TailorCoPilot: Enabling Agentic Pattern Making with Version-Controlled State Tracking

Yuexin Sun
Donghua University
Shanghai, China
ysun@mail.dhu.edu.cn

Zhaohui Wang*
Donghua University
Shanghai, China
wzh_sh2007@dhu.edu.cn

Ruiyang Liu†
Style3D Research
HangZhou, China
novemree@gmail.com

Demian Kong
Style3D Research
Hangzhou, China
kongdemian@linctex.com

Qian He
Style3D Research
HangZhou, China
heqianhailie@gmail.com

Gaofeng He
Style3D Research
HangZhou, China
hegaofeng@linctex.com

Huamin Wang
Style3D Research
HangZhou, China
wanghmin@gmail.com

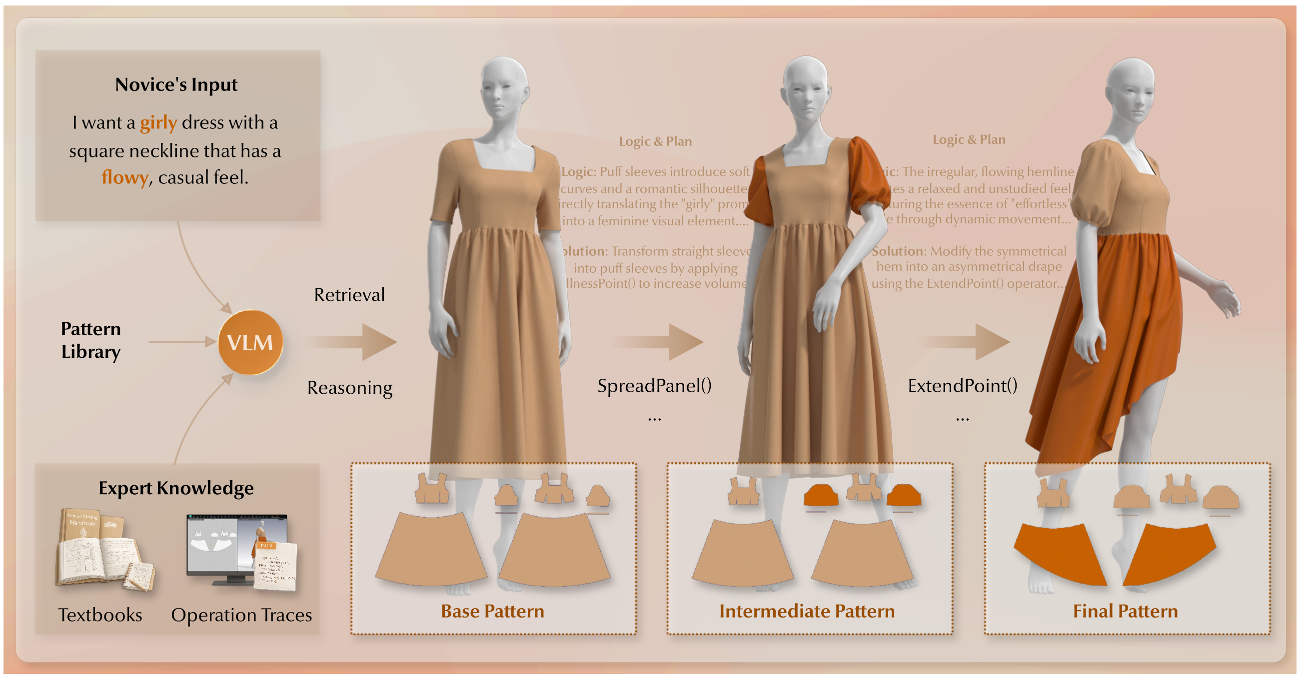


Figure 1: Overview of TailorCoPilot for novice garment pattern making. Given a novice's natural-language design intent, TailorCoPilot retrieves a compatible base pattern from a pattern library and uses a vision-language model (VLM) to reason based on embedded expert knowledge from textbooks and operation traces. It then proposes stepwise editable pattern operations, such as `SpreadPanel()` and `ExtendPoint()`, that progressively transform the base pattern into intermediate and final patterns, with corresponding changes in the 3D garment form.

Abstract

Experience-driven manufacturing, such as garment pattern making, faces a severe generational skills gap because its core expertise relies on undocumented tacit knowledge forged through day-to-day practice. To address this challenge, we present TailorCoPilot, an agentic pattern-making system built upon a specially designed version-control backend TailorTrace. TailorTrace models sewing patterns as structured, discrete states and records their transformations during the pattern-making process as explicit operation sequences defined upon the geometry primitives in the sewing pattern (panels, edges, vertices and stitches). Integrated into a conventional

*Corresponding author.
†Project leader.
Yuexin Sun, Zhaohui Wang are affiliated with College of Fashion and Design, Donghua University, and Key Laboratory of Clothing Design & Technology, Ministry of Education. This research work is conducted at Style3D Research.

UIST '26, Detroit, MI, USA

ACM ISBN 979-8-4007-2856-3/2026/11
https://doi.org/10.1145/3830398.3830502

pattern-making GUI, TailorTrace enables seamless documentation of senior experts' tacit pattern-making knowledge without breaking their daily workflow. The documented knowledge further offers interactive, pedagogical scaffolding for novices, while providing a robust foundation to power TailorCoPilot and train future generative AI models. In a user study with novices and advanced novices, TailorCoPilot improved task completion rates, reduced time and perceived workload, and yielded higher-quality artifacts compared to skill-appropriate baselines. Ultimately, TailorCoPilot demonstrates a viable pathway to capture practice-based expertise, operationalizing it to support both generative AI advancements and human apprenticeship.

CCS Concepts

• **Human-centered computing** → **Interactive systems and tools**; • **Applied computing** → *Computers in other domains*; • **Computing methodologies** → *Knowledge representation and reasoning*.

Keywords

Garment Pattern Making, Expert Process Knowledge, State Tracking, Traceable Workflows, Human-AI Collaboration

ACM Reference Format:
Yuexin Sun, Zhaohui Wang, Ruiyang Liu, Demian Kong, Qian He, Gaofeng He, and Huamin Wang. 2026. TailorCoPilot: Enabling Agentic Pattern Making with Version-Controlled State Tracking. In *The 39th Annual ACM Symposium on User Interface Software and Technology (UIST '26), November 02–05, 2026, Detroit, MI, USA.* UIST '26, Detroit, MI, USA, 13 pages. https://doi.org/10.1145/3830398.3830502

1 Introduction

Experience-driven manufacturing domains rely heavily on expertise built through repeated practice and only rarely formalized in textbooks [51]. As senior practitioners retire and fewer newcomers enter these trades, a large body of practical knowledge faces the risk of being lost before it can be passed on [18, 27, 44]. This problem is especially visible in garment pattern making. Experienced pattern makers develop judgment through years of hands-on work, learning how to translate design intent into production-ready pattern structures while preserving fit, balance, and sewability [19, 39]. Much of this knowledge is embedded in procedural habits, intermediate decisions, and localized repair strategies, making it nearly impossible to reverse-engineer from finished artifacts alone [25]. Addressing this gap requires a dedicated, trace-capturing infrastructure to document the nuanced steps of expert workflows—an objective that serves as the core motivation for this research.

Existing computational tools mostly focus on modeling final geometric outcomes, while failing to capture the crucial intermediate steps of pattern making. Professional Computer-Aided Design (CAD) systems provide robust precision and editing power, but they function largely as blank canvases. They operate under the assumption that users have already mastered the tacit drafting logic and established conventions necessary to build a pattern from scratch [61]. To lower this learning barrier, some providers have introduced parametric, template-driven systems; however, these inherently sacrifice creative degrees of freedom by restricting designers to predefined structural boundaries. More recently, generative AI has attempted to automate the craft by predicting system parameters or directly outputting final vector layouts [17, 38]. Yet, due to the probabilistic nature of generative models, their outputs struggle to consistently meet the strict geometric tolerances required for industrial manufacturing, while the absence of intermediate drafting states means that human practitioners lack the procedural context needed to interpret, adjust, or locally repair the AI-generated results.

To address these challenges, we introduce **TailorTrace**, a version-controlled computational framework designed to track and structure state changes throughout garment pattern development. By representing sewing patterns as discrete states and recording transitions as explicit geometric operations, TailorTrace renders the inherently opaque pattern-making process legible at the level of states, actions, and revisions. This architecture enables expert workflows to be captured as interpretable traces, revisited step-by-step, and reused as operational resources. Crucially, by documenting tacit expert knowledge as explicit operational traces, TailorTrace establishes a robust data foundation for future generative AI, allowing models to learn from expert-defined transformations rather than relying solely on static, final-outcome examples.

Building upon this trace-capturing infrastructure, we present **TailorCoPilot**, an agentic co-creation system that leverages these recorded workflows to provide state-aware, interactive assistance. During the drafting process, TailorCoPilot actively surfaces relevant revision histories and intermediate structures, offering essential procedural context and guided scaffolding for digital apprenticeship [10]. We evaluate TailorCoPilot through a user study involving novice and advanced-novice pattern makers, comparing its efficacy against baseline tools matched to their respective skill levels. Our results demonstrate quantitative and qualitative improvements in task completion rates, time efficiency, perceived workload, and final artifact quality. We summarize the primary contributions of this paper as follows:

- A process-centered framing of garment pattern making as traceable iterative revision.
- TailorTrace, a CAD-integrated, version-controlled representation of validated pattern states and domain-specific transformations[1].
- TailorCoPilot, an interactive agentic system that provides stepwise, editable, and human-controllable support through traceable states.
- Evidence from user and diagnostic studies that trace-based support improves the performance of novices and advanced novices.

2 Related Work

2.1 Garment Pattern-Making Tools

Digital support for garment pattern making has developed along several directions. Industrial CAD systems such as Lectra and ET [52, 53] support precise drafting and production workflows, but they typically assume substantial knowledge of geometric

[1]The current evidence is limited to bounded pattern-revision tasks in a controlled setting

operations, drafting logic, and professional conventions [5, 31, 47, 61]. Parametric methods further improve efficiency through measurement-driven rules and constrained variables [24, 28, 35], and recent surveys highlight the growing role of pattern-centered workflows in digital fashion design [38, 39].

Recent work has explored structured representations and learning-based approaches for garment generation. GarmentCode represents sewing patterns as executable programs [30], and GarmentCodeData provides large-scale paired datasets [29]. SewFormer and other generative models support garment and sewing-pattern reconstruction or generation from visual, textual, or multimodal input [8, 21, 33, 34, 36, 37, 42]. These end-to-end approaches can produce complete pattern outputs, but their intermediate decisions and revision paths are often difficult to inspect, edit, or reuse. ChatGarment [4] and Design2GarmentCode [65] use VLMs to infer higher-level design parameters for constrained pattern generation. Such parametric AI methods improve controllability for predefined garment variations, yet their flexibility is often bounded by the available templates and control variables. In contrast, TailorTrace and TailorCoPilot represent pattern making as traceable meta-operations over validated pattern states, supporting finer-grained revision while preserving interpretable transitions between states.

2.2 Capturing Expert Processes in Design Work

A central challenge in the garment industry is that much of the relevant expertise is tacit and situated in practice. Prior work [2, 19, 43] notes that expert knowledge often resides in judgment, collaboration, and context-sensitive decision making that is difficult to document in formal rules alone. This challenge is especially acute in pattern making, where local edits often need to be evaluated in relation to downstream construction constraints and interactions among multiple pieces.

In computational design, structured representations have been used to encode geometry, constraints, and design procedures. DeepCAD models CAD construction sequences [60], SketchGraphs [49] represents relational sketch structure, and Vitruvion [50] generates parametric CAD sketches. More recent approaches [7, 26] such as solver-aided CAD languages, scene programs [64], and workflow-capture systems like ClearFairy [54] further demonstrate that procedural steps and design rationales can be represented explicitly. However, these approaches primarily encode geometric structure, executable programs, or generic workflows, and do not address how expert revision knowledge can be captured as validated, operation-level traces for coordinated garment-pattern modification.

2.3 Human-AI Support

Recent HCI research increasingly frames AI systems as interactive partners rather than purely generative tools. Work on co-creative systems and human-AI collaboration highlights the importance of user agency, control, and iterative interaction [23, 48, 63]. Related studies [12, 14, 59] further emphasize transparency, task-dependent interfaces, and support for user steering during generation. More recently, agentic AI design systems have begun to incorporate multi-step reasoning, planning, and tool use in creative workflows [1, 54].

Recent work also focuses on making intermediate decisions inspectable and revisable. ArtKrit [40] introduces computational scaffolding for structured artistic workflows, while DesignTrace [45], GenTune [58], and ImaginationVellum [41] support traceable refinement and iterative design. However, these systems target general creative refinement and do not address manufacturing-oriented revision, where each operation must preserve geometric, topological, and physical validity. Reviews of interactive and user-centered AI argue for moving beyond post-hoc explanations toward interaction-time support [22, 46], and work in fashion design shows that graphical interfaces can better support exploration than text prompting alone [11]. Together, these studies highlight the importance of inspectability, controllability, and iterative refinement in human-AI design systems. We extend this perspective to garment pattern making through validated operation traces that support interactive revision and novice learning.

3 Formative Study

Garment pattern making is fundamentally a complex process of inverse spatial deconstruction. It converts a conceptual design, typically a 2D stylized illustration of the garment's desired 3D draped form on a human avatar, into a precise set of flat 2D geometric primitives. Unlike traditional engineering CAD where 3D modeling is largely "what-you-see-is-what-you-get", pattern making is highly indirect and requires profound spatial intuition from the pattern makers to mentally deconstruct the 3D illustration into flat pattern pieces with notch notations for assembly, while accounting for physical properties of fabrics and strict dimensional precision. For novice practitioners, cultivating this dynamic mental model purely from static textbooks is extremely difficult, not to mention that such complex 2D-to-3D geometric reasoning remains a significant blind spot for current generative AI systems.

TailorCoPilot is designed to bridge the expertise gap between novice and expert pattern makers and provide the missing infrastructure required to integrate generative AI into the complex pattern-making workflows. Before delving into TailorCoPilot, we conducted a formative study to identify how expert pattern makers manage interconnected geometric constraints between 2D patterns and 3D draping, where novices fail to anticipate downstream consequences, and what foundational barriers underlie this gap.

3.1 Participants and Procedure

We recruited 13 participants through purposive sampling, following the early stages of the Dreyfus model of skill acquisition [15]. The sample included **6 novices** (PF1–PF6) with limited or no pattern-making experience, **4 junior pattern-making students as advanced novices** (PF7–PF10) who had acquired basic pattern-making skills via textbooks, and **3 experts** (PF11–PF13) who were senior pattern makers or pattern-making educators with more than 5 years of industry experience. Each session lasted about 60 minutes, including a 30-minute semi-structured interview and a 30-minute task session. In the interview, we asked about participants' workflows, how they learned or taught pattern-making knowledge, how they decomposed garment changes, common breakdowns, and the criteria they used to judge whether a modification was structurally and aesthetically sound.

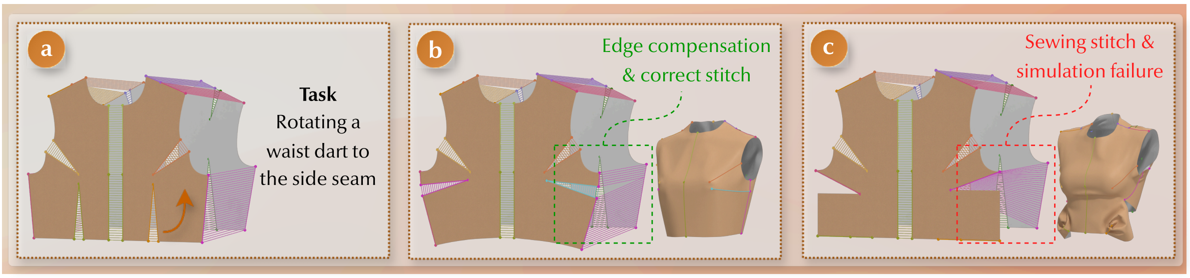
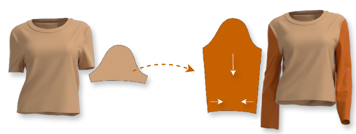
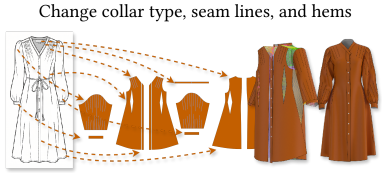


Figure 2: Example from the formative study. (a) Base patterns for a dart-rotation task. (b) Correct result, which requires both dart relocation and corresponding seam adjustment and edge compensation. (c) A novice participant's actual result. Only the dart position was changed, while the stitch relationship was left unresolved, leading to sewing and simulation failure.

Table 1: Representative tasks categorized by difficulty. Simple tasks involve local parametric refinements to base patterns, such as length or width adjustments. Medium tasks encompass modular component synthesis and constrained topological operations, such as dart transfers, pleating and gathering. Complex tasks require reference-driven deconstruction from design sketches or advanced structural transformations.

Difficulty	Count	Typical Case
Simple	15	 Convert short sleeves to long sleeves
Medium	10	 Change collar type, seam lines, and hems
Complex	5	 Create a dress from a reference image

In the task sessions, participants performed representative tasks across three difficulty levels (Simple, Medium, and Difficult), encompassing operations ranging from local refinements to structural modifications and reference-driven creations (Table 1). To ensure ecological validity, these tasks were validated by expert educators to mirror authentic practitioner workflows, specifically mapping design initiation, progression trajectories, and the exact points where participants encountered friction. During the session, participants were allowed to use their preferred CAD environments, supplemented by Style3D [55] for virtual draping and AI-based pattern assistants [42, 65]. To minimize cognitive interference during the session, we employed retrospective think-aloud protocols [13] and screen recordings to isolate critical decision points, workflow breakdowns, and repair strategies.

We collected interview transcripts, task recordings, and resulting 2D and 3D artifacts, and analyzed them using reflexive thematic analysis [6]. Preliminary clustering was used to organize recurring issues in the textual data [62], after which two researchers reviewed the materials, refined the themes, and derived implications for system design.

3.2 Findings

Our formative study yielded four findings about how pattern making begins, why expert knowledge remains hard to access, where less experienced users struggle, and how participants judge whether a revision is working. Together, these findings motivate a system that starts from an initial base pattern, digitizes expert pattern-making knowledge into reusable traces, and keeps revision inspectable and revisable.

3.2.1 Workflow. Among all difficulty levels, unless necessary, participants rarely draft sewing patterns from scratch. Instead, they usually started from a pattern close to the target and modified it through comparison. Advanced novices and experts often drew this starting point from a pattern library or AI-assisted generation, then continued editing in CAD. Novices lacked such resources and could not operate CAD, so they relied almost entirely on end-to-end AI generation. As one expert explained, "We usually begin from the closest block we can find, then modify toward the target" (PF11). This suggests that revision support should first establish an editable base pattern.

3.2.2 Digitizing Expert Knowledge. Experts did not frame pattern making as fixed rules alone. They combined structural principles, prior cases, embodied judgment, and process habits to decide what to change, in what order, and whether a result remained sound. For less experienced users, this knowledge was hard to access because it was scattered across textbooks, teaching, examples, and personal practice. Revisions were also path-dependent, since later decisions relied on earlier ones and experts often reasoned through intermediate states. As one participant noted, "The know-how is not just the rule itself, but knowing when to use it, what to adjust next, and how to tell if it is going wrong" (PF13). This suggests that a central design challenge is how to digitize expert knowledge as reusable traces, decision criteria, and revision records, rather than preserving only final outcomes.

3.2.3 Barriers From Intents to Edits. A central difficulty was turning intended appearance into executable edits. Novices could often describe a desired effect, but could not determine what part of the pattern to change or what operation would produce it. For example, one participant wanted a silhouette that felt "softer and more expanded" (PF2), but could not identify the geometric changes needed for the sewing pattern. Advanced novices could understand concepts such as dart transfer and perform local edits to the sewing pattern, yet still struggled with precision, sequencing, and more complex modifications. As one participant put it, "I can describe the style I want, but I still do not know what to adjust first on the pattern" (PF5). These observations indicate that expert support becomes useful only when intent can be translated into inspectable steps and editable parameters.

3.2.4 3D Feedback and Intervention. Novice participants generally relied heavily on 3D try-on and other visual feedback to judge whether a revision was working. They often could not assess the consequences of a local 2D change until sewing or simulation exposed them. For example, one novice (PF3) completed a dart change without resolving the corresponding seam relationship, leading to simulation failure (Figure 2). Experts used intermediate visualizations to check proportion, balance, seam compatibility, and wearing effect, while less experienced users needed to inspect the evolving result before deciding what to do next. As one advanced novice remarked, "I need to see the garment update step by step before I can trust the modification" (PF8). This suggests that support tools should allow users to inspect and intervene in the revision process itself.

4 System Design

Guided by our formative study, we introduce TailorTrace (Sec. 4.1), a novel interaction paradigm for pattern making that bridges tacit, physical-world pattern-making expertise with a structured, machine-readable version control backend. Designed as a transparent layer beneath a traditional CAD interface, TailorTrace captures discrete semantic actions, rigorously validates physical constraints, and records these traces as multi-modal commits. Apart from facilitating knowledge transfer between senior and junior pattern makers, the high-quality, structured trajectory data recorded with TailorTrace further enables the design of our agentic pattern-making system TailorCoPilot (Sec. 4.2). Further details of the system design are provided in Supplementary Section 2.

4.1 TailorTrace Design

4.1.1 Unified State Representation. Unlike traditional software version control where intermediate commits may contain broken or non-compiling code, TailorTrace enforces a strict "Valid State" paradigm. Every commit represents a functionally viable garment-pattern state modeled as a structured discrete state encompassing both geometric primitives and rich semantic metadata, with draping information included when available. Formally, we define a garment state as a tuple $\Sigma = \langle \mathcal{P}, \mathcal{G}, \mathcal{M} \rangle$ involving:

- **Structured 2D Geometry** ($\mathcal{P} = (P, S)$) with a set of 2D panels $P = \{p_1, \ldots, p_N\}$ and stitches S. Each panel $p_i = (e_{i,1}, \ldots, e_{i,k})$ is a closed loop of curved edges while each stitch $s \in S$ is defined upon a pair of compatible edges (e_a, e_b, ρ_s) with parameters ρ_s.
- **3D Draped Geometry** ($\mathcal{G}$) of the pattern, conditionally instantiated (i.e., $\mathcal{G} \neq \emptyset$) if and only if the initial state or the user's subsequent workflow incorporates 3D simulation.
- **Multi-Modal Metadata** ($\mathcal{M}$): A rich auxiliary representation that encapsulates design descriptions of the garment, rendered images of the 3D garment and physical fabric parameters (e.g., density, bending, and shearing stiffness) if available.

4.1.2 Semantic Action Space and Transformations. To digitize tacit pattern-making knowledge, TailorTrace follows neuro-symbolic design and tracks semantic design intent through explicit transformation sequences [3]. We define a comprehensive action space over the geometric primitives, such as `AddPoint(...)`, `SpreadPanel(...)`, or `MovePoint(...)` with parameters defined by mouse or keyboard input events. By recording the transformation between commits, the system captures the "how" and "where" of a pattern's evolution, creating a highly interpretable revision history.

4.1.3 Validation Engine. To maintain the integrity of the "Valid State" paradigm, we implemented a robust, multi-tiered evaluator that operates at commit time rather than after each individual edit. Users may freely explore and repair intermediate configurations between commits. TailorTrace stores a state only after it passes validation, together with the full sequence of semantic operations that produced it, including repair operations. This design supports reliable revision histories and curated training trajectories for downstream learning, while excluding abandoned or unresolved invalid configurations as standalone states.

In the 2D domain, the evaluator verifies that all contour loops are closed, no geometric self-intersections are present, and topological stitching is valid (e.g., no repeated stitch assignments or unmatched seam edges). When 3D simulation is enabled, the evaluator performs a preliminary draping simulation onto a target avatar, checking for severe self-intersections, inverted normals, or obvious body collision. The 3D checks use the simulator's default collision and normal-validity tolerances. Commits that violate these constraints are rejected, and users can continue editing until the state becomes valid.

4.1.4 TailorTrace Interaction Design. A core interaction goal of TailorTrace is to introduce minimal additional friction into existing workflows [56]. Senior pattern makers possess deep tacit knowledge but often resist disruptive shifts in their tooling. Therefore, the version control mechanics (branching, committing, rebasing) are abstracted away and embedded invisibly behind a standard CAD GUI. Users maintain their existing drafting workflows, utilizing familiar input modalities like standard mouse peripherals or stylus interfaces, while TailorTrace records these workflows as traces in the background, ensuring that cognitive load remains entirely on the creative task rather than on managing version control trees.

4.2 TailorCoPilot: Agentic Pattern Making

Leveraging the structured semantic traces captured by TailorTrace, we developed TailorCoPilot, an agentic pattern-making system

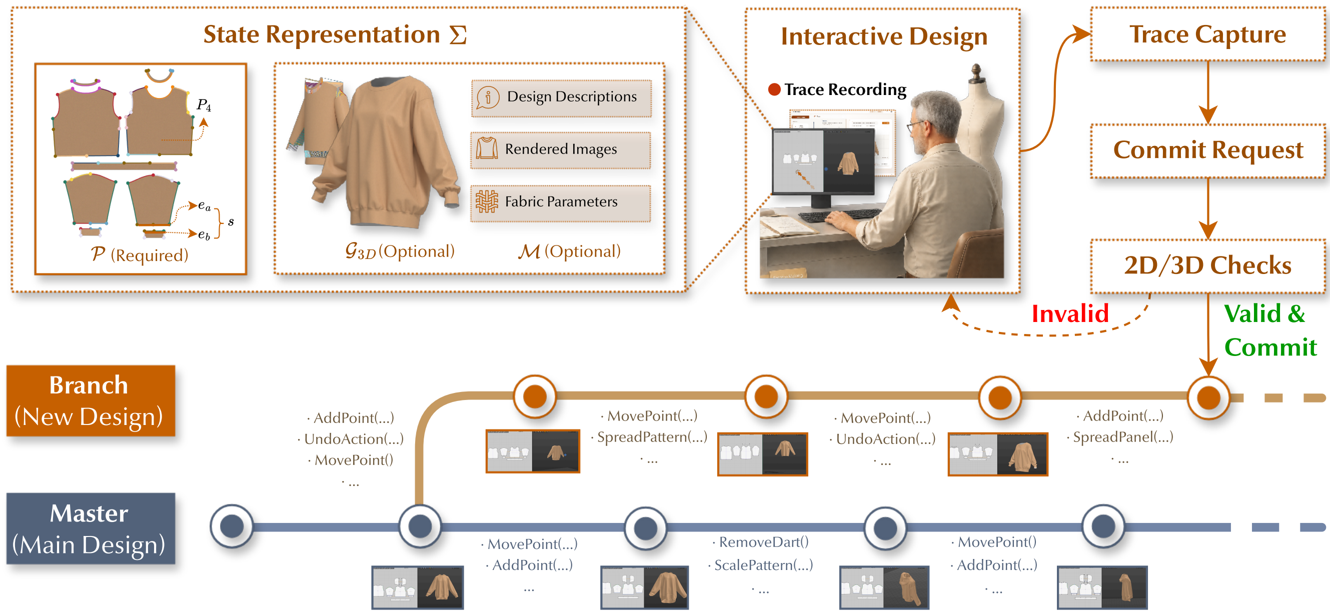


Figure 3: Overview of TailorTrace. Experts begin by exploring and modifying garment designs through an interactive frontend. As they edit, the backend captures their semantic operations as traces and maintains each design state using a unified representation that includes required 2D pattern geometry, optional 3D draping results, and multimodal metadata. Once a design reaches a milestone stage, the expert can request a commit. TailorTrace first performs checks on the 2D pattern and, when 3D geometry is available, further evaluates the draping result. A valid state is committed as a new revision in a branching history, allowing experts to revisit earlier states and continue exploring alternative design directions. An invalid state is returned to the frontend for further editing before another commit is attempted.

capable of translating unstructured design intent into executable operations. To train this system, we curated a dataset of 93 high-quality design traces, comprising 56 foundational traces sourced from pattern-making textbooks (spanning 3 to 10 states each) and 37 real-production traces captured live through TailorTrace as industry experts worked in standard CAD software, rather than reconstructed afterward (spanning 10 to 30 states each). TailorTrace ran as a background CAD plugin that automatically recorded editing operations, while explicit trace-management actions such as commits and branches were performed through button clicks or keyboard shortcuts.

To prevent data leakage and evaluate operation synthesis on held-out traces, we strictly maintained a trace-level split for our training and validation sets. Within the training traces, we employed a distance-based stratified sampling strategy to capture a diverse distribution of both short-range, atomic edits and long-range, compound modifications. For each sampled state pair, we utilized Gemini [9, 16] to generate unstructured, natural language descriptions of the visual and intentional design differences, which domain experts manually verified and corrected when necessary for technical accuracy and authentic terminology. By pairing these natural language descriptions with the exact, discrete operation sequences extracted from TailorTrace, we constructed a comprehensive dataset of approximately 2,500 translation pairs. Finally, we utilized this dataset to fine-tune the Qwen3-VL-8B [57] model, yielding the core reasoning engine for TailorCoPilot.

Figure 4 shows the overall workflow of TailorCoPilot. A user begins by presenting their design intent through either a textual description of the garment style or a reference image/sketch. In response to Finding 3.2.1, TailorCoPilot first initializes a base pattern either via generative model or by retrieving one from a provided pattern library with VLM [32] extracted features. The system then compares the initialized pattern against the user's intent and identifies the pattern components that require change.

Given the current pattern state and relevant expert resources, the reasoning module then analyzes the desired modifications to be made and proposes a sequence of stepwise editable operations, where each step specifies what part of the pattern should change, which operator should be applied, and what parameters should be adjusted in the current context, as shown in Figure 5. These operations progressively transform the base pattern into intermediate and final states, with corresponding changes shown in the 3D garment form. Throughout this process, the interface exposes the current logic, editable parameters, and resulting 2D pattern changes so that novice users can inspect, replay, and intervene in the revision process to enhance their own knowledge.

5 Evaluation

We evaluate our system from two complementary perspectives to validate both the interactive benefits of TailorCoPilot and the foundational utility of TailorTrace. First, we conduct a user study to

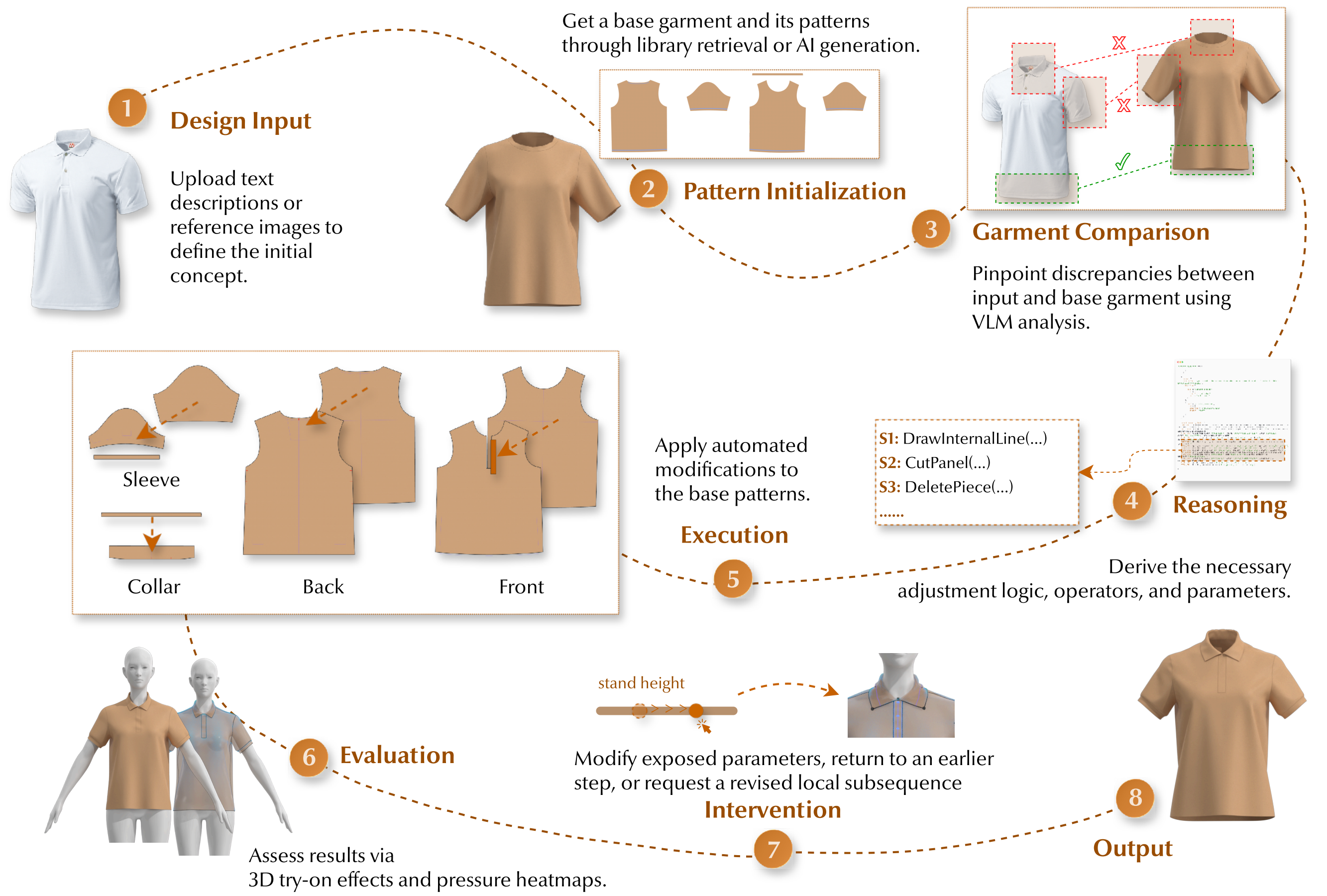


Figure 4: Novice walkthrough of TailorCoPilot. The workflow proceeds through eight stages: (1) Design Input, where the user provides a text description or reference image; (2) Pattern Initialization, where the system gets a base garment and its patterns by retrieving from a user-provided pattern library or AI generation; (3) Garment Comparison, where a VLM pinpoints discrepancies between the input and the retrieved garment; (4) Reasoning, where the system derives adjustment logic, symbolic operators, and editable parameters; (5) Execution, where the proposed modifications are applied to the base pattern; (6) Evaluation, where results are assessed through synchronized 3D try-on effects and pressure heatmaps; (7) Intervention, where users can modify exposed parameters, return to earlier steps, or request a revised local subsequence; and (8) Output, the final revised garment pattern and corresponding garment form.

assess whether the trace-based support embedded within the TailorCoPilot system improves interactive pattern making for novice users, evaluating its capacity to close the generational knowledge gap between experts and beginners. We then conduct a diagnostic study comparing VLM-only, textbook-trace-only, and expert-trace-augmented models, demonstrating that TailorTrace captures critical expert knowledge that lies beyond static LLM knowledge and textbook materials.

5.1 Evaluation Settings

To avoid carry-over effects from the formative study, we recruited a new pool of 20 participants, including **10 novices** (PU1–PU10) and **10 advanced novices** (PU11–PU20). We used a mixed design with *Workflow* (TailorCoPilot vs. baseline) and *Difficulty* (simple, medium, complex) as within-participant factors, and *Expertise* (novice vs. advanced novice) as a between-participant factor. As in the formative study, participants were allowed to choose the baseline pattern-making tools they could use, including AI [42, 65] and CAD [55].

Each participant completed 12 trials across two sessions held on separate days, with at least one day between sessions. Each session included 6 trials, with 3 simple, 2 medium, and 1 complex task. Trials were divided into two 3-trial blocks separated by a short break. Workflow order was counterbalanced across participants using an AB/BA scheme at the block level to reduce context switching between workflows. Each participant completed 6 simple, 4 medium, and 2 complex trials in total. Within each difficulty level, trials were split evenly between TailorCoPilot and the baseline. Task assignment was randomized and balanced so that each task

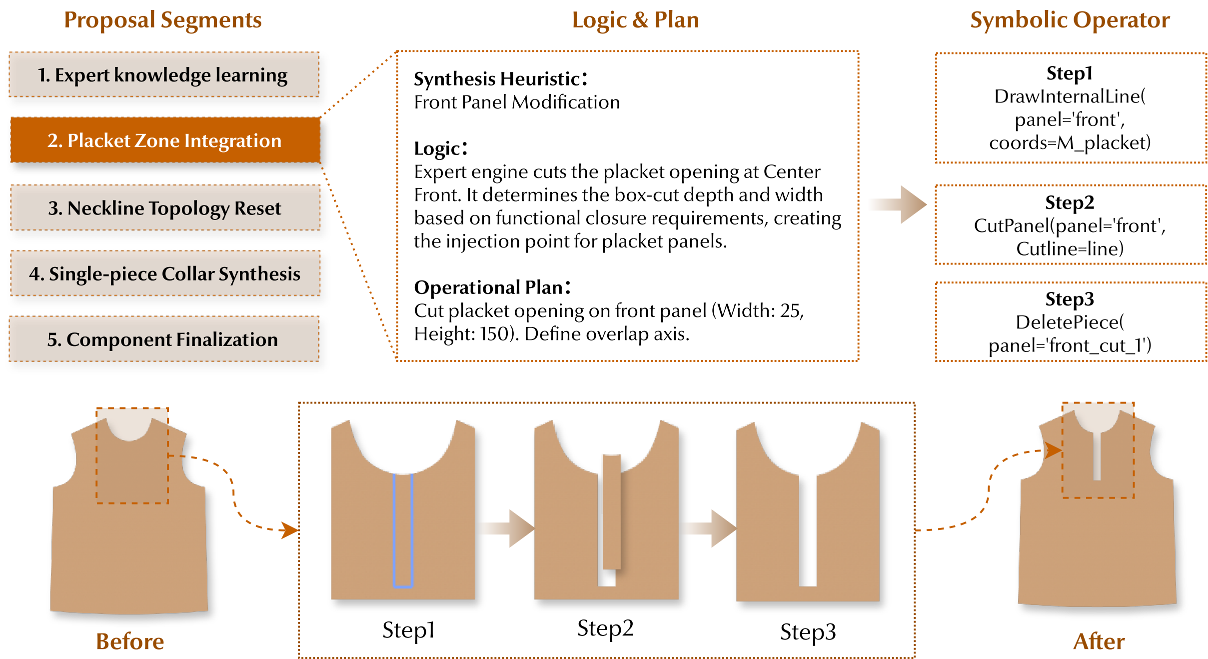


Figure 5: Neuro-symbolic reasoning and execution flow. The system translates design intent into structural heuristics, compiles them into an operational plan, and executes symbolic edits on the pattern topology. In the illustrated placket integration step, the engine determines the center-front opening logic before applying the corresponding backend operators.

was evaluated under both workflows overall, while no participant encountered the same task twice.

We measured *task success*, *completion time*, *expert-rated artifact quality*, and *NASA-TLX* [20]. Success was defined as producing a structurally valid final pattern within the allotted time. To reduce rater burden, artifact quality was assessed on a stratified random sample. Five blinded experts took part in the evaluation, and each sampled artifact was independently rated by two experts on *sewability*, *structural balance*, and *reference conformity*. We averaged these ratings into a composite score. NASA-TLX was administered immediately after each trial, enabling trial-level workload analysis. To account for repeated observations, we used mixed-effects logistic regression for task success and linear mixed-effects models for completion time, artifact quality, and NASA-TLX. After the study, we conducted a brief exit interview with each participant about their experience with the two workflows.

5.2 Results

5.2.1 Performance benefits across task complexity. Figure 6 summarizes task success and completion time by workflow, difficulty, and expertise group. TailorCoPilot significantly increased the odds of task success relative to the baseline ($OR = 2.31$, 95% CI $[1.42, 3.86]$, $p = .001$). The performance pattern differed by expertise. For novices, TailorCoPilot maintained a clear success-rate advantage across all difficulty levels, including simple tasks. For advanced novices, TailorCoPilot and the baseline were relatively close on simple tasks, but the gap widened on medium and complex tasks. This pattern suggests that TailorCoPilot was especially helpful when revision required multi-step coordination across related pattern pieces, while still providing substantial support for novices on simpler tasks.

Completion times were significantly shorter with TailorCoPilot compared to the baseline ($p < .001$), with the effect size varied by task difficulty ($p = .018$). TailorCoPilot was generally faster on medium and complex tasks in both groups. Among advanced novices, the CAD baseline remained competitive on some simple tasks, which is consistent with the efficiency of familiar tools when revision demands were still limited. Because the baselines differed by expertise group, we do not interpret absolute between-group differences as direct estimates of system effect.

5.2.2 Revision became more efficient and manageable. TailorCoPilot not only improved artifact quality but also reduced perceived workload. Table 2 reports blinded expert ratings of final artifacts, averaged across task difficulties. These ratings demonstrated strong inter-rater reliability ($ICC(1,1) = 0.81$). Artifact quality was significantly higher with TailorCoPilot ($p = .001$). The clearest gains appeared in *structural balance* and *reference conformity*, suggesting better preservation of cross-piece consistency and closer alignment with the target design. Gains in *sewability* were also consistent, though relatively smaller. Representative qualitative results further illustrate this pattern (Figure 7). In the complex dress task, the baseline showed asymmetry, jagged edges, incorrect sleeve shape, and incorrect skirt length, while TailorCoPilot better preserved the target V-neck, waist pleats, and symmetry, producing a result that was structurally closer to the reference.

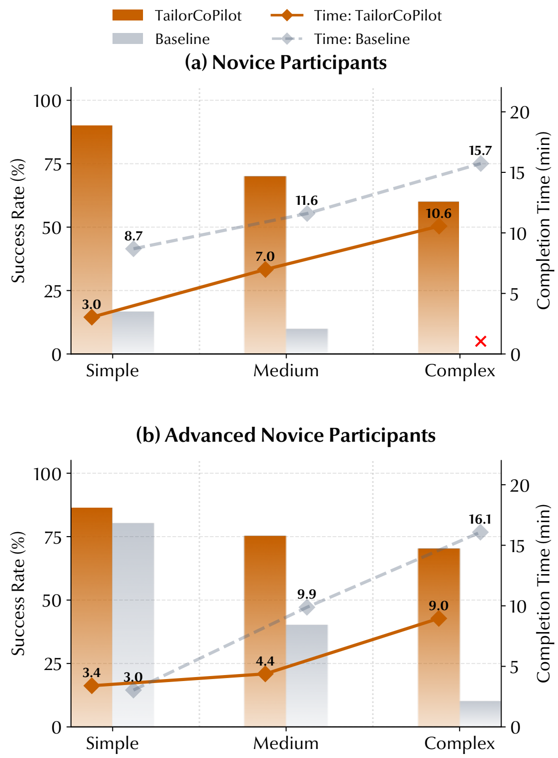


Figure 6: Task success and completion time by workflow, difficulty, and expertise group. For novices, TailorCoPilot maintained a clear success-rate advantage across all task difficulties. For advanced novices, TailorCoPilot and the baseline were closer on simple tasks, while TailorCoPilot achieved higher success and lower completion time on medium and complex tasks.

Perceived workload was also reduced with TailorCoPilot, as measured by NASA-TLX scores ($p < .001$). Exit interviews further indicated that participants retained a sense of agency and ownership during revision. One participant described the result as "still my pattern because the final decisions were mine" (PU6), while an advanced novice remarked, "I stopped second-guessing every edit" (PU13). These findings indicate that TailorCoPilot made pattern revision more manageable for less experienced users without diminishing their sense of control.

5.3 Diagnostic Study on TailorTrace

To systematically examine the contribution of trace-based scaffolding, we expanded the diagnostic study to 15 complex pattern-making tasks and compared three conditions. **TailorCoPilot (V)** relied solely on the pre-trained Qwen3-VL-8B, without access to TailorTrace data. **TailorCoPilot (T)** and **TailorCoPilot (F)** both utilized textbook-based traces, with the latter also incorporating expert-authored traces capturing industrial pattern-making practices. All conditions used identical target references, base patterns, and inference budgets, and all outputs were expert-evaluated using the same protocol as in the main study.

The results showed that **V** achieved no success across the 15 tasks and received an average expert rating of $M = 1.3$ ($SD = 0.5$). **T** met the success criterion in 3 tasks (20.0%) with an average rating of $M = 1.9$ ($SD = 0.8$). In contrast, **F** met the criterion in 10 tasks (66.7%) and scored significantly higher on expert evaluation ($M = 4.1$, $SD = 0.6$). A paired Wilcoxon signed-rank test confirmed a statistically significant difference between **T** and **F** ($W = 1$, $p < .001$).

Figure 8 illustrates a representative qualitative example of these differences. While **V** completely failed the revision task and **T** only managed localized modifications without maintaining overall structural integrity, **F** successfully generated a reference-consistent pattern. These results indicate that expert-authored TailorTrace data offers critical procedural guidance that extends beyond static LLM knowledge and textbook materials, particularly for complex tasks requiring coordinated changes across multiple pattern components.

6 Discussion

6.1 Reusable Traces as Expert Process Knowledge

Our results suggest that one central contribution of TailorCoPilot lies in making expert process knowledge usable during pattern making. In garment pattern making, difficulty often comes from deciding how design intent should be translated into concrete edits, how a change to one piece propagates to related pieces, and how to continue when an intermediate state is only partially correct. TailorTrace addresses these challenges by representing pattern development as discrete states linked by explicit geometric operations. This gives TailorCoPilot a process-aware basis for planning, revision, and recovery.

This also clarifies what agentic pattern making means in our setting. TailorCoPilot does more than generate a final result from a prompt. It operates over traceable states, proposes multi-step revisions, and supports users in inspecting and adjusting the process as it unfolds. Its agentic behavior therefore depends on a structured revision space that remains visible and editable to the user.

Our findings also show why static construction knowledge alone is often insufficient. Handbooks can describe principles and canonical techniques, but they do not fully preserve how experts sequence edits, maintain cross-piece consistency, or carry partial results forward through complex revision. The diagnostic study points to this gap directly. In the textbook-only variant, the system could still produce locally plausible edits, but it was less reliable at coordinating structurally valid changes across pieces. This suggests that expert-authored traces contribute procedural information that is difficult to recover from rules or final examples alone. It also provides a foundation for collecting process-rich data that may support future generative AI models in this domain.

6.2 Supporting Apprenticeship Without Replacing Expertise

TailorCoPilot supported novices and advanced novices in related but not identical ways. For novices, it made the work more approachable by reducing the amount of CAD fluency and pattern-making knowledge required to begin making meaningful changes.

Table 2: Expert ratings of final artifacts (mean ± SD) on a 1–5 scale, grouped by user expertise. Values are averaged across simple, medium, and complex tasks. Across both expertise groups, TailorCoPilot received higher ratings than the baseline on all evaluation dimensions.

	Sewability		Structural balance		Reference conformity		Overall quality	
Expertise	TailorCoPilot	Baseline	TailorCoPilot	Baseline	TailorCoPilot	Baseline	TailorCoPilot	Baseline
Novice	3.82 ± 0.86	2.02 ± 1.01	3.97 ± 0.81	1.48 ± 0.59	3.97 ± 0.83	1.35 ± 0.53	3.92 ± 0.82	1.62 ± 0.70
Advanced novice	4.17 ± 0.74	3.48 ± 0.93	4.32 ± 0.70	2.48 ± 0.99	4.33 ± 0.71	2.63 ± 0.97	4.27 ± 0.71	2.87 ± 0.95

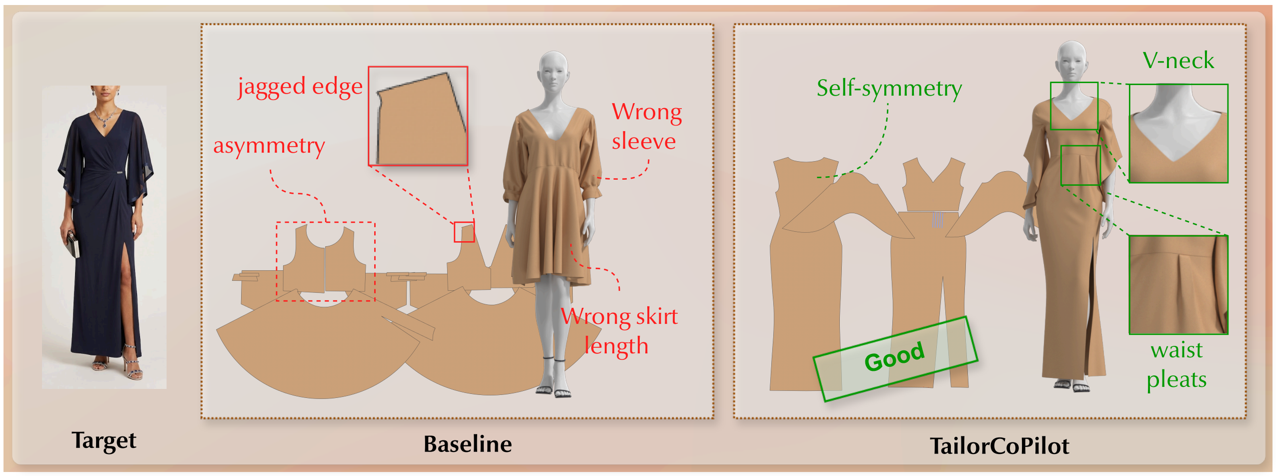


Figure 7: Representative qualitative result from the user study on the complex dress task. The baseline exhibited asymmetry, jagged edges, incorrect sleeve shape, and incorrect skirt length, while TailorCoPilot better preserved the target V-neck, waist pleats, and symmetry, producing a result that was structurally closer to the reference.

For advanced novices, its value was less about entry and more about continuity. Instead of having to reconstruct expert procedures step by step, they could follow a more legible path through the revision process, with key operations, dependencies, and constraints surfaced at the moment they mattered. Across both groups, this made complex revision easier to start and easier to continue.

At the same time, these gains did not erase the gap between assisted participation and expert practice. Pattern making still relies on forms of judgment that resist full formalization, such as evaluating balance, assessing silhouette quality, and deciding whether a local correction preserves the coherence of the overall construction. What shifted was not the need for expertise itself, but the distribution of effort. TailorCoPilot reduced procedural overhead and made recovery from mistakes less demanding, while higher-level evaluation and final decisions remained in the user's hands.

This suggests a broader role for systems of this kind in experience-dependent workflows shaped by iteration and expert judgment. The aim is not to flatten the distinction between novice and professional performance, but to make expert process knowledge more available during learning. Seen this way, TailorCoPilot is better understood as support for apprenticeship than as a substitute for expertise. By preserving senior experts' workflows as reusable traces, it points to one possible way of addressing the generational skills gap that motivates this work.

6.3 Limitations and Future Directions

This work has three main limitations. First, the evaluation uses a modest sample and measures short-term performance in a controlled setting. Although the results suggest that trace-based support improves revision performance and usability, we do not yet know whether these benefits translate into durable learning or long-term skill development. Longitudinal studies will be needed to examine whether repeated use supports independent pattern-making ability over time.

Second, TailorCoPilot depends on a scoped representation of the domain. The current system assumes a predefined operator vocabulary, and expert-authored traces that can be represented through TailorTrace. Given the modest dataset of 93 traces and approximately 2,500 derived training pairs, our claims are limited to operation synthesis on held-out traces within this bounded garment domain; broader deployment in professional prototyping and small-batch production remains to be validated. Figure 9 shows a representative failure case, where the system only partially recovered the intended structure under these constraints.

Third, to ensure comparability across conditions, we used a unified set of default fabric-simulation parameters. This simplifies evaluation, but leaves material-specific behavior and some forms of production variability for future work. Extending TailorTrace to

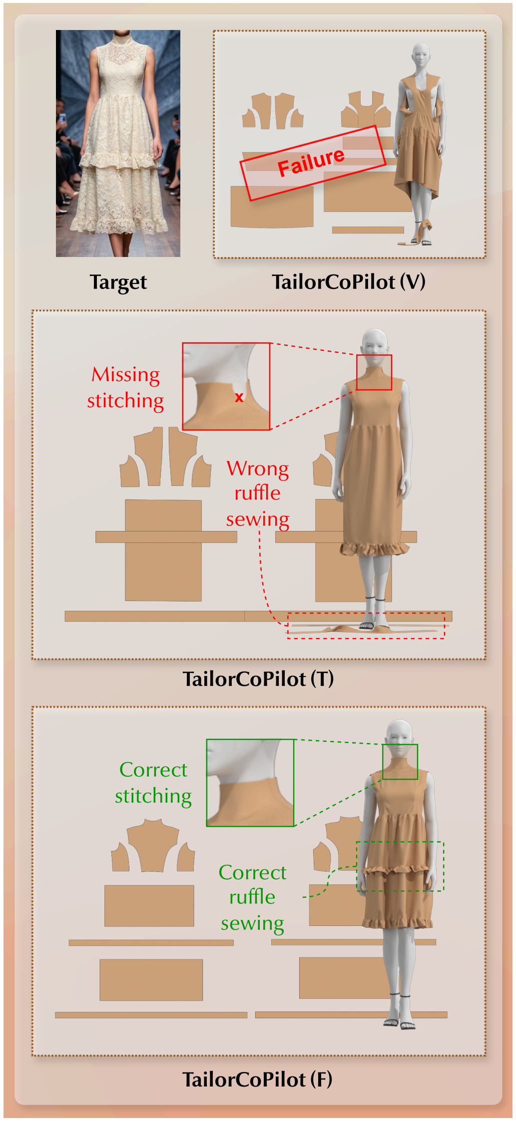


Figure 8: Qualitative comparison from the diagnostic study using identical references, base patterns, and inference budgets. TailorCoPilot (V) fails to execute the revision. TailorCoPilot (T) produces locally plausible edits but breaks cross-piece structural relations (e.g., missing stitching and incorrect ruffle construction). By incorporating expert traces, TailorCoPilot (F) generates a structurally valid pattern that accurately matches the target design.

richer material settings and broader garment categories remains an important direction.

7 Conclusion

This paper presented **TailorCoPilot**, a system that supports garment pattern making by grounding interaction in **TailorTrace**, a version-controlled representation of pattern states and transformations. Our findings show that representing expert process knowledge as reusable traces can make pattern making more accessible to novices and advanced novices, especially in revisions that require coordination across multiple pieces and intermediate states. More broadly, this work suggests that practice-based expertise may become more usable when it is captured as revisitable traces. Within this bounded garment-pattern domain, our findings point to a promising direction for supporting both apprenticeship and future AI systems in workflows where expert knowledge is deeply procedural and difficult to formalize.

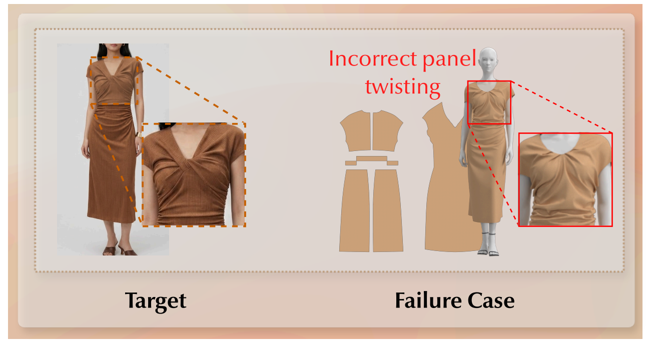


Figure 9: Failure Case. As TailorTrace and TailorCoPilot are primarily defined by 2D topological transformations, they currently struggle to model intricate physical manipulations, such as the twisted knot on the target bust.

Acknowledgments

The authors gratefully acknowledge the support provided by Style3D Research. The authors would also like to acknowledge the financial support from the Fundamental Research Funds for the Central Universities (Grant Nos. CUSF-DH-T-2025007 and 2232026G-08) and the International Cooperation Fund of the Science and Technology Commission of Shanghai Municipality (Grant No. 21130750100).

References

[1] Deepak Bhaskar Acharya, Karthigeyan Kuppan, and Divya Bhaskaracharya. 2025. Agentic AI: Autonomous Intelligence for Complex Goals—A Comprehensive Survey. *IEEE Access* 13 (2025), 18912–18936. https://api.semanticscholar.org/CorpusID:275830329

[2] Fatma Baytar, Yoon Yang, and Mona Maher. 2022. The Importance of Deciphering Tacit Knowledge in Apparel Product Development for Explainable AI: Proposed steps for increasing digitalization in the apparel industry. In *Proceedings of the 12th Hellenic Conference on Artificial Intelligence* (Corfu, Greece) *(SETN '22)*. Association for Computing Machinery, New York, NY, USA, Article 65, 5 pages. doi:10.1145/3549737.3549806

[3] Bikram Pratim Bhuyan, Amar Ramdane-Cherif, Ravi Tomar, and T. P. Singh. 2024. Neuro-symbolic artificial intelligence: a survey. *Neural Comput. Appl.* 36, 21 (June 2024), 12809–12844. doi:10.1007/s00521-024-09960-z

[4] 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. *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2024), 2924–2934. https://api.semanticscholar.org/CorpusID:274992065

[5] Ophelia Boateng, Solomon Ayesu, Benjamin Asinyo, and Timothy Crentsil. 2025. The use of apparel cad technology in pedagogical training among technical universities in Ghana. *Indonesian Journal of Education and Social Sciences* 4 (01 2025), 97–112. doi:10.56916/ijess.v4i1.878

[6] Virginia Braun and Victoria Clarke. 2006. Using thematic analysis in psychology. *Qualitative research in psychology* 3, 2 (2006), 77–101.

[7] Evan Casey, Tianyu Zhang, Shu Ishida, John Roger Thompson, Amir Hosein Khasahmadi, J. Lambourne, Pradeep Kumar Jayaraman, and Karl D. D. Willis. 2025.

Aligning Constraint Generation with Design Intent in Parametric CAD. *ArXiv* abs/2504.13178 (2025). https://api.semanticscholar.org/CorpusID:277856734
[8] Cheng-Hsiu Chen, Jheng-Wei Su, Min-Chun Hu, Chih-Yuan Yao, and Hung-Kuo Chu. 2024. Panelformer: Sewing Pattern Reconstruction from 2D Garment Images. *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)* (2024), 443–452. https://api.semanticscholar.org/CorpusID:266729549
[9] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261* (2025).
[10] Valdemar Danry, Pat Pataranutaporn, Yaoli Mao, and Pattie Maes. 2023. Don't Just Tell Me, Ask Me: AI Systems that Intelligently Frame Explanations as Questions Improve Human Logical Discernment Accuracy over Causal AI explanations. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 352, 13 pages. doi:10.1145/3544548.3580672
[11] Richard Lee Davis, Thiemo Wambsganss, Weijin Jiang, Kevin Gonyop Kim, Tanja Käser, and Pierre Dillenbourg. 2024. Fashioning Creative Expertise with Generative AI: Graphical Interfaces for Design Space Exploration Better Support Ideation Than Text Prompts. *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (2024). https://api.semanticscholar.org/CorpusID:269748507
[12] Sebastiaan De Peuter, Antti Oulasvirta, and Samuel Kaski. 2023. Toward AI assistants that let designers design. *AI Magazine* 44, 1 (2023), 85–96. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/aaai.12077 doi:10.1002/aaai.12077
[13] Maaike J. Van den Haak, Menno de Jong, and Peter Jan Schellens. 2003. Retrospective vs. concurrent think-aloud protocols: Testing the usability of an online library catalogue. *Behaviour & Information Technology* 22 (2003), 339 – 351. https://api.semanticscholar.org/CorpusID:18239370
[14] Zijian Ding. 2024. Advancing GUI for Generative AI: Charting the Design Space of Human-AI Interactions through Task Creativity and Complexity. In *Companion Proceedings of the 29th International Conference on Intelligent User Interfaces* (Greenville, SC, USA) *(IUI '24 Companion)*. Association for Computing Machinery, New York, NY, USA, 140–143. doi:10.1145/3640544.3645241
[15] Stuart E. Dreyfus. 2004. The Five-Stage Model of Adult Skill Acquisition. *Bulletin of Science, Technology & Society* 24, 3 (2004), 177–181. doi:10.1177/0270467604264992
[16] Google. 2026. *Gemini*. https://gemini.google.com Accessed: 2026-01-01.
[17] Hongquan Gui, Zhanpeng Yang, Arjun Rachana Harish, Cheng Ren, Yishu Yang, and Ming Li. 2025. GenPattern: dual-graph enhanced sewing pattern generation via multimodal large language model. *Journal of Manufacturing Systems* (2025). https://api.semanticscholar.org/CorpusID:283088676
[18] Yanhong Guo and Li Wei. 2025. AI technology adoption and intergenerational knowledge transfer among older employees. *Frontiers in Psychology* Volume 16 - 2025 (2025). doi:10.3389/fpsyg.2025.1673730
[19] Umer Hameed. 2020. Effects of Pattern Maker's Work Experience with Designer on Clothing Creation. *Journal of Textile Science & Fashion Technology* (2020). https://api.semanticscholar.org/CorpusID:234609949
[20] Sandra G. Hart and Lowell E. Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. In *Human Mental Workload*, Peter A. Hancock and Najmedin Meshkati (Eds.). Advances in Psychology, Vol. 52. North-Holland, 139–183. doi:10.1016/S0166-4115(08)62386-9
[21] 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 (2024), 1 – 13. https://api.semanticscholar.org/CorpusID:267320968
[22] San Hong and Woojin Park. 2025. Developing user-centered system design guidelines for explainable AI: a systematic literature review. *Artificial Intelligence Review* 58 (2025). https://api.semanticscholar.org/CorpusID:282241726
[23] Xi Hu, Yiwen Xing, Xudong Cai, Yihang Zhao, Michael Cook, Rita Borgo, and Timothy Neate. 2025. Designing Interactions with Generative AI for Art and Creativity: A Systematic Review and Taxonomy. In *Proceedings of the 2025 ACM Designing Interactive Systems Conference (DIS '25)*. Association for Computing Machinery, New York, NY, USA, 1126–1155. doi:10.1145/3715336.3735843
[24] Peng Jin, Jintu Fan, Rong Zheng, Qing Chen, Le Liu, Runtian Jiang, and Hui Zhang. 2023. Design and Research of Automatic Garment-Pattern-Generation System Based on Parameterized Design. *Sustainability* (2023). https://api.semanticscholar.org/CorpusID:255676005
[25] Jeyeon Jo and Fatma Baytar. 2022. Generative Pants Pattern Making for Optimized Fit in Mass Customization. *Innovate to Elevate* (2022). https://api.semanticscholar.org/CorpusID:255897220
[26] Benjamin T. Jones, Zihan Zhang, Felix Hähnlein, Wojciech Matusik, Maaz Bin Safeer Ahmad, Vladimir G. Kim, and Adriana Schulz. 2025. A Solver-Aided Hierarchical Language for LLM-Driven CAD Design. *Computer Graphics Forum* 44 (2025). https://api.semanticscholar.org/CorpusID:282051728
[27] Minna Kaipainen and Sinikka Pöllänen. 2025. Why Don't They Sew Anymore?—Exploring the Decline of Home Garment Sewing. *Journal of Community & Applied Social Psychology* 35, 5 (2025), e70172. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/casp.70172 doi:10.1002/casp.70172 e70172 CASP-24-797.
[28] Narae Kim and Jinhee Park. 2025. Development of a parametric production jacket pattern for an automated pattern-making system. *Fashion and Textiles* 12, 1 (July 2025), 20. doi:10.1186/s40691-025-00419-w
[29] 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 *European Conference on Computer Vision*. https://api.semanticscholar.org/CorpusID:270067826
[30] Maria Korosteleva and Olga Sorkine-Hornung. 2023. GarmentCode: Programming Parametric Sewing Patterns. *ACM Transactions on Graphics (TOG)* 42 (2023), 1 – 15. https://api.semanticscholar.org/CorpusID:259089061
[31] Ah Lam Lee and Hyunsook Han. 2024. A review of parametric clothing pattern CAD software methodology. *International Journal of Clothing Science and Technology* 36, 1 (01 2024), 102–116. arXiv:https://www.emerald.com/ijcst/article-pdf/36/1/102/9479146/ijcst-01-2023-0002.pdf doi:10.1108/IJCST-01-2023-0002
[32] Mingxin Li, Yanzhao Zhang, Dingkun Long, Keqin Chen, Sibo Song, Shuai Bai, Zhibo Yang, Pengjun Xie, An Yang, Dayiheng Liu, Jingren Zhou, and Junyang Lin. 2026. Qwen3-VL-Embedding and Qwen3-VL-Reranker: A Unified Framework for State-of-the-Art Multimodal Retrieval and Ranking. *arXiv* (2026).
[33] Siran Li, Ruiyang Liu, Chen Liu, Zhendong Wang, Gaofeng He, Yong-Lu Li, Xiaogang Jin, and Huamin Wang. 2025. GarmageNet: A Multimodal Generative Framework for Sewing Pattern Design and Generic Garment Modeling. *ACM Trans. Graph.* 44, 6, Article 216 (Dec. 2025), 23 pages. doi:10.1145/3763271
[34] Xinyu Li, Qingqing Yao, and Yuanda Wang. 2025. GarmentDiffusion: 3D Garment Sewing Pattern Generation with Multimodal Diffusion Transformers. *ArXiv* abs/2504.21476 (2025). https://api.semanticscholar.org/CorpusID:278207472
[35] Kaixuan Liu, Chun Zhu, Xuyuan Tao, Pascal Bruniaux, and Xianyi Zeng. 2019. Parametric design of garment pattern based on body dimensions. *International Journal of Industrial Ergonomics* (2019). https://api.semanticscholar.org/CorpusID:195410639
[36] 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).
[37] Shengqi Liu, Yuhao Cheng, Zhuo Chen, Xingyu Ren, Wenhan Zhu, Lincheng Li, Mengxiao Bi, Xiaokang Yang, and Yichao Yan. 2024. Multimodal Latent Diffusion Model for Complex Sewing Pattern Generation. *ArXiv* abs/2412.14453 (2024). https://api.semanticscholar.org/CorpusID:274859745
[38] Jinbo Luo, Hong Qu, Yujie Zhao, Jie Zhang, and Yadie Yang. 2025. Deep Learning for 3D Fashion Design: A Survey from a Sewing Pattern-Driven Perspective. *IEEE Transactions on Circuits and Systems for Video Technology* (2025). https://api.semanticscholar.org/CorpusID:283371952
[39] Yingrui Lyu, Zhaohui Wang, Qinwen Ye, Yuexin Sun, and Jing Chao. 2025. Body data-driven garment pattern construction in digital fashion innovations: a review. *Textile Research Journal* (2025). doi:10.1177/00405175251352800
[40] Jiaju Ma, Chau Vu, Asya Lyubavina, Catherine Liu, and Jingyi Li. 2025. Computational Scaffolding of Composition, Value, and Color for Disciplined Drawing. *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology* (2025). https://api.semanticscholar.org/CorpusID:281421347
[41] Nicolai Marquardt, Asta Roseway, Hugo Romat, Payod Panda, Michel Pahud, Gonzalo Ramos, Steven Mark Drucker, Andrew D. Wilson, Ken Hinckley, and Nathalie Riche. 2025. ImaginationVellum: Generative-AI Ideation Canvas with Spatial Prompts, Generative Strokes, and Ideation History. *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology* (2025). https://api.semanticscholar.org/CorpusID:281683078
[42] Kiyohiro Nakayama, Jan Ackermann, Timur Levent Kesdogan, Yang Zheng, Maria Korosteleva, Olga Sorkine-Hornung, Leonidas Guibas, Guandao Yang, and Gordon Wetzstein. 2025. AIpparel: A Multimodal Foundation Model for Digital Garments. *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2025), 8138–8149. https://api.semanticscholar.org/CorpusID:277621364
[43] Evridiki Papahristou and Nefeli Zolota Tatsi. 2024. A review of 3D design knowledge and its impact on creativity in fashion design education. *Communications in Development and Assembling of Textile Products* 5, 2 (Dec. 2024), 266–277. doi:10.25367/cdatp.2024.5.p266-277
[44] Nikolaos Partarakis, Xenophon Zabulis, Carlo Meghini, Arnaud Dubois, Ines Moreno, Chistodoulos Ringas, Aikaterini Ziova, Danai Kaplanidi, David Arnaud, Noël Crescenzo, Patricia Hee, Juan José Ortega, Josefina Garrido, Marie-Adelaide Benvenuti, and Jelena Krivokapic. 2025. A Review, Analysis, and Roadmap to Support the Short-Term and Long-Term Sustainability of the European Crafts Sector. *Heritage* 8, 2 (2025). doi:10.3390/heritage8020070
[45] Xiaohan Peng, Debanjana Haldar, Janin Koch, and Wendy E. Mackay. 2026. DesignTrace: Exploring, Iterating and Tracking Design Alternatives with GenAI. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems* (Barcelona, Spain) *(CHI '26)*. Association for Computing Machinery, New York,

NY, USA. doi:10.1145/3772318.3791036

[46] Muhammad Raees, Inge Meijerink, Ioanna Lykourentzou, Vassilis-Javed Khan, and Konstantinos Papangelis. 2024. From explainable to interactive AI: A literature review on current trends in human-AI interaction. *International Journal of Human-Computer Studies* 189 (2024), 103301. doi:10.1016/j.ijhcs.2024.103301

[47] Mustafijur Rahman, Rishad Rayyaan, Md Golam Nur, Sk Nazmus Saaqib, and Md Moynul Hassan Shibly. 2015. An exploratory study on modern 3D computerised body scanning system and various types of pattern making software's with their constructive implementation in apparel industry. *European Scientific Journal* 11, 15 (2015), 120–125.

[48] Jeba Rezwana and Mary Lou Maher. 2023. Designing Creative AI Partners with COFI: A Framework for Modeling Interaction in Human-AI Co-Creative Systems. *ACM Trans. Comput.-Hum. Interact.* 30, 5, Article 67 (Sept. 2023), 28 pages. doi:10.1145/3519026

[49] Ari Seff, Yaniv Ovadia, Wenda Zhou, and Ryan P. Adams. 2020. SketchGraphs: A Large-Scale Dataset for Modeling Relational Geometry in Computer-Aided Design. In *ICML 2020 Workshop on Object-Oriented Learning*.

[50] Ari Seff, Wenda Zhou, Nick Richardson, and Ryan P. Adams. 2021. Vitruvion: A Generative Model of Parametric CAD Sketches. *ArXiv* abs/2109.14124 (2021). https://api.semanticscholar.org/CorpusID:238215827

[51] Ouissale Zaoui Seghroucheni, Mohammed Al Achhab, and Mohamed Lazaar. 2023. Systematic Review on the Conversion of Tacit Knowledge. *2023 7th IEEE Congress on Information Science and Technology (CiSt)* (2023), 123–128. https://api.semanticscholar.org/CorpusID:267516460

[52] ET Software. [n. d.]. ET Official Website. https://www.etsystem.cn/en/

[53] Lectra Software. 2025. Industry 4.0 transformation and innovation. https://www.lectra.com/en

[54] Kihoon Son, DaEun Choi, Tae Soo Kim, Young-Ho Kim, Sangdoo Yun, and Juho Kim. 2025. ClearFairy: Capturing Creative Workflows through Decision Structuring, In-Situ Questioning, and Rationale Inference. *ArXiv* abs/2509.14537 (2025). https://api.semanticscholar.org/CorpusID:281394589

[55] Style3D. [n. d.]. *Style3D Studio*. https://www.style3d.com/products/studio

[56] John Sweller. 1988. Cognitive Load During Problem Solving: Effects on Learning. *Cogn. Sci.* 12 (1988), 257–285. https://api.semanticscholar.org/CorpusID:9585835

[57] Qwen Team. 2025. Qwen3 Technical Report. arXiv:2505.09388 [cs.CL] https://arxiv.org/abs/2505.09388

[58] Wen-Fan Wang, Ting-Ying Lee, Chien-Ting Lu, Che-Wei Hsu, Nil Ponsa i Campanyà, Yu Chen, Mike Y. Chen, and Bing-Yu Chen. 2025. GenTune: Toward Traceable Prompts to Improve Controllability of Image Refinement in Environment Design. In *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology (UIST '25)*. Association for Computing Machinery, New York, NY, USA, Article 11, 21 pages. doi:10.1145/3746059.3747774

[59] Mikael Wiberg and Erik Stolterman Bergqvist. 2023. Automation of interaction - interaction design at the crossroads of user experience (UX) and artificial intelligence (AI). *Pers. Ubiquitous Comput.* 27 (2023), 2281–2290. https://api.semanticscholar.org/CorpusID:265193227

[60] Rundi Wu, Chang Xiao, and Changxi Zheng. 2021. DeepCAD: A Deep Generative Network for Computer-Aided Design Models. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)* (2021), 6752–6762. https://api.semanticscholar.org/CorpusID:234789948

[61] Yi Xiu and Zhen-Kai Wan. 2013. A survey on pattern-making technologies in Garment CAD. In *IEEE Conference Anthology*. 1–6. doi:10.1109/ANTHOLOGY.2013.6784694

[62] Huimin Xu, Seungjun Yi, Terence Lim, Jiawei Xu, Andrew Well, Carlos Mery, Aidong Zhang, Yuji Zhang, Heng Ji, Keshav Pingali, et al. 2025. Tama: A human-ai collaborative thematic analysis framework using multi-agent llms for clinical interviews. *arXiv preprint arXiv:2503.20666* (2025).

[63] Shuning Zhang, Hui Wang, and Xin Yi. 2025. Exploring Collaboration Patterns and Strategies in Human-AI Co-creation through the Lens of Agency: A Scoping Review of the Top-tier HCI Literature. *Proc. ACM Hum.-Comput. Interact.* 9, 7, Article CSCW413 (Oct. 2025), 43 pages. doi:10.1145/3757594

[64] Yunzhi Zhang, Zizhang Li, Matt Zhou, Shangzhe Wu, and Jiajun Wu. 2024. The Scene Language: Representing Scenes with Programs, Words, and Embeddings. *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2024), 24625–24634. https://api.semanticscholar.org/CorpusID:273507447

[65] Feng Zhou, Ruiyang Liu, Chen Liu, Gaofeng He, Yong-Lu Li, Xiaogang Jin, and Huamin Wang. 2025. Design2GarmentCode: Turning Design Concepts to Tangible Garments Through Program Synthesis. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 23712–23722. doi:10.1109/CVPR52734.2025.02208