MotionSmith: A Sketch-Based Design System for Automata Making

DoangJoo (Alan) Synn
alansynn@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia, USA

Zhifan Guo
zhifanguo@ucsb.edu
University of California, Santa Barbara
Goleta, California, USA

Sehoon Ha
sehoonha@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia, USA

HyunJoo Oh
hyunjoo.oh@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia, USA

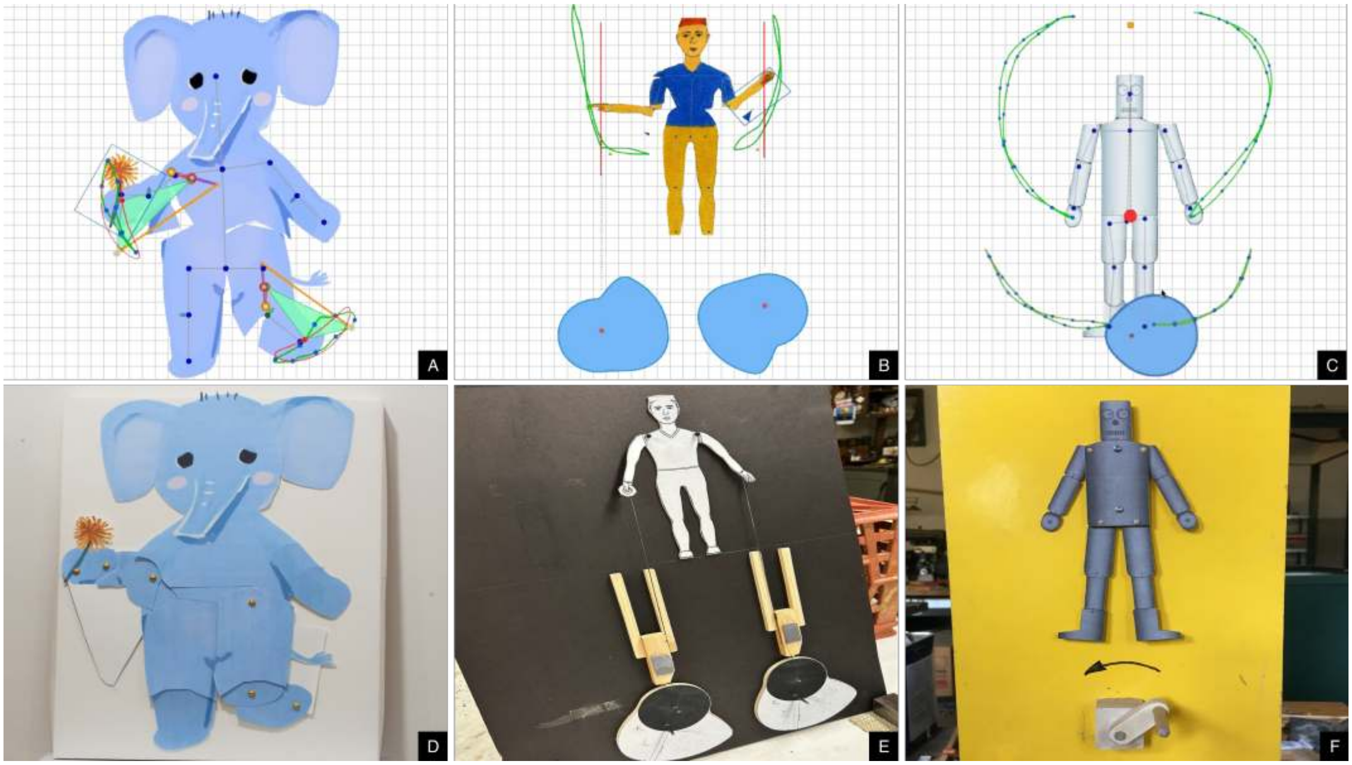


Figure 1: *MotionSmith* is a computational design system informed by participatory design with experienced makers that supports the creation of mechanical craft through an iterative, sketch-based workflow for designing and fabricating working automata. The figure shows three artist-designed automata, each with a digital sketch on top and a fabricated model on bottom. (A,D) A shy blue elephant holding a flower, animated with four-bar linkages. (B,E) A man with upward-swinging arms, actuated by cams. (C,F) A tin robot with synchronized arm and leg motions, driven by a compound cam mechanism.

Abstract

We present MotionSmith, a sketch-based computational design system developed through participatory design (PD) with three experienced makers of mechanical crafts (automata). MotionSmith enables users to sketch and iteratively identify a desired motion, explore system-generated mechanisms to realize the motion, refine the chosen mechanism, and export fabrication-ready files. We developed the system through a year-long iterative process, including PD with three artists. Insights from this collaboration emphasized the importance of prioritizing creative intent over literal sketch fidelity, enabling fluent iteration across design stages, and ensuring mechanically simple, fabricable outcomes. The resulting system

---

CHI '26, Barcelona, Spain

ACM ISBN 979-8-4007-2278-3/2026/04
https://doi.org/10.1145/3772318.3791545

leverages computational support while maintaining user agency across sketching, mechanism synthesis, and blueprint generation. Our deployment with experts shows how this workflow bridges user intent and mechanical realization and surfaces opportunities to expand educational scaffolding and fabrication guidance for a broader community of makers.

CCS Concepts

• **Human-centered computing → Participatory design**; **Interaction design**; **Interactive systems and tools**.

Keywords

Participatory design, Experienced creators, Sketch-based design, Automata making, Digital fabrication

ACM Reference Format:
DoangJoo (Alan) Synn, Zhifan Guo, Sehoon Ha, and HyunJoo Oh. 2026. MotionSmith: A Sketch-Based Design System for Automata Making. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems (CHI '26), April 13–17, 2026, Barcelona, Spain.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3772318.3791545

1 Introduction

Mechanical crafts, commonly known as automata, are an interdisciplinary medium that combines mechanical engineering, art, craft, and storytelling [?]. Historically rooted in the intricate craftsmanship of clockwork and timepieces [?], automata have evolved into modern forms such as kinetic art [? ? ?], animatronics [? ?], and hands-on learning tools [? ? ? ?]. Their interdisciplinary nature makes automata a compelling medium for creative expression and education, appealing to makers of all ages. However, designing and building automata remains a complex process. Creating mechanical motion requires understanding gears, linkages, and their combinations, and physically realizing these designs involves challenges in selecting appropriate materials, preparing custom parts, and assembling them with precision.

To lower these barriers, researchers in computational design have explored ways to assist makers in creating mechanical movements. One line of work translates user-specified motion, such as a hand-drawn curve, into candidate mechanisms through library-based search or continuous optimization [? ? ? ?]. While effective, these systems typically treat the input motion as a fixed target, which may not capture users' intent or align with the exploratory nature of early-stage design. Others have examined how users define motion through interactive modalities, including sketching [? ? ? ? ? ? ? ? ? ?], motion capture [? ?], and tangible interfaces [? ? ?]. Of these, sketch-based systems stand out for their expressive and approachable qualities in automata making, as they align with established practices of sketching ideas and working in CAD programs to develop mechanical components.

We developed a sketch-based computational design system through participatory design with experienced makers to support the design and fabrication of creative automata. Our system *MotionSmith* enables makers to sketch motion paths, articulate intent, simulate and refine their ideas, generate multiple candidate mechanism candidates, select and modify their options, and export design files optimized for their fabrication methods. We conducted a year-long iterative process that included exploratory prototyping and a literature review to identify key gaps, followed by six months of participatory design (PD) with three expert automata artists (Figure 1). Through successive PD sessions with artists, each with over seven years of practice, we identified key design needs: supporting creative intent rather than literal input, enabling fluent iteration across design stages, and ensuring mechanically simple, fabricable outcomes.

In the final PD phase, artists used MotionSmith to articulate creative goals, explore mechanism options, customize mechanical configurations, and fabricate their intended automata models. Our findings show how MotionSmith sustains agency and supports exploratory, non-linear workflows, while also revealing opportunities for educational scaffolding and fabrication support.

In summary, our contributions include:

- **Participatory design insights** from collaborations with automata artists, identifying which computational features are needed and preferred, why they matter within existing creative workflows, and how system design can scaffold goal-setting and mechanism generation with fine-tuning.
- **MotionSmith**, a sketch-based computational design system that embodies these insights by supporting intent articulation and an iterative, tinkerable workflow for progressively translating user intent in motion design into concrete mechanisms and fabrication-ready outputs.
- **A set of automata artifacts** created in collaboration with artists, demonstrating the expressive and technical range enabled by MotionSmith's design-to-fabrication support.

2 Related Work

2.1 Automata as the Art and Engineering of Mechanical Movement

Automata are mechanical figures designed to follow a set of movements, often evoking lifelike actions or storytelling scenes. Their history is long yet fragmented, with early surviving examples traced to Egyptian tombs from around 2000 BC, suggesting a longstanding human fascination with movable models [?]. Over time, automata evolved from scientific and technological demonstrations such as intricate clockwork engineering to contemporary forms in kinetic art, animatronics, and educational toys [? ? ?].

Creating automata is inherently interdisciplinary, requiring knowledge of mechanical components such as gears and linkages, engineering expertise to fabricate and assemble parts with precision, and creative insight to envision expressive mechanical movements. This combination has been used as an engaging educational context across levels from K-12 [?] to university contexts [?]. For example, Paper Mechatronics [? ?] offers resources such as the FoldMecha simulator [?], which enables makers to design mechanical movements, adjust parameters, download parts, and follow tutorials to construct working mechanical movements, demonstrating the potential as a creative educational medium. While these educational efforts inspire our long-term vision of making automata design accessible to a wide range of makers, in this paper we focus on advancing participatory approaches to computational design in the context of automata making. Specifically, we work closely with

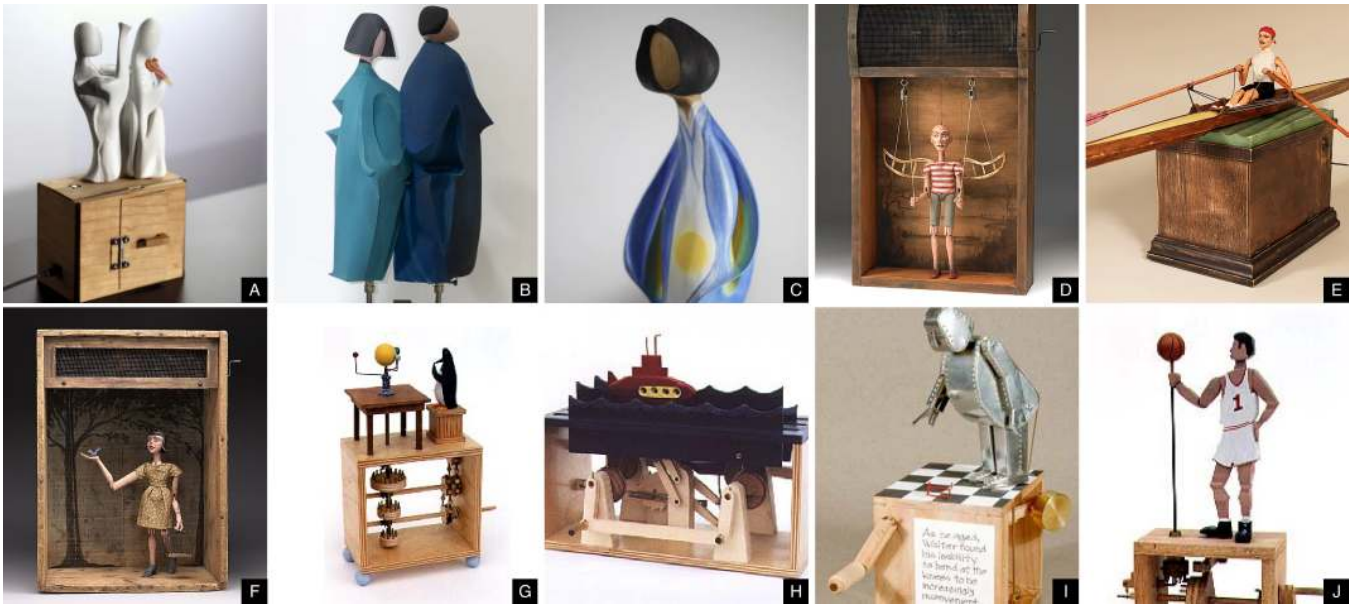


Figure 2: Prior work of collaborating artists. Lu Lyu: (A) *Engagement I* — minimalist ceramic-like duet on a crank box, (B) *Engagement II* — painted abstract pair with offset pivots, and (C) *Hontaru* — figurative solo sculpture with tapered body and head bob. Tom Haney: (D) *Visionary II* — a suspended figure with counterweighted arms pulling wing-strings, (E) *Rower* — a rowing automaton, and (F) *Fly Away Home* — narrative box with bird interaction. Marc Horovitz: (G) *Jasper Contemplates the Cosmos* — planetary orrery automaton with independent orbits, (H) *Submarines* — dual crank-driven subs alternately rising and submerging, and (I) *Walter* — tinplate figure bending unsuccessfully to retrieve glasses. (J) *The Basketball Player* — multitasking motion with ball spin, arm, and foot tapping.

professional automata artists through PD to examine how their practices and tacit knowledge can inform the integration of sketch-based interaction with mechanical synthesis, as demonstrated in MotionSmith.

2.2 Computational Design Systems for Mechanical Movements

Researchers in computational design systems have long aimed to support the creation of mechanical movements, given the complexity of designing and fabricating automata. Prior work can be broadly categorized into two streams: one focused on kinematic synthesis, which investigates mechanism synthesis algorithms to realize specific inputs, and another on interactive modalities, which enable users to express and refine their design goals.

In the kinematic synthesis literature, we draw an approach that translates user-specified motions into mechanisms such as linkages and gears through a combination of library search and continuous optimization. Motion-guided synthesis systems have advanced automated mechanism synthesis across various domains, ranging from linkage-based characters [**? ? ?**] and mechanical toys [**?**] to robotic devices [**?**] and mocap-driven automata [**?**]. For example, **?**] demonstrated this potential with a system that enables users to sketch motion trajectories and automatically generates the corresponding mechanical linkages and joints to realize them, while **?**] offers symbolic editing for refining existing linkage topologies. These systems demonstrate substantial computational capabilities once a target motion has been specified. However, they typically treat the user-defined motion path as a precise expression of intent, assuming users begin with a clearly articulated motion goal and focusing optimization on realizing that path. In MotionSmith, we adapt this computational strategy for generating candidate mechanisms but intentionally broaden the design space. We provide features that allow users to iteratively explore and refine their motion ideas, articulate intent as it emerges, and maintain agency in selecting among generated options. This approach is informed by insights from our collaboration with automata makers, who emphasized the importance of fluent iteration across design stages and a preference for mechanically simple, fabricable outcomes.

The second stream explores various interaction modalities that capture and communicate creative intent, such as sketches, which leverages the expressive qualities of drawing [**?**], motion capture, which digitally records embodied movement [**? ?**], and tangible user interfaces, which allow users to manipulate physical objects to define envisioned motions [**? ? ?**]. Among these, sketch-based interfaces are especially compelling because they enable intuitive and free-form exploration and align with long-standing creative practices. For example, sketch-driven systems have been used to couple freehand input with procedural semantics for tasks ranging from animating illustrations [**? ? ?**] and hand-drawn characters [**? ?**] to creating dynamic data visualizations [**? ? ? ?**].

These works demonstrate how sketching supports ideation and exploration while offloading technical complexity to the system. In the context of mechanical movement design, M.Sketch [**? ?**] enabled users to draw linkage-based mechanisms, simulate their motion, and iteratively refine them with solver support. This approach was later extended into AutomataStage [?], which combined AR-based video see-through applications with hardware configurations for interactive automata-making in education. Our system is aligned with this spirit of empowering makers through sketch-based expression, but extends beyond linkage mechanisms and contextualizes the design process by beginning with a figure image that anchors the maker's creative goal.

2.3 Fabrication-Aware Design Strategies

With the growing accessibility of CAD tools and digital fabrication technologies, creating 2D or 3D models and translating them into physical artifacts has become increasingly approachable for a wide range of makers. However, bridging the gap between digital design and physical fabrication remains challenging, particularly in domains such as making mechanical movements that demand precise assembly. Real-world considerations such as material thickness, friction, and tolerances often diverge from or are omitted in digital simulations [? ?]. Research in fabrication-aware computational design has proposed three main strategies to address this challenge. First, pre-validated parts restrict users to components guaranteed to be fabricatable (e.g., planar assemblies [?], modular electronic kits [?]). Second, constraint-guided design incorporates physical constraints into the design environment to provide instant feasibility checks, whether for structural stability [?], material deformation [?], or kinematic rules such as Grashof's condition [?]. Lastly, post-processing methods detect and correct fabrication issues in completed designs, such as verifying assembly [?], optimizing deformable characters [?], or generating printable connectors [?]. Our system builds primarily on the pre-validated parts approach but extends it by supporting parameterized modifications that preserve fabrication feasibility. For instance, by restricting components to pre-defined depth layers, we prevent physical interference without requiring complex 3D simulations.

2.4 Engaging Creative Practitioners as Domain Experts

While traditional participatory design (PD) involves diverse non-designer stakeholders (e.g., policy makers, healthcare professionals), research on creative tool development has increasingly recognized the importance of engaging crafters, artists, and designers as domain experts in shaping new technologies that empower and expand diverse creative practices [?]. Such collaborations often involve experienced practitioners exploring new technologies, demonstrating expected uses, and generating novel creative examples [? ? ?]. More recently, emerging approaches engage creative practitioners as long-term partners, rather than one-time evaluators, leveraging extended collaborations throughout the design process. Sustained partnerships draw on practitioners' tacit, embodied, and material knowledge accumulated through years of hands-on practice, offering insights that go beyond conventional user requirements in domains such as ceramics [? ?], digital pottery [? ?], and metal-working [? ?]. Examples include textile [?] and ceramic [? ?] residency programs, which integrate domain-specific artists' embodied knowledge into the technology design process [?]. In the same spirit, our participatory design with automata artists seeks to incorporate these insights through deeper engagement, enabling our technology to support broader creative practices.

3 Design Process and Goals

To design and build MotionSmith, we adapted a participatory design (PD) approach that centered collaboration with three automata artists. Through this process, we iteratively developed and refined the system prototype and ultimately conducted a multi-session study in which the participating artists used the system to design and fabricate their envisioned projects.

Before beginning the PD sessions, we spent 3-4 months on exploratory work, including iterative prototyping of mechanical models and a review of related literature. Over the following month and a half, we surveyed automata-maker communities, refined our list of expert practitioners, and submitted our study protocols for review by the Georgia Institute of Technology's Institutional Review Board (IRB). After receiving IRB approval, we initiated recruitment and confirmed our participants by the sixth month of the project.

3.1 Recruiting Collaborators

The study took place remotely using Zoom video conferencing platform. Despite the virtual platform used, we decided to scope our search within the United States for a possibility that we perhaps need to ship materials as moving to the final design & fabrication study phase.

To engage with the unique and complex design space of automata, we decided to work with established automata artists. We identified potential collaborators by reviewing online sources, including the *AutomataCon* website (https://www.automatacon.org), a biennial convention for automata makers and enthusiasts. From this search, we compiled a list of ten experts established in the field and sought to recruit 3-4 participants. Selection prioritized diversity across two dimensions: (1) artists' use of computational tools (e.g., CAD systems or digital fabrication machines), ranging from extensive to none, to account for varying expectations of CAD support; and (2) creative aesthetics in their portfolios, spanning descriptive to abstract, to avoid bias toward supporting a single visual style. In addition, we confirmed that participants had prior experience teaching novices or sharing instructional materials with the broader public. This ensured that they were not only skilled practitioners but also adept at articulating technical aspects of their work and reflecting on its relevance to wider maker communities. This consideration was important given our project's open-ended goal of exploring expert practices while also considering potential pathways for non-expert use. Based on these criteria, we invited four artists, all of whom initially consented to participate. One participant withdrew after the first interview due to scheduling conflicts, leaving three artists who completed the full PD process. Each received a modest stipend for their contributions.

Here we briefly introduce our three collaborators:

- **Lu Lyu** (LL: 30s, female) is an artist whose work explores fluid geometric transformations and abstract figurative forms realized through motor-driven linkages. She has developed

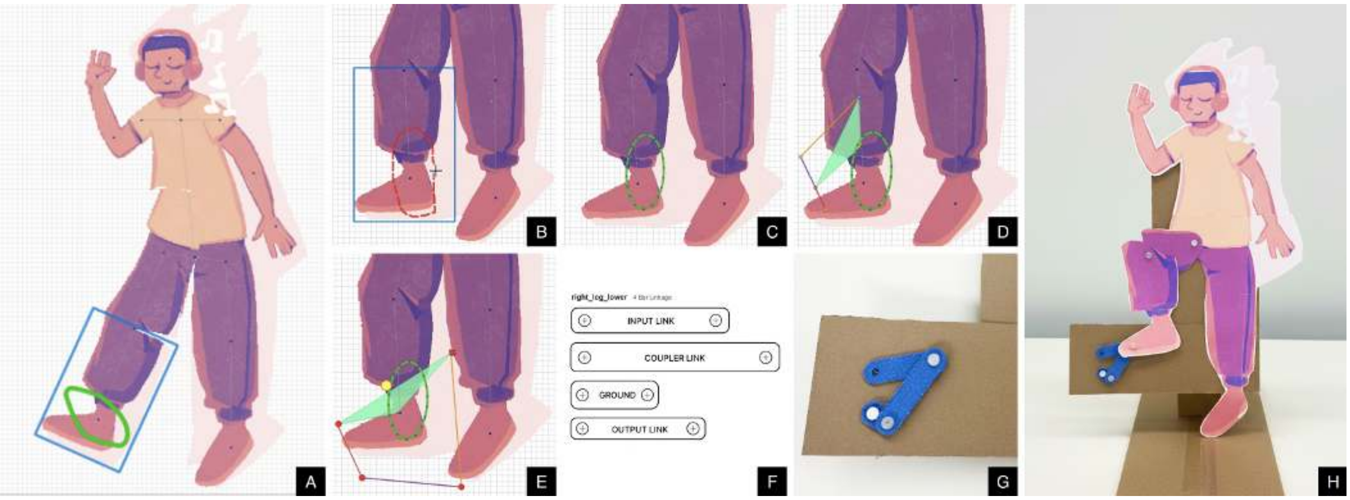


Figure 3: MotionSmith workflow from design to fabrication. (A) Users sketch a rough motion path on the canvas; the blue bounding box indicates the active component selection. (B) The system smooths the user drawn path (red curve) while preserving extremes. (C) Users adjust the smoothed path by repositioning vertices on the green curve. (D) The system applies a selected four-bar linkage candidate to support the specified motion path. (E) Editing mechanism with a motion preview that shows linkage alignment. (F) The system exports fabrication-ready blueprints of component profiles. (G) Physical parts fabricated from the blueprint (e.g., 3D-printed linkages). (H) Assembled automaton prototype demonstrating the leg-lifting motion. This pipeline scaffolds the three stages detailed below (Stage 1: A–C; Stage 2: D–E; Stage 3: F–H).

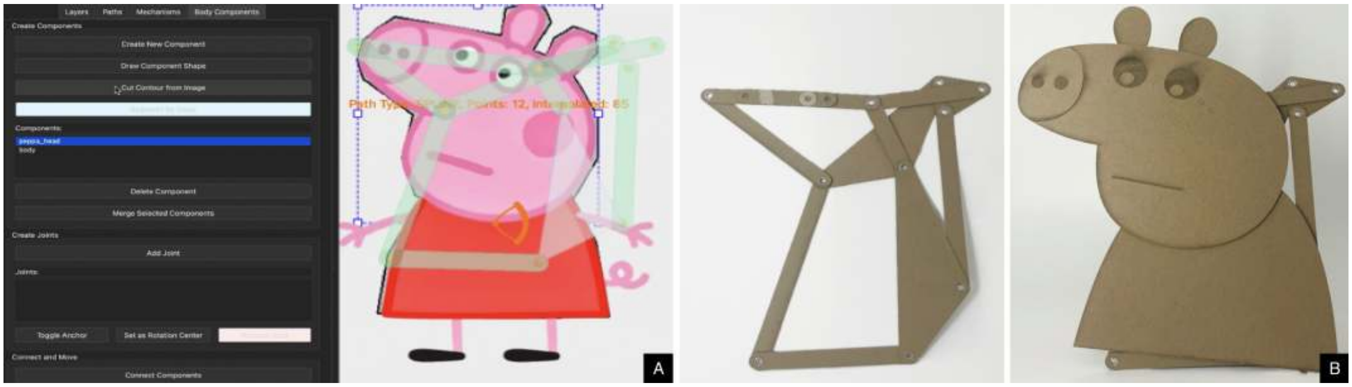


Figure 4: End-to-end early prototype of MotionSmith used in PD Phase 1. (A) Prototype interface where users sketch a motion path under an end-to-end assumption, triggering automatic mechanism synthesis. (B) Resulting linkage mechanism exported for fabrication. Because the solver treated the sketch as a rigid constraint, it produced dense, complex structures to match even minor hand-drawn irregularities. Expert feedback on this workflow highlighted the loss of user agency and the need for iterative interpretation rather than literal execution, motivating our shift to a staged, intent-aware system.

a practice that integrates CAD systems with digital fabrication such as laser cutting and 3D printing, producing precisely engineered components like gears and joints. For over seven years, she has produced commissioned installations for clients including Google, Baidu, Shopify, Stripe, and Macy's. Alongside her practice, she teaches workshops at a local makerspace, supporting young students and hobbyist makers in kinetic design (Figure 2A-C).

- **Tom Haney** (TH: 60s, male) is a kinetic artist who began his career building props, models, and miniatures for film and television commercials. His portfolio is broad, but much of his work emphasizes figurative forms and lifelike motion. While interested in exploring modern CAD tools, he currently makes little use of CAD or digital fabrication, relying instead on hand sketches and traditional craft techniques such as wood carving, polymer clay, and fabric sewing, preferring direct material engagement. He also leads workshops for maker communities, sharing his expertise in automata building (Figure 2D-F).

- **Marc Horovitz** (MH: 70s, male) has been building automata since the mid-1990s, inspired by the traditions of British automatists and the Cabaret Mechanical Theatre [**?**]. His practice centers on handcrafted mechanisms in wood and metal, where every cam, gear, lever, and linkage is individually fabricated. He works iteratively from sketches and occasional CAD drafting, refining designs through repeated fitting and assembly. In addition to his studio practice, he has edited a long-running magazine dedicated to automata makers, helping to connect practitioners worldwide and document the field's history and innovations (Figure 2G-J).

3.2 PD Overview

Our study followed three phases of participatory design (PD) sessions: (1) learning from artists' creative processes (~60 mins), (2) introducing the MotionSmith prototype and inviting ideas and feedback (~90 mins), and (3) multi-session hands-on design and fabrication using the revised system (8-44 hours in total). Between sessions, the first author maintained active communication with the three artists, following up with clarifying questions and gathering additional insights. We also allocated one to one and a half months between phases for system development, integrating lessons learned from each session. The PD process concluded with post-interviews and surveys to capture participants' reflections. All sessions were conducted individually and remotely via Zoom, with video recordings collected for analysis. Below we report on the first two phases that shaped the design goals for MotionSmith; the third phase is presented in Section 5.

3.2.1 Phase 1: learning from artists' creative processes. To ground our ideation in artists' practices, we conducted semi-structured individual interviews (~60 min each) with our collaborators. These conversations helped us understand how artists generate ideas, what factors shape their design decisions, and what tools and processes they use in design and fabrication. We also shared a rough sketch and short concept video of our initial system direction, which focused on identifying motion paths and generating underlying mechanisms (drawing on Coros et al. [**?**]), and asked for their feedback on feasibility and refinement. At this stage, we adopted a fully automated mechanism-generation approach. We expected that artists would specify a motion path and the system would directly solve for an underlying mechanism, treating the input motion as final without requiring further refinement. Finally, we explored participants' perspectives on developing sketch-based computational tools for automata making, including potential excitement, concerns, and desired features.

Results. Participants described diverse approaches to ideation and design. Their processes varied by project: sometimes beginning with sketches of specific mechanical movements (e.g., four-bar linkages), other times generating and refining ideas while directly interacting with materials and parts. When presented with our sketch-based system direction, all participants expressed enthusiasm and offered constructive suggestions. Two key requirements emerged: (1) iterative sketching of motion paths with the ability to select different supporting mechanisms, and (2) visualization to see immediate changes. LL and MH also emphasized that the system should only present fabricable designs to avoid misleading users with infeasible movements or mechanisms to build.

Perspectives on computational design tools diverged among participants. LL, who actively uses CAD, expressed strong enthusiasm and curiosity about computational design affordances. By contrast, TH and MH, who rely less on digital tools, were more neutral, with MH voicing concerns. In particular, during discussions of computational tools for creatives in the AI era, MH emphasized that computational tools should function as a coach rather than a replacement, guiding learning and preserving creative agency. This emphasis on maintaining agency resonated across the group. Similarly, regarding our initial system sketch, all three participants felt that the early prototype's automated solver distanced them from their workflow. They pointed out that the dense, optimized mechanisms were hard to fabricate and interpret, and more importantly, removed opportunities for learning and constrained their creative agency within the design loop.

Participants also reflected on the affordances and challenges of different mechanical mechanisms. For example, TH and MH highlighted cams as a foundational mechanism for automata makers, noting their transparent operation and the wide range of expressive variations they enable. When discussing where computational tools felt most needed, LL and TH pointed to four-bar linkages as a prime example. Because this mechanism is conceptually complex and demands high precision, they reported using digital tools (e.g., Adobe Illustrator) during design.

Outcomes. Insights from this phase informed four design goals for our initial system prototype: (1) enabling iterative editing of both motion-path sketches and supporting mechanisms, (2) providing real-time simulation of changes, (3) setting the scope to fabricable mechanical movements only, and (4) leveraging computational design to generate editable design options while leaving decisions to the user. These insights drove a fundamental shift in our design approach. We moved away from the initial goal of instantly calculating a complex mechanism from a single sketch. Instead, we aimed to support a process where users can continually refine their intents, narrowing the system's scope to three specific mechanism families (four-bar linkages, cam-followers, and spur/planetary gears) that align with expert needs.

3.2.2 Phase 2: Introducing the MotionSmith early prototype and inviting ideas. To validate our design direction and surface potential usability issues, we conducted 90-minute design sessions with our collaborators using an early prototype of MotionSmith. This prototype was developed in response to collaborators' feedback on our Phase 1 study, implementing an initial version of the intent-aware system while still retaining some of our original end-to-end assumptions. The prototype supported a workflow in which users selected a character image from a small repository (three humanoid styles: kid, alien, astronaut). The system then embedded a skeleton into the image, enabling users to select joints, sketch motion paths, and review three alternative mechanism designs before applying one.

Each session began with a brief introduction and reminder of project goals (~5 min), followed by a short demonstration of the prototype (~5 min). Participants then explored the system through a self-defined design task using Zoom's remote-control while thinking aloud (~40 min). Next, we presented envisioned features for the next iteration (e.g., synthesizing multiple regional movements by interlocking gears driven by a single force; supporting fabrication by auto-generating a case or background sheet to separate mechanisms from objects) and invited feedback (~20 min). Sessions concluded with reflective discussions on potential applications and refinements (~20 min).

Results. All participants responded positively to the overall workflow of sketching motion paths and generating supporting mechanisms. They engaged in multiple cycles of refinement, often revising intended motions while sketching and observing the resulting character movement. Participants frequently returned to the sketching phase after reviewing system-generated mechanisms or adjusting which parts of the character were animated.

A recurring concern was limited control over joint behavior. Because the system auto-generated a human-like skeleton with default joint rotations, LL and MH noted that their automata often required exaggerated or non-naturalistic motions for creative expression. They requested features to redefine both direction and range of joint rotations. Another usability challenge involved mismatches between user intent and system interpretation. While the system translated sketched paths literally, hand-drawn irregularities (e.g., jaggedness when depicting a raised hand) were reproduced in the generated mechanisms, yielding overly complex results that did not match participants' intended motions. Once participants understood how the system had interpreted their sketches, they clarified their inputs, but emphasized the need for greater interpretive flexibility. Finally, participants critiqued the mechanism suggestions: the three alternatives often reflected small variations within a single gear-based design rather than distinct options. Because these were typically dense with gears and linkages, artists questioned practical feasibility, describing them as difficult and unstable to fabricate. They expressed a strong preference for simpler and more diverse options (e.g., incorporating varied cam shapes), which they described as both easier to build and versatile for creative variation.

Outcomes. Based on this feedback, we identified three directions for system development: (1) enhance user agency by expanding freedom in sketching motion paths and editing system-generated mechanisms, (2) provide options to simplify design intent for fabricable automata, and (3) restructure the interface to reduce navigation friction by consolidating key stages into a unified view (e.g., moving between motion-path sketching, mechanism generation, and mechanism editing without repeated menu selections).

3.3 MotionSmith System Design Goals

Drawing on insights from the PD sessions, our experiences building and testing early prototypes, and relevant prior work, we synthesized the following Design Goals (DG1–DG3) to guide system development for a broader community of automata makers.

DG1. Tinkerable. Support exploratory design by embracing the iterative, nonlinear nature of creative practice. Users can move across stages, from defining motions to modifying mechanisms, while retaining agency to refine and redirect their process. Building on the tinkerable design literature [**?**], we frame tinkering as a playful, iterative mode in which makers reassess goals, explore alternatives, and imagine new possibilities. Our system supports this approach by providing immediate feedback, visualizing the interactive design process, and enabling fluid experimentation.

DG2. Fabricatable. Bridge digital design and physical making. The system encourages models that are realizable and stable, suggesting mechanisms that are simple to construct yet effective in approximating intended motion. Where possible, the system exposes constraints (e.g., kerf, tolerance) and offers simplification levers to reduce complexity without erasing intent.

DG3. Interpretation over Direct Adaptation. Realize makers' creative vision without constraining them to current technical skill. Rather than interpreting input literally, the system should infer intended motion and provide controls for expressing and refining ideas (e.g., smoothing or idealizing sketches, altering joint axes/limits, selecting qualitatively distinct mechanism families) while keeping users in charge of final decisions.

4 The MotionSmith System

MotionSmith is a sketch-based computational design system for creating automata. The system combines direct on-canvas manipulation with solver-based reasoning, in which constraint solvers compute valid kinematics and mechanism configurations. Users can sketch and edit directly on a canvas over an uploaded image, while real-time simulation renders constraints and kinematics. MotionSmith is designed to support exploratory, tinkering-like workflows in which makers continually iterate on their ideas, revisiting earlier stages of design to test alternatives and pursue new possibilities. To ensure the outputs remain fabricable, the system proposes mechanism solutions that simplify construction by reducing part count and precision needs. The system also incorporates features that help interpret and scaffold makers' creative intent, enabling articulation and refinement of ideas such as intended motion paths, expressive pacing, and fabricable mechanism constraints. Figure 3 shows MotionSmith's end-to-end pipeline from sketching a target motion (A–C; Stage 1 features in Table 2), through mechanism synthesis and simulation (D, E; Stage 2 features), to fabrication-ready output and assembly (F–H; Stage 3 features).

The system provides these key functionalities across three stages:

- *Stage 1. Identifying a goal:* Users upload an image, configure the skeleton, and sketch a motion path to define the intended movement. (see Figure 3 A–C)
- *Stage 2. Generating supporting mechanism candidates:* The system generates three alternative mechanism designs, which users can simulate and refine through interactive editing. (see Figure 3 D–E)
- *Stage 3. Supporting fabrication:* The finalized design is converted into fabrication-ready blueprints that can be exported as SVG or PDF files. (see Figure 3 F-H)

Across these stages, MotionSmith supports a tinkerable workflow by distributing control over motion intent along the creative process rather than concentrating it at a single point: Stage 1 focuses on externalizing qualitative motion ideas as sketches, rig edits, and pacing; Stage 2 refines that intent into parametric mechanisms under feasibility guardrails; and Stage 3 encodes fabrication decisions in blueprint form. This staging mirrors phases repeatedly observed in our PD sessions with automata artists, and provides fine-grained handles on intent where experts reported making the most consequential creative and mechanical choices.

While Figure 5 illustrating interfaces, Table 2 and Figure 8 enumerate the key features by stage, coupling specific user actions with the forms of agency the system is designed to support.

4.1 Stage 1. Identifying a goal

In the first stage, the maker begins by creating an articulated 2D character and specifying a desirable animation as a motion curve, which serves as the goal for the subsequent stage.

The process starts with uploading a rasterized image of the character to be built. Our system then proposes a human-like rig using a template-based pose-estimation initializer inspired by Animated Drawings [?] and CharSegNet [?]. This initializer segments the drawing into named parts through a lightweight stroke–pixel graph seeded by joint locations, yielding editable part masks and attachment handles suitable for later anchoring and fabrication (Figure 6A). The output is a set of named joints (e.g., `head`, `torso`, `L/R-arm-upper/lower`, and `L/R-leg-upper/lower`) and their 2D positions, while following the pre-defined hierarchy of the template. While the default template assumes a human-like character, the maker may switch to alternative templates or define their own for non-human characters.

Once the articulated character is established, the maker specifies the desired motion of selected joints. Users first indicate the target joint to be animated and then draw a motion curve as time-indexed keypoints $(t_i, \mathbf{x}_i)$ over the duration $0 \leq t_i \leq T$. As a result, the curve captures both positional movements and their velocity profiles. After the keypoints are defined, tolerance-based smoothing is applied while preserving extreme positions, followed by fitting Catmull–Rom splines for interpolation (Figure 6B).

The system supports iterative refinement, allowing users to review visualizations of the motion (Figure 6D), adjust keypoints, or modify parameterized features that influence smoothness and scale. In some cases, a motion curve may fall outside the feasible range of the specified joint, making the motion mechanically infeasible. To address this, the system automatically snaps the curve to the nearest realizable trajectory while still displaying the original curve for reference (Figure 6C). This iterative editing with automatic correction is designed to maintain user intent while increasing the feasibility of mechanism synthesis in the next stage. Together, these interactions instantiate the Stage 1 features ("Initialization," "Motion path sketching," and "Simulation and pacing") summarized in Table 2.

4.2 Stage 2. Generating and editing supporting mechanism candidates

After refining the motion path, the maker clicks the *Get Mechanism* button in the left panel, prompting the system to generate three distinct mechanism candidates. Our mechanism selection algorithm follows a template-based approach inspired by ?], which combines mechanism selection from pre-generated designs followed by parameter refinement. Figure 7 illustrates the three families we surface to users (A–C) complementing the editable parameters and guardrails summarized in Table 1.

The system begins by retrieving promising mechanisms from the pre-computed database that can closely approximate the maker-specified motion curve. To ensure coverage, we populate candidates across three mechanism families: four-bar linkages, cam-followers, and spur/planetary gears. During offline generation, we filter out infeasible designs (e.g., self-intersecting linkages) to ensure that the database contains only viable starting points. At runtime, the system queries this database to find candidates that approximate the user's target motion. The details of the parameters are illustrated in Table 1.

We select three top edit-stable candidates, one from each family, by pairing perceptual-faithful curve matching with feasibility-aware refinement: we minimize the normalized directed Hausdorff distance between the maker's motion curve $\mathbf{x}_{1,\cdots,N}$ and the candidate curve $\hat{\mathbf{x}}_{1,\cdots,N}$. We resample time indices with a fixed timestep to match velocity profile, ensuring that candidates are compared under a shared pacing. Each selected mechanism is then further refined through constrained optimization of the parameter θ.

$$\min_{\theta} \; d_{\text{haus}}\big(\mathbf{x}_{1,\cdots,N}, \hat{\mathbf{x}}_{1,\cdots,N}(\theta)\big) \quad \text{s.t. } c_m(\theta) > 0, \quad m = 1, \cdots, M. \tag{1}$$

While the primary objective is to minimize Hausdorff distance d_{haus}, the optimization also incorporates feasibility constraints c_m, such as ensuring minimal curvature exceeds a specified threshold for Cam-followers. The further details are described in Table 1. These constraints are expressed as soft penalties with L2 regularization and a weighting parameter λ, and the resulting objective is solved using the sampling-based CMA-ES algorithm [?]. After optimization, the system reports an accuracy score, defined as

$$\text{Acc} = 100 \cdot \left(1 - \frac{d^{\star}_{\text{haus}}}{D_{\text{norm}}}\right),$$

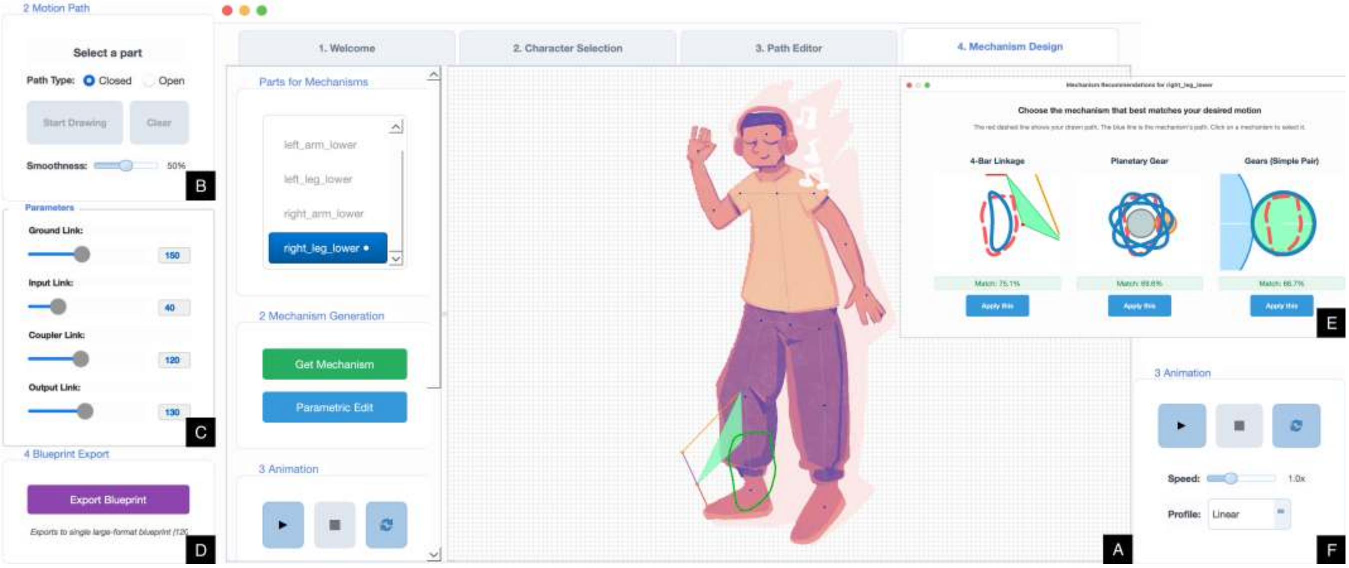
where $d^{\star}_{\text{haus}}$ is the Hausdorff distance after alignment between the user path and a candidate trajectory, and D_{norm} is a shape-intrinsic normalizer, where we use the path's bounding box diagonal. These scores are presented as "Match %" in the interface (Figure 5E).

Once the initial candidate is selected, the maker may continue to edit the trajectory (Stage 1) or mechanism parameters directly (Stage 2) on the canvas. Each edit triggers a re-execution of the constrained optimization with a warm start, allowing the system to refine the candidate mechanism while preserving feasibility. This process enables the maker to iteratively explore, verify, and adjust their design within the system and ensures that mechanism synthesis and creative intent remain coupled throughout the workflow. These interactions instantiate the Stage 2 features ("Candidate mechanism generation," "On-canvas mechanism editing," and "Diagnostics overlays") in Table 2.

Our system supports three mechanism families: cams, four-bar linkages, and gears. As noted in section 3.2.1, artists identified cams as compelling for their operational transparency and four-bar linkages as high-precision candidates benefiting from computational support. We included gears as an additional mechanism family to ensure broad applicability across diverse automata designs.

4.3 Stage 3. Supporting fabrication

As the final stage of the workflow, the system enables the maker to generate a fabrication-ready blueprint of the design(see Table 2). To



Path Type: ● Closed ○ Open

Smoothness: 50%

Ground Link: 150

Input Link: 40

Coupler Link: 120

Output Link: 130

Mechanism Recommendations for right_leg_lower

Choose the mechanism that best matches your desired motion

The red dashed line shows your drawn path. The blue line is the mechanism's path. Click on a mechanism to select it.

Match: 75.1%

Match: 69.6%

Match: 66.7%

Speed: 1.0x

Profile: Linear

Figure 5: MotionSmith interface. (A) Main canvas for on-image authoring: users select a body part and draw a motion path; live simulation shows the traced trajectory and kinematic overlays (magnified inset). (B) Path drawing with smoothing: a slider toggles raw versus idealized curves while preserving extreme poses. (C) Parametric control (example: four-bar): bounded sliders expose ground/input/coupler/output lengths and anchors; edits are guard-aware (e.g., transmission angle, branch consistency). (D) Blueprint export: generates character/mechanism packets (SVG/PDF) with minimal dimensions for fabrication. (E) Mechanism candidates: a dialog presents ranked alternatives (four-bar, cam–follower, gears) with similarity scores; a chosen candidate is instantiated on the canvas for editing. (F) Animation control: play/stop, speed, and timing profile (e.g., linear/eased) for pacing review independent of geometry. Collectively, these panels foreground a tinkerable yet guard-aware workflow.

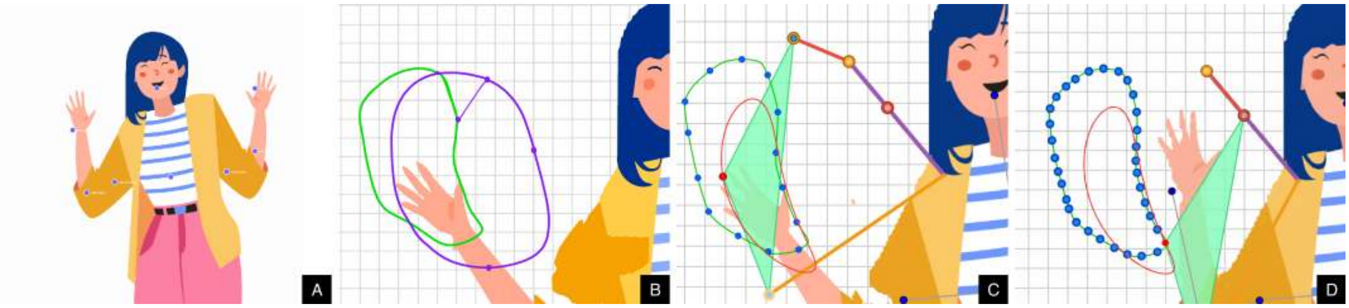


Figure 6: Authoring motion. (A) The initializer proposes a skeleton with editable part masks over the character image. (B) Users draw motion paths, shown with dual-tracks: raw paths (green) and max smoothed curves (purple). (C) The system generates a mechanism aligned with the path and retargets motion to the nearest feasible pose while preserving the target curve. (D) Sampled poses illustrate pacing along the path, normalized to the chosen animation duration.

support verification before fabrication, the system provides multiple views such as orthographic projections for dimension accuracy, isometric perspectives for overall form, and exploded or sectional views that reveal internal assemblies. These views allow the maker to inspect alignment, fit, and assembly feasibility before committing to fabrication. Panels G–I in Figure 8 depict the exported packets and their translation to partial and final assemblies.

The resulting blueprint is exported as an SVG or PDF file, ready for use with digital fabrication tools. To ensure persistence and precise export, projects are saved as JSON with explicit units. At export time, the system generates two distinct vector packets: a character packet that contains silhouette layers and attachment holes, and a mechanism packet that contains layered drawings with a minimal dimension set as well as optional isometric or exploded inserts. To prevent physical interference, the system employs a deterministic Z-layering strategy. Mechanisms and character parts are assigned to pre-defined depth layers, such as ground, mechanism layers, and character plane, and the constraint solver inherently rejects topologies with potential collisions. Fabrication-specific parameters such as kerf width, spacing, and minimum size thresholds are

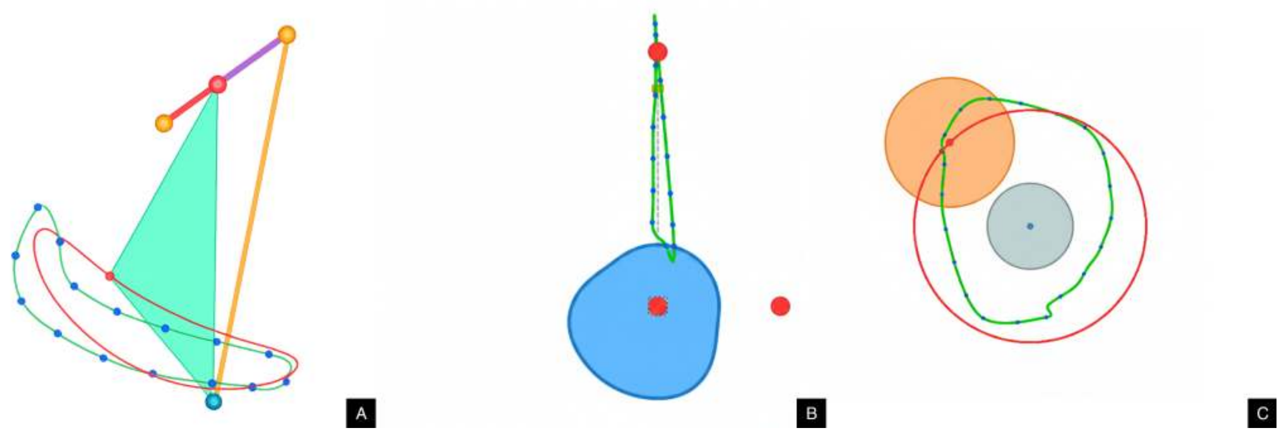


Figure 7: Mechanism families in MotionSmith. (A) Four-bar linkage generated from a user-drawn path. (B) Cam–follower configuration mapping motion to rotation. (C) Gear pair in motion, showing driver and follower gears.

Family	Editable Parameters	Invariants	Constraints
Four-bar linkage	Ground placement Moving pivot locations Bar lengths	Joint types Planar assembly	Branch consistency Transmission angle band Non-degenerate lengths
Cam–follower	Lift/dwell control points Follower tip type	Guided follower Layer planarity	Curvature minima
Spur/Planetary gears	Module Tooth count(s) Center distances	Pressure-angle range Integer teeth	Module/center-distance compatibility

Table 1: Mechanism parameters. For each family, we list: *Editable parameters*, which expose user-facing degrees of freedom for interactive tinkering; *Invariants*, which remain fixed by definition of the mechanism type and enforce consistency during solver-based editing; and *Constraints*, which are automatically checked to ensure mechanical feasibility (e.g., branch consistency and transmission angle limits for linkages, curvature bounds for cam profiles, and compatibility between module and center distance for gears). These constraints act as guardrails that maintain fabricability while still affording creative flexibility in parameter exploration.

initialized with empirical defaults derived from prior builds, and makers can adjust these parameters at export time to match their fabrication context. Together with panels E–H in Figure 3, these views substantiate the 'fabricatable by design' claim that anchors our system goals.

4.4 Implementation Details

MotionSmith is implemented as a cross-platform desktop application for macOS and Windows. The interface is implemented with `PyQt6`, while real-time rendering uses `QtQuick`/OpenGL. A C++/Python core, exposed through `pybind11`, provides geometry and kinematics utilities, mechanism search, and feasibility checks. All inference and simulation run locally, maintaining interactive performance at 60 FPS on laptops.

5 System Deployment in Practice

In the final phase of PD, we investigated the use of the MotionSmith system for a full cycle of automata design and fabrication. Each artist engaged over multiple sessions, ranging from 8 to 44 hours to assess the feasibility of MotionSmith in practice and its alignment with our design goals (Section 3.3).

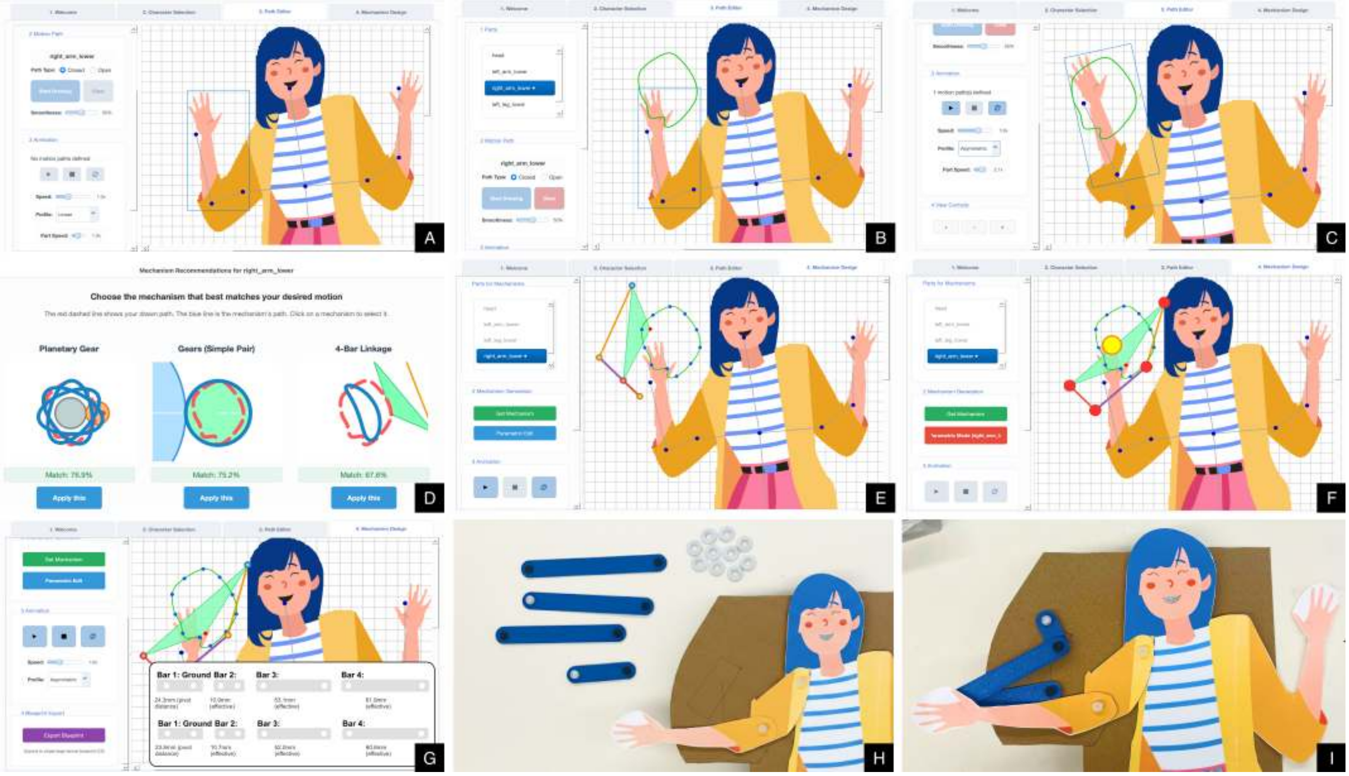
We examined two primary research questions:

RQ 1. What controls are essential to support agency and better capture makers' creative intent? Here we investigate controls over motion-path authoring, anchor/pivot placement, joint axes and limits, per-part pacing, and parameter edits to mechanism candidates (e.g., bar lengths/anchors; cam lift/dwell) within single-DoF actuation. These controls correspond to the Stage 1 and Stage 2 features summarized in Table 2.

RQ 2. How do makers negotiate the trade-offs between simplicity and motion fidelity? We examine how participants set design goals, whether and how they simplify motion-path sketches,

System feature	User interaction	Intended support for agency
	Stage 1: Identifying a goal	
Initialization	Upload an image; generates a skeleton with editable parts and handles.	Quickly set up an articulated character while retaining flexibility for changes.
Motion path sketching	Draw or edit a curve for a target joint; adjust keypoints or use smoothing slider.	Externalize intended motion in a sketchable form that can be refined iteratively.
Simulation and pacing	Play back motion with live preview; adjust speed and select linear/eased timing.	Review expressive pacing independently of geometry and explore variations.
	Stage 2: Mechanism exploration	
Candidate mechanism generation	Click *Get Mechanism* to see three families (four-bar, cam–follower, gears).	Compare different feasible realizations of the same intent before committing.
On-canvas mechanism editing	Adjust anchors, pivots, or lengths through bounded sliders with guards.	Safely tinker with mechanism parameters while solver maintains feasibility.
Diagnostics overlays	Visual overlays show infeasible poses, conflicts, and unreachable segments.	Prevent hidden errors and support a "learn by doing" style of tinkering.
	Stage 3: Fabrication support	
Blueprint export	Generate fabrication-ready outputs as SVG/PDF with bill of materials included.	Bridge digital design to tangible automata with fabrication-safe exports.

Table 2: Key features of MotionSmith organized across the three workflow stages. Stage 1 establishes the design goal by initializing a rig and sketching desired motion paths. Stage 2 supports exploration and refinement through system-generated mechanism candidates, interactive parameter editing, and visual diagnostics. Stage 3 bridges digital and physical domains by exporting fabrication-ready blueprints, ensuring that exploratory designs remain realizable as tangible automata.



The red dashed line shows your drawn path. The blue line is the mechanism's path. Click on a mechanism to select it.

Figure 8: System features of MotionSmith. Panels (A–G) illustrate the features listed in Table 2; panels (H–I) depict fabrication outcomes: partial assembly (H) and the final prototype demonstrating the intended motion (I).

and how they select among three system-generated mechanism candidates, balancing ease of fabrication with accurate realization of envisioned motion. While our central focus was on the design process with MotionSmith, we also examine **fabrication-support needs**, investigating whether and how the system-generated blueprints integrate into makers' preferred fabrication workflows.

We ground these questions in three artist-led cases, each foregrounding different control stacks (RQ1) and mechanism–fabrication choices (RQ2) progressing from digital authoring (Subfigure A), through mechanism design and fabrication processes (Subfigure B–E), to completed outcomes (Subfigure F) in Figures 9 to 11. Throughout all sessions, we asked participants to think aloud, enabling us to understand their decision-making considerations and articulate the reasoning behind their actions.

5.1 Procedure

Before the first session, the first author asked collaborators to sketch the figures they intended to create and provided support for installing the system on their own computers. While participants primarily used their local machines for design and exploration, they later switched to Zoom's remote control due to rendering incompatibilities between macOS and Windows.

Each session began with an overview of the MotionSmith workflow: (1) load an image of the figure to be animated; (2) select a target part and sketch its intended motion path; (3) browse and select a candidate mechanism generated by the system; (4) refine the mechanism through optimization; and (5) generate and download a blueprint for fabrication.

We initially reserved a two-day window (8-10 hours in total) for the final phase, but we reminded all artists that the schedule could be flexible, allowing the phase to end earlier or extend longer in a manner consistent with their typical workflow. As a result, LL spent approximately 13 hours, TH about 8 hours, and MH a total of 44 hours over 16 days on design and fabrication. All participants noted that the time required for projects is inherently variable and often unpredictable, and they felt that their engagement in this study closely mirrored their normal practice workflows.

LL and MH began with specific design goals (LL: a jumping motion using folded legs; MH: raising both arms), whereas TH developed ideas iteratively through exploration. Across sessions, throughout their design exploration with MotionSmith, all participants actively revised their designs. Some extended original goals—for instance, LL added emotional nuance through "shy" gestures (Figure 9A), and MH expanded from arm to full-body motion (Figure 11A). Others embraced unexpected opportunities, such as TH recognizing the humor in arm-raising (Figure 10A). This iteration was not linear but emerged through cycles of revisiting multiple workflow stages (Sections 4.1 to 4.3). Accordingly, LL's piece emphasizes subtle expressivity via a hybrid of 3D-printed linkages and paper craft (Figure 9), TH spotlights asymmetric cam profiles hand-tuned for a lively 'cheer' (Figure 10), and MH scales to compound cam assemblies balancing simplicity and presence (Figure 11).

Each participant also concluded with distinct fabrication strategies. LL exported an SVG, extruded it into a 3D model (Autodesk Fusion 360), and 3D-printed linkage components. TH and MH instead printed paper blueprints, using character parts directly and fabricating cams to blueprint dimensions. Figures 9 to 11 depict this handoff: linkages exported and extruded for printing (LL), and 1:1 cam/linkage packets used as cutting jigs in woodshops (TH, MH).

5.2 Findings

5.2.1 *Iterative Sketching and Mechanism Refinement Controls (RQ1).*

Throughout Stages 1 & 2, artists actively used rig controls, motion path sketching, pacing, and mechanism editing to sustain agency. Accuracy scores of candidate mechanisms and diagnostics played little role; instead, participants valued direct manipulation that supported iterative creative exploration. These interactions align with the Stage 1 and Stage 2 features in Table 2.

In Stage 1, all artists actively engaged with the provided features. Each began by importing figure drawings they wished to animate and, upon loading these images, sketched motion paths directly on the canvas while checking results through simulation. During this process, they frequently revised rigs by relocating joints or adjusting rotation ranges. For example, LL repositioned joints multiple times to better articulate a foot-lifting pose, explaining that she was searching for a specific "shy rhythm." MH similarly adjusted arms and legs to develop the concept of an "angel pose." Motion path sketching was highly iterative, often prompting artists to rethink their initial ideas. LL shifted from a simple rotational arm motion to a shy elephant lifting its leg while holding a flower, noting that the emotional tone emerged through tinkering. TH explored multiple paths, using simulation feedback to judge whether the movement was "funny enough." MH produced extensive sketches both before and after committing to the angel pose, refining the motion until it aligned with his envisioned image. Artists also experimented with the motion path smoothing feature. LL and MH applied it to reduce jitter in hand-drawn paths, whereas TH deliberately avoided it after trials, choosing instead to preserve a playful roughness. Pacing was explored by all participants as a means of refining expression. LL applied it to accentuate movement, TH iteratively tuned timing for humor, and MH synchronized limb motions to strengthen the angel pose. Across cases, pacing functioned as a final adjustment to solidify expressive intent.

In Stage 2, when displaying mechanism candidates, the system also indicated accuracy scores showing how well each mechanism matched the identified motion. However, artists paid little attention to these accuracy levels, focusing instead on the expressive qualities of the motion. After selecting a mechanism, all participants further modified it using on-canvas editing features. For example, LL adjusted linkage positions to prepare for fabrication, while TH and MH repeatedly repositioned cam anchors to refine alignment. Although the diagnostic feature that flagged infeasible poses and conflicts was continuously active in the background, none of the artists explicitly sought it out or relied on it when making design decisions.

In summary, the MotionSmith system supported artists in identifying and articulating creative intent, enabling hands-on, iterative exploration of motion and mechanisms. Analytical features, such as accuracy scores and diagnostics, played a secondary role, as artists primarily relied on direct manipulation to refine motion

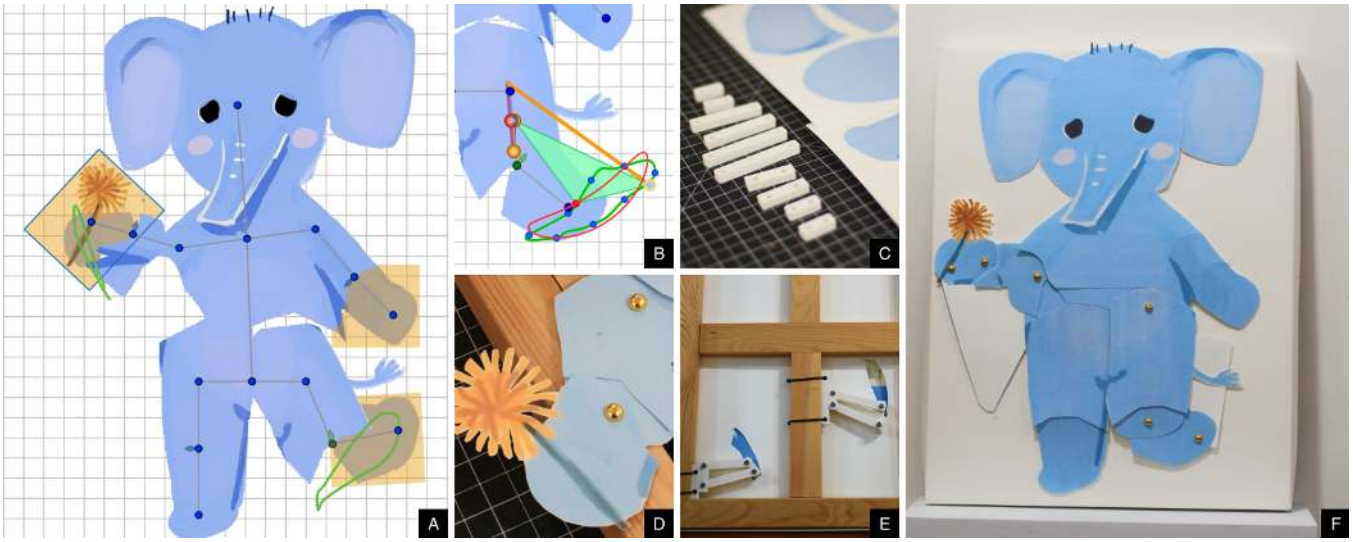


Figure 9: Design and fabrication process by LL: a shy elephant holding a flower. Starting with digital authoring in MotionSmith (A), LL defined arm and leg gestures using editable skeleton rigs and motion paths. The system then generated and refined a candidate four-bar linkage for the leg motion (B). These designs were realized as 3D-printed linkage components and cardstock character layers (C), which were assembled with brass fasteners to integrate ornamental details such as the flower (D). The components were mounted on a wooden frame for stability and actuation (E). The process culminated in a completed automaton (F) that blends 3D-printed linkages and papercraft to convey subtle, shy expressions.

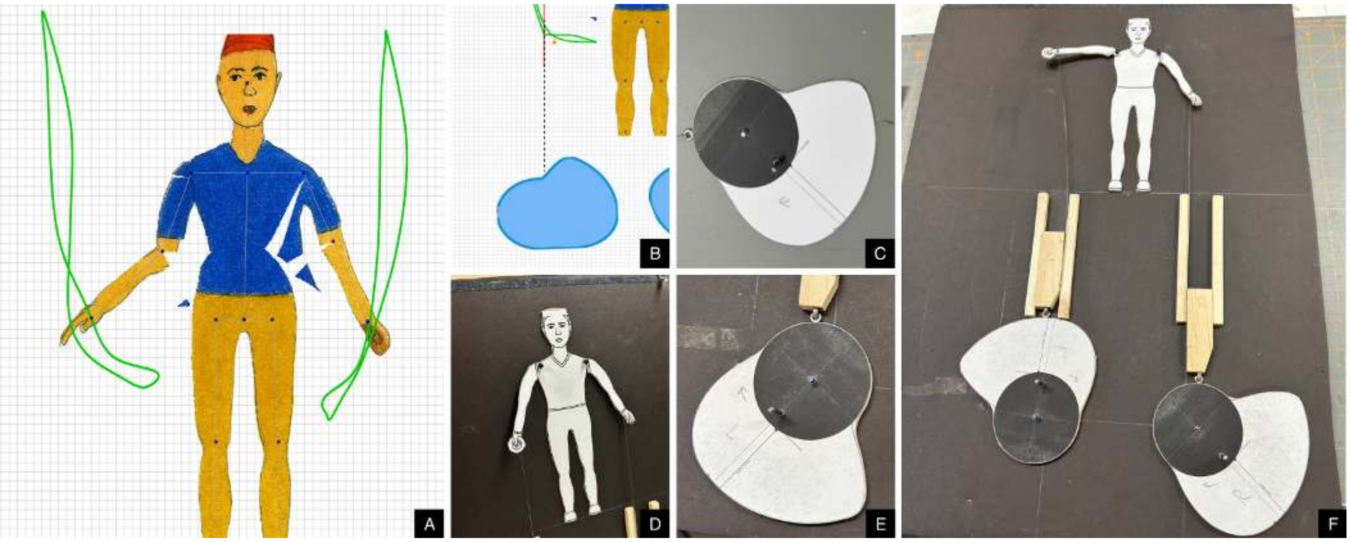


Figure 10: Design and fabrication process by TH: a cheering man. Starting with digital authoring in MotionSmith (A), TH sketched upward motion paths for both arms. The system then generated candidate cam profiles to approximate the desired trajectories (B), which were realized as asymmetric paper cam templates (C). These templates were connected to the character's arms using wire linkages (D), and mounted with wooden followers into a frame for synchronized actuation (E). The process culminated in a completed automaton (F) that conveys a lively cheering gesture through paired cam-driven motions.

paths, pacing, and mechanisms in alignment with their expressive goals.

5.2.2 Negotiating trade-offs in design decisions (RQ2). Throughout the design process, makers continuously weighed trade-offs that extended beyond achieving precise motion accuracy. We observed three recurring considerations guiding their design decisions: comprehensibility (does the motion make sense at a glance?), expressiveness (does the motion convey an aesthetic quality?), and

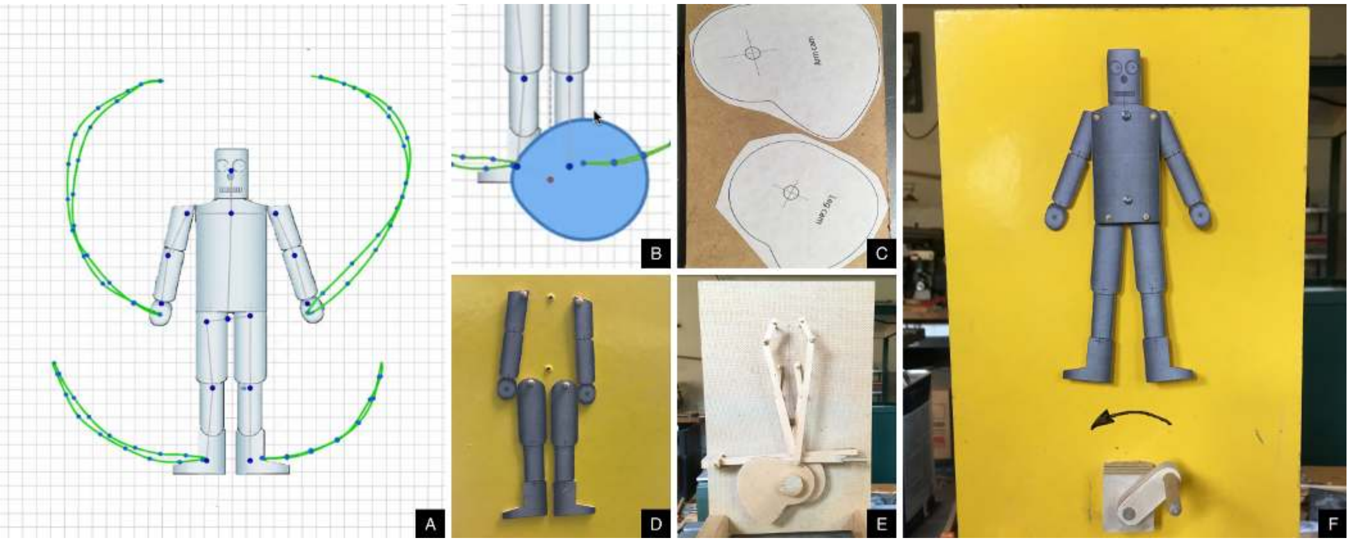


Figure 11: Design and fabrication process by MH: a tin robot with angel pose. Starting with digital authoring in MotionSmith (A), MH specified symmetric motion paths for the robot's arms and legs. The system then generated candidate cam profiles aligned with these trajectories (B), which were realized as paper templates for arm and leg cams (C). Additional limb components were cut and prepared for assembly (D), and mounted together with the cam mechanism on a wooden backing board (E). The process culminated in a completed automaton (F), where a tin robot figure performs an angel-like gesture through a crank-actuated cam system.

fabricability (can the design be built reliably?). Rather than adjusting parameters to reach a numeric optimum, participants treated the system as a tool for shaping design direction.

Each mechanism family exposed a small, interpretable set of controls (e.g., lift/dwell for cams; bar lengths and anchors for four-bars), which participants swept interactively to assess traces and finalize their designs. While this process sometimes improved geometric accuracy, it more often revealed motions that were both comprehensible and expressive. Notably, these dimensions were interdependent: artists' sense of what appeared "elegant" often hinged on whether the motion was easily comprehensible. For example, TH and MH preferred cams because they were both straightforward to interpret and elegant as mechanisms, particularly in how they employed gravity. Further, across all participants, familiarity with certain mechanisms and prior experience shaped their assessment of how fabricable a design would be. For example, LL privileges four-bar 'readability' over a closer gear fit (Figure 9E), TH favors cams for a clear apex accent consistent with his practice (Figure 10E), and MH composes multi-track cams (5-inch arms; 2-inch legs) to differentiate accents while keeping construction straightforward (Figure 11E)

The artists' final designs illustrate how these considerations converged. LL adopted a four-bar linkage for an arm gesture (Figure 9B-E), despite earlier trials with spur gears that offered closer motion accuracy. She favored the four-bar because its coupler swing produced a more comprehensible "raise–pause–settle" sequence once pacing was adjusted, creating a specific "shy" rhythm that aligned better with her artistic intent than the mechanically precise gear motion. This example highlights how makers often prioritize expressive intent over mechanical optimization with MotionSmith supporting this creative agency throughout the design process. TH chose a cam for an arms-up movement (Figure 10B-E), citing both familiarity and the cleaner visual accent it produced at the top, even though a refined four-bar more closely reproduced his digital sketch. He described preferring "elegant" designs—simple, clear, and aesthetically pleasing. MH implemented four cam motions (arms and legs) differentiated by cam sizes (5-inch for arms, 2-inch for legs) to produce distinct motion accents (Figure 11B-E), compiling them into a compound cam (multi-track cam) driven by a single driver. Favoring hand-aligned timing over a global timing synchronization, he valued cams for their simplicity and versatility. By layering two cams on a single driver, he balanced comprehensibility, expressiveness, and fabricability.

5.2.3 Blueprints supporting diverse fabrication strategies. The system's blueprint generation function, which allows users to download blueprints aligned with their selected fabrication methods, was validated through participants' actual use of the function during their fabrication process. Artists fabricated their final designs from the blueprint files, which contained both character and mechanism parts, but adapted them differently based on their fabrication methods. For instance, LL cut the character parts as silhouettes, imported the mechanism file into Autodesk Fusion 360 (her preferred CAD software) to extract line drawings of the four-bar linkages, 3D-printed and assembled the mechanical components, and then mounted the character layers on top of the mechanisms (Figure 9B,E). TH and MH affixed the mechanism printouts onto wood for cutting in their woodshops then attached the printed character parts directly to the resulting mechanisms (Figure 10B,E, Figure 11B,E). Participants described the blueprint documents as "just enough to get it right" (TH) and, in some cases, as exceeding expectations; MH, for example, had anticipated receiving only "reference sketches." LL noted that the blueprint including the assembly instructions reduced the trial-and-error she would typically expect, saving her considerable time. While participants saw the packets as already sufficient for pragmatic handoff, they proposed incremental improvements, such as exploded views for selected assemblies and a toggle for clearance annotations (LL). Notably, MH's compound cam design required manual completion, as the system does not yet support multiple-motion synthesis.

6 Limitation and Future Work

This paper aims to advance computational design systems by exploring the unique design space of the automata making with expert makers. While working closely with three experts allowed us to investigate high-level creative possibilities and set ambitious expectations, it also limited our understanding of beginners' challenges and the types of support novices may require. The small number of participants (N=3) further limits the generalizability of our findings. In this section, we critically reflect on the limitations of our approach and outline opportunities for future work.

6.1 Supporting Learning through Exploratory Design

MH noted that designing functional mechanical movements requires a foundational understanding and actively made suggestions on ways the system could support learning rather than merely offering solutions. Currently, the system presents three auto-generated mechanism suggestions in a pop-up dialog, which are applied to the canvas upon selection. Expanding this feature could create a richer learning experience by explaining underlying physics, mechanism strengths and weaknesses, and trade-offs. This motivates the addition of mechanism libraries and community resources, enabling users to explore examples and best practices.

Maker education literature highlights the value of tinkering and messy, bottom-up exploration as pathways to creative, hands-on learning [**?**]. MotionSmith supports this kind of iterative exploration, yet the current workflow assumes a top-down approach: makers begin with a figurative image they wish to animate. However, as observed in PD Phase 1, creative inspirations often emerge over time rather than being specified in advance (e.g., LL often drew from abstract geometric motion, TH from natural phenomena like a falling leaf). With the potential addition of mechanism libraries, we envision an alternative workflow in which users first explore and experiment with mechanisms, learning through hands-on design, and then specify an image to represent their concept.

6.2 Enhancing Interpretability via Diagnostics and Mechanism Coverage

While our expert collaborators did not actively rely on the system's real-time diagnostics (Stage 2), we believe these features hold significant potential for supporting less experienced makers, particularly if made more interpretable. At present, the solver silently rejects infeasible parameters with explaining the reason. To better support learning-by-doing, future iterations should surface these constraints through explainable diagnostics. Visual cues could not only flag a configuration as invalid but explain why it failed (e.g., "input crank too short for rotation"). By shifting from silent rejection to active explanation, the system can transform errors into pedagogical moments, helping users build the intuition that experts like MH possess implicitly. Finally, while our current implementation supports only three mechanism families (four-bar linkages, cams, and gears), we plan to expand this library to include additional kinematics such as cranks and non-planar mechanisms. A richer mechanism library would enable the system to design, diagnose, and suggest broader creative possibilities.

6.3 Expanding Blueprints to Scaffold Fabrication

Currently, the system-generated blueprint includes only mechanical components and image elements, omitting important considerations such as casing or enclosure. While our collaborating artists did not explicitly request support in this area, their installation methods revealed that such decisions are critical and could benefit from system guidance. For instance, artists made deliberate choices about whether to expose the underlying mechanisms as part of the intended aesthetics (e.g., TH, Figure 10) or conceal them to foreground the narrative (e.g., LL, Figure 9). The choice of power source further shaped these decisions: MH synthesized motions into a compound cam driven by hand, whereas LL planned to connect servos to individual parts and sequencing them with a microcontroller. These examples suggest that operation methods directly influence casing and enclosure needs, pointing to opportunities for system support. Furthermore, the blueprint feature could be expanded to incorporate fabrication methods (e.g., 3D printing, laser cutting) and material preferences (e.g., wood, cardboard). While expert collaborators readily adapted without this support, such flexibility will likely be indispensable for novice makers with limited fabrication experience.

Beyond casing, the system manages physical complexity using a *planar layering* strategy. Instead of simulating complex 3D collisions, MotionSmith enforces a valid layer order (Z-layering). This ensures designs are immediately fabricable while keeping the workflow flexible for tinkering. While future iterations could support more complex kinematics and fabrication support, these affordances will need to be carefully designed to avoid over-constraining creative expression.

6.4 Engaging Makers Beyond Three Experts

In line with the goal of achieving a "low floor and high ceiling," [**?**] we recognize the importance of involving both experts and beginners in the system design process. However, practical constraints led us to prioritize deep collaboration with three expert automata artists. Through PD sessions, we learned from their established practices and workflows, using MotionSmith as both an ideation tool and a way to probe fabrication bottlenecks such as material choices and mechanism visibility. To partially offset the absence of a novice study, we recruited experts who also teach novices and share instructional materials with wider maker communities, enabling them to speak directly about how their practices translate to less experienced makers. Nevertheless, this expert cohort cannot capture the full diversity of automata-making workflows, and future work should extend the system with novice makers and additional experts across different traditions to test how well our motion-intent framing and fabrication scaffolds generalize.

7 Conclusion

This paper presents a sketch-based computational design system *MotionSmith* developed through participatory design (PD) with domain experts to support the design and fabrication of creative automata. PD sessions with three professional automata artists surfaced critical needs for sustaining agency in creative workflows: iterative, tinkering-like exploration, fabricable mechanism candidates, and design features that prioritize creative intent. Our final phase demonstrated *MotionSmith*'s feasibility in artists' projects and highlighted opportunities to further scaffold learning and fabrication for broader maker communities. Beyond mechanical movement design, this work contributes to HCI and computational design system literature by demonstrating how engaging expert practitioners through participatory design can lead to more usable, useful, and contextually grounded design tools.

Acknowledgments

We thank Lu Lyu, Tom Haney and Marc Horovitz for their engagement and valuable insights throughout the process as domain experts. This paper was supported by the National Science Foundation (NSF) under Grant No. 2339076. Additional support was provided by seed funding from the Institute for People and Technology (IPaT) at the Georgia Institute of Technology, co-sponsored by the Georgia Tech Research Institute (GTRI) and the Brook Byers Institute for Sustainable Systems, and by the Dolby Gift Fund, provided by Dolby Laboratories, Inc.