling Cloth. *ACM Trans. Graph. (SIGGRAPH)* 22, 3 (July 2003), 862–870.
- Aric Bartle, Alla Sheffer, Vladimir G. Kim, Danny M. Kaufman, Nicholas Vining, and Floraine Berthouzo. 2016. Physics-Driven Pattern Adjustment for Direct 3D Garment Editing. *ACM Trans. Graph. (SIGGRAPH)* 35, 4, Article 50 (July 2016), 11 pages.
- Miklos Bergou, Max Wardetzky, David Harmon, Denis Zorin, and Eitan Grinspun. 2006. A Quadratic Bending Model for Inextensible Surfaces. In *Proceedings of SGP*. 227–230.
- Floraine Berthouzo, Akash Garg, Danny M. Kaufman, Eitan Grinspun, and Maneesh Agrawala. 2013. Parsing Sewing Patterns into 3D Garments. *ACM Trans. Graph. (SIGGRAPH)* 32, 4, Article 85 (July 2013), 12 pages.
- Sofien Bouaziz, Sebastian Martin, Tiantian Liu, Ladislav Kavan, and Mark Pauly. 2014. Projective Dynamics: Fusing Constraint Projections for Fast Simulation. *ACM Trans. Graph. (SIGGRAPH)* 33, 4, Article 154 (July 2014), 11 pages.
- Robert Bridson, Ronald Fedkiw, and John Anderson. 2002. Robust Treatment of Collisions, Contact and Friction for Cloth Animation. *ACM Trans. Graph. (SIGGRAPH)* 21, 3 (July 2002), 594–603.
- Robert Bridson, Sebastian Marino, and Ronald Fedkiw. 2003. Simulation of Clothing with Folds and Wrinkles. In *Proceedings of SCA*. 28–36.
- Tyson Brochu, Essex Edwards, and Robert Bridson. 2012. Efficient Geometrically Exact Continuous Collision Detection. *ACM Trans. Graph. (SIGGRAPH)* 31, 4, Article 96 (July 2012), 7 pages.
- Thomas Buffet, Damien Rohmer, Loïc Barthe, Laurence Boissieux, and Marie-Paule Cani. 2019. Implicit Untangling: A Robust Solution for Modeling Layered Clothing. *ACM Trans. Graph. (SIGGRAPH)* 38, 4, Article 120 (July 2019), 12 pages.
- Yunuo Chen, Tianyi Xie, Cem Yuksel, Danny Kaufman, Yin Yang, Chenfanfu Jiang, and Minchen Li. 2023. Multi-Layer Thick Shells. In *ACM SIGGRAPH 2023 Conference Proceedings (Los Angeles, CA, USA) (SIGGRAPH '23)*. Article 25, 9 pages.
- Kwang-Jin Choi and Hyeong-Seok Ko. 2002. Stable but Responsive Cloth. *ACM Trans. Graph. (SIGGRAPH)* 21, 3 (July 2002), 604–611.
- Gabriel Cirio, Jorge Lopez-Moreno, David Miraut, and Miguel A. Otaduy. 2014. Yarn-Level Simulation of Woven Cloth. *ACM Trans. Graph. (SIGGRAPH Asia)* 33, 6, Article 207 (Nov. 2014), 11 pages.
- Chihiro Goto and Nobuyuki Umetani. 2021. Data-Driven Garment Pattern Estimation from 3D Geometries. In *Eurographics 2021 - Short Papers*. The Eurographics Association.
- Takeo Igarashi and John F. Hughes. 2002. Clothing Manipulation. In *Proceedings of UIST (Paris, France)*. 91–100.
- Jonathan M. Kaldor, Doug L. James, and Steve Marschner. 2008. Simulating Knitted Cloth at the Yarn Level. *ACM Trans. Graph. (SIGGRAPH)* 27, 3, Article 65 (Aug. 2008), 9 pages.
- Jonathan M. Kaldor, Doug L. James, and Steve Marschner. 2010. Efficient Yarn-Based Cloth with Adaptive Contact Linearization. *ACM Trans. Graph. (SIGGRAPH)* 29, 4, Article 105 (July 2010), 10 pages.
- Maria Korosteleva and Sung-Hee Lee. 2021. Generating Datasets of 3D Garments with Sewing Patterns. In *Proceedings of NeurIPS*, J. Vanschoren and S. Yeung (Eds.), Vol. 1. Maria Korosteleva and Sung-Hee Lee. 2022. NeuralTailor: Reconstructing Sewing Pattern Structures from 3D Point Clouds of Garments. *ACM Trans. Graph. (SIGGRAPH)* 41, 4, Article 158 (July 2022), 16 pages.
- Maria Korosteleva and Olga Sorkine-Hornung. 2023. GarmentCode: Programming Parametric Sewing Patterns. *ACM Trans. Graph. (SIGGRAPH Asia)* 42, 6, Article 199 (Dec. 2023), 15 pages.
- Jie Li, Gilles Daviet, Rahul Narain, Florence Bertails-Descoubes, Matthew Overby, George E. Brown, and Laurence Boissieux. 2018. An Implicit Frictional Contact

- Solver for Adaptive Cloth Simulation. *ACM Trans. Graph. (SIGGRAPH)* 37, 4, Article 52 (July 2018), 15 pages.
- Minchen Li, Zachary Ferguson, Teseo Schneider, Timothy Langlois, Denis Zorin, Daniele Panozzo, Chenfanfu Jiang, and Danny M. Kaufman. 2020. Incremental Potential Contact: Intersection- and Inversion-Free Large Deformation Dynamics. *ACM Trans. Graph. (SIGGRAPH)* 39, 4, Article 49 (July 2020), 20 pages.
- Lijuan Liu, Xiangyu Xu, Zhijie Lin, Jiabin Liang, and Shuicheng Yan. 2023. Towards Garment Sewing Pattern Reconstruction from a Single Image. *ACM Trans. Graph. (SIGGRAPH Asia)* 42, 6, Article 200 (Dec. 2023), 15 pages.
- Tiantian Liu, Adam W. Bargteil, James F. O'Brien, and Ladislav Kavan. 2013. Fast Simulation of Mass-Spring Systems. *ACM Trans. Graph. (SIGGRAPH Asia)* 32, 6, Article 214 (Nov. 2013), 7 pages.
- Jiaxin Lu, Yifan Sun, and Qixing Huang. 2023. Jigsaw: Learning to Assemble Multiple Fractured Objects. In *Proceedings of NeurIPS*.
- Mickaël Ly, Romain Casati, Florence Bertails-Descoubes, Mélina Skouras, and Laurence Boissieux. 2018. Inverse Elastic Shell Design with Contact and Friction. *ACM Trans. Graph. (SIGGRAPH Asia)* 37, 6, Article 201 (Dec. 2018), 16 pages.
- Mickaël Ly, Jean Jouve, Laurence Boissieux, and Florence Bertails-Descoubes. 2020. Projective Dynamics with Dry Frictional Contact. *ACM Trans. Graph. (SIGGRAPH)* 39, 4, Article 57 (July 2020), 8 pages.
- Eder Miguel, Derek Bradley, Bernhard Thomaszewski, Bernd Bickel, Wojciech Matusik, Miguel A. Otaduy, and Steve Marschner. 2012. Data-Driven Estimation of Cloth Simulation Models. *Comput. Graph. Forum (Eurographics)* 31, 2 (May 2012).
- Eder Miguel, Rasmus Tamstorf, Derek Bradley, Sara C. Schwartzman, Bernhard Thomaszewski, Bernd Bickel, Wojciech Matusik, Steve Marschner, and Miguel A. Otaduy. 2013. Modeling and Estimation of Internal Friction in Cloth. *ACM Trans. Graph. (SIGGRAPH Asia)* 32, 6, Article 212 (Nov. 2013), 10 pages.
- Matthias Müller, Bruno Heidelberger, Matthias Teschner, and Markus Gross. 2005. Meshless Deformations Based on Shape Matching. *ACM Trans. Graph. (SIGGRAPH)* 24, 3 (July 2005), 471–478.
- Rahul Narain, Matthew Overby, and George E. Brown. 2016. ADMM \supseteq Projective Dynamics: Fast Simulation of General Constitutive Models. In *Proceedings of SCA*. 21–28.
- Rahul Narain, Armin Samii, and James F. O'Brien. 2012. Adaptive Anisotropic Remeshing for Cloth Simulation. *ACM Trans. Graph. (SIGGRAPH Asia)* 31, 6, Article 152 (Nov. 2012), 10 pages.
- Stanley Osher and Ronald Fedkiw. 2002. *Level Set Methods and Dynamic Implicit Surfaces (Applied Mathematical Sciences, 153)*. Springer.
- Nico Pietroni, Corentin Dumery, Raphael Falque, Mark Liu, Teresa Vidal-Calleja, and Olga Sorkine-Hornung. 2022. Computational Pattern Making from 3D Garment Models. *ACM Trans. Graph. (SIGGRAPH)* 41, 4, Article 157 (July 2022), 14 pages.
- Silvia Sellán, Heng Yi Cheng, Yuming Ma, Mitchell Dembowski, and Alec Jacobson. 2019. Solid Geometry Processing on Deconstructed Domains. *Comput. Graph. Forum* 38, 1 (Feb. 2019), 564–579.
- Georg Sperl, Rosa M. Sánchez-Banderas, Manwen Li, Chris Wojtan, and Miguel A. Otaduy. 2022. Estimation of Yarn-Level Simulation Models for Production Fabrics. *ACM Trans. Graph. (SIGGRAPH)* 41, 4, Article 65 (July 2022), 15 pages.
- Rasmus Tamstorf, Toby Jones, and Stephen F. McCormick. 2015. Smoothed Aggregation Multigrid for Cloth Simulation. *ACM Trans. Graph. (SIGGRAPH Asia)* 34, 6, Article 245 (Oct. 2015), 13 pages.
- Min Tang, tongtong wang, Zhongyuan Liu, Ruofeng Tong, and Dinesh Manocha. 2018. I-Cloth: Incremental Collision Handling for GPU-Based Interactive Cloth Simulation. *ACM Trans. Graph. (SIGGRAPH Asia)* 37, 6, Article 204 (Dec. 2018), 10 pages.
- Nobuyuki Umetani, Danny M. Kaufman, Takeo Igarashi, and Eitan Grinspun. 2011. Sensitive Couture for Interactive Garment Modeling and Editing. *ACM Trans. Graph. (SIGGRAPH)* 30, 4, Article 90 (July 2011), 12 pages.
- Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. In *International Conference on Learning Representations*. 1–12.
- Pascal Volino and Nadia Magnenat-Thalmann. 2006. Resolving Surface Collisions through Intersection Contour Minimization. *ACM Trans. Graph. (SIGGRAPH)* 25, 3 (Jul 2006), 1154–1159.
- Pascal Volino, Nadia Magnenat-Thalmann, and Francois Faure. 2009. A Simple Approach to Nonlinear Tensile Stiffness for Accurate Cloth Simulation. *ACM Trans. Graph.* 28, 4, Article 105 (Sept. 2009), 16 pages.
- Pascal Volino and Nadia Magnenat Thalmann. 2000. Implementing Fast Cloth Simulation with Collision Response. In *Proceedings of CGI*. 257–266.
- Huamin Wang. 2018. Rule-Free Sewing Pattern Adjustment with Precision and Efficiency. *ACM Trans. Graph. (SIGGRAPH)* 37, 4, Article 53 (July 2018), 13 pages.
- Huamin Wang, James F. O'Brien, and Ravi Ramamoorthi. 2011. Data-Driven Elastic Models for Cloth: Modeling and Measurement. *ACM Trans. Graph. (SIGGRAPH)* 30, 4, Article 71 (July 2011), 9 pages.
- Huamin Wang and Yin Yang. 2016. Descent Methods for Elastic Body Simulation on the GPU. *ACM Trans. Graph. (SIGGRAPH Asia)* 35, 6, Article 212 (Nov. 2016), 10 pages.
- Tuanfeng Y. Wang, Duygu Ceylan, Jovan Popović, and Niloy J. Mitra. 2018. Learning a Shared Shape Space for Multimodal Garment Design. *ACM Trans. Graph. (SIGGRAPH Asia)* 37, 6, Article 203 (Dec. 2018), 13 pages.
- Zhendong Wang, Yin Yang, and Huamin Wang. 2023. Stable Discrete Bending by Analytic Eigensystem and Adaptive Orthotropic Geometric Stiffness. *ACM Trans. Graph. (SIGGRAPH Asia)* 42, 6, Article 183 (Dec. 2023), 16 pages.
- Haomiao Wu and Theodore Kim. 2023. An Eigenanalysis of Angle-Based Deformation Energies. *Proc. ACM Comput. Graph. Interact. Tech. (SCA)* 6, 3, Article 40 (Aug. 2023), 19 pages.
- Longhua Wu, Botao Wu, Yin Yang, and Huamin Wang. 2020. A Safe and Fast Repulsion Method for GPU-Based Cloth Self Collisions. *ACM Trans. Graph.* 40, 1, Article 5 (Dec. 2020), 18 pages.
- Shan Yang, Zherong Pan, Tanya Amert, Ke Wang, Licheng Yu, Tamara Berg, and Ming C. Lin. 2018. Physics-Inspired Garment Recovery from a Single-View Image. *ACM Trans. Graph.* 37, 5, Article 170 (Nov. 2018), 14 pages.
- Zaiwei Zhang, Zhenpei Yang, Chongyang Ma, Linjie Luo, Alexander Huth, Etienne Vouga, and Qixing Huang. 2020. Deep Generative Modeling for Scene Synthesis via Hybrid Representations. *ACM Trans. Graph.* 39, 2, Article 17 (April 2020), 21 pages.



Fig. 13. The digital garments produced by a physics-based simulator following our initialization process. Note that our initialization excludes accessory details like zippers, buttons, and laces, as well as simulation specifics like filling and folding. These elements require additional user editing during or after simulation. ACM Trans. Graph., Vol. 43, No. 4, Article 74. Publication date: July 2024.