

Room-Scale Robotic Surfaces

Jesse T. Gonzalez

CMU-HCII-26-100

February 2026

Human-Computer Interaction Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, Pennsylvania, 15212

Committee:

Scott E. Hudson (Co-Chair)
Carnegie Mellon University

Alexandra Ion (Co-Chair)
Carnegie Mellon University

Nikolas Martelaro
Carnegie Mellon University

Nadya Peek
University of Washington

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy

Keywords: shape-changing interfaces, human-robot interaction, ubiquitous computing, digital fabrication, interactive architecture, digital twins, human-building interaction, origami-inspired structures, robotic metamaterials, physical and embodied AI

Abstract

What might our surroundings look like, in a world of ubiquitous robots? This dissertation reimagines the distinction between robots and spaces, introducing an architectural integration paradigm where robotic systems become one with our built environment. Rather than existing as distinct entities, these shape-changing surfaces can be integrated into walls and floors, enabling environmental adaptations in response to contextual requirements. While prior work has demonstrated the promise of such dynamic systems, pre-existing strategies for fabrication and actuation often fall short when it comes to constructing shape-changing interfaces that span larger areas.

This research addresses three fundamental challenges in realizing room-scale robotic surfaces: the development of scalable materials and fabrication methodologies, the implementation of natural human-surface interaction paradigms, and the engineering of large-scale shape-transformation mechanisms. Specifically, we introduce: (1) a technique for creating “flat-fabricated” electro-mechanical components via a PCB fabrication pipeline; (2) a geometric state estimation system that enables surfaces to sense and reconstruct their own shape during human interaction; (3) a constraint-driven actuation approach that achieves robust transformations with minimal power requirements; and (4) a pair of design and elicitation studies that uncover natural interaction techniques for human-robot collaboration in adaptive environments. We conclude with a discussion of how these contributions can lead to fluent collaboration between people and their surroundings.

Contents

Abstract.....	3
I. Introduction	9
1. Room-Scale Robotic Affordances.....	10
2. Shape-Changing Surfaces at Large Scales	13
Room-Scale Materials and Fabrication	13
Room-Scale Tangible Inputs.....	15
Room-Scale Physical Outputs	16
Room-Scale Interactions.....	18
3. Structure of this Document.....	18
II. Related Work	19
1. Interactive Environments.....	19
From Calm Computing to Tangible Bits.....	19
Chairs, Drawers, and Garbage Cans: Robots in Disguise	20
Sensing and Shaping of Room-Scale Structures	21
2. Shape-Changing Surfaces.....	22
Modular Shape Displays.....	22
Configurable Constraints.....	23
3. Modular and Reconfigurable Robots	24
Chains, Snakes, and Tentacles.....	24
Blocks and Swarms	26
Flat-Fabricated, Foldable Robots	27
Machines that Morph to Move.....	27
4. Room-Scale Fabrication	28
Deployable Structures.....	28

Room-Scale Fabrication in HCI	29
5. Towards Room-Scale Robotic Surfaces.....	30
III. From Machinery to Materials: Layered Fabrication of Programmable Surfaces	32
1. Introduction	32
2. Layer-Driven Design.....	34
3. Background	35
Soft and Foldable Robots	35
Electrostatic Actuators.....	36
4. Patterned Valves for Programmable Surfaces	38
Working Principle	41
Insulating Dielectric and Driving Voltage.....	43
Daisy-Chained Electronic Communication	44
Pneumatic Push-Pull Control Scheme	45
5. Fabrication Process	47
Cutting: Diaphragm and Spacer.....	47
Casting: Upper Gasket and Soft Transducers	48
Printing: Global Reservoir and Upper Manifold	49
6. Actuation Characteristics	50
Opening Voltage	52
Holding Voltage	54
Programmable Stoma-Board Surfaces	54
Tactile Patterns from a Bistable Membrane	55
Bellows-Style Robotic Actuators.....	56
Spoke-Propelled Rimless Wheel.....	57
7. Discussion, Limitations, and Future Work	59
Long-Term Use.....	59
Towards Larger Surfaces.....	59
Towards Denser Surfaces.....	60
Bidirectional Valves.....	60
Conclusion.....	61

IV. Sculptable Mesh Structures for Large-Scale Form-Finding	63
1. Introduction	63
2. Background	66
3. Room-Scale Form-Finding with an Interactive Mesh.....	69
From Physical, to Digital, to Physical Again.....	70
End-to-End Workflow	72
4. Overview and Mechanical Design	73
Curvature and Connection Scheme.....	75
Materials and Fabrication.....	76
Topology and Anchors	77
5. Electronics and Firmware	77
Length Sensing	79
Curvature Detection	80
Power and Communication.....	81
Mesh Representation and Traversal	81
Physical-to-Digital Link	82
6. Digital Reconstruction	84
Complete Reconstruction Pipeline.....	84
Geometric Representation.....	85
Energy Terms	86
Handling Bistability	87
Local Coplanarity.....	87
Axial Bending.....	88
Post-Processing for Final Form.....	88
7. Discussion	89
Limitations.....	89
Comparison with Existing Digitization Techniques	90
Future Directions in Hardware and Sensing	90
Topologies and Configurations.....	91
Bidirectional Information Flow.....	92
Applications Beyond Architecture	93

8. Conclusion	94
V. Constraint-Driven Robotic Surfaces, at Human Scale	95
1. Introduction	95
Constraint-Driven Robotic Surfaces	96
Contributions.....	97
2. Living with Robotic Surfaces	97
3. Background	101
Serpentine and Continuum Robots.....	102
4. Building Robotic Surfaces at Human-Scale	102
System Overview	103
Cell Fabrication and Assembly.....	104
Endpoint Actuation	106
Power, Modularity, and Mounting	107
Software Pipeline.....	108
5. Heuristics for Control	109
Deploying Shapes	110
Retracting Shapes	111
Axial Movement.....	112
6. Mechanical Characterization	113
7. Discussion, Future Work, and Conclusion	115
VI. Fluent Interaction with Cyber-Physical Architecture	118
1. Introduction	118
2. Related Work	121
Foundations	121
3. Design Workshops	125
Resource Optimization.....	129
Cognitive Offloading.....	131
Preserving Autonomy	131
Privacy and Personal Boundaries	132
Trust, Dependability, and Safety.....	133
Physical Extension and Support	134

Personalization.....	136
Aesthetic Harmony	137
Engagement and Dynamism.....	137
Surprise and Emergence	138
Comfort and Convenience.....	139
Preliminary Implications.....	140
4. Elicitation Study	141
Participants and Procedures	142
Analysis Method	143
Overview of Findings	144
Voice.....	150
Gesture and Touch	155
Modal Entropy	160
5. So, What Should You Build?	163
“Raise” a Robotic Environment.....	164
Directly Malleable Surfaces	164
Crafting Legible Intent.....	165
Co-Design for Accessibility	165
Large-Scale, Load-Bearing Structures	166
6. Conclusion: Implicit Context Necessitates a Model of User Intent.....	167
Acknowledgements	169
References.....	170

I. Introduction

Will we live in a world of ubiquitous robots?

We should start by clarifying the term. Drawing from popular culture, the phrase “ubiquitous robotics” can evoke an image that is quite busy — fleets of autonomous machines, walking or rolling amongst humans, ready to be recruited for one task or another. This picture, while a bit crowded, is not necessarily negative; one can imagine multiple tasks for which such free-roaming agents would be useful.

But contrast this with the concept of ubiquitous computing¹ — Mark Weiser's vision of “calm technology”, introduced at the dawn of the mainstream Internet. Weiser proposed that, rather than competing for our attention, computation should operate in the background, weaving itself seamlessly into the environment². Calm technology becomes invisible — not in the literal sense, but by becoming so well-integrated into daily life that it requires minimal conscious interaction.

Viewing “ubiquitous robotics” through this lens, we might reconsider our initial picture. What does it mean for robots to be embedded in the environment so seamlessly that they practically vanish from notice?

Instead of imagining a robot as an entity acting *within* a room, we might imagine a situation in which the robot *is itself* the room. Under this paradigm, robotic systems are embedded into walls, floors, and infrastructure, creating a responsive space, adaptive to human needs.

Consider the example in Figure 1, taken from Chapter 5 of this document. A person walks into their small apartment, wearing a jacket and carrying a heavy bag of groceries. Their living room wall, seeing that the person is struggling, transforms into a shelf, allowing the person to place down their bag. A different part of the wall transforms into a coat hook, allowing the person to hang up their jacket. Once the person grabs

the grocery bag again (walking into the kitchen), the shelf disappears back into the wall, its purpose served.



Figure 1. An interaction with a room-scale robotic surface (specifically, the robotic wall in Chapter 5). In the default state (a), the surface is flat, embedded in the wall. When a human approaches with a grocery bag, the surface morphs into a shelf (b) and a coat hook (c). When the human takes the bag and leaves, the shelf folds back into the wall (d).

Robots are machines that sense, plan, and act. Our room-scale robotic surfaces should do the same: process information (similar to Abowd’s “computational materials”³), and also physically act on their environment. We can consider these surfaces to be a subset of “robotic materials” as defined by Correll⁴ — “materials that couple sensing, actuation, computation, and communication”.

In this document, we focus specifically on surfaces that change shape, and how that shape change can support human activities in the built environment. Key to this problem is tackling the issue of *scale*. Scalable robotic surfaces require particular material choices and fabrication strategies — which in turn inform the design of physical inputs, physical outputs, and human-room interactions.

1. Room-Scale Robotic Affordances

When materials and geometries are thoughtfully combined, they create intuitive cues that can guide user behavior without the need for explicit

instructions⁵. These affordances are fundamental building blocks of interaction design.

Just as a door handle's form suggests how it should be grasped and operated⁶, material and geometric affordances play a crucial role in communicating a robot's capabilities and intentions^{7,8}. A social robot with soft, rounded surfaces suggests approachability and safe interaction; while one with exposed wiring and hard metal components implies a need for distance. A robotic gripper that resembles a human hand might lead users to expect human-like dexterity; while a simple two-fingered device more accurately communicates a limited grasping ability. This alignment between a robot's apparent capabilities (suggested by its materials and geometry) and its actual abilities is crucial for successful human-robot interaction. Mismatches can lead to confusion, frustration, or unsafe situations^{9–11}.

In the context of shape-changing interfaces, these principles take on new dimensions, as geometries transform (and can *be* transformed) dynamically. A fabric or silicone surface invites stretching and sculpting, suggesting direct manipulation and reconfiguration¹². In contrast, rigid but jointed structures suggest more prescribed transformations — e.g. a surface that folds along specific lines, or rotates around fixed points.

An important consideration when designing large shape-changing interfaces is the *scalability* of material and geometric choices. Not all fabrication approaches that work at a small scale will be practical or effective when implemented at room size. For instance, a surface made of high-resolution motorized linear actuators can create compelling interactions on a tabletop; but the assembly, cost, and maintenance requirements make it impractical to embed in walls or floors¹³.

Scaling up shape-changing interfaces requires careful consideration of both materials and geometry — it's not simply a matter of enlarging existing designs. Since these structural choices can define the space of potential interactions, we must keep these material constraints in mind from the onset of the design process.

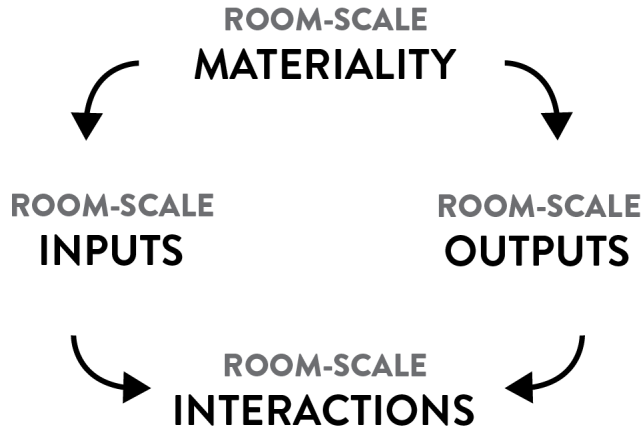


Figure 2. Our approach to creating scalable robotic surfaces. Materials and fabrication processes are foregrounded, and inform the design of physical inputs and outputs. These inputs and outputs are then used for human interaction with robotic surfaces.

Our approach is outlined in Figure 2, above. In our work, we foreground materials and fabrication processes, which in turn inform the design of physical inputs, outputs, and interactions. We explore:

- *Materiality, via layup-based fabrication.* Instead of assembling individual modules and actuators, we develop “flat-fabricated” structures with embedded electrical and mechanical functionality. These initially-flat structures consist of multiple layers, created using planar manufacturing techniques (e.g. laser cutting, photo-etching). By stacking and laminating these layers together, we can create actuatable structures while maintaining scalable manufacturing processes.
- *Inputs, via geometric state estimation.* Since we expect these surfaces to be deformed and reconfigured by humans, we implement a software routine to sense and reconstruct surface geometry. Given a coarse, mesh-like surface, for example, we use embedded sensors to track the length changes of individual edges, allowing us to computationally reconstruct the overall shape.
- *Outputs, via constraint-driven actuation.* Instead of using high-power actuators to drive shape change, we use low-power toggles to set and remove geometric constraints. When combined with a global force (like gravity or mechanical pressure), these strategically placed constraints allow us to achieve complex shape transformations with minimal power consumption.

At first, this approach may appear overly engineering-oriented — focused on the underlying technology rather than the user experience. However, we argue that it is in fact compatible with a human-centered design philosophy. By emphasizing the underlying materials and fabrication processes, we ensure that the fundamental capabilities and constraints of our robotic surfaces are well-understood from the start. This allows us to then design meaningful, contextually-appropriate interactions that leverage the unique affordances of scalable shape-changing structures.

2. Shape-Changing Surfaces at Large Scales

In this section, we summarize how our work addresses some of the key technical challenges and design considerations involved in developing large-scale, reconfigurable interfaces.

Room-Scale Materials and Fabrication

A robotic surface consists of multiple actuatable components. If these components and actuators are built and assembled individually, then the surface becomes difficult to manufacture at large scales.

In our work, we solve this by leveraging “flat fabrication” methods — techniques such as patterning, layup, cutting, and folding — to create complex robotic surfaces using mature printed circuit board (PCB) manufacturing processes. Our aim is to enable the production of integrated mechanical, electrical, and “actuatable” components within a single unified fabrication pipeline.

For an example of this technique, see the surface shown in Figure 3. This is an array of electrostatic valves, neatly contained within a single flat-fabricated structure. These valves regulate air through pneumatic channels within the surface, allowing us to control individual areas of a cast silicone membrane (or any other air-activated transducer).

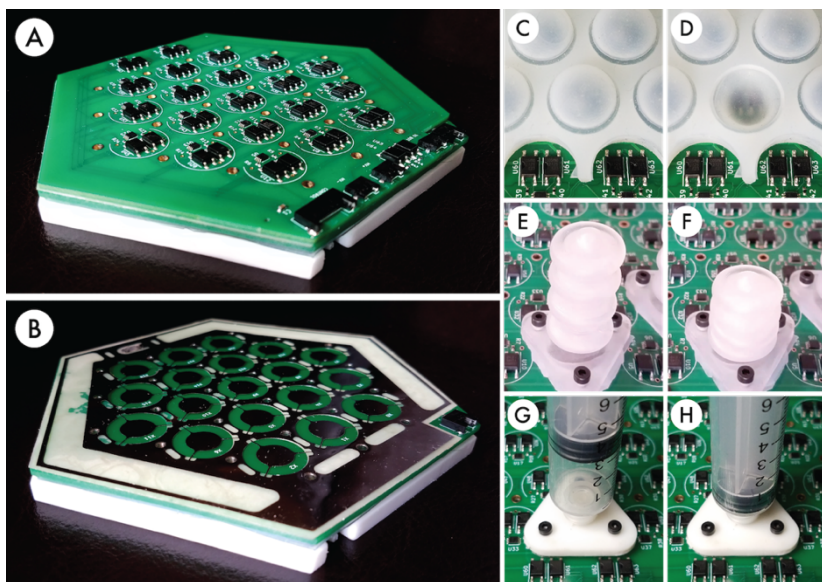


Figure 3. An array of electrostatic valves, composed of several layers (a). One of these layers is a flexible, conductive diaphragm (b), which regulates air flow through the structure. When a transducer layer is placed on top of the surface, the valves can control the flow of air in and out of individual sections, deflating bubbles (c,d), contracting bellows (e,f), or controlling syringes (g,h).

The valve array is made from a stack of laminated layers. At the center is a thin, conductive diaphragm — laser-cut with sections that subtly flex in and out of plane. It is sandwiched between two grids of patterned electrodes; and a flat pneumatic manifold sits underneath this “valve sandwich”. Together, these layers form an electro-pneumatic surface that can be used to actuate a variety of components.

Critically, the assembly process (stacking layers) is independent from the number of actuators. Additionally, the 2D layer fabrication techniques themselves (cutting, etching) are generally fast, compared to 3D additive or subtractive methods.

In addition to patterning electrodes for actuation, we can pattern electrodes for sensing (Figure 4). The mesh structure in the following section, for instance, is composed of thin strips of FR4, which serve as the structural components of a room-scale, bending-active mesh structure. Each patterned electrode is connected to distinct node of a resistive ladder. As edges of the mesh are lengthened or shortened by a user, this deformation is sensed by a scanning head that contacts the electrodes.

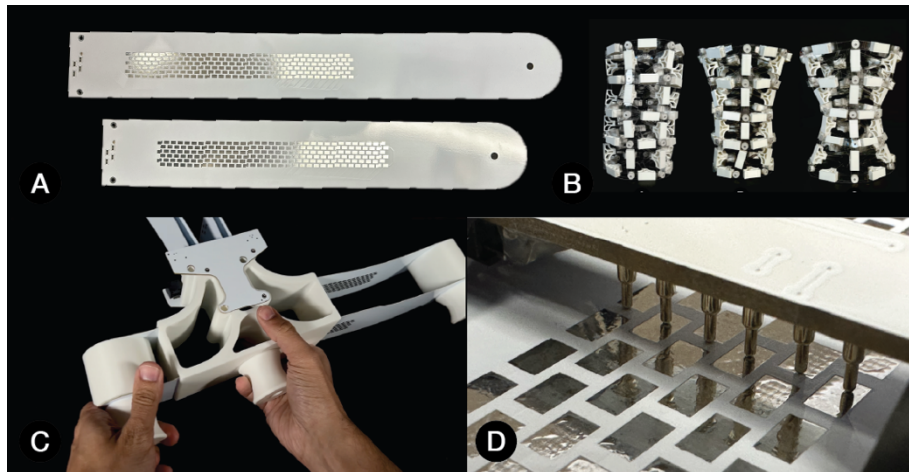


Figure 4. By linking together adjustable-length modules (c), we can create user-reconfigurable mesh structures, similar to those in (b). The edges of these mesh structures are made from strips of FR4, patterned with a resistive code (a). We can read this code (d) to detect the length of each module.

We can also use these flat-fabrication techniques to define the geometry of structures, such as the robotic wall presented earlier. In Figure 5, we use a layup technique inspired by rigid-flex PCBs to create a cube net with integrated hinges, which can then be folded into a configurable column. (The outer layers These origami-inspired techniques have become popular amongst some robotics researchers in recent years^{14,15}, as it allows for rapid fabrication at multiple scales¹⁶.

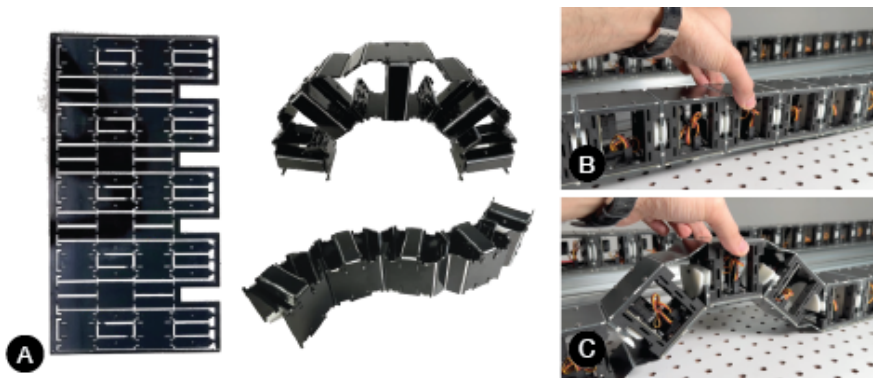


Figure 5. A patterned, rigid-flex sheet can be folded and assembled into a reconfigurable column structure (a). This structure is similar to the columns of our robotic wall (b,c), described in Chapter 5.

Room-Scale Tangible Inputs

These surfaces exist in the physical world, and we should expect for them to be physically manipulated by humans. After all, if a user knows that a surface is *capable* of being reconfigured, and it's not *currently* configured to their liking, then the user may want to physically sculpt or adjust the surface to reach their desired shape¹⁷.

This means that these surfaces may need to exhibit some compliance, and, upon reconfiguration, be able to detect their new geometric state.

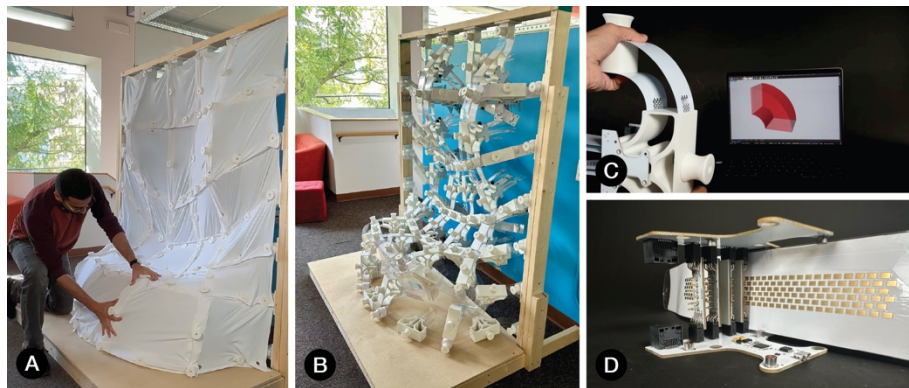


Figure 6. A user sculpts a room-scale surface into a desirable shape, as a means of low-fidelity prototyping (a). Underneath the fabric is a mesh structure (b) composed of flexible bands with embedded length sensors (d). By sensing the length of these bands, (c) we can reconstruct the surface.

Figure 6 depicts a user-adjustable, room-scale, “shape-aware” mesh structure. A user can physically manipulate the mesh by lengthening and shortening the edges, altering the overall curvature and sculpting coarse forms. The mesh edges are made of thin FR4, a fiberglass laminate commonly used as a PCB substrate. They are patterned with resistive length sensors, and transmit their configuration to a central computer. We can then reconstruct this geometry in software through an energy-minimization routine.

Specifically, Chapter 4 explores this concept within the context of an architectural design task. It can be hard to design a physical structure entirely within the confines of a computer monitor — practitioners will often go through a sequence of low-fidelity physical prototypes (using paper, clay, or foam¹⁸) before arriving at a form that satisfies both functional and aesthetic concerns. Here, the robotic surface is used as a tool that connects the physical prototyping stage to an architect’s pre-existing computational design pipeline.

Room-Scale Physical Outputs

A robotic surface contains a collection of actuators. In a typical implementation, each actuator directly controls a section of the surface — for instance, a linear actuator might push a rectangular patch out-of-plane, or a motorized rotary joint might adjust the angle of a fold in the surface.

However, this direct actuation approach faces significant practical limitations when scaled to larger surfaces. As the area increases, each actuator must support more material weight, increasing the power requirements and necessary actuator strength. The growing structural loads also introduce mechanical challenges — actuators must be more robustly mounted and may require additional support structures, adding further weight and complexity.

As a result, large-scale implementations of these directly-actuated surfaces are not typically seen outside of high-budget artistic installations¹⁹, where practical considerations such as energy efficiency are secondary to achieving dramatic visual effects. In research settings, experimenters exploring room-scale interfaces are often relegated to using digital projections²⁰ or virtual reality²¹ to simulate shape change.

In Chapter 5, we describe an alternate approach to physical output. Instead of using actuators to directly drive the curvature of the surface (i.e. motorized joints), we use actuators as lightweight *toggles*. These toggles constrain sections of the surface, so that when a global force is applied, the structure deforms in a pre-programmed, predictable manner. We call this approach *constraint-driven actuation*.



Figure 7. A shape-changing robotic wall (left) composed of multiple reconfigurable columns. Each joint can be clamped in either a valley orientation, rigid orientation, or mountain orientation (right). Then, when edges are squeezed from the surface, shapes pop out of the plane.

In Figure 7, we show a robotic surface composed of multiple columns. Each column is a series of cubes, connected by hinges, which can be constrained in either a rigid configuration, a mountain configuration, or a valley configuration. Once the columns are constrained, two larger

actuators squeeze them from the endpoints, causing our programmed structure to pop out of plane.

Room-Scale Interactions

Chapter 3–5 cover room-scale materiality, input, and outputs. Chapter 6 focuses on room-scale interactions. We present findings from two studies: (1) a series of speculative design workshops that uncovered aspirational visions for these spaces, and (2) a task-based Wizard-of-Oz elicitation study that grounded these visions in the challenges of practical interaction. Our workshop findings reveal a complex landscape of user desires, exposing critical tensions between proactive automation and the preservation of user autonomy, and between personalization and public ownership. Our elicitation study reveals a set of core interaction challenges related to multimodal collaboration; and, most critically: suggests the need for a modality-agnostic model of evolving user intent.

3. Structure of this Document

In Chapter 2, we review the state-of-the-art in four areas relevant to this dissertation: interactive built environments, shape-changing interfaces, modular robotics, and room-scale fabrication techniques.

In Chapter 3, we describe a *layer-based fabrication* technique, which allows for the scalable production of programmable surfaces.

In Chapter 4, we show methods for input sensing and *geometric state estimation*, that allow us to reconstruct the curvature of a room-scale surface after a user deforms it.

In Chapter 5, we introduce *constraint-driven actuation* as means of triggering large-scale shape-change with low-power toggles.

In Chapter 6, we explore the design of *human-robot interaction* for architectural-scale, shape-changing environments. This is done through a design workshop and elicitation study. We then conclude the dissertation, presenting a roadmap for future work in *user intent recognition for personal robotic environments*.

II. Related Work

In this section we survey related work in the fields of interactive environments, shape-changing surfaces, modular robotics, and room-scale fabrication.

1. Interactive Environments

From Calm Computing to Tangible Bits

A designer succeeds when their creations provide clear affordances – when doors make it obvious how to open them, or when switches clearly indicate what they control. This principle (made famous by Don Norman⁵), was of particular relevance at the dawn of the mainstream Internet, as researchers began imagining how computation might move beyond the desktop monitor and into the physical world.

Mark Weiser pioneered the concept of ubiquitous computing^{1,2}, arguing that “the most profound technologies are those that disappear”. He advocated for computers that were as unremarkable as written text – present everywhere but noticed nowhere. Researchers such as Ishii embraced this framework, bridging digital and physical space through tangible user interfaces (TUIs)²². While Weiser emphasized invisible integration, Ishii focused on making digital information directly manipulable through physical objects, tokens, and ambient media²³.

The ambientROOM²⁴ project, for instance, demonstrated an environment where digital information was conveyed through subtle changes in sound, light, and movement. This trend toward embedding computation in everyday objects²⁵ (the original “Internet-of-Things”²⁶) represented a shift in how researchers navigated the relationship between physical and digital worlds^{25,26}.

Theories of embodied interaction²⁷ help explain why Weiser's invisible computers and Ishii's graspable interfaces feel more intuitive than traditional computing. Consciously or not, people create meaning through direct interactions with their environment — and interfaces that take advantage of this can be quite powerful.

Chairs, Drawers, and Garbage Cans: Robots in Disguise

While not traditionally thought of as a “user interface”, there has been recent interest in using the built environment (i.e. architecture) as an interactive medium²⁸. Shape-shifting walls, floors, and furniture can greatly impact how we experience environments²⁹, collaborate with others^{30,31}, adjust our posture³², and move through physical space³³. Walls, in particular, are ubiquitous in our daily life, and have the potential to support rich, engaging interactions — transforming into multi-use structures²⁰.

Within HCI, research in actuated architecture has mostly centered around controlling the in-room position of free-standing walls. When attached to motorized platforms³⁴, these roving structures can support automatic room partitioning³⁵, or form “just-in-time” surfaces for virtual reality experiences^{36,37}. In these instances, however, shape-change is limited to adjusting the overall height of the wall.

Beyond walls, researchers have explored how everyday furniture can be augmented with robotic capabilities to create more expressive and interactive environments. These investigations sit at the intersection of human-robot interaction and ambient computing, examining how people respond when familiar objects exhibit autonomous behaviors.

Hoffman's robotic desk lamp³⁸ was an eye-catching exploration in this space. Through careful choreography of head tilts, pivots, and light intensity, the lamp could express attention, curiosity, and even personality — leading to more engaging human-robot interactions⁸.

Similar principles have been applied to other furniture pieces, such as drawers^{39,40}, cabinets⁴¹, and chairs^{32,42–44}. These pieces can influence social

dynamics and encourage activity participation. For instance, when chairs autonomously repositioned themselves to form conversation circles or learning configurations, participants reported feeling subtle social pressure to join group activities.

Even trash cans⁴⁵ have been reimaged as interactive agents, actively seeking litter while engaging passersby through playful movements⁴⁶. Initial studies showed that people were more likely to properly dispose of waste when bins exhibited agency and personality, suggesting that anthropomorphized everyday objects could promote positive behavioral change. (It should be noted, however, that some find these motions annoying or off-putting.)

Sensing and Shaping of Room-Scale Structures

While room-scale actuation techniques tend to be more coarse (involving either simple shape transformations or locomoting objects), room-scale sensing can be quite high-resolution. There are various approaches to embedding sensing and interactivity into fundamental building components like walls and floors.

Several promising approaches have emerged for integrating sensing capabilities into building materials and structures. Some researchers have successfully embedded fiber optic sensors directly into cast materials like concrete and polymer resins, enabling vibration sensing and activity recognition at room scale without requiring internal power⁴⁷. Others have transformed ordinary walls into interactive surfaces capable of detecting touch, gestures, and even body poses, while simultaneously monitoring electromagnetic signatures from nearby appliances⁴⁸. Floor-based sensing systems have also proven effective, leveraging the inevitable vibrations and loads generated by human movement to track user locations and recognize foot gestures⁴⁹. Some provide tactile feedback as well⁵⁰, and can be made to support accessibility and wayfinding⁵¹.

Activity monitoring in large buildings has also become increasingly sophisticated, with distributed sensor networks capable of tracking occupancy patterns, energy usage, and environmental conditions across multiple floors and zones⁵².

The field has also seen innovations in deployable and reconfigurable interactive structures. Swaminathan et al.⁵³ introduced pneumatic, truss-like structures with embedded sensing and actuation capabilities, allowing for room-scale installations that can respond to user input through various interaction modalities, such as touch. By dynamically controlling air pressure in individual truss members, these structures can deform as well.

2. Shape-Changing Surfaces

Modular Shape Displays

Perhaps the most recognizable style of shape-shifting interface is the “pin display” and its variants⁵⁴. These devices consist of individually controlled “physical pixels”, arranged in a two-dimensional array, that can each move up and down to alter the overall topography of a surface. The pins are typically motorized, driven by linear actuators⁵⁵, though pneumatic control schemes have been studied as well⁵⁶. The handheld⁵⁷ and tabletop⁵⁸ demonstrations of these interfaces are compelling, and there is interest in expanding such shape-shifting surfaces to the architectural scale^{20,59}.

However, the construction of pin displays makes this level of scalability a challenge. In these interfaces, there is a unique actuator for each physical bit, and the shape-change capabilities of the surface are dictated by the size and strength of those actuators. Practically, this often limits us to smaller areas, and lightweight object manipulation.

At larger scales, researchers have tried to solve this problem by using latching mechanisms. Je et al.⁶⁰ demonstrate a shape-shifting floor — a single large actuator sweeps across the surface, adjusting the height of physical bits one row at a time. After each configuration step, a brake (one for each bit) locks the piece in place. The resulting surface is robust (a user can stand on it), but the transformation process takes a long time. Researchers have also explored large-scale inflatable surfaces^{56,59}. These surfaces can interact with bigger objects (included humans), but are still difficult to scale because the modules must be individually assembled.

Work by Zárte et al.⁶¹ hints at a solution to the problem of scalability. Their tactile display, while still consisting of individual (magnetic) pins, is driven by coils that are patterned on a printed circuit board. Since these multiple coils are all formed as part of a single fabrication step (photo-etching, during the printed circuit board manufacturing process), this act of patterning reduces the ultimate assembly time for the device. These copper traces can similarly be used for within-plane steering of magnetic markers^{62,63}, and stand in contrast to prior approaches that employed beds of individually packaged solenoids^{64,65}. Notably, there is a trade-off here — while these layered coils can be replicated across large surfaces with relative ease, they are generally less powerful than their discrete counterparts (owing to the reduced number of turns).

Configurable Constraints

Some researchers achieve shape change through reconfigurable modules, while others rely on material characteristics (i.e. deformation due to heat^{66–68} or moisture⁶⁹). A third, hybrid approach relies on using lower-power actuators to make changes to the surface geometry, and then allowing this new geometry to dictate the shape change.

One way researchers have achieved this is by using patterned elements as a means of “tapping in” to a larger energy source. Under this paradigm, power and control are fully decoupled, and the actuators serve as gateways to a global reservoir. Qiu et al.⁷⁰ employ this approach, spray-coating an array of resistive traces onto an electroactive polymer film, which is suspended above a pressurized pneumatic chamber. Through targeted Joule-heating of the film (via the resistive traces), they force heat-softened sections of the membrane to deflect outward and raise a layer of pins.

Previously, Besse et al.^{71,72} demonstrated a similar design, fabricating a shape-memory polymer membrane with integrated heating elements. This membrane was mounted over an alternating pressure source — positive pressure pushes the selected areas upward, and negative pressure pulls them back down.

Constraints can also be controlled via electrostatics. Rauf et al.⁷³ introduced a kirigami-inspired auxetic skin, which when suspended over an inflatable membrane, acts as a configurable shape display. By selectively

locking individual sections of the skin (through electrostatic adhesive), the membrane is able to expand in some areas, while being prevented from expanding in others.

3. Modular and Reconfigurable Robots

While HCI explores shape-change primarily through the lens of user interfaces, there exists a vast body of research in robotics that investigates shape-transformation for achieving locomotion and manipulation. In this section, we survey some popular configurations.

Chains, Snakes, and Tentacles

A common modular architecture is the chain-like configuration, where multiple modules are connected in series. Typically, each module has at least one degree of freedom (e.g. a yaw rotation). The coordinated actuation of these interconnected sections enables complex shape transformations of the overall structure, forming the basis for various locomotion and manipulation strategies⁷⁴.

Biological inspiration, particularly from snakes, has significantly influenced the development of chain-type modular robots⁷⁵. These systems achieve locomotion through the orchestrated movement of 2-DOF universal joints connecting adjacent modules^{76,77}. Some designs incorporate additional functionality through helical actuators that encompass individual modules, providing roll-axis rotation to enhance mobility⁷⁸. This biomimetic approach has proven effective in replicating snake-like movement patterns and adapting to diverse terrains⁷⁹.

However, the serial nature of these systems presents notable challenges in power management. As each joint requires independent actuation, the energy consumption increases with the number of modules⁸⁰, potentially limiting the practical deployment of longer chains in real-world applications.

Beyond locomotion, researchers have explored the potential of these systems for dynamic shape transformation. Notable examples include

Nakagaki's LineFORM⁸¹ and ChainFORM⁸² projects, which demonstrate the versatility of serially-connected 1-DOF actuators in creating programmable shapes and interfaces.

While the chain-like architecture naturally suggests applications in manipulation (e.g. robotic arms), there are issues that arise when direct-drive motors are placed at each joint. The cumulative weight of these motors creates substantial torque requirements, particularly at proximal joints, which must support the load of all distal components.

Continuum robots are one solution to this problem. Instead of the traditional rigid-link construction, these systems use a flexible backbone actuated by internal tendons⁸³ or pneumatic structures⁸⁴. Drawing inspiration from biological systems such as elephant trunks and octopus tentacles, continuum robots achieve complex shapes through tension-driven deformation of a compliant backbone⁸⁵.

However, continuum designs typically offer more limited control over their shape compared to their rigid-link counterparts. Most implementations utilize only one or two sets of tendons, typically anchored at the end effector and occasionally at an intermediate point along the structure, resulting in relatively constrained shape possibilities.

To address this control limitation while maintaining the advantages of tendon-driven systems, researchers have investigated variable-stiffness approaches^{86,87}. By selectively modulating the rigidity of different segments along the robot's length while maintaining a single set of actuation tendons, these systems achieve enhanced shape control without the weight penalties associated with distributed actuation. This stiffness adjustment can be done pneumatically⁸⁸, mechanically, or with active materials such as shape-memory polymers⁸⁹. This is a promising technique, balancing the competing demands of precise control and system complexity.

Blocks and Swarms

Grid or lattice-like designs can be seen as a generalization of the chain-like architecture. These systems typically incorporate latching mechanisms that enable modules to dynamically connect to and disconnect from neighboring units, facilitating structural reconfiguration⁹⁰.

Several notable implementations demonstrate the versatility of this approach. The SMORES⁹¹ platform features modules connected through universal joints, with integrated pan and tilt mechanisms. Alternative designs, such as the Molecube system⁹², employ oblique swivel joints to achieve reconfiguration. These joint mechanisms serve dual purposes: they enable both articulation (i.e. the structure can move like a rigid-link arm) and they facilitate reconfiguration (i.e. one module can be moved and attached to a different area of the structure). The Roombots^{93–95} project exemplifies practical applications of such systems, with modules capable of self-assembling into functional furniture pieces like tables and chairs.

A particularly innovative approach to block-based modular robotics is demonstrated by M-Blocks^{96,97}. These cubic modules achieve movement and reconfiguration through internal flywheels and mechanical brakes, utilizing momentum transfer for locomotion and self-assembly. The sudden braking of high-speed flywheels generates sufficient angular momentum to allow blocks to flip and pivot, while permanent magnets facilitate secure connections between modules.

Module interconnection has been addressed through various coupling mechanisms. Often, this is achieved through electromagnets^{98,99}. However, while electromagnetic coupling offers precise control over module interactions, its significant power requirements present a notable limitation. Addressing this challenge, researchers have developed alternative approaches using mechanical coupling with compliant latches¹⁰⁰. These passive latching mechanisms engage automatically when blocks are brought into contact, and require specific disengagement movements for decoupling. This reduces power consumption while maintaining reliable connections.

Some systems, such as ElectroVoxel¹⁰¹, employ electromagnetic coupling not only for attachment but also for inter-module movement, enabling blocks to pivot relative to each other. This approach works well in

microgravity environments, or specialized low-friction surfaces such as air tables.

Recent advances in this field include the development of specialized assembler robots¹⁰². These robots are designed to traverse and modify lattice-like structures while assembling voxel-based components. This approach separates the tasks of locomotion and assembly, potentially offering more efficient and reliable construction of large-scale structures in outer space applications.

There is overlap here with the field of swarm robotics as well. Within HCI, researchers have explored using individual mobile modules for swarm user interfaces^{103,104}. These modules can assemble into 2D, or (in some cases¹⁰⁵, 2.5D) shapes.

Flat-Fabricated, Foldable Robots

One constraint of pin-like shape displays (both motorized and pneumatic) is that their rendering capabilities are limited to “2.5” dimensions — overhangs and voids are impossible to achieve¹⁰⁶. A different approach leverages origami-inspired techniques (bending and folding)¹⁰⁷. Like pin displays, these “self-folding” surfaces also start in a flat state. But instead of actuating two-dimensional subsections (moving them orthogonally), these techniques instead actuate one-dimensional hinges and folds, deforming the surface.

These joints can be motorized¹⁰⁸, but many researchers instead leverage material properties (i.e. shape memory polymers) in order to trigger a shape change¹⁰⁹. These composite materials also lend themselves well to miniaturization¹¹⁰.

Machines that Morph to Move

So far in this section, we have looked at modular robots that change shape by reconfiguring or rotating rigid components. However, another class of robots accomplishes locomotion and shape change through deformation of their constituent elements themselves.

Truss-inspired robots are a compelling example of this approach. Gu et al.¹¹¹, for instance, developed a robotic system composed of pneumatic linear actuators, where controlled actuation of these elements enables both deformation and locomotion of the entire structure. At larger scales, Kovacs et. al.¹¹² employed combination of active and passive truss elements to create adaptive, shape-changing structures.

A related avenue research approach explores the design of truss-like robots with configurable joints. Usevitch et al.¹¹³ introduced an inflatable tube structure with adjustable pinch points — by modifying the location of these constrictions, researchers can induce controlled deformation of the overall structure. These systems have been dubbed “balloon animal robots”¹¹⁴ due to their resemblance to inflatable sculptures, and the fact that their geometry (i.e. edge lengths) is determined by the movement of pinch points.

Tensegrity-based robots represent another significant development in this space¹¹⁵. Tensegrity structures, which maintain their shape through a balance of elements either purely in tension or in compression, have inspired both large and small-scale robotic systems. NASA's SUPERball^{116,117} exemplifies larger implementations, achieving movement and shape change through cable tension adjustments, causing their compressive elements to translate and rotate. At smaller scales, researchers have successfully substituted springs for tensile elements and induced locomotion through attached vibration motors¹¹⁸.

4. Room-Scale Fabrication

Deployable Structures

Deployable structures combine compact storage with rapid deployment capabilities. As this “deployment” can be seen as a form of shape-change or reconfiguration, these systems can give us insights on how to design large-shape shape-changing structures.

The fundamental challenge in deployable structure design lies in achieving both efficient packability and structural integrity post-deployment^{119,120}. Traditional approaches typically sacrificed one attribute for the other —

inflatable structures offered excellent compression ratios but required constant pressure maintenance, while mechanical systems provided stability but often lacked enclosed protection. Recent innovations in bistable origami-based designs have begun to bridge this gap, demonstrating that deployable structures can achieve both efficient packaging and structural stability through innovative geometric configurations¹²¹.

Emergency response applications have emerged as a particularly promising domain for deployable structures. Scissor-based designs have proven especially valuable in this context, offering not only rapid deployment but also adaptability through different phases of disaster recovery¹²². However, the transition from theoretical design to practical implementation remains challenging, particularly regarding structural analysis and real-world performance validation¹²³.

Room-Scale Fabrication in HCI

Within HCI, researchers have explored several methods for the design and construction of room-scale structures. Kovacs et al.^{112,124} focused on truss-like configurations, with developing computational tools that can generate both static and kinetic configurations. These systems incorporate passive structural members alongside active elements (i.e. linear actuators), enabling transformable architectural elements.

Researchers have also investigated inflatable and deployable structures. Similar to the “balloon-animal-robots”¹¹⁴ described earlier in this chapter, Rambold et al.¹²⁵ developed a system for on-site fabrication of large-scale pneumatic trusses. Their system controls the inflation of large tubes with precisely calculated constriction points, creating stable yet easily deployable structures. This is particularly valuable for temporary installations and rapid prototyping of spatial designs.

Large structures, reconfigurable at build-time, can be fabricated through the assembly of modular components as well¹²⁶. Lyu et al.¹²⁷ developed methods for manufacturing large structures through the assembly of planar fabricated pieces, incorporating bending-active components that achieve complex curves through strategic material deformation.

An alternative to modular assembly are systems that allow for rapid prototyping at room scale. Protopiper¹²⁸, for instance, is a handheld fabrication device that extrudes hollow tubes from plastic tape to rapidly prototype room-scale structures. This tool enables quick iteration of spatial designs through direct physical manipulation. Researchers have also explored rapid fabrication through the use of mobile 3D printers, both alone¹²⁹, and working collaboratively¹³⁰.

5. Towards Room-Scale Robotic Surfaces

In the literature surveyed above, we've seen how researchers have made significant advances in interactive environments, shape-changing interfaces, modular robotics, and room-scale fabrication. However, the integration of these approaches into cohesive architectural-scale robotic surfaces remains an open challenge.

Within the tangible computing literature, there is a progression from Weiser's vision of calm technology to increasingly sophisticated implementations of computation in physical space. While early work focused on embedding sensing and simple actuation into everyday objects, recent advances in actuated furniture and interactive walls demonstrate the potential for more complex architectural interventions. However, most of these systems either rely on coarse actuation (like moving entire walls) or are limited to sensing without significant physical reconfiguration capabilities.

Similarly, while shape-changing interfaces offer promising approaches for creating dynamic surfaces, traditional implementations requiring individual actuators become impractical at room scale. More promising are approaches using configurable constraints with global energy sources and local control.

Modular robotics research provides valuable insights into how complex shape changes can be achieved through the coordination of simple elements. Particularly relevant are the advances in flat-fabricated, foldable robots, which leverage integrated hinges as opposed to individually assembled joints. These approaches align well with the requirements for a room-scale implementation.

Overall, this literature review reveals several gaps that this document aims to address:

1. While individual actuator technologies are well-developed, there remains a need for scalable approaches to coordinating multiple actuators across large surfaces. The flat fabrication methods described in Chapter 3 address this challenge.
2. Current sensing techniques for room-scale structures focus primarily on occupancy and touch detection. The geometric state estimation methods presented in Chapter 4 extend this capability to include shape sensing for deformable surfaces.
3. Existing approaches to large-scale actuation typically rely on high-power direct drive systems. The constraint-driven actuation strategy introduced in Chapter 5 offers a more efficient alternative by leveraging global forces with locally controlled constraints.
4. Most room-scale interactive systems require explicit user input or follow pre-programmed behaviors. The proposed work in Chapter 6 aims to develop more sophisticated interaction models that can anticipate and respond to user needs.

By addressing these gaps, we aim to advance the field toward truly integrated room-scale robotic surfaces that can seamlessly adapt to human needs, while maintaining practical feasibility for real-world implementation.

III. From Machinery to Materials: Layered Fabrication of Programmable Surfaces

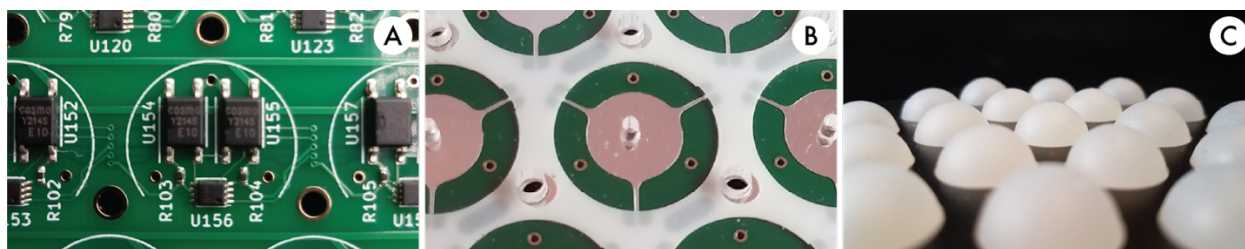


Figure 1. Through a combination of bulk fabrication processes and layered assembly, we arrive at a composite material that contains both computational elements (a) and embedded, mechanical mesostructures. One such mesostructure is an electrostatic valve (b), which enables pneumatic control of soft surfaces (c). We refer to this particular composite (the electrostatic valve array) as *Stoma-Board*, an allusion to the pores in plant leaves (stomata) that regulate air exchange.

1. Introduction

Every day, we interact with dozens of physical surfaces that are largely static. Forward-looking designers have perceived an opportunity in this space — anticipating a suite of tangible, shape-changing interfaces that are as mutable as their digital relatives¹³¹. But such surfaces, clearly, are far from commonplace today.

The few existing objects that *do* adapt to our inputs (tactile Braille displays, for instance) are specialized devices, only able to achieve a physical state change through the complex manipulation of multiple independent elements^{132,133}. This modular design, while workable in cost-agnostic settings, is limited in practice by a labor-intensive construction process¹³. If we are interested in making shape-shifting surfaces pervasive

within our world, then this discretely-assembled machinery must be replaced with a more fabrication-friendly alternative.

Layered construction provides a solution. Among some mechanical engineers, there is an emerging migration from multi-piece mechanisms to cleverly-designed composites --- soft and flexible structures that arise from the stacking of patterned films^{14,134}. In this chapter, we adopt layer-driven design as a method for building programmable pneumatic surfaces that are inherently scalable. We introduce an electrostatic valve array (Figure 1), which we refer to as *Stoma-Board* — an allusion to the micro-scale pores (stomata) that regulate air intake from the surface of plant leaves.

By orienting our fabrication process around bulk procedures — actions that can be performed in one or more "sweeps", such as casting, photo-etching, and laser cutting — we eliminate the many individual manufacturing steps that are common in more module-driven designs. These modular pneumatic surfaces consist of separate cells that are each tied to independent solenoid valves^{135,136}; in contrast, our approach allows for the creation of variable-sized arrays within a fixed number of fabrication steps.

This layer-driven design integrates well with the type of composite transducers that have taken hold within the soft robotics community¹³⁷, providing a scalable bed of actuators that can replace networks of individually connected tubes. Such a system may be well-suited for the production of room-scale pneumatic surfaces^{56,59}, which, like current toolkits^{138–140}, have so far relied on a modular approach. Composite arrays such as *Stoma-Board*, which simplify manufacturing through the use of bulk-fabricated layers, may bring large-scale structures and high-resolution surfaces within reach.

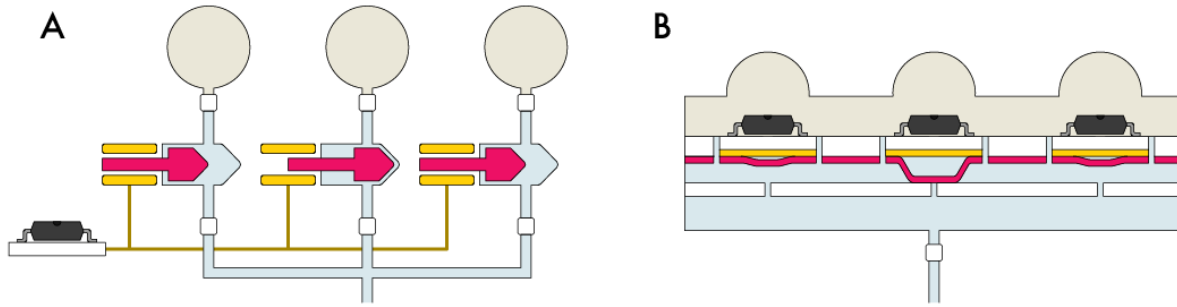


Figure 2. A module-driven design (a), in the form of a solenoid array, is contrasted with a layer-driven design (b), in the form of an electrostatic valve array. Actuators in (a) are distinct mechanisms; actuators in (b) are cells in a common substrate. Communication in (a) is centralized (one-to-many); communication in (b) is distributed (neighbor-to-neighbor). Energy in (a) is transmitted via independent branches; energy in (b) is accessible through a global, unified reservoir.

2. Layer-Driven Design

In Figure 2, we illustrate the driving principle of our design process: functionality arises from a stacked configuration of bulk-fabricated layers, rather than the assembly of many individual parts. (We observe that sufficiently mature robotic processes, such as “pick-and-place” or 3D-printing, may also be considered bulk procedures.) Though this imposes an additional engineering challenge during the development of the primary actuation mechanism, the resulting material cell will necessarily be scalable under this framework. Such designs are distinguished by four characteristics:

1. *Constituent actuators are not distinct components, but instead cells within a common substrate.* In our implementation, this substrate is a patterned electrode array, with each node able to exert an attractive force on one section of a shared diaphragm.
2. *Communication, and in some cases computation, is not centralized, but instead distributed throughout the surface.* Neighboring cells exchange messages with each other, which our system exhibits by means of a daisy-chained control scheme.
3. *Energy is not delivered through individual connections, but instead made available through a globally accessible reservoir.* Analogous to the power and ground planes that have long been a staple of multi-

layer circuit boards, the base layer of our system is a vacuum-connected cavity that acts as a global pneumatic sink.

4. *Pattern generation precedes fabrication and assembly.* In a module-driven design, actuators may be built independently, and then placed in positions across a surface. In our system, patterning must take place earlier in the design process, as all actuators are built concurrently during the layer-stacking assembly procedure. We implement a simple computational tool to generate these layouts.

We demonstrate the validity of this approach through the fabrication of a programmable surface, which serves as a “raw” base layer for mounting multiple types of pneumatic transducers.

3. Background

In this section, we provide some background information on soft robotics (in particular, pneumatically actuated surfaces), as well as electrostatic actuators.

Soft and Foldable Robots

Roboticians, in recent years, have been drawn to shape-changing composites — with some eschewing traditional linkages in favor of flat materials that fold or stretch^{14,134}. By stacking layers with dissimilar mechanical properties, or by cutting patterns into overlapping sheets¹⁴¹, sections of the resultant material can be constrained to move along predefined paths when actuated. In a soft robot, for example, a bending motion might be achieved by depositing a compliant silicone layer on top of a more rigid, strain-resisting substrate. When subjected to an axial force, either via piezoelectric¹⁴², electrostatic¹⁴³, or pneumatic actuation¹⁴⁴, the material curves in the direction of the less-stiff layer.

By micro-machining these individual layers (selectively removing material), bending behaviors can be further restricted to specific regions of the composite. This enables “pop-up” mechanisms, such as the origami-inspired joystick by Mintchev et al.¹⁴⁵, or the millimeter-scale delta robot by McClintock et al.¹⁴⁶. In each of these systems, the composites consist of

a flexible layer (polyimide), sandwiched between two rigid layers (fiberglass, or carbon-fiber). Strategic cuts in these layers give rise to Sarrus linkages, which lift sections of the composite out-of-plane. (These particular mechanisms are both driven by external base-stations, though other researchers have begun to explore foldable composites with self-actuating inner layers¹⁴⁷.

Further embracing deformation, roboticists such as Kellaris et al.¹⁴⁸ achieve out-of-plane motion through the manipulation of flexible, fluid-filled membranes. These sandwich-style actuators are characterized by “zippering” motions — as portions of the outer membranes are squeezed together, an inner layer of fluid flows into a predefined pocket, causing the membranes to bulge outward. This squeezing is initiated by an applied electrostatic force, similar to the operation of dielectric elastomer actuators. Artificial muscles such as these can be fabricated from stacked thermoplastic sheets, which are then heat-sealed, filled with fluid, and patterned with compliant electrodes¹⁴⁹. Other researchers have used similar constructions to produce wearable haptic interfaces¹⁵⁰, indicating that this may be a viable technique for the creation of programmable surfaces.

For these devices, electrostatic actuation is a reasonable choice. It is a low-power method, only drawing current during transient periods; and at small distances, these actuators can provide a significant amount of force. We discuss this further in the next section.

Electrostatic Actuators

Two charged objects of opposite polarity, when placed in close enough proximity, will begin to move towards each other as they experience an attractive electrostatic force. This seemingly straightforward transformation of electrical energy into mechanical work has seeded a significant body of research relating to electrostatic actuators — perhaps most widely in the MEMS community. Here, investigators have leveraged this principle to develop micro-scale (sub-millimeter) sensors, switches, pumps, and valves¹⁵¹. Many of these devices incorporate some displacement mechanism, consisting of an anchored electrode that pushes or pulls a moveable plate.

At the meso-scale (centimeter) and larger, this attractive electrostatic force has been harnessed for braking and locking — modulating the frictional force between two electrodes that slide against one another. Diller et al. explored this through the construction of an electrostatic clutch, engaging and disengaging spring elements in a leg exoskeleton¹⁵². Hinchet et al. developed a similar concept, for grasping in virtual reality¹⁵³. Their thinner strips were worn on the fingers, constraining hand movement when a voltage was applied. Recent work by Zhang et al.¹⁴² has adapted this technique for use in for shape-changing pin displays — individual braking mechanisms will lock brass pins in place as a global upward force is applied, resulting in a reconfigurable height field.

An advantage of this friction-based mechanism is that during operation, the components are always in close proximity, where electrostatic forces are strongest. (These forces fall off with the square of the distance between charged elements.) Consequently, such systems have been able to bear human-scale forces (up to 100 N for the 10 cm wide electrode pairs in the Diller et. al clutch¹⁵²). A key consideration is that the direction of actuation is normal to the applied electrostatic force (i.e. attraction up and down prevents sliding from left to right).

This has not typically been the case with regard to valves, for which electrostatic forces must resist an applied pressure directly. In the most basic construction¹⁵⁴, a flexible conductive diaphragm sits on top of an insulated electrode, covering a hole in the center of that electrode. When a voltage is applied, the attractive electrostatic force creates a seal over the hole, restricting air flow (Figure 6).

The difficulty here is that as soon as a small gap emerges, a peeling effect rapidly dislodges the diaphragm. As a result, valves with this construction are often constrained to operate within low pressure conditions¹⁵⁵ (1 kPa), or to be driven with high voltages¹⁵⁶ (>600 V).

To address this limitation, researchers have used additional pneumatic ports, which serve to balance the applied pressure and hold the diaphragm in place^{157–159}. This, however, requires a second pneumatic source (one that sits just below the input pressure), which increases the overall fabrication complexity. In the next section, we introduce a novel valve design that overcomes this problem without the addition of extra pneumatic channels. By inverting the diaphragm actuation mechanism,

such that it acts as a check valve rather than a lid or a cap, we are able to regulate pressures 80 times larger than comparable implementations¹⁵⁵ — a necessary performance increase for working with human-scale forces.

4. Patterned Valves for Programmable Surfaces

In the remainder of this chapter, we describe the design and fabrication of a material that emerges from our layer-driven framework: *Stoma-Board*. This is an array of patterned, electrostatic valves, situated on top of a global cavity (Figure 3a, 3b). By switching individual valves on and off, we are able to control the flow of air into a transducer layer on top of the board. For instance, as shown in Figure 3d, we can regulate airflow into discrete pockets of an elastic substrate (causing these cells to inflate, buckle, or bend, depending on the geometry).

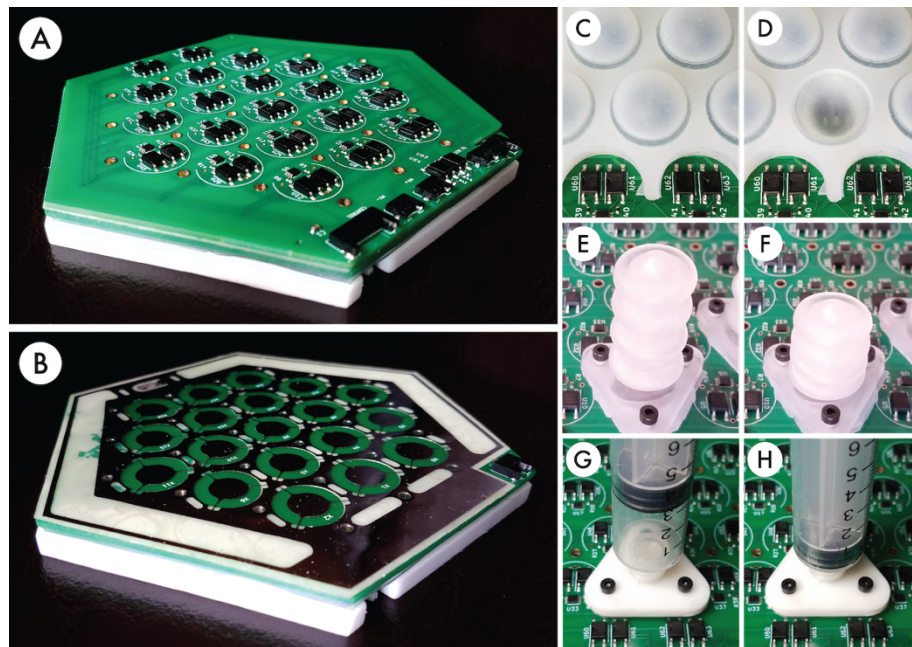


Figure 3. One of our 19-cell Stoma-Boards (a). Directly below the top circuit board is a conductive diaphragm (b), which regulates air flow into a global reservoir. Cells are individually addressable. We treat Stoma-Board as a “raw material”, using it with different transducer layers to build a variety of devices. Example transducers are: (c,d) Bistable dome, part of a larger cast membrane. (e,f) 3D-printed bellows-style transducer, which contracts under negative pressure. (g,h) Rigid Luer-lock attachment, connected to a syringe for linear actuation.

Many programmable surfaces, if they are discretely actuated, more often resemble collections of machines than they do continuous matter. This is due in part to the limitations of a module-driven design — if actuators are assembled as independent components, increased resolution comes at a high manufacturing cost. Our aim at the outset of this work was to solve this by identifying a fabrication process for which assembly time is

proportional to the number of layers, rather than the number of actuators. This decoupling allows us to pack more actuators into the same area with little or no additional time loss.

Researchers in MEMS have long embraced lithography as a means of achieving this type of feature density¹⁶⁰. However, industrial processes in this area are usually optimized for the production of singular devices at the micro-scale (accelerometers, gyroscopes, etc.), as opposed to networks of actuators that extend over a centimeter-scale surface area or larger. For this more expansive application, we looked towards the already-mature process of printed circuit board (PCB) manufacturing, and focused on integration into that pipeline.

PCB manufacturing is characterized by patterning, etching, and layup (stacking and bonding layers)¹⁶¹. Most fabricators start with a copper-clad fiberglass sheet, and begin production by applying a photosensitive coating and printed mask. They follow these steps with a chemical procedure that selectively removes exposed copper from the substrate. For boards with more than two layers, additional material is then stacked around the core and laminated together — usually with an insulating dielectric on both sides of the sheet, and copper foil on the outer layers. This new material can be etched again, and the process may be repeated as more layers are added (such that a multi-layer board is effectively a sandwich of smaller sub-boards). Some complex PCBs, particularly those with embedded electronic components, are actually composites of other fully drilled, plated, and soldered circuit boards¹⁶².

From this standpoint, the Stoma-Board layers that we introduce can be seen as an extension of this manufacturing process — a few extra steps during the layup and bonding stage. The result of our procedure is a composite material with integrated electronic and pneumatic components (Figure 4).

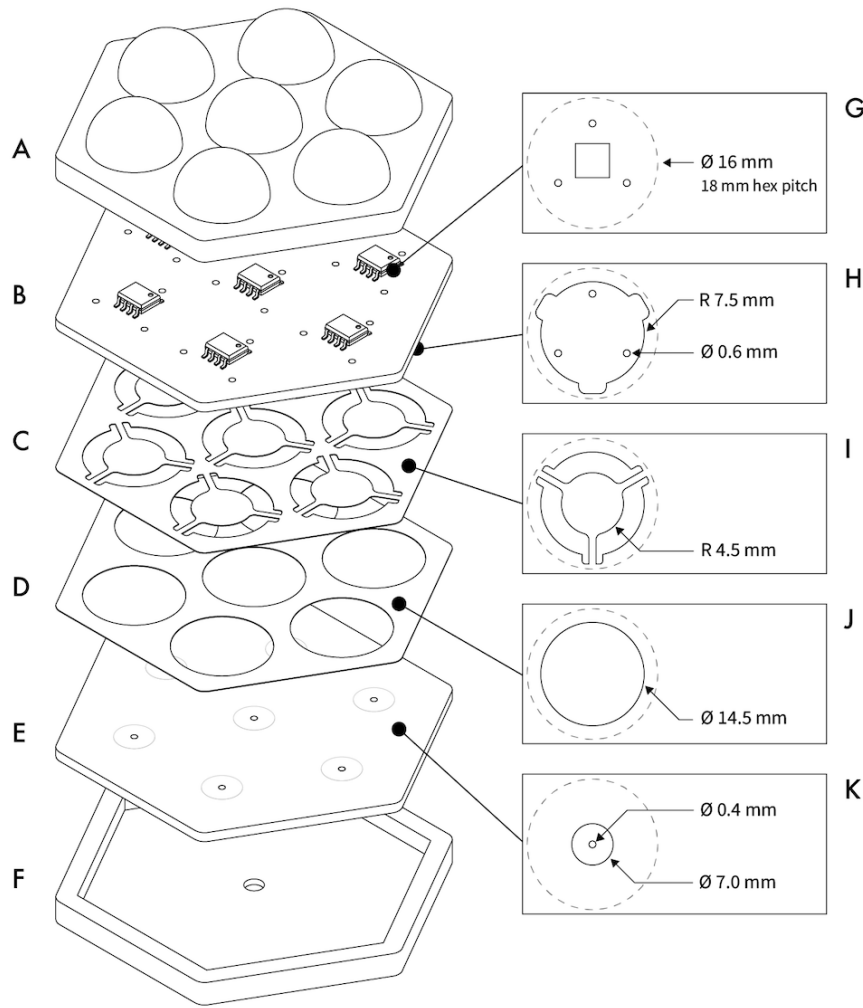


Figure 4. Exploded view of the Stoma-Board structure. The transducer layer (a), in this case, is an elastic surface, composed of multiple pockets that can inflate, buckle, or bend independently. Air flow into this layer is regulated by a flexible, conductive diaphragm (c, i). When a vacuum is applied, this diaphragm deflects downward through a rubber spacer layer (d, j), and is pulled flat against an outlet (e, k). This closes the valve and (in this example) traps air inside the pockets of (a). To open the valve and allow air to flow, a electric charge is applied to an electrode (h) on the underside of a printed circuit board (b). Electrostatic forces retract the diaphragm from the valve outlet, allowing air to move from the transducer layer (a) to a global sink (f). Cells are hexagonally packed, with an 18mm pitch.

From top to bottom, these layers are:

1. An application-dependent transducer. (For example, an elastic surface, with air-filled pockets that buckle or bend when actuated.)
2. A printed circuit board, with control circuitry on top, and patterned electrodes on the underside.
3. A metallized polyester diaphragm (Mylar, 25 μm), which regulates air flow.
4. A natural rubber spacer (150 μm), through which the diaphragm travels.
5. A second printed circuit board, with patterned electrodes on top.
6. An SLA 3D-printed cavity, which serves as a global pneumatic sink.

Note that although each layer is manufactured by a bulk process (casting, etching, laser-cutting, or machine placement), we still retain the ability to dynamically actuate discrete subsections of the finished surface.

Working Principle

Air flow through each valve is regulated by a metallized polyester diaphragm, which rests underneath an insulated copper electrode (Figure 4). This diaphragm cell is connected to the greater membrane layer by three hinges (Figure 4i), which allow it to flex out of plane.

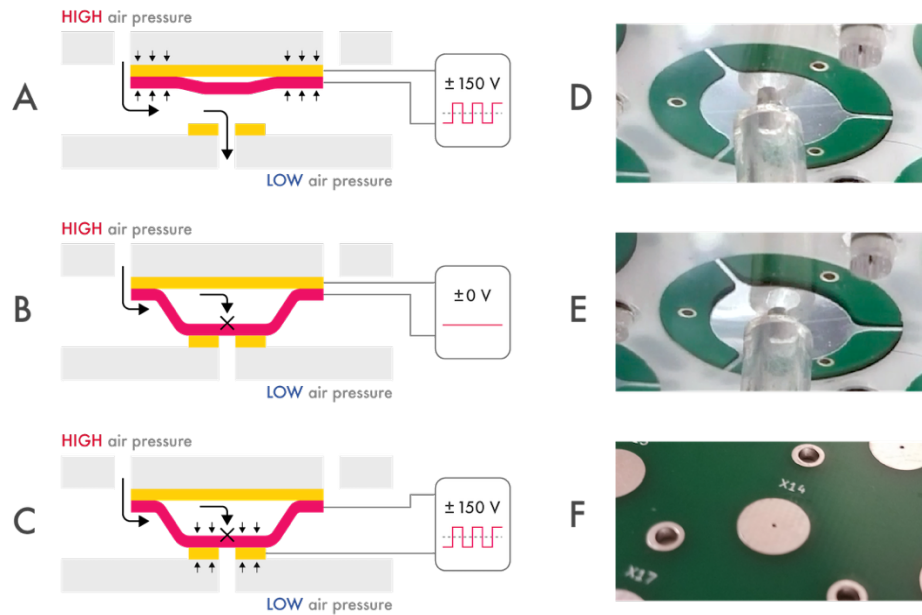


Figure 5. The three states of operation. (a) *Active state*: A voltage is applied between the valve-opening electrode and the conductive diaphragm. The resulting electrostatic force causes the diaphragm to adhere to this electrode, allowing air to pass underneath. (b) *Non-active state*: If no voltage is applied, a pressure differential will pull the diaphragm flat against the valve outlet, restricting air flow. (c) *Sealed state*: A voltage is applied between the conductive diaphragm and the valve-sealing electrode. Electrostatic forces pull the diaphragm tight against the outlet, mitigating leakage. (d,e) Diaphragm and valve-opening electrode in active (open) and non-active (blocking flow) states. To show actuation in these photographs, a clear acrylic plate is used in place of the opaque valve-sealing electrode. (f) An image of the valve-sealing electrode, by itself.

In the active state (when a voltage is applied), electrostatic forces hold the diaphragm in place, and air passes freely from inlet to outlet (Figure 5a). In the non-active state, negative pressure at the valve outlet instead pulls the diaphragm downstream, blocking the outlet and impeding further flow (Figure 5b). To mitigate leakage in the non-active state, an optional secondary electrode at the outlet can be activated and used to further seal

the valve (Figure 5c). Throughout this chapter, we will refer to the first electrode as the valve-opening electrode, and the second electrode as the valve-sealing electrode.

The attractive force between (either) electrode and the valve diaphragm is expressed by the relationship:

$$F \propto \frac{V^2 A}{d}$$

where F is the electrostatic force, A is the overlapping electrode area, V is the applied voltage, and d is the distance between the electrode and the diaphragm¹⁵⁶. For an electrostatic valve, the parameter of greatest concern is the distance d (the other two right-side variables can be readily controlled, but this distance will vary during operation). Since the attractive force decreases with the square of this value, small fluctuations in distance have large consequences for valve behavior.

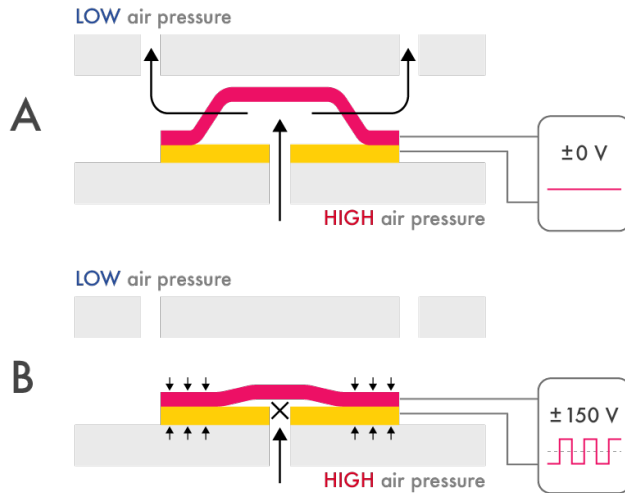


Figure 6. Early Stoma-Board prototypes used an alternate valve configuration, inspired by common MEMS implementations¹⁵⁴. The diaphragm rests above the valve inlet, and when actuated, blocks air flow from below (b). At pressures beyond 1 kPa, small gaps begin to emerge at the valve inlet, which propagate across the surface and eventually dislodge the diaphragm¹⁵⁵. When no voltage is applied, air passes through the valve unimpeded (a).

In particular, electrostatic latching mechanisms such as these are often susceptible to “peeling”¹⁶³; and some valve geometries are more vulnerable than others. One of these configurations is shown in Figure 6, in which the diaphragm is positioned to block incoming flow from below, as opposed to sealing outgoing leakage from above. As soon as a minor gap emerges under the diaphragm (due to pressure fluctuations, surface roughness, or other environmental factors), the electrostatic force at that interface is greatly diminished. This reduction in force means that a section

of the diaphragm may begin to balloon or lift away, propagating until the seal is fully broken. Our earliest prototypes were built in this style, and (just as in similar implementations¹⁵⁵), they were unable to regulate pressures above several kilopascals.

It is possible to address this limitation by adding additional pneumatic sources and channels, in order to keep the pressure differential around the diaphragm within a small threshold^{157–159}. But seeking a simpler construction, we took an alternate approach — by inverting the diaphragm configuration as described earlier (Figure 5a–c), we were able to take advantage of a “zippering” effect¹⁶⁴ to actuate the valve.

“Zippering” motions, in our system, start at the hinges of the diaphragm and move inward. Consider the valve in the non-active state (Figure 5b): negative pressure has pulled the diaphragm flat against the outlet. This is 150 μm below the valve-opening electrode — small on a human scale, but large enough for electrostatic forces to be noticeably weakened. The diaphragm hinges, by contrast, are ten times closer — separated only by a thin dielectric layer, just 15 μm thick. When we switch to the active state, strong electrostatic forces near the hinges allow us to retract a diaphragm that would otherwise be out of reach at high negative pressures (> 80 kPa).

Insulating Dielectric and Driving Voltage

When we apply a voltage between an electrode and the valve diaphragm, an insulating layer is needed in the middle to prevent electrical shorts. In early prototypes, we used 25 μm polyimide sheets, hand-laminated onto bare copper electrodes. (This technique is similar to Dextres¹⁵³, though we did not use any adhesive.) Polyimide is a natural choice for this layer — with a breakdown voltage in excess of 250 V/ μm , we can administer several kilovolts before the insulating layer fails and conducts.

Our system, however, operates between 150V and 300V (drawing a low and safe 50 μA). At these levels, a thin layer of solder mask (breakdown voltage around 25–30 V/ μm) is sufficient insulation, with the additional benefit of a smaller minimum distance d between conductors.

Consequently, we use only solder mask to separate the valve-opening electrode from the metallized side of the diaphragm.

The valve-sealing electrode is left exposed (bare copper). It does not contact the metallized side of the diaphragm — the 25 μm Mylar sheet itself provides adequate insulation.

Daisy-Chained Electronic Communication

Fully distributed computation is a target that arises from the layer-driven model (and is addressed briefly in Section 7). This is desirable not only from a fabrication perspective (eliminating multiple physical connections to a central hub), but also important for making robust programmable materials that scale without regard to the processing power of a master controller.

A first step towards this objective is casting each material subsection as a distinct computational node, capable of communication with its immediate neighbors. We implement this (somewhat simply) in our system by means of a daisy-chained control scheme.

Physically, this communication takes place in the layer above the valve diaphragm (Figure 4b). Each material cell contains several discrete electronic components, as well as the patterned copper circuitry that connects them, which includes the valve-opening electrode. A shared clock line and global power sources (high-voltage rails for actuation and low-voltage rails for signaling) run across the surface and are accessible by each cell.

A simplified schematic is shown in Figure 7. An electrode may be connected to either of the high voltage rails through a pair of opto-isolated solid-state relays^a. A discrete flip-flop^b drives the LEDs of the opto-isolator, and also serves as the main communication element. On every clock pulse, the cell as a whole follows two rules:

1. Pass the current cell state to the right-side neighbor.
2. Receive a new cell state from the left-side neighbor.

^a Cosmo Electronics KAQY214S

^b Texas Instruments SN74LCV1G74

With this configuration, an external microcontroller upstream can quickly send data to all nodes in the chain. This operation is similar to that of the addressable LED strings (such as the APA102 or WS2812) that in recent years have served a material-like role in maker and DIY communities¹⁶⁵. Our clock signal runs at 300 kHz, which allows us to set the state of 300 valves per millisecond. This rapid switching means that in practice, the overall refresh rate of the surface is generally independent from the number of valves, and instead a function of the pneumatic control scheme (see Section 4.4).

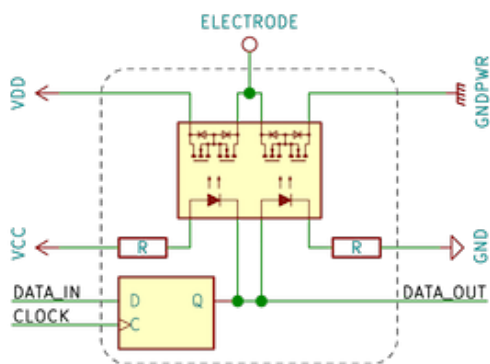


Figure 7. Electronic components within a Stoma-Board cell. Cells are daisy-chained, and change state in response to a global clock signal.

Opto-isolated relays are not the only driver option. In earlier prototypes, we used resistor-transistor logic to control the valve array — electrodes were either actively switched low or passively pulled high. Though smaller in footprint, which is desirable, this design wastes power on the actuation side, continuously sending current through the resistor when the electrode is held low. A more efficient half-bridge topology was also considered, but ultimately discarded due to the need for additional high-voltage bias supplies. While the package size of the solid-state relays did limit our electrode density, we decided that the reduced complexity, low-power actuation, and isolation benefits made them a reasonable choice for this implementation.

Pneumatic Push-Pull Control Scheme

The bottom layer of our Stoma-Board (Figure 4f) is a global reservoir, which typically acts as a pneumatic sink (negative pressure, pulling air down through the material). The presence of this common driver has implications for the design of the transducer layer, as well as for the actuation control scheme.

Consider the following example, relating to the silicone domes in Figure 3c. An open valve allows air to flow from a pocket in the top layer to the global sink, deflating or depressurizing that top pocket in the process. Closing the valve will restrict that air flow, preventing further deflation — but this stoppage alone can not put the top pocket back in the inflated or neutral position.

In order for this layer to “rebound”, spring forces in the elastomer must overcome the negative air pressure within the cell. One way to address this is by strategically adding drain holes in the top layer, allowing outside air to leak into the cell so that the pressure can begin to equalize. However, for more predictable behavior, we can instead use the "push-pull" actuation scheme outlined in Figure 8. Under this scheme, a global reservoir is connected to an alternating pressure source (such as the syringe pump in Figure 9). During the "pull" stroke, air flows out of the Stoma-Board, and actuated cells buckle under the negative pressure. During the "push" stroke, positive pressure restores all cells to their upright positions.

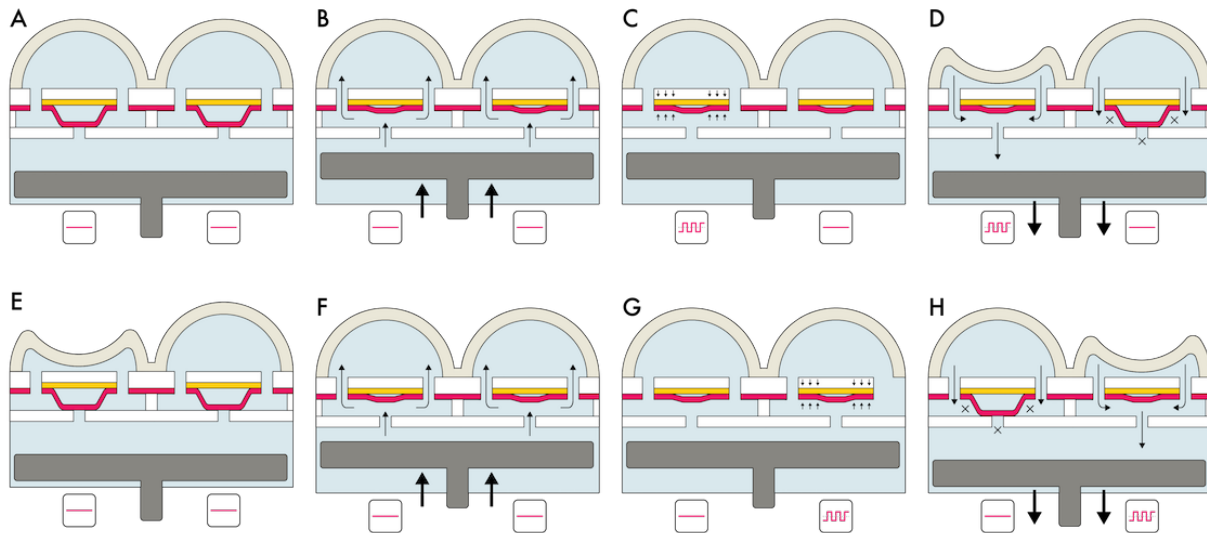


Figure 8. The "push-pull" actuation sequence for driving Stoma-Board prototypes using bi-stable silicone domes. (The alternating pressure source is represented by a moving piston, in gray.) Initially, the system is at rest (a). When positive pressure is created in the global reservoir, all diaphragm cells are pushed upward towards the top electrodes (b). At this stage, we may apply a voltage to latch one or more diaphragm cells in place (c, left). When negative pressure is created in the global reservoir, air exits the latched cells, and the elastic structures buckle (d, left). Note that in the unlatched cells, the elastic structures remains upright, as the diaphragm prevents air from escaping (d, right). If we remove the applied voltage, the valves may close, but the buckled cells remain unchanged (e, left). To restore these cells, we can again create a positive pressure (f), at which point we may choose to actuate another group of cells by the same process (f-h). The global pressure oscillations effectively determine the "refresh rate" of the Stoma-Board.

For this procedure, the global reservoir is connected to a variable source, which alternates between positive and negative pressure. To generate this behavior, we use a custom syringe pump (Figure 9). In the “push” stage, air is forced upward through the valves, into every cell of the top layer. In the “pull” stage, air is drawn out of the system — but only through valves that are in the active state.

Note that since every cell is affected by the "push" stroke, the transducer layer is fully inflated between refreshes. For some applications, such as refreshable Braille displays, this is sufficient¹⁶⁶. However, if more dynamic control is desired, then a secondary valve layer is necessary to block the positive pressure from the "push" stroke. (We describe such a construction in Section 8.4.)

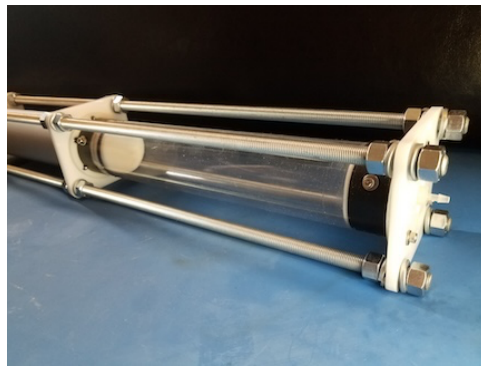


Figure 9. The linear-actuator syringe pump used to pneumatically drive our Stoma-Board surfaces

While we use a syringe pump in our applications, any alternating pneumatic source will suffice. Ultimately, it is the frequency of this source that determines the refresh rate of a Stoma-Board surface. If fast refreshes are necessary, then the flow rate of the pump must large enough to quickly pressurize and depressurize the global cavity.

5. Fabrication Process

Cutting: Diaphragm and Spacer

The Stoma-Board diaphragm is laser-cut from a 25 μm thick Mylar film, metallized on one side (CS Hyde 48-1F-1M). It is important that this film is held flat during the cutting process -- if manufactured imprecisely, it may kink or buckle out-of-plane when aligned with the electrode array. We solve this by first laminating the uncut Mylar film onto a sacrificial acrylic sheet (using isopropyl alcohol and a rubber squeegee), which lies

flat on the laser bed. After the cut, we peel the finished diaphragm off of the acrylic substrate in a single step, with no weeding or deburring necessary.

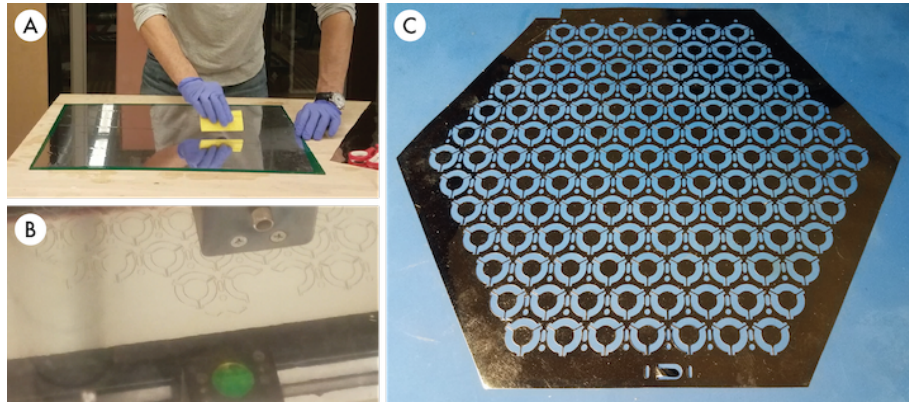


Figure 10. To fabricate the diaphragm, we first use isopropyl alcohol and a rubber squeegee to laminate a metallized Mylar film onto an acrylic sheet (a). The diaphragm pattern is then laser cut (b), and the finished diaphragm is removed (c).

Notice that our diaphragm is cut with a small tab near the bottom edge (Figure 10c). Later, in the layup process, we use silver epoxy (MG Chemicals 8331) to adhere this tab to an exposed copper pad on our circuit board.

The spacer (Figure 4d) that separates the diaphragm from the valve outlet is cut from a sheet of natural gum rubber, 150 μm thick (McMaster-Carr 8611K11). The fabrication process is similar to the one described above — the rubber is laid flat on an acrylic sheet prior to the cut, and peeled away afterwards in a single motion. We first experimented with rigid spacers (polyester sheets ranging from 50 to 250 μm), but ultimately found that a more compliant material led to better valve performance, perhaps due to a tighter pneumatic seal.

For some prototypes, we used this same rubber material as a bottom gasket layer, which sits between the valve-sealing electrodes (Figure 4e) and the common cavity or printed piping structure (Figure 4f). These gaskets were cut in the same manner as the spacer.

Casting: Upper Gasket and Soft Transducers

For the top layer of our system, we tested a variety of custom-made transducers. Some of these were 3D-printed (Formlabs Elastic Resin 50A), and some of them were cast (Smooth-On Dragon Skin 30 and Mold Star

30). Additionally, some experiments required a rigid manifold, which sat on top of a cast gasket layer.

To begin the casting process, we first spray a release agent (Mann Ease Release 200) onto a two-piece 3D-printed mold (Figure 11). This mold is left to sit for ten minutes. In that interval, the uncured silicone is mixed and degassed at -27 inHg, in order to remove any large air pockets. The silicone is then poured into the mold, and left to cure at room temperature for 16 hours.

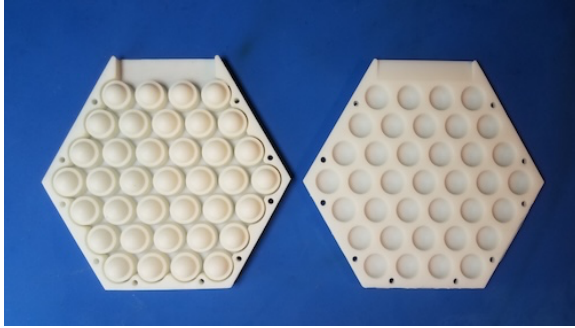


Figure 11. A 3D-printed mold used to cast one of our soft transducer layers (the bistable dome array).

Printing: Global Reservoir and Upper Manifold

Our Stoma-Board valves rest on common reservoir, used as a global pneumatic sink (Figure 12). While this could be a single, open cavity (CNC-milled, for example), we chose to 3D-print a connected pipe structure, in order to minimize the reservoir volume and allow for quicker pressure changes.

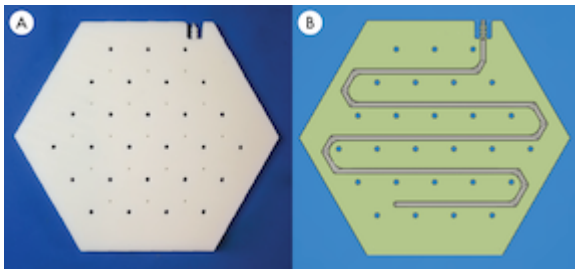


Figure 12. 3D-printed reservoir (a) that serves as a global pneumatic sink. Cross section (b).

We also printed rigid manifold layers (Figure 19a) for mounting some of our elastic transducers. While these transducers could be fixed to the upper PCB directly, we found that the manifold helped to ease prototyping, as transducers of different geometries could be more easily swapped and tested.

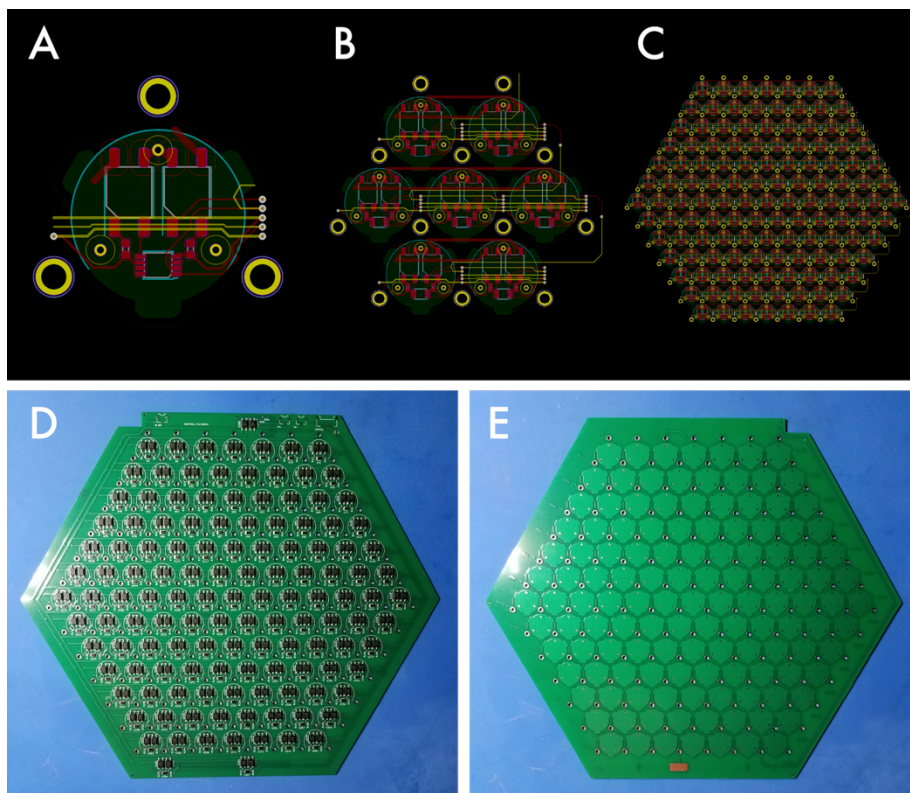


Figure 13. Our command-line interface auto-generates KiCad-compatible design files, for hexagonal Stoma-Board patterns at multiple scales. (a) A single cell, with valve-opening electrode in green, pads for surface-mount components in red, valve inlets in yellow (small), and transducer mounting holes in yellow (large). (b,c) Patterns of 7 and 127 cells. (d,e) Fabricated circuit board, with electronic components on top (d) and patterned electrodes on bottom (e). Note that while our design software can produce Stoma-Board patterns at various scales, all actuation characteristics are measured from a 19-valve test rig, not the 127-valve board.

6. Actuation Characteristics

There are two ways we might wish to actuate a cell of our Stoma-Board. We can either start with the global vacuum active and try to *lift* the valve diaphragm upwards; or we can first *latch* the valve diaphragm in the raised state, and apply a vacuum only after the valve is already open.

We make the distinction because Stoma-Board valves exhibit a degree of hysteresis (Figure 14). Once the diaphragm has been pulled close to the valve-opening electrode, the newly strengthened electrostatic forces tend to hold it in place, even if the voltage drops. (Recall that the electrostatic force between two objects is inversely proportional to the squared distance.)

While the lifting mode is arguably more dynamic, the latching mode allows us drive the board at lower voltages, and circumvents some of the dielectric charging issues that we discuss in Section 7. This is also the mode used in our push-pull control scheme (Figure 8). However, for

completeness, we report the actuation characteristics for both lifting and latching modes.

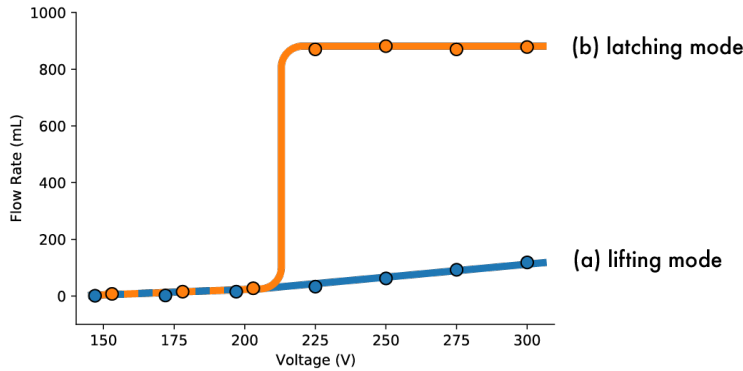


Figure 14. Average flow rate through a single Stoma-Board valve ($\Delta P = 30$ kPa, $n = 3$) over a range of applied voltages. The behavior differs in lifting (vacuum-first) and latching (voltage-first) modes. (a) Lifting: Below 200V, negative pressure dominates this interaction and keeps the valve diaphragm closed. Beyond this point, air flow grows linearly as electrostatic forces lift the diaphragm away from the valve outlet. (b) Latching: If a valve is latched open before negative pressure is applied, electrostatic forces are stronger due to the reduced diaphragm distance. Above 200V, the valve remains fully open. Below this value, the diaphragm is pulled away from the valve-opening electrode, and behavior is similar to the lifting mode.

To facilitate our measurements, we attached a rigid manifold to the top of a 19-cell Stoma-Board (in place of the transducer layer). With Luer lock fittings for each cell, we were able to quickly add and remove mass flow sensors for evaluating actuator performance (Figure 15).

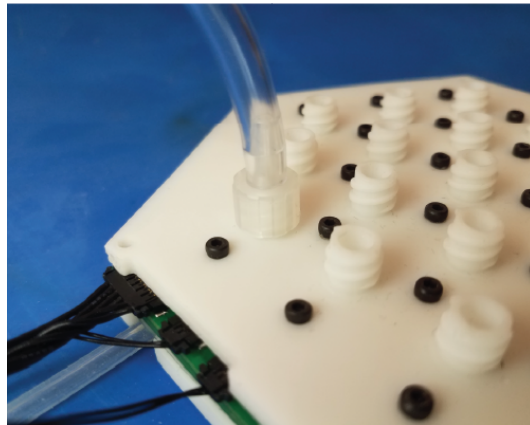


Figure 15. Test rig for measuring the flow characteristics of a 19-cell Stoma-Board. A 3D-printed manifold with Luer-lock fittings allows for individual connections to flow sensors. A vacuum pump acts as a global negative pressure source, which can be manually adjusted with a breaker valve. Valve inlets are left at atmospheric pressure.

Opening Voltage

The opening voltage is associated with the lifting mode — it is the point at which electrostatic forces upward surpass pneumatic forces downward, and air begins to pass through the cell.

To measure the opening voltage for a given valve, we first applied a global negative pressure, which pulled the valve diaphragm closed. We then increased the voltage between the diaphragm and valve-opening electrode, in increments of 10V, until the diaphragm was attracted upwards and significant air flow through the valve was observed (25 mL/min).

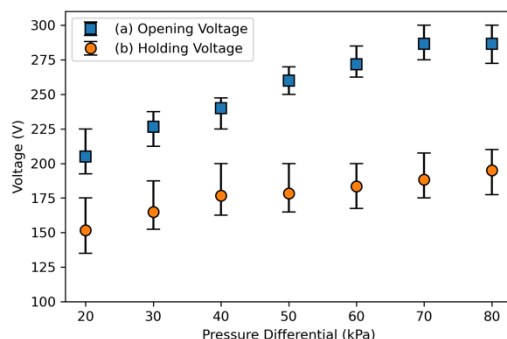


Figure 16. Average opening voltage (lifting mode) and holding voltage (latching mode) across a range of pressure differentials. Valves were sampled uniformly from the 19-cell test rig in Figure 15. Error bars represent 25th and 75th percentiles ($n=7$).

Figure 16a charts the Stoma-Board opening voltage as a function of pressure (the differential between valve inlets and the global reservoir). At higher pressures (80 kPa), this approaches 300V, which we chose as the limit of our system. Beyond this point, we noticed an increase in parasitic effects.

Though the valves in our application examples all occupy discrete “on” and “off” states, we can achieve variable flow rates (between “fully on” and “fully off”) by driving the system at higher or lower voltages respectively. Figure 17 shows this relationship for six valves at several operating pressures (sampled randomly from our test rig). Below 150 to 200 volts, the applied vacuum dominates this interaction. As the voltage increases beyond this point, air flow through the valve increases as electrostatic forces lift the diaphragm away from the valve outlet. We did not observe any pneumatic crosstalk during these tests — flow rates through neighboring valves stood unchanged.

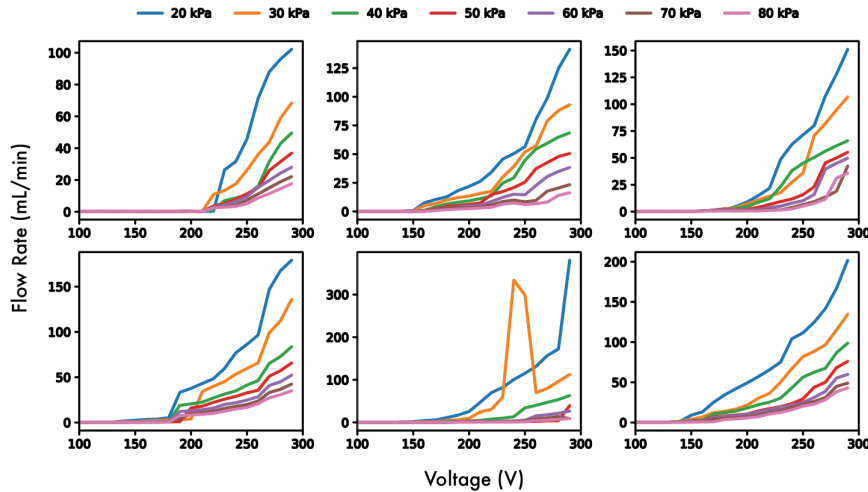


Figure 17. Flow rates for six randomly sampled Stoma-Board valves, as a function of opening voltage. Note that for one valve (middle, bottom) there was a flow rate spike at 30 kPa and 240 V. This anomaly did not reoccur during a second test of the same valve, but we include it here for transparency.

Stoma-Board cells remain robust throughout repeated actuations. In Figure 18, we plot the average opening times for an additional six valves, over the course of one hundred trials. Operating at 300 volts, and subject to 80 kPa, the average valve opens in less than 40 milliseconds. Notice that there is no degradation over repeated trials — the average opening time for the final 25 actuations (36 ms) is comparable to the average opening time for the first 25 actuations (38 ms). We discuss reliability further in Section 8 of this chapter.

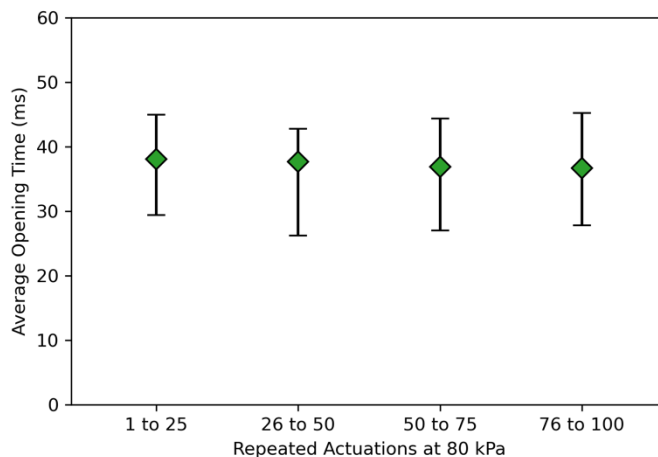


Figure 18. Average opening times for six randomly sampled Stoma-Board valves, over 100 trials. (Opening times are averaged on a per-valve basis, and then these values are averaged to produce the green data points.) Note that opening times remain consistent throughout the experiment. Operating conditions: 80 kPa, 300 V. Error bars represent 25th and 75th percentiles.

Holding Voltage

Since our applications use an alternating pressure source, we are able to take advantage of the latching mode, and activate valves before a negative pressure is applied. In this context, the quantity of interest is the holding voltage -- the minimum voltage required to keep the valve from closing (once the pressure in the global reservoir drops). This value is smaller than the opening voltage, due to the reduced distance between the diaphragm and the valve-opening electrode in the initial state.

To measure the holding voltage (Figure 16b), we first applied 300 volts to the valve of interest. We then applied a negative pressure, and recorded the steady-state flow rate. Finally, we progressively lowered the valve voltage (in increments of 10 volts), until the flow rate dropped below ten percent of the steady-state value. This drop-off does not happen gradually — as shown in Figure 14, there is a "cliff" at which point the valve transitions from fully-open to nearly-closed.

Programmable Stoma-Board Surfaces

From a stock piece of wood or plastic, a craftsperson can construct a multitude of objects, each with mechanical properties that are (in part) a function of the original workpiece. Porous sheets of fiberboard, for instance, can be shaped into a vacuum table; and compliant strips of bamboo can be woven into a flexible basket. Similarly, we envision a process in which a fabricator may start with a sheet of “raw” Stoma-Board, and through the addition of custom transducers, arrive at a variety of programmable surfaces and mechanisms. Importantly, the scalable production of this base material (i.e. patterned valves) is only made possible by the layer-driven manufacturing techniques that we employ.

In this section, we showcase four surfaces (using three types of transducers) that are built upon our “raw” Stoma-Board surface.

Tactile Patterns from a Bistable Membrane

Standard Pitch.

Bistable mechanisms do not require an external energy source in order to maintain their state, which makes them an attractive transducer option in low-power applications¹⁶⁷. For our first demo, we cast an elastic silicone membrane (Smooth-On Dragon Skin 30), composed of bistable, hemispherical cells (Figure 19). While it's possible to attach to this membrane to the raw Stoma-Board directly, we opted to use a 3D-printed manifold as an intermediate layer for ease of mounting.

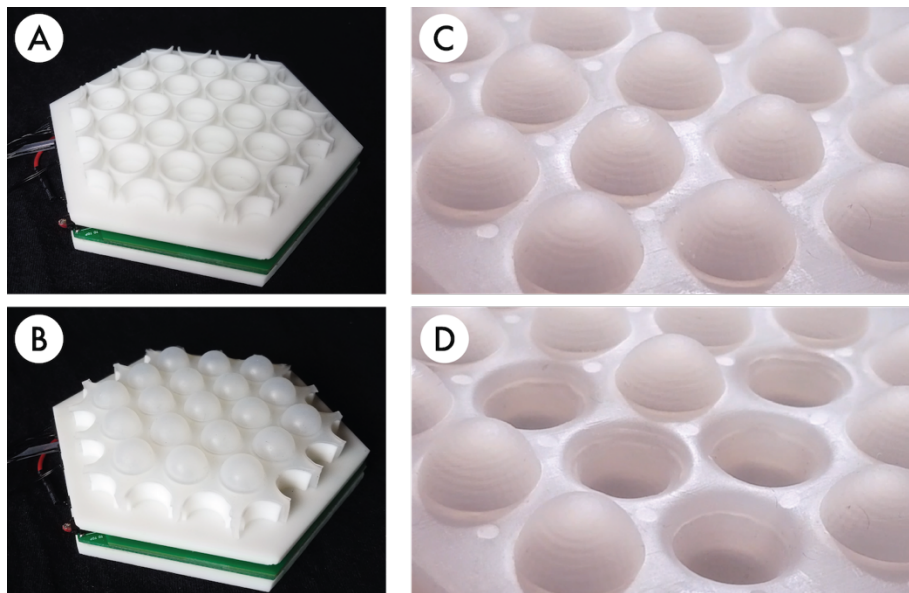


Figure 19. We mount a cast array of bistable domes (b) on a 3D-printed manifold (a). By activating specific valves, we can render tactile patterns (c,d). Domes are 14 mm in diameter.

Initially, all cells are in the raised state. To render a tactile pattern, we latch open the relevant valves, and apply a global negative pressure — this causes the selected cells to buckle into the collapsed state. To return the cells to the raised state, we apply a global positive pressure.

Compact Pitch.

By altering the channel structure of the upper manifold, we can modify the apparent density of the Stoma-Board surface. In our second demo (Figure 20), a set of converging pipes decreases the cell pitch to 2.5 mm (Braille spacing), such that six transducers can fit within the area of a human fingertip. At this scale, the raised domes are more readily perceived as a texture, as opposed to a group of distinguishable objects¹⁶⁸. While the

perimeter of the Stoma-Board is still determined by the electrode and membrane layers, this flexibility in piping can hasten the prototyping of fine-grained (and variable-pitch) tactile interactions.

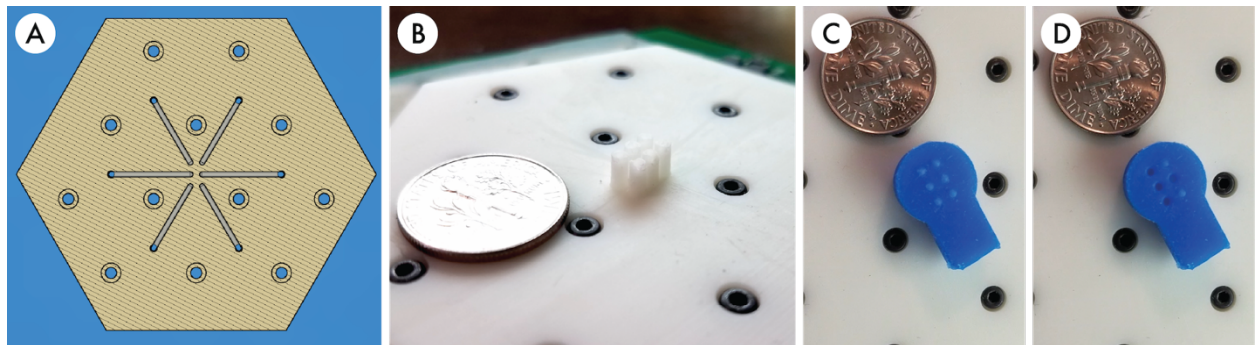


Figure 20. A set of converging pipes, as seen in the manifold cross-section (a), shrink the transducer pitch to 2.5 mm (b). The Braille transducer (c,d) is cast silicone (Smooth-On Mold Star 30), and individual dots can be lowered when the corresponding Stoma-Board valves are opened.

Bellows-Style Robotic Actuators

Among soft roboticists, the bellows-style actuator (Figure 3e, 3f) is well-established as a means of converting pneumatic energy into linear motion¹⁶⁹. When pressurized, it expands along a single axis. When depressurized, it contracts. If two or more are connected in parallel, an applied pressure differential will result in a bending motion, which forms the basis for many soft robotic joints¹⁷⁰.

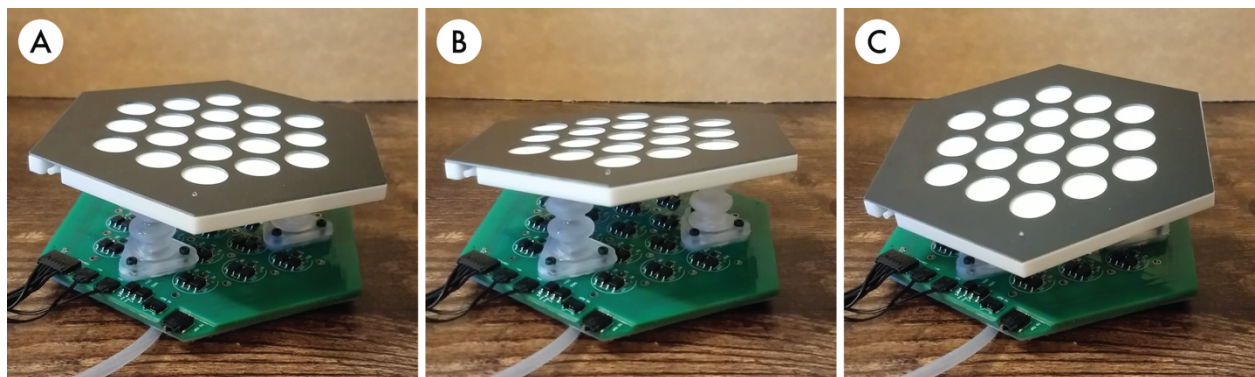


Figure 21. 3D-printed bellows-style actuators control the orientation of an attached platform.

Pneumatic routing is often a challenge when integrating these types of soft actuators. Since every bellows structure needs to be pressurized independently, many research prototypes rely on bundles of tubes, which tether the individual actuators to an external bank of valves.

With Stoma-Board as a base material, these valves can instead be embedded in the robot structure directly. In our third demo (Figure 21), three bellows-style transducers (SLA 3D-printed with Formlabs Elastic Resin 50A) are fixed to a raw Stoma-Board and used to drive a 2-DOF joint. By selectively opening valves and contracting individual transducers, we can determine the roll and pitch of an attached platform. These embedded valves may enable modular stacking, in a manner similar to the continuum robots by Robertson and Paik⁸⁸.

Spoke-Propelled Rimless Wheel

The "rimless wheel" is a rudimentary legged robot, suitable for traversing rough and uneven terrain¹⁷¹. In a typical implementation, the spokes of the wheel are passive elements, and a hub-mounted motor is used to drive the structure forward or backward¹⁷². To enable additional modes of locomotion, variants of this design have incorporated spokes that are independently actuated, which propel the robot by extending and contracting} on command^{173,174}. This modification eliminates the heavy, motorized hub, but requires a separate mechanism for each leg of the robot. Building such an assembly can be cumbersome.

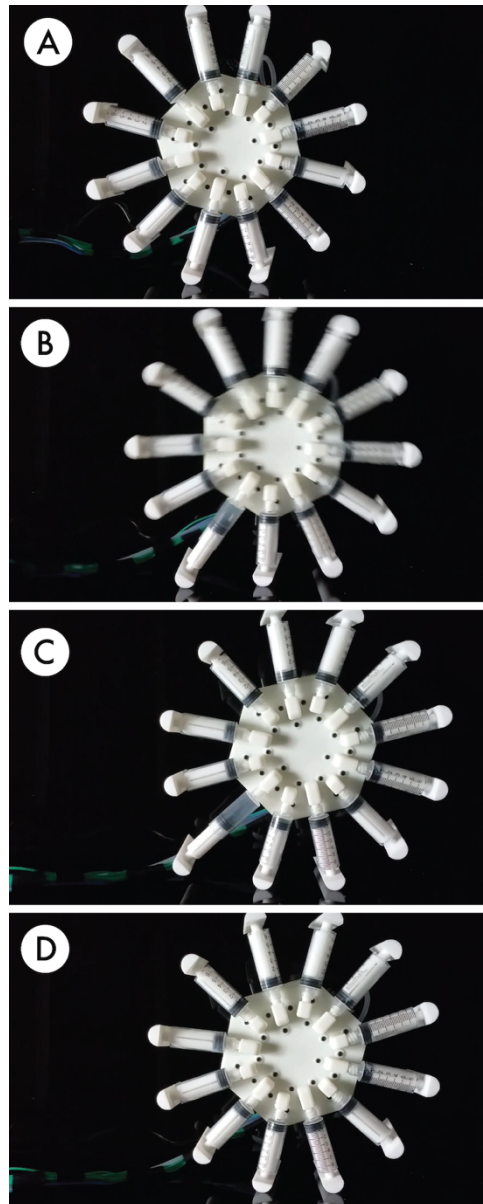


Figure 22. An actuated rimless wheel, with spokes (syringes) that can extend independently. The syringe manifold is attached to a hexagonal Stoma-Board, which is connected to a single pressure source. In the stable position (a), two spokes maintain contact with the ground. When the left spoke is extended (by opening the appropriate Stoma-Board valve) the wheel tilts to the right (b). The wheel falls to a new stable position (c), and the extended spoke is retracted (d).

With a raw base of patterned valves, constructing this prototype becomes as simple as adding a custom transducer layer. In our fourth demo, we showcase a spoke-propelled rimless wheel (Figure 22), which can "walk" forward via a sequential extension of pneumatically-actuated legs. Note that in this example, the transducer layer and common cavity are swapped — we use a global pressure *source* instead of a pressure sink.

7. Discussion, Limitations, and Future Work

Long-Term Use

From a distance, electrostatic actuation appears deceptively simple — a straightforward mapping of an applied voltage to a resultant force. In reality, there are parasitic effects which can reduce the long-term reliability of such systems. The most significant of these, well-recognized by the MEMS community, is dielectric charging¹⁷⁵. This is a molecular-scale phenomenon, which results from a displacement of polarized components within the material. If unaccounted for, this can lead to “sticky” switches (or in this context, valves).

In our Stoma-Board actuators, the dielectric layer is a thin coating of solder mask, which separates the conductive diaphragm from the valve-opening electrode. If a permanent charge develops in this dielectric layer, the diaphragm will always cling to the solder mask, and the valve will be unable to close. This parasitic charging often occurs after an extended period of exposure to a constant, applied voltage¹⁵³.

We were able to mitigate this effect by driving our actuators with a bipolar square wave (as opposed to a DC source), and reducing the peak voltage (no higher than 300 V). While this balancing of polarity over time was sufficient for our applications, it is possible that such adjustments may only reduce or delay this effect. Further reliability testing is needed in order to assess valve performance beyond the 100 trials described in Section 6.1.

Towards Larger Surfaces

A second concern relates to pneumatic actuation, and arises when working with large transducer membranes. Since these membranes are cast from 3D-printed molds, the printer's working area determines the maximum size of a cast membrane. Larger membranes can be assembled from these smaller sections, but discontinuities at the section boundaries can lead to air leakage, and unpredictable actuation behavior. Future revisions may benefit from a less-dense actuator packing (allowing for a thicker, more robust section boundary) or flexures between cells to mitigate PCB

bowing (which can become significant at larger scales, such as our 127 valve array).

In addition, the global cavity, instead of being 3D-printed, could itself be fabricated via a layer-driven approach. (Indeed, commercial manifolds exist that utilize this technique.) Stacking and bonding milled sheets may scale more easily than a purely additive procedure.}

Towards Denser Surfaces

The electronic components that we use — in particular, the opto-isolated solid state relays — determine the minimum pitch of the raw Stoma-Board surface (18 mm). It is worth noting that these particular components are overpowered for our application, capable of switching currents that are orders of magnitude larger than necessary. With more refined electrical engineering, it is conceivable that this pitch could be reduced. However, as the size of the electrostatic actuators decrease, so too does the force. The tradeoffs in this area should be evaluated.

Bidirectional Valves

As noted in Section 4.4, valves in our system are designed to withstand pressure in only one direction. For many applications, such as those demonstrated in Section 7, this is sufficient. However, if bidirectional switching is necessary, a variation of this construction can be used. By adding an additional membrane (Figure 23), effectively placing two flap valves "back-to-back", we can block (and switch) pressure in both directions. Though we have built some early prototypes with this construction, these new valves have yet to be fully characterized — this is an area of future work.

Stoma-Board as a "Raw Material"

Ultimately, we envision programmable surfaces that can function as "raw materials" (i.e. available for broad use by designers, not necessarily engineered for one particular application). An LED strip, for example, can be purchased, cut, and applied to an object — in the hands of a user, it is

closer to a roll of decorative tape than a flexible circuit board. Such a future may be possible for tactile programmable materials as well, but not within the constraints of a module-driven design.

We see the four characteristics of these layer-driven programmable surfaces (outlined in Section 1) as branching points for subsequent research. For instance, while cell communication within Stoma-Board is neighbor-to-neighbor, the original messages are still generated upstream — a truly information-responsive material could perform this act internally. Future implementations might benefit from a transition towards fully distributed computation. Onboard pressure sources, such as electrostatic or piezoelectric micropumps, can also bring this closer to becoming a fully untethered programmable material.

Conclusion

We observe that a layer-driven fabrication process, characterized by patterning and stacking, can give rise to programmable surfaces that may scale more readily than their module-driven counterparts. Instead of building and assembling actuators individually, actuators in our framework are formed all at once, during the layup process. The resulting cells are composite mesostructures, cross-sections of which are distributed across the functional layers of our material.

We introduce Stoma-Board as a programmable surface that exemplifies this model. In this particular implementation, the actuator cells are electrostatic valves, which communicate digitally with their neighbors, and regulate the flow of air between a pneumatic transducer layer and a global reservoir. The valves themselves are a novel structure that can resist human-scale pressures of 80 kPa (an order of magnitude larger than earlier variants). As the cells are ultimately responsive to information sources upstream, we are able to use Stoma-Board as a "raw material" to construct several dynamic, pneumatically-powered devices.

Moving forward, research in discretely-actuated programmable surfaces may benefit from this layer-driven lens, adopting composite fabrication techniques and distributed computation paradigms from materials science and robotics.

IV. Sculptable Mesh Structures for Large-Scale Form-Finding

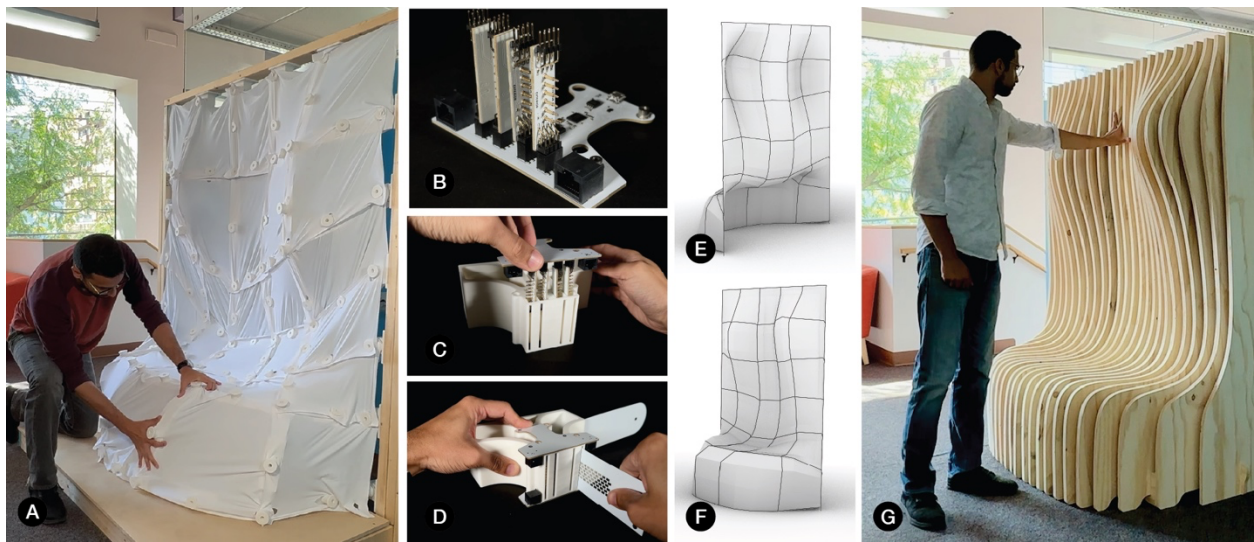


Figure 1. A large-scale mesh structure (a) composed of user-adjustable, flexible beams. A person can manipulate the structure by hand in order to sculpt the mesh into different forms. Each mesh edge is equipped with a length sensor (b-d), and the modules can transmit their configuration to an external computer, where the geometry can be reconstructed (e-f). The now-digitized geometry can serve as input to computer-controlled manufacturing processes, resulting in permanent, room-scale installations that have been “sculpted” by non-CAD users (g).

1. Introduction

From architecture to engineering, computers allow us to construct physical forms that define our environments and captivate the eye. In the modern world, digital tools touch nearly every aspect of this process: from parametric modeling, to global file-sharing, to toolpath generation for computer-controlled fabrication. But the initial steps of a physical design — form-finding in particular — often benefit from a tactile approach that software cannot always provide. Designers need to assess how real-world

objects will integrate with their creations with respect to both usability (“Is the luggage rack too high?” “Do I have enough leg room?”) and aesthetic quality. For this reason, practitioners will often make low-fidelity prototypes — many of them — out of foam, paper, or clay¹⁷⁶. These mock-ups deliver practical insights in real-time, and in some cases (e.g. clay) allow the designer to make in-situ adjustments¹⁷⁷.

This hands-on step, however, is still quite siloed from the rest of the computational design pipeline. To bridge this gap, we introduce a room-scale, “shape-aware” mesh structure that can be hand-sculpted into a variety of forms — and can transmit its configuration to an accompanying software system in real-time. The structure itself is a flexible gridshell with adjustable-length members, capable of composing both single- and double-curved surfaces. It serves as a tangible interface to computer-based tools.

Interacting with our system is straightforward: a user walks up to the mesh and physically manipulates the edges and vertices, until they are satisfied with the resulting shape. Because the members are flexible (constructed from thin fiberglass panels), our mesh is well-suited for modeling curved, “organic” surfaces (Figure 2). Resistive length sensors, embedded into each member, are used to capture this new geometry, smoothly connecting it to the full power of a computational modeling pipeline.

This type of direct interaction, outside of a computer, has a number of benefits. For user researchers, it allows for rapid iteration of tangible prototypes, combined with the state-saving advantages of a digital tool. For non-expert CAD users, it opens access to surface modeling, facilitating the co-design of architectural spaces.

Concretely, our contributions are:

1. An extensible, room-scale mesh surface composed of user-adjustable, “length-aware” members.
2. An energy-minimization routine for modeling bending-active structures composed of variable-length, flexible beams.
3. An end-to-end demonstration of the workflow that our system enables, in the context of an architectural design and construction task.



Figure 2. A user can shape our flexible gridshell structure by hand (a-c), as real-time sensor data is captured. For instance, a planar mesh (d) can be sculpted into an organic bench form (e). Since the members of this mesh are equipped with length sensors, the geometry can be reconstructed digitally (f,g) and fabricated (h).

We particularly intend for our system to serve as a form-finding tool for architects and designers. A typical studio will have many tools for converting digital models into physical artifacts (laser cutters, 3D printers, etc.); but the physical-to-digital workflow is less well-established. Our system adds another tool to the studio's arsenal: allowing designers to sculpt tangible forms, evaluate the prototypes on-location, and then use the sensed geometry to construct permanent installations.

More broadly, HCI researchers have been increasingly interested in advancing tangible, shape-aware interfaces that provide direct manipulation capabilities and real-time feedback during the design process^{178–180}. We advance this space by introducing a system that combines multiple deformation capabilities (bending, length adjustment, and rotation) within a single unified interface.

2. Background

Our approach bridges multiple research domains: design processes, tangible interfaces, and digital twins. To properly situate our work, we first examine the broader context of physical prototyping in design before addressing the specific gap our system targets.

Physical Prototyping in Design Processes

Design processes typically progress through phases of increasing fidelity and specificity¹⁸¹. Within this progression, physical prototyping serves diverse purposes that vary by design stage and objective. Early explorations may use physical artifacts to inspire ideation, while subsequent stages might employ them to test functionality¹⁸². These low-fidelity iterations are important for understanding spatial relationships, ergonomics, and user interactions in ways that digital models alone cannot convey^{183,184}. While architectural designers do often build smaller scale models of their designs, the complexities of human body-scale interactions with architecture¹⁸⁵ can be much more directly investigated with full-scale models¹⁸⁶.

Crucially, designers receive feedback from the properties and constraints of physical media. This material “backtalk” often guides design decisions¹⁸⁷ in ways difficult to replicate in purely digital environments. (For instance, when an architect manually bends a thin sheet of metal to explore a curved roof form, the physical resistance suggests natural resting states that inform the final design.) This type of embodied knowledge acquisition has been recognized as fundamental to architectural thinking¹⁸⁸.

We do not position our system as a replacement for all physical prototyping methods, many of which serve important purposes outside our scope. Instead, we focus specifically on the transition point between form-finding and digital capture — a persistent challenge in design workflows that has been identified as a significant bottleneck in the creative process¹⁸⁹.

Digital Twins and Shape-Aware Materials

Digital twins pair physical objects with virtual representations¹⁹⁰ — in our case, embedded sensors capture the mesh's geometry for software reconstruction. Traditionally, digital twins are used in the latter stages of the product life cycle, to track usage patterns¹⁹¹ or mechanical wear-and-tear¹⁹². Our work instead leverages this concept earlier in the design process, bringing real-time digital capture into the initial prototyping phase.

In the HCI community, researchers have explored similar ideas through the development of “shape-aware” materials and structures¹⁹³. These are physical objects that detect user manipulation (i.e., extension, compression), and can stream this data to a connected computer. This can be accomplished with directly-embedded sensors¹⁹⁴, or by leveraging proxy objects¹⁹⁵ that work in conjunction with external cameras to track deformation¹⁹⁶.

These structures have proven valuable as creative input devices¹⁹⁷. In puppetry applications¹⁸⁰, for example, users can control digital characters by manipulating sensor-equipped “Tinkertoy-like” modules. Advances in piezoresistive materials¹⁹⁸ have further expanded these possibilities, enabling sensors to detect subtle physical interactions like squeezing or bending, and update digital models in real-time. Subtractive design approaches¹⁹⁹ explore how digital models can update in response to physical cutting or carving.

Particularly relevant to this work are devices designed to measure curves. ShapeTape^{200,201} is an early example: a handheld rubber strip with fiber-optic bend sensors, which a user can manipulate in order to model and capture curves in 3D space. More recent methods have leveraged flexible circuit boards with onboard IMUs²⁰² (to capture twist), and capacitive sensing of shifting, offset electrodes²⁰³.

This has also led to the development of physical tools that also have a degree of spatial awareness. HandSCAPE²⁰⁴ is an orientation-aware digital tape measure that captures both distance and direction, enabling users to digitize field measurements. Similarly, the SPATA tools²⁰⁵ provide bidirectional transfer between physical and virtual measurements through actuated calipers and protractors.

Further leveraging this physical-digital link, researchers also have created tabletop construction kits that augment hands-on prototyping at the centimeter-scale. Systems like StrutModeling¹⁷⁸ and NurbsForms¹⁷⁹ allow users to assemble deformable physical components, which generate corresponding digital models.

Architectural design, however, often benefits from whole-body interaction and collaborative shaping. Our work scales up shape-aware principles to room-sized surfaces — addressing challenges in sensing and mechanical design to enable multiple users to physically sculpt and evaluate their creations at room-scale. This allows for the kind of intuitive, hands-on exploration that is crucial for processes such as participatory design.

Participatory Design in Architecture

Participatory design actively involves stakeholders in the design process²⁰⁶. Instead of designing *for* users, architects using participatory techniques will design *with* users, acknowledging that communities have unique insight into their own needs and daily patterns of life²⁰⁷.

When spaces are designed this way, communities often feel greater ownership over the final result²⁰⁸. This collaborative approach can strengthen the connection between people and place. It has been shown to be an especially effective way of designing schools²⁰⁸, housing developments²⁰⁹, and public areas²¹⁰.

The success of participatory design hinges on clear communication between architects and community participants. Here, physical prototypes serve as crucial tools. While architectural drawings and digital renderings can be challenging for non-professionals to interpret, physical prototypes provide an intuitive medium for spatial exploration. These tools bridge the expertise gap between architects and community members²¹¹ —helping to focus discussions and test design ideas¹⁸¹.

Particularly significant are “experiential prototypes” — physical models that allow people to directly interact with and experience spatial concepts²¹². Full-scale mock-ups enable users to directly experience spatial relationships, helping them evaluate factors like ceiling heights, circulation

paths, and furniture arrangements that might be unclear in traditional drawings²⁰⁶.

By providing a tangible interface for spatial design while simultaneously capturing digital information, our system addresses a specific challenge in participatory design: how to involve community members meaningfully while maintaining a connection to the computational design pipeline that dominates modern architectural practice. This integration of physical exploration with digital design tools has the potential to enhance the participatory design process, making it more accessible and effective for everyone involved.

3. Room-Scale Form-Finding with an Interactive Mesh

Our large-scale, interactive gridshell is shown in Figure 2. This is a mesh-like structure composed of bendable fiberglass members, whose lengths can be manually adjusted by a human operator. By manipulating these members, users can “sculpt” the mesh, transforming it from an initial configuration (i.e a flat plane) to a new one, analogous to how an artisan molds a block of clay. Though not load-bearing, the resulting shapes are stable, serving as a physical wireframe that designers can use to evaluate their forms. Onboard sensors detect the lengths of these wireframe edges, allowing a paired software system to capture and reconstruct the sculpted geometry for later use.

We envision an interaction workflow that follows the general pattern below:

1. First, the user chooses an initial topology. In our examples, this is a quad mesh, bounded on three sides by a rigid frame.
2. Then, the user physically adjusts edges and vertices, while the accompanying software periodically saves the intermediate forms.
3. Finally, the user leverages the digitized form to augment their design process. They may choose to review “snapshots” of their sculpted geometries, or import the digital mesh into a CAD tool for further processing.

We detail this workflow in the following section, where we explore how a group of architects might use our tool to complete a small design and construction task.

From Physical, to Digital, to Physical Again

Consider the following example: a local library is undergoing a large renovation, and the architects involved would like to engage the community in the design process. They bring a few participants into the building, and direct them to an area with a large mesh wall. Here, the participants are instructed to shape the space into a structure that suits their needs, experimenting with different forms.

Looking towards a nearby window, one participant notes that this site receives a large amount of sunlight. They think it would be great spot to keep some plants, and begin shaping the mesh accordingly (Figure 3). First, they pull out a section of the mesh and flatten it, turning it into a shelf. Then, inspired by the manner in which the plant leaves hang, they sculpt a second, lower shelf beside it. They choose to have this shelf taper off, so as not to obstruct the nearby stairway. Because this low-fidelity prototyping is taking place on-site, the participants can use features of surrounding environment (e.g. windows, sunlight, stairway) to stimulate and inform their design.



Figure 3. A person, inspired by the sunlight from a nearby window, sculpts a couple shelves to hold plants. The mesh holds its form and can support lightweight objects (such as the plants in b and d) without additional reinforcements.

Another participant decides to sculpt a small alcove, reasoning that children might enjoy hiding within the nook (Figure 4). After an initial "draft", other participants sit inside, testing the fit. One participant feels that the nook is too shallow, so they drag a piece of the mesh outward, enveloping themselves for a cozier atmosphere (Figure 4c). The architects take note of this and save the configuration digitally.



Figure 4. One person approaches the mesh wall and sculpts a child-sized alcove (a,b). A second person sits inside and modifies the alcove. She pulls the mesh around her so as to cover herself more fully (c), in an attempt to create a cozier environment (d).

The site is near the entrance of the library. Some participants point out that they often see people standing in this area, waiting for friends or family to arrive. They suggest that some additional seating could be useful here, and begin to sculpt a bench (Figure 5). The participants also add a wave-like pattern to the wall, which they think compliments the shape of the seat.



Figure 5. In its default state, our sculptable mesh surface is a flat plane (a). A person can pull on the edges and vertices to create forms such as a bench (b,c). They can use physical items, such as a stool (f) to interact with their creations. In this example, a person notices that their sculpted bench is uncomfortable (d), and so they make a modification (e).

The mesh itself is not meant to be load-bearing, so to test the bench, one participant grabs a nearby stool and places it inside of the mesh structure (Figure 5f). When they sit down, they discover that the sculpted wave bulges out too far, and is uncomfortable. Though it appeared fine at first, physical testing made this design flaw apparent. The participant pushes the

bulge inward, creating a cavity that they can lean back against (Figure 5e). They note that this would be a nice spot to sit and read a book. Ultimately, the participants conclude that — this being a library — the reading bench is the best choice for the space.

End-to-End Workflow

With the community participants having finalized the bench design through physical sculpting and testing, the architects now proceed to transform this temporary prototype into a permanent installation. This process demonstrates how our system bridges the gap between intuitive physical prototyping and precise digital fabrication.

Since each edge of the sculpted mesh is equipped with length sensors, the architects can capture the exact geometry through a software interface connected to the mesh. Length readings for all modules are continuously streamed to Rhino CAD software via this interface. The physical adjustments that the participants made (the height of the bench, the waves on the wall, the sculpted cavity) are preserved in a digital wireframe model.

The reconstruction algorithm (detailed in Section 6) processes this wireframe, first by applying energy minimization to establish the correct form, then by fitting cubic splines along each member and generating smooth Coons patches between them. This results in a continuous NURBS surface that captures the essential ergonomic features while creating a clean, fabrication-ready model (Figure 2g).

For fabrication, the architects decide to slice the bench into panels that can be cut from standard plywood sheets — a common fabrication strategy for complex curved forms. Using contour tools in Rhino, they generate evenly spaced vertical sections that, when stacked, will recreate the 3D form. These contours are arranged to maximize material efficiency and include alignment features to ensure accurate assembly.

The prepared profiles are exported as vector files and imported into CAM software for toolpath generation (Figure 6a). After the panels are cut, they are sanded, assembled according to their designed sequence, and press-fit together (Figure 6b,c).

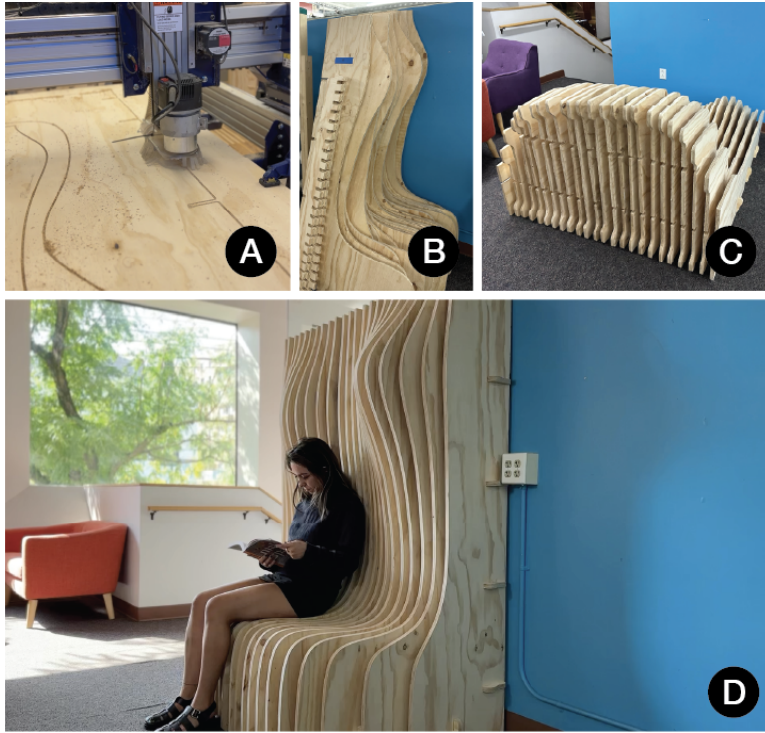


Figure 6. Using data from the length sensors, the sculpted bench is reconstructed in software, and can be connected to the rest of the digital manufacturing pipeline. In this example, the digital mesh is used to create CNC toolpaths (a), which then results in a permanent installation (d).

The completed bench is installed in the exact location where the initial prototyping took place (Figure 6d). The library now has a permanent installation that was collaboratively sculpted by stakeholders in the community.

4. Overview and Mechanical Design

The T-shaped structure shown in Figure 8 is the “unit cell” of our sculptable mesh, and we refer to it as a “T-Module”. It is an adjustable-length beam, made of four flexible FR4 (fiberglass) bands which can slide freely through a plastic guide. A user can pull on these bands to expand a particular edge (Figure 9b), or slide them inward to shrink the edge, tucking the excess material away (Figure 9a).

These modules can be linked together to form a mesh, as shown in Figure 7. Mechanically, the T-Modules are connected with simple fasteners (plastic cylinders, with a metal bolt as an axle). Electrically, they are networked via flexible patch cables in a winged-edge configuration (see Section 5).

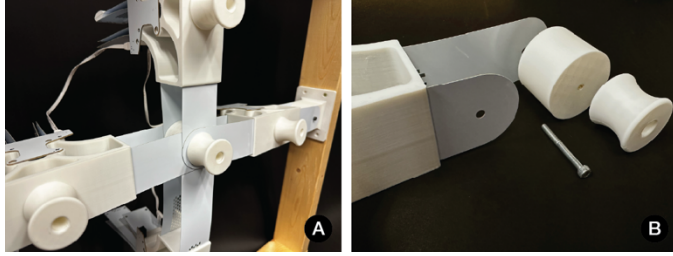


Figure 7. Modules can be joined together at the vertices (a) by using passive mechanical fasteners (b).

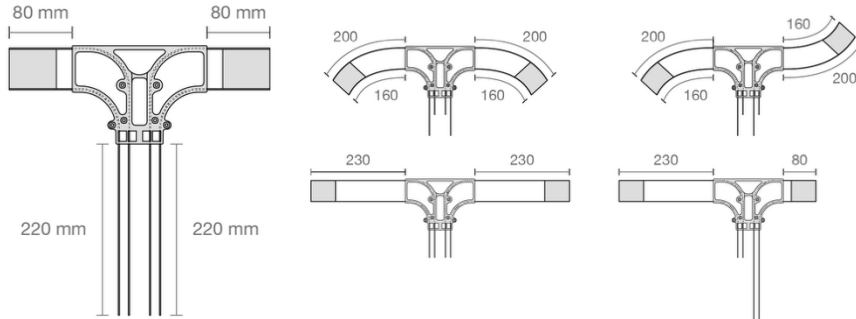


Figure 8. The “T-Module” is the unit cell of our sculptable mesh structure, acting as a variable-length edge. It consists of flexible bands routed through a rigid guide. When fully contracted, the module measures 22 cm from vertex to vertex. When fully expanded, it reaches 56 cm. Excess material is directed towards the interior of the mesh, away from the user.

Individually, the FR4 bands of the T-Module are flexible enough to pass through the plastic guide without much resistance, but together (in a “two-tier” configuration), they form a prismatic structure with enough rigidity to keep the parent mesh from collapsing. Note that for the curved forms in Figure 8, there is a length differential between the inner and outer bands, which allows us to detect the direction of curvature (Figure 9c,d).

The four FR4 strips are circuit boards, 0.8~mm thick. By adding these boards, we transform this passive module into a “length-aware” mesh edge. Electrically, each circuit operates much like a slide potentiometer. The flexible boards are patterned with a resistive code, which, when read by a scanning head at the “neck” of the module, can be mapped to a physical displacement. A microcontroller, connected to this scanning head, records this length data and transmits it to an external computer. (This is done via neighbor-to-neighbor communication within the mesh — the message is passed from module to module until it reaches a pre-selected relay unit). The strips contain no onboard power source — instead, the scanning head sends a small current through the strip's resistor ladder, and takes a reading.

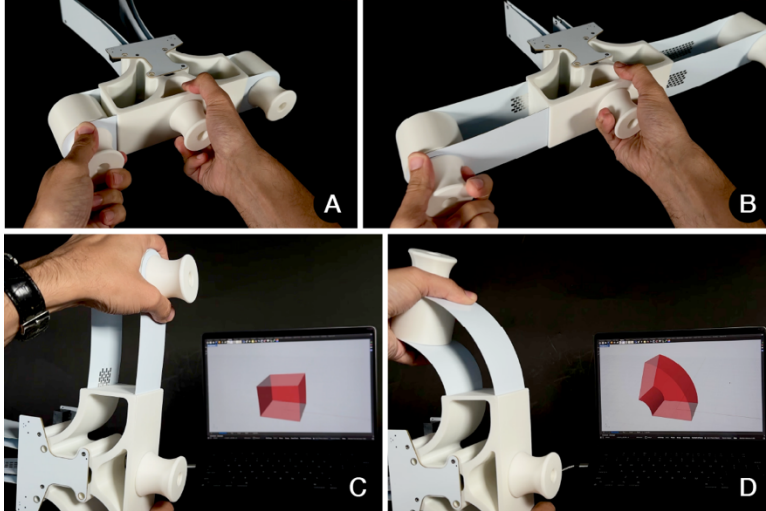


Figure 9. A user can alter the length and curvature of the T-Module by pulling on its handles (a,b). Each side of the module can be adjusted individually. Embedded length sensors capture the module geometry (c,d). By composing many of these members together, we can create sculptable forms that have an inherent digital twin.

The modules are joined mechanically through passive vertices — plastic cylinders with a metal axle. They are joined electrically through flexible, Cat6 patch cables, in a winged edge configuration. Any individual module can serve as a relay device, collecting length data from the parent mesh and transmitting it to our software. (In practice, we choose a module near the outer perimeter to perform this function).

Curvature and Connection Scheme

Our mesh structures mimic a biaxial weave (Figure 10), with four edges meeting at every interior vertex. Compared to other patterns, this configuration gives us a good balance between user manipulability and structural stability. We found that denser structures, such as a valence-6 triangular mesh, were stiff near the vertices and difficult to physically adjust. Sparser structures, such as a valence-3 hexagonal mesh, too often fell into configurations that were underconstrained. A valence-4 configuration gave us a good balance between these alternatives.

Our mesh structure is able to transition smoothly between single and double curvature (Figure 11). By adjusting the length of the flexible members, the surface can "stretch" in ways that would normally cause kinks or distortions in conventional materials. This allows us to adapt to both single-curved and double-curved surfaces, without compromising the continuity of the form.

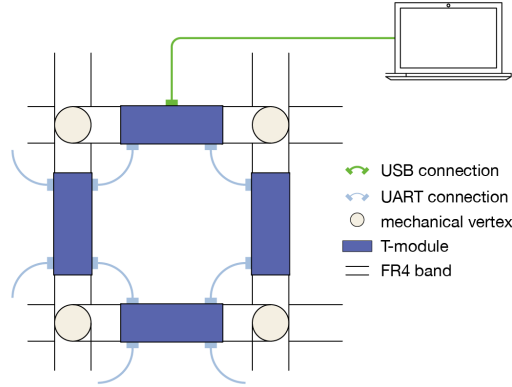


Figure 10. Connection scheme for our sculptable mesh surface. T-Modules are joined to their neighbors using Cat6 cables, and exchange messages via a UART peripheral. One T-Module is connected to a computer, where it transmits the mesh edge lengths to our reconstruction software.

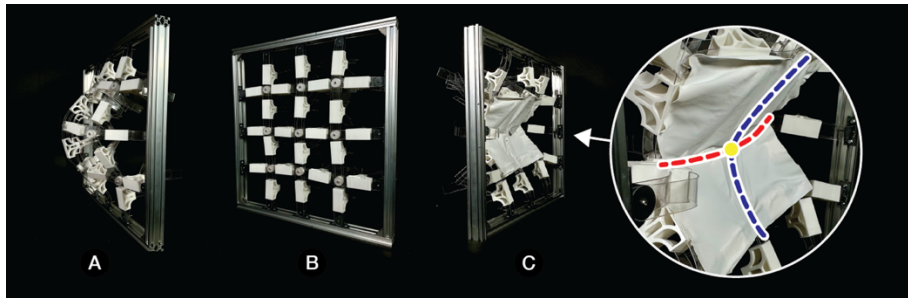


Figure 11. By expanding, contracting, and bending modules, our mesh can transition from a state of zero curvature (b) to a state of either positive curvature (a) or negative curvature (c). We have attached a piece of fabric to the modules in (c) to highlight the saddle point.

Though quad meshes are theoretically susceptible to shearing, we did not experience noticeable issues in our prototypes. This assumes that the boundaries of the mesh are anchored to fixed points, which provides sufficient constraint to prevent unwanted deformations.

Materials and Fabrication

At the core of each module is a 3D-printed, PLA plastic guide. It contains four curved slots, each of which constrain a thin strip of FR4 at a bend radius of 50 mm. This radius was chosen to mitigate plastic deformation — at smaller radii, this deformation becomes noticeable by eye, and introduces additional stresses into the system (making it difficult to model in software). The strips themselves are 60 mm wide and 0.8 mm thick. The inner and outer strips are spaced 50 mm apart.

An early engineering challenge was optimizing the stiffness of the T-Modules. Overly stiff beams limit our ability to form curved surfaces; but

if the beams are too flexible, the sculpted mesh won't hold its shape. Stiffness also impacts the extension ratio of the modules — as the strips pass through the rigid guide, stiffer strips must do so at a larger bend radius. A larger bend radius requires a longer plastic guide, which ultimately reduces the overall extension ratio.

The “two-tier” prismatic structure helps solve this problem: instead of directly optimizing the flexibility of the beam, we can optimize the distance between the outer and inner strips. This allows us to experiment with different stiffness levels while using strips of a constant thickness — a larger distance between strips corresponds to a stiffer beam.

Additionally, using two thin bands instead of a single thick band permits a smaller bend radius at the rigid guide, which in turn increases the dynamic range of the adjustable edge.

For ease of manipulation, we attach plastic knobs to the both the T-modules and mechanical vertices. Once a user adjusts the length of the beam, static friction keeps it in place. (We experimented with mechanical locking mechanisms for the beams, but found friction sufficient for furniture-scale modeling.)

Topology and Anchors

In our current prototype, the user must provide a starting topology — in the form of an adjacency matrix — before we can reconstruct the mesh. The user also specifies which (if any) vertices should act as “anchors”, along with their starting positions. For instance, in the mesh shown in Section 2, the user must specify that the 16 vertices along the top and sides of the frame are anchor points. In the current prototype, the user supplies this information via a C# script. In future versions it should not be hard to do this with a simple interface.

5. Electronics and Firmware

Every active module in our mesh structure is equipped with a small microcontroller (RP2040), which allows for local length sensing and

neighbor-to-neighbor communication. Adjacent edges are connected by flexible cables, which transmit both power and data, allowing us to interface with the system via a single external connection.

Figure 12 shows an interior view of the module. Within, there are four components of interest:

1. A faceplate, housing the microcontroller, sensing circuitry, a USB interface, and two power/communication ports.
2. A backplate, providing two additional power/communication ports.
3. Two pairs of flexible circuits with a resistive ladder, each on an 0.8mm thick FR4 panel.
4. Two pairs of scanning heads, which read a resistive code from the flexible strips. These also facilitate power and data transmission between the faceplate and backplate.

Note that while only one pair of strips (and scanning heads) are required for measuring the module length, an additional pair helps us determine the direction of curvature.

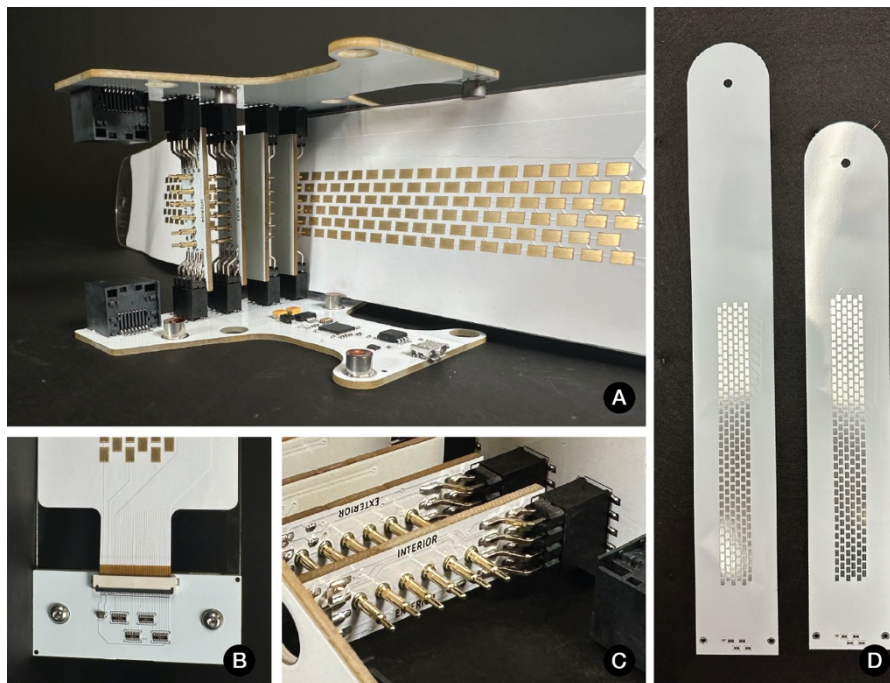


Figure 12. Interior of T-Module (a), which uses flexible FR4 strips (d) as length sensors. The resistive code is read by a scanning head (c). Earlier prototypes used flexible printed circuit boards, attached to a rigid resistor bank (b).

Length Sensing

When our software system first connects to the physical mesh, the length of each edge is unknown. This situation is similar to many machines found in an architecture studio, such as laser cutters or 3D printers, which address the issue by performing a homing sequence upon startup. However, unlike those machines, we cannot expect the user to manually “zero” each variable member before beginning the sculpting process — this would be far too tedious. As a result, absolute positioning becomes a critical requirement.

We need to sense changes in length along each strip, detecting variations of up to 200 mm. Commercial sensors that can achieve this over such a range are either difficult to find or prohibitively expensive.

To meet this need, we implement a resistive ladder on each FR4 strip. The ladder itself is a chain of discrete 100-ohm resistors, connected in series at the bottom edge of the strip. Each connection point between resistors is then routed via a copper trace to an exposed pad positioned along the length of the strip.

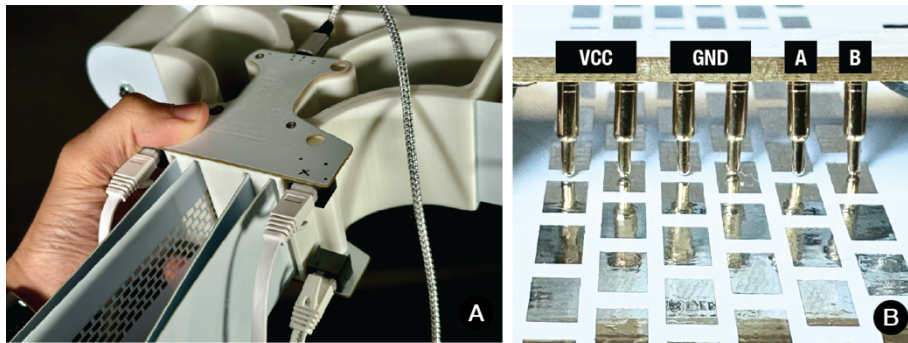


Figure 13. We detect module curvature by sensing the length of resistive strips (b). Having two probes on the scanning head allows us to avoid dead zones on the resistive ladder (Figure 14).

A scanning head with spring-loaded probes maintains contact with these copper pads. It operates much like a slide potentiometer, measuring the voltage at whichever copper pad it currently contacts (Figure 14a). Since each pad corresponds to a specific position along the strip, the measured voltage directly indicates the module's length. This offers an inexpensive method for absolute length sensing, as the resistor ladder can be patterned on a flexible substrate (e.g., thin FR4 or polyimide) over significant distances.

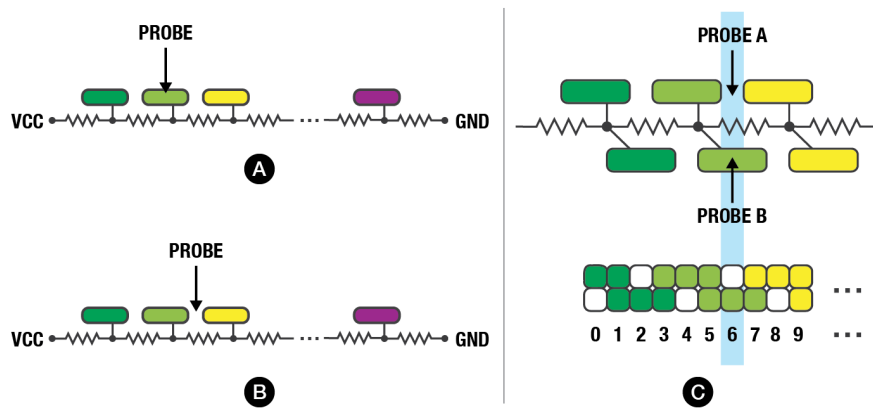


Figure 14. On the left (a,b) we show a standard resistive ladder; on the right (c) we show our configuration. Each node on our resistive ladder is connected to two copper pads, which are offset. This configuration ensures that there are no “dead zones” on the strip — areas between pads where the pins of the scanning head are not in contact with any part of the resistive ladder.

However, using only a single probe can cause issues when the scanning head crosses between pads (Figure 14b), as the probe is left floating. This ambiguity may be acceptable for incremental sensors, but it is insufficient for applications requiring absolute positioning, since the probe may start in an undefined state.

We resolve this by introducing a second set of conductive pads, slightly offset from the first (Figure 14b). In this staggered configuration, at least one probe is always in contact with a pad. This has the added benefit of increasing our resolution by a factor of four. With pad lengths of 6 mm, our resolution improves to 2 mm across a 200 mm active area.

Curvature Detection

With sensors on only the outer strips, we can detect edge length, but not the direction of curvature. For some scenarios, this is sufficient (for instance, if we know that the mesh is against a wall, we can assume that most edges will bend outward when lengthened, and then bias the simulation accordingly). However, in more underconstrained scenarios, we need to use sensors on both the top and bottom strips — which our system does indeed support.

With two probes per strip, we require a total of eight ADC channels to sense the length on all four strips. This actually exceeds the amount available on the RP2040 (which is limited to four), so we use an analog switch (Vishay DG2788A) to de-multiplex the inputs. By comparing the

inner and outer strip lengths, we can determine which way the module bends.

Power and Communication

On the underside of each T-Module, there are four physical ports (Figure 12a) for transmitting power and data between adjacent units. These connections are made through flexible Cat6 cables — although we use UART, rather than Ethernet, to communicate. Each unit passes messages directly to its neighbors, which allows us to easily extend and reconfigure the mesh network.

The RP2040 only features two hardware UARTs, but up to four additional interfaces can be implemented by using the onboard PIO state machines. We leverage all of them, and create four full duplex UART peripherals, so that each T-Module can communicate with its immediate neighbors. These four ports are sufficient, since no module will have more than four neighbors (as we explain in the following section).

Power can be injected at any point in the structure, or at multiple points. This approach helps distribute the load in larger configurations, reducing the current required through any individual cable and minimizing potential voltage drops across the system.

Mesh Representation and Traversal

In our implementation, neighboring modules are defined as mesh edges that (1) share a vertex and (2) belong to the same planar face. This definition allows us to efficiently represent our mesh using a winged-edge data structure²¹³. In this structure, each edge stores references not only to its two endpoints, but also to the two adjacent faces and the four edges that are incident to them (Figure 15a).

In contrast to other mesh representations (e.g. face-vertex or vertex-vertex), this representation provides several advantages for our system. First, it facilitates traversal of the mesh using only information encoded in the edges — important because the edges are the only active elements, capable of sending and receiving messages. Second, this structure supports

efficient updates to the mesh topology, which is crucial as edges are adjusted during the sculpting process, enabling real-time interaction with the physical mesh.

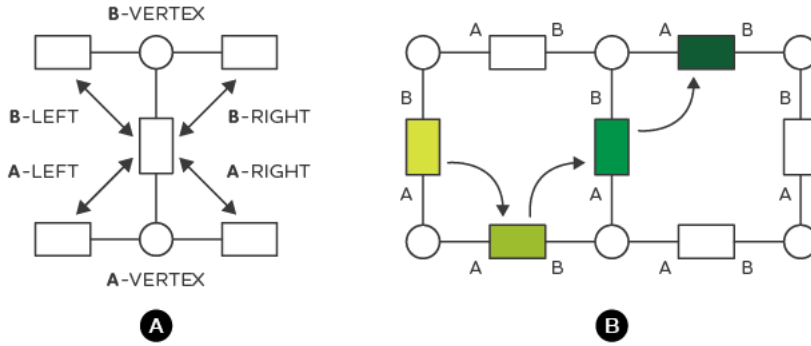


Figure 15. Our mesh can be represented by a winged edge data structure (a). Modules can send messages to each other by passing data from neighbor to neighbor (b). Along this path, modules construct a return address, so the receiver module can send a message back to the sender.

One module sends a message to another by attaching an address — a list of hops — to a data packet and passing it to a neighboring module. Each module along the path tracks the number of hops and forwards the message to the next neighbor specified in the address list (Figure 15b). As the message passes through, each module also notes the port through which it received the message and appends this information to the packet, effectively creating a return address. When the target module receives the message, it can reply to the original sender by following the return address generated during the message's journey.

Physical-to-Digital Link

To ensure smooth interaction between the physical mesh and its digital twin, we must establish a direct connection between the two systems. This process involves linking each physical module to the corresponding edge in the software model, allowing real-time communication and synchronization between the virtual and physical representations of the structure.

First, the user defines the mesh topology in software using a C# script. Next, we need to align the physical mesh with the software-defined topology, meaning each physical module must be mapped to a corresponding edge in the software

For all of our applications, we choose one module to serve as the “relay unit”, and connect it to a central computer via USB. With this module as the root, we first perform a breadth-first traversal of the mesh, obtaining the IDs and addresses (routes relative to the root) of all connected units. After this initialization, the central computer begins polling each module, requesting sensor data. Each polled module responds by sending its ID and length data to the relay unit.

With the physical and digital meshes now synchronized, the user can manipulate the length of each edge in the physical structure, and these changes will be reflected within the digital model.

6. Digital Reconstruction

Once the lengths of each mesh member are known, we can reconstruct the overall geometry to create a digital twin of the physical structure. We approach this as a dynamic relaxation problem, using the Kangaroo physics engine²¹⁴ within Rhino 8 to calculate and minimize the residual energy in our mesh structure.

For the mesh wall demonstrated in Section 2 (consisting of 54 T-Modules), our simulation creates approximately 300 particles. These particles correspond to physical points on the sculpted structure, and their positions are optimized through energy minimization.

At each iteration of our solver, a weighted sum of energies acting on each particle is calculated, a correction vector (pointing towards the lowest energy state) is constructed, and the particles are moved accordingly. This process continues until the system reaches a stable configuration that best matches the sensed physical state.

Complete Reconstruction Pipeline

Our digital reconstruction follows a multi-stage process:

1. First, we initialize the mesh topology with known anchor positions (e.g. the wooden frame in which the physical mesh is suspended).
2. We then perform a coarse solving pass using only length constraints between vertices, establishing the basic shape.
3. Once this coarse mesh is established, we interpolate intermediate points along each member to represent the full T-Module geometry.
4. Next, we add additional energy terms to refine the curvature and mechanical behavior of the mesh.
5. Finally, we iteratively solve until the mesh represents a physically plausible configuration that satisfies our measured constraints.

This approach allows us to capture the general physical form, preserving important distances relevant to human users (e.g., depth and width of a seat, height of an overhang).

Geometric Representation

We represent each adjustable-length member with a polyline consisting of both fixed and variable-length segments (Figure 16). Six points define this polyline, each mapping to a particle in our simulation.

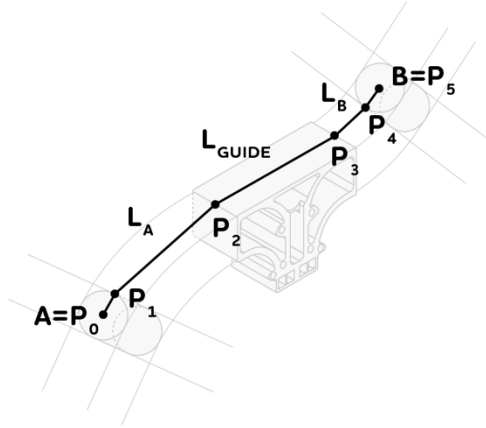


Figure 16. In our simulation, each T-Module is represented by a polyline consisting of six points. The points correspond to particles in our solver.

The two endpoints, p_0 and p_5 , correspond to the “A” and “B” vertices of the T-module. Two intermediate points, p_1 and p_4 , are placed on the perimeters of the mechanical vertices (the plastic cylinders where are T-modules joined together). The remaining two points, p_2 and p_3 , represent the edges of the rigid guide.

In our simulation, particles are shared between mesh members. Specifically, this occurs at the “A” and “B” vertices. For example, p_5 for one mesh member may be p_0 for another. Our solver does not distinguish between these shared particles and simply sums all energies acting on each one, resulting in a cohesive mesh structure.

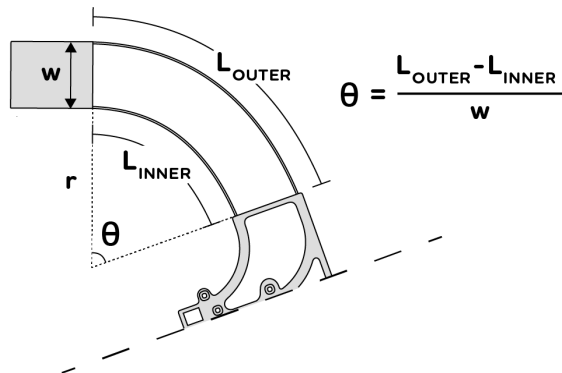


Figure 17. We can compare the measured lengths of the inner and outer bands to detect the direction and amount of curvature.

Energy Terms

Our simulation incorporates several types of energy terms that collectively determine the final configuration:

1. *Length Energies*. These act like springs between vertices, pushing or pulling particles until the distance between them matches the sensed physical length.
2. *Coplanarity Energies*. These maintain local smoothness at mechanical vertices by ensuring surrounding points lie approximately in the same plane.
3. *Bending Energies*. These model the physical stiffness of the FR4 strips, restricting how sharply the mesh can curve.
4. *Anchor Constraints*. These fix certain points in space, typically along the boundary of the mesh.
5. *Bias Energies*. These help resolve bistable configurations by pushing vertices in the correct curvature direction.

These energies work in concert, with length energies primarily determining the overall shape, while the others refine local behavior to match physical constraints.

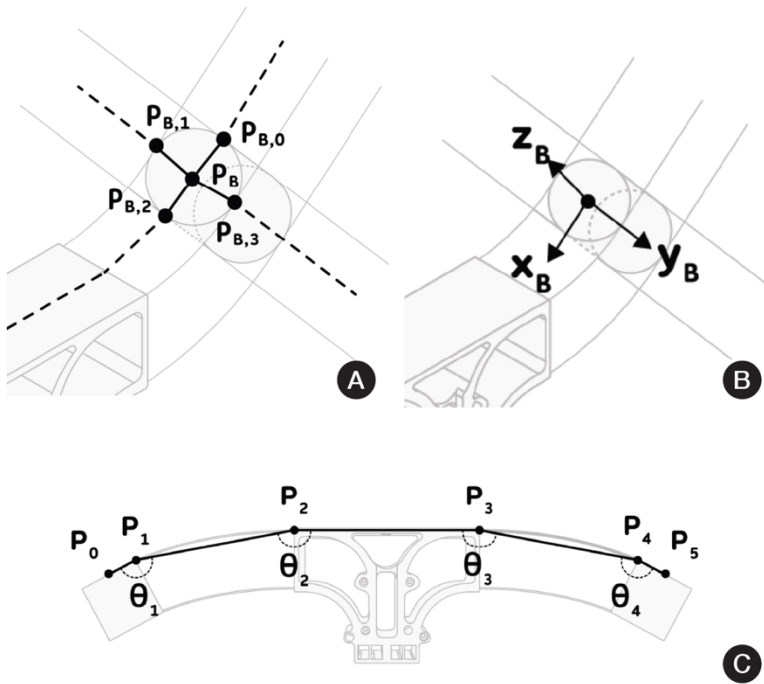


Figure 18. Illustration of energies minimized by our solver. (a) Vertex particles, along with their four immediate neighboring particles, are softly constrained to be coplanar. (b) A reference frame is constructed at each module vertex. (c) A bending energy attempts to minimize the angle between adjacent line segments.

Handling Bistability

When attempting to reconstruct the mesh from length measurements alone, we encounter a fundamental ambiguity: knowing only the distance between endpoints does not tell us whether a curved edge bends inward or outward. This bistability creates a challenge for our reconstruction algorithm.

To address this, we begin with a coarse solver that considers only "length energies" and optimizes just the particles corresponding to the "A" and "B" vertices. These length energies function like springs — when the distance between p_0 and p_5 equals the sensed, real-world distance, the particles exist in a zero-energy state. Otherwise, a correction vector guides the particles toward the correct distance.

However, with length constraints alone, the solver may converge to configurations where areas that should be convex appear concave, or vice versa. When a T-Module is equipped with all four length sensors (both inner and outer bands), we can determine the direction of curvature by comparing these readings (Figure 16). When the outer band is longer than the inner band, we know the module curves away from the outer band.

Once the curvature direction is determined, we apply a constant-magnitude bias force in that direction. This force is "artificial" in that it doesn't model a physical phenomenon but instead helps guide the solver toward the correct solution by breaking the symmetry of bistable configurations. The magnitude of this force remains constant regardless of how much curvature is detected—it merely indicates the correct direction.

After the coarse solver has converged, points p_1 through p_4 are interpolated between p_0 and p_5 as an initial guess for their final positions. We then proceed to reconstruct the curvature by incorporating additional energy terms.

Local Coplanarity

The mechanical vertices of our mesh structure are rigid cylinders, 6 cm in diameter. This physical constraint requires that all points along the

circumference of each vertex remain approximately coplanar. We enforce this with a “coplanarity energy” term in our simulation.

For each vertex, we gather the four surrounding particles and fit a plane to all five points (the vertex itself plus its four neighbors). The best-fit plane is calculated using standard least-squares methods. For each of these points, we generate a correction vector perpendicular to the plane, with magnitude equal to the point's distance from the plane.

When applied to our simulation, this coplanarity constraint has the effect of smoothing the sharp features that may result from the coarse solver. It also effectively “clamps” the ends of the flexible bands, constraining the bending energy that we add in the next section and better reflecting the physical behavior of our T-Modules.

Axial Bending

For every trio of adjacent points in our polyline representation, we calculate the bending stresses using the method described by Adriaenssens and Barnes²¹⁵. This approach models each segment as a rod with internal bending resistance, yielding a shearing transformation that, when applied, drives the member towards its natural curvature.

This method, however, assumes a circular cross-section with a uniform moment of inertia — an assumption not met by our system. To account for this, we constrain the bending to occur only in the plane where the physical module can actually flex, eliminating forces that would attempt to bend the module in physically impossible ways. Specifically, we project the particle correction vectors onto the plane defined by p_0, p_5 , and the average of the vertex normals at those points. While this adaptation simplifies the true mechanical behavior, it captures the essential characteristics needed for our reconstruction.

Post-Processing for Final Form

Once the energy minimization simulation converges to a stable configuration that satisfies our measured constraints, we apply several post-processing steps to create a smoother, more usable digital model.

First, we fit natural cubic splines through the polyline representations of each member. This smooths any remaining discontinuities in the simulated mesh while preserving the overall shape and measured dimensions.

After creating splines for all members, we generate Coons patches between adjacent splines. These surface patches interpolate between the boundary curves, creating smooth surface transitions across the entire mesh structure. Finally, the individual Coons patches are joined together to create a unified surface representation of the sculpted form.

This post-processed model can then be directly imported into CAD software for further refinement or manufacturing preparation, as demonstrated in the library bench example in Section 2.

7. Discussion

Our shape-aware mesh system enables low-fidelity form-finding through tangible interaction, serving as an important link between hands-on prototyping and the digital manufacturing pipeline. While our current implementation demonstrates the potential of this approach, there are several areas where future work could enhance functionality and expand applications.

Limitations

Perhaps the most visible limitation of our system is resolution. While our tool is well-suited for room-scale applications (e.g., benches, alcoves, facades), more detailed forms may demand a larger number of small-scale modules. The current module size creates a tradeoff between workspace coverage and feature resolution — a challenge when designing objects that contain both broad surfaces and fine details.

Another consideration is the material properties of the mesh. Although the flexible FR4 panels used in our structure are well-suited for modeling curved surfaces, they are subject to small amounts of plastic deformation when held in place over time. This did not noticeably affect the shapes we were able to sculpt during our testing, but it is worth noting as a potential

long-term failure point, particularly for installations that might remain in a fixed configuration for extended periods.

Finally, the accuracy of our reconstruction routine is contingent on several assumptions about material behavior. In practice, small inconsistencies in the flexibility of the members, or external forces like gravity, may cause deviations from the predicted shapes.

Comparison with Existing Digitization Techniques

Unlike real-time 3D scanning approaches that capture surface geometry through point clouds, our shape-aware mesh provides direct structural information through edge length measurements. This distinction offers several advantages: our system eliminates occlusion issues common in optical scanning, requires no post-processing to clean up point cloud data, and seamlessly integrates with CAD workflows through a direct data pipeline. Additionally, while scanning approaches passively record geometry, our system actively captures designer intent through direct manipulation of structural elements, creating a more deliberate bridge between physical intuition and digital representation.

Furthermore, traditional form-finding techniques typically operate as one-way processes, either moving from physical models to digital representations (through scanning) or from digital designs to physical models (through fabrication). Our approach establishes a continuous feedback loop where physical manipulations can be immediately visualized digitally, evaluated, and refined. This bidirectional relationship transforms the mesh from a simple digitization tool into an interactive medium that combines the tactile benefits of physical prototyping with the precision and iterative capabilities of digital design.

Future Directions in Hardware and Sensing

To address the resolution limitations, future implementations could incorporate modules of different sizes within the same structure --- allowing for higher-resolution patches alongside coarser features. This multi-scale approach would enable designers to focus on detail where

needed, while maintaining manageable complexity in areas requiring less definition.

Alternate materials (with better fatigue resistance) could be explored as well. Composites that combine the resistive properties of our FR4 strips with improved mechanical characteristics might offer an ideal balance between sensing capability and physical behavior.

Lastly, the neighbor-to-neighbor communication in our mesh could potentially introduce latency for structures that are larger than room-scale (as the number of hops increases). A potential solution is to introduce localized hubs that all share a communication bus, and then hop from hub to hub instead of module to module.

Topologies and Configurations

Although our current implementation requires users to predefine the mesh topology, our system is also capable of supporting automatic topology discovery. By leveraging the communication capabilities of the modules, each unit is able to identify its connected neighbors by exchanging messages and recording connection points. Future implementations can leverage this existing feature — dynamically constructing the mesh topology without manual input, and streamlining the user experience.

Beyond flat meshes, our system can support various starting geometries, such as the cylindrical configurations in Figure 18. Future work could explore more complex initial topologies: spheres, toroidal surfaces, or other structures — expanding the design space for architectural applications.

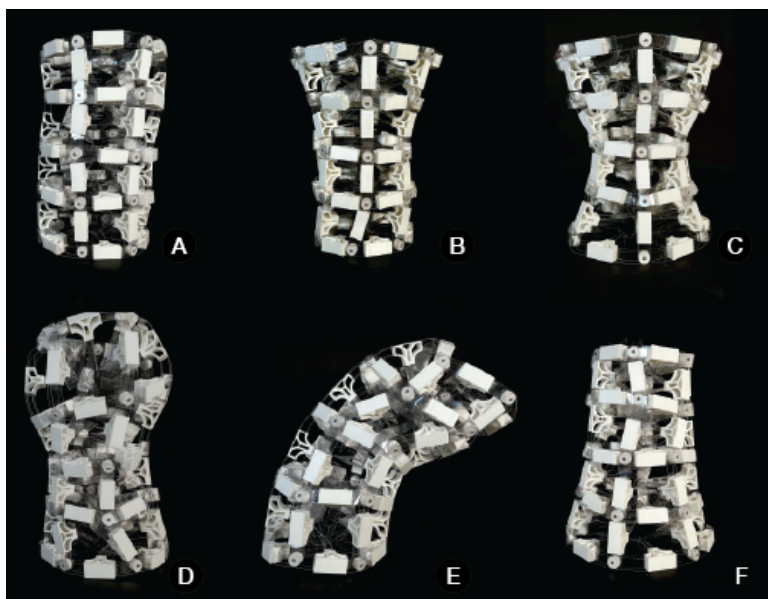


Figure 19. Our modules are capable of starting from geometries other than flat planes. Here we explore various forms based on a cylinder.

Bidirectional Information Flow

In this work, we have focused primarily on the flow of information from the physical to the digital. Moving forward, it may also be valuable to explore the reverse: enabling mesh structures with actuated, length-changing edges. Such a system would allow a seasoned CAD user to first create a form in software, and then experience it in physical space—perhaps making hands-on refinements when design issues become apparent. Since each module is already equipped with absolute length sensors, closed-loop control of this shape-changing mesh may be feasible without significant hardware modifications.

A system capable of physical-to-digital and digital-to-physical transformations could serve as a platform for exploring the interplay between computational optimization and human intuition. Algorithms could suggest modifications to human-created forms to improve structural performance or material efficiency, then physically manifest these changes for designer evaluation.

Such a system could also allow for collaborative design processes within geographically distributed teams. Architects in different locations could manipulate the same structure — changes made to a physical mesh in one location could be transmitted digitally and reproduced elsewhere (e.g. on an actuated mesh, or in virtual reality).

Applications Beyond Architecture

The system also shows promise for accessibility and inclusive design. For individuals who may struggle with traditional CAD interfaces, a tangible modeling system provides an alternative pathway to spatial design. This could be particularly valuable in participatory design sessions involving diverse stakeholders with varying technical backgrounds, as illustrated in the bottom of Figure 20, where children can directly engage in designing play spaces without needing to understand scale models or interpret virtual renderings.

In museum and exhibition design, our system offers unique advantages for working with difficult-to-digitize objects. As shown in the top of Figure 20, exhibition designers can arrange physical items like plants directly on a shape-aware surface, sculpting display fixtures around them in real time. The resulting configurations can be immediately captured in digital form, bypassing the typical challenges of 3D scanning organic objects with complex geometries.

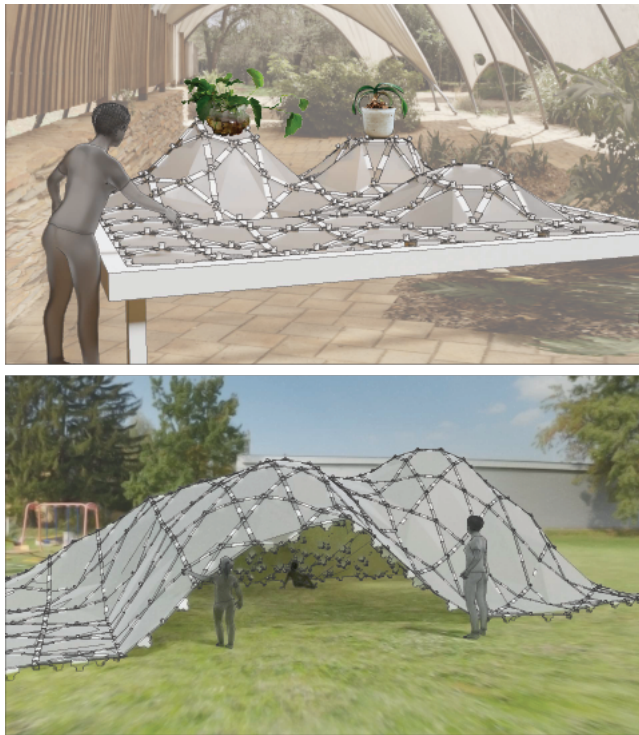


Figure 20. Envisioned uses.
Top: An exhibit designer working with difficult-to-digitize items, such as plants, can arrange them directly on a physical display table.
Bottom: Children can help design a play space without needing to understand scale models or virtual renderings.

8. Conclusion

In this chapter, we introduced an interactive, large-scale sculptable mesh structure that bridges the gap between physical form-finding and digital design processes in architecture. Our system consists of adjustable-length members equipped with onboard sensors that detect changes in edge lengths, allowing for real-time digital reconstruction of the sculpted geometry. This enables architects, designers, and even community participants to physically manipulate mesh structures and immediately see the impact of their changes in a digital model.

Through a fictional participatory design scenario involving a library renovation, we demonstrated how users can intuitively shape the mesh to create various architectural elements, such as alcoves, shelves, and seating areas. The immediate physical feedback provided by this low-fidelity prototyping allows users to test and refine their ideas on location. Physical adjustments made to the sculpted structure can be captured, reconstructed by our energy-minimization routine, and subsequently imported into CAD software for post-processing and fabrication. Through the type of enhanced user engagement that our system facilitates, we hope to further enrich the built environment.

V. Constraint-Driven Robotic Surfaces, at Human Scale

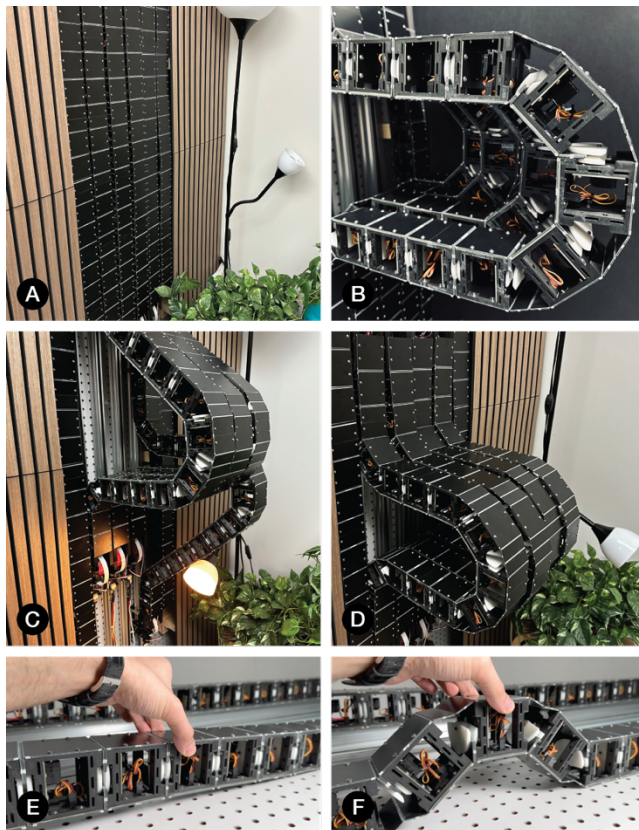


Figure 1. By constraining the cells of our robotic surface in various patterns, we are able to create robust, human-scale structures that "pop out" of a living room wall (a-d). These constraints are added and removed through the computer-controlled rotation of a rigid clamp, which adds mountain and valley folds (f) to an otherwise rigid surface (e). Once constrained, a squeezing force (located at the top and bottom edges of the structure) pushes the cell columns out-of-plane, and locks them in position. Our surface can render shapes in a few seconds or less, allowing for the possibility of real-time user interactions.

1. Introduction

Shape-changing structures enable tangible interfaces and interactive environments. Since their form, and hence function, can be established under computational control, they offer interesting new possibilities for highly adaptive objects. Fittingly, these shape-shifting surfaces have a

diverse history of implementations. To name a few: 4D-printed structures, that deform in response to an external trigger (e.g., heat⁶⁸); foldable origami-inspired materials, that can be bent into 3D shapes using embedded actuators (e.g., shape memory polymers¹⁶); and robotic constructs known as shape displays^{55,216–218}, which consist of many (perhaps hundreds of) actuators that drive the change in shape by repositioning their parts.

Such shape displays offer the highest flexibility in terms of the number of shapes that can be formed. However, this comes at a cost. Beyond requiring an actuator (e.g., motor) at each configurable bit, the actuator must be strong enough to hold its position under load. Such actuators are not only pricey, but also power-hungry, which further require expensive equipment, such as sophisticated power supplies. To mitigate the cost and implementation issues, the demonstrated capabilities are often limited to low-force and often small-scale applications (data physicalization²¹⁹, or manipulation of lightweight, centimeter-scale objects).

In this chapter, we propose a shape-changing robotic surface that is designed for large-scale applications, but still uses small, power-efficient actuators to reconfigure itself dynamically. The key to our material is this: instead of driving the shape change at many points (i.e. at each hinge of our structure), we simply constrain its shape, and relocate the transformative actuation force to the outer edges of our surface.

Constraint-Driven Robotic Surfaces

We introduce a reconfigurable surface which is able to begin in a neutral — in this case flat — state, and then under computational control, assume a desired functional shape. The material can later be returned to its neutral state, and then assume a completely different functional shape. Our surface works at human-scale — larger than the typical hand-scale of many HCI devices, but still designed to fit alongside room-scale structures. It uses a series of actuated columns each made up of a linear collection of identical cells, with bending joints between each pair of cells. Rather than directly actuating each bend, which would require powerful, heavy, and expensive motors, cells establish constraints on the bend angle at each joint using small, lightweight, and inexpensive actuators. Once constraints have been established, bending actuation is then provided by a

simple but more powerful system which presses from the ends of the columns in order to achieve and hold the final desired shape.

The work presented here details the design and implementation of our material, which we refer to as a "constraint-driven robotic surface". We include techniques that make partial use of a layered fabrication method which simplifies construction. We consider a number of example functional forms that can be achieved, and present some simple technical evaluations that test strength and stiffness of the material when deployed.

Contributions

The main contribution of this work is the concept of constraint-driven robotic surfaces. We make the following specific contributions:

1. A technique for surface reconfiguration through programmable constraints.
2. A novel structure for a bendable column, and the associated fabrication method.
3. An exploration of the application space, within the context of interactive architecture.

2. Living with Robotic Surfaces

Lee is an imagined person, living in a small apartment full of robotic surfaces. Their space is too cramped for roving robotic agents, but nevertheless, Lee's apartment is filled with functionality — neatly concealed within the walls and floors of their home. Though Lee's story is fictional, the shape-changing surfaces that they will encounter are all realizable by our system today.

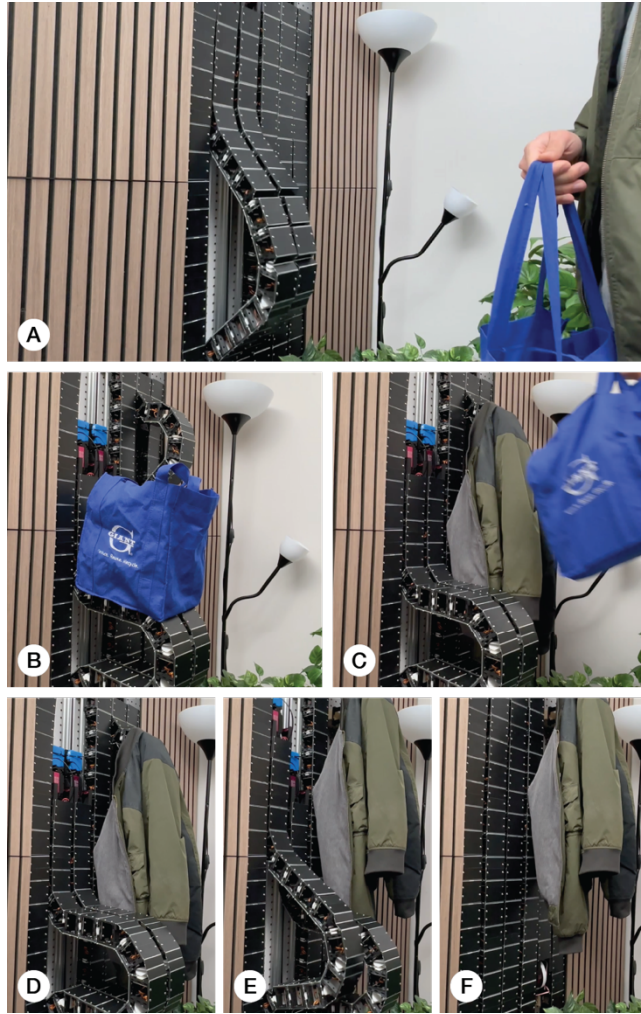


Figure 2. From the wall, a small shelf emerges (a), able to support loads such as the grocery bag in (b). This temporary holding spot allows a human user to (for example) remove their coat, and hang it on a hook that pops out of the wall (c). When the shelf is no longer needed, it folds back into the wall (d,e,f), and the coat hook rises out of the way (f).

Wearing a coat and carrying a bag of groceries, Lee enters their apartment and is greeted by a subtle waving gesture from the wall across the room. Lee would like to take off their coat, but they first need a spot to set down their bag of groceries — fortunately, a shelf emerges from a portion of the wall beside them (Figure 2). As Lee sets down the bag, a coat hook descends from the wall (2b). Lee hangs up their jacket, grabs the groceries again (2c), and goes into the kitchen. As Lee leaves, the coat is raised upwards, out of the way, and the shelf collapses back into the wall, its purpose served (Figure 2d-f).

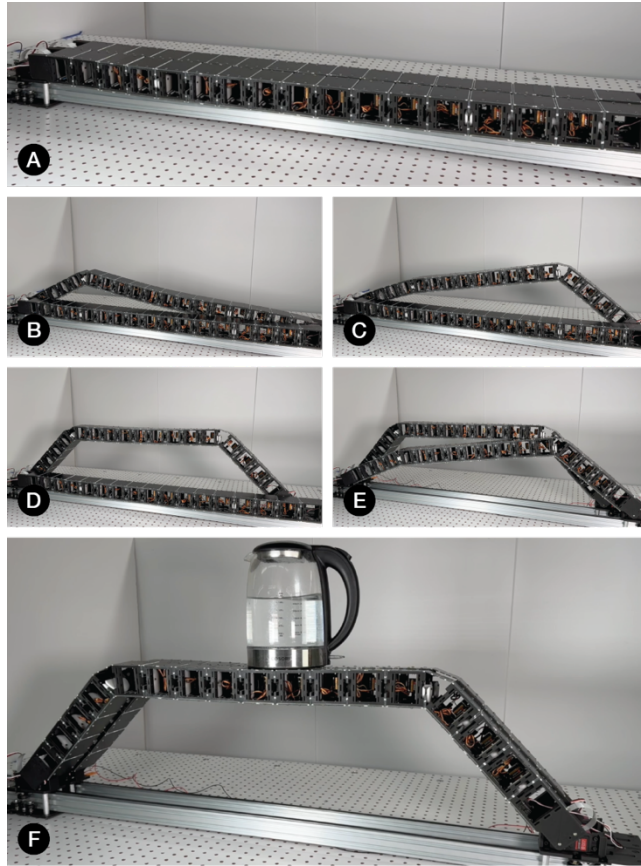


Figure 3. A table rises from the floor, shifting into place in just a few seconds (a-e). It is able to hold the 2 kg pitcher of water (f) without any visible deflection.

A bit later, Lee returns to the living room with a meal in hand, ready to eat. A small table rises from the ground – a bit low, but Lee doesn't mind sitting on the floor (Figure 3). The furniture folds flat once it's no longer in use.

Lee has some work to catch up on, so they turn back towards their wall and solicit a small desk for their laptop (Figure 4). Lee has been sitting for a while and would like to work while standing up, so the wall adjusts the height of the desk.

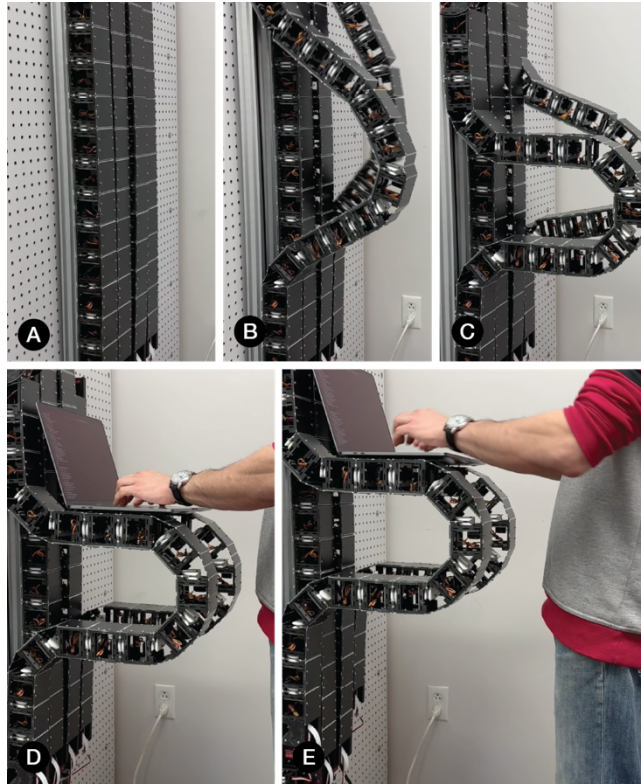


Figure 4. A lightweight desk emerges from the wall in seconds (a-c), easily able to support a laptop computer (d). In this example, the user would prefer to work while standing, so the desk raises the laptop to a comfortable height (e).

Afterwards, Lee decides to settle in for some reading. As they lean against the wall behind them, the vertical surface transforms into a reading nook (Figure 5). An "appendage" emerges as well, angling a nearby gooseneck lamp to illuminate Lee's book for the evening (5b,c).



Figure 5. Our robotic surface forms a small canopy — a person can sit underneath, and use it as a cozy reading nook (a). To signal the user to engage with the space, one column of the robotic surface forms an appendage and pushes a nearby gooseneck lamp (b-d), angling it downward and illuminating the seating area.

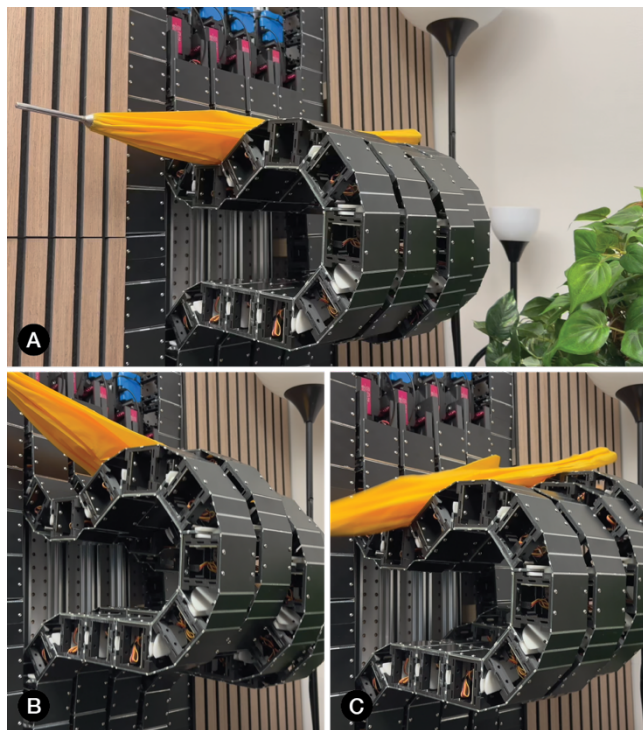


Figure 2. A physical ripple effect (resembling a stadium wave) propagates across the wall, reminding a nearby user to grab their umbrella. The makeshift shelf has a small divot to cradle the umbrella.

The next day, when rain is forecast, Lee's wall reminds them to take their umbrella with them before they leave (Figure 6).

The robotic functionality in Lee's apartment is seamlessly integrated into their environment — it lies in the space between "material" and "machine". Such surfaces, however, are difficult to construct if we insist on directly actuating every joint. If we want something that scales as a material does, then we instead might want to examine a constraint-driven solution.

3. Background

There is an ongoing fascination, shared by many researchers, around bringing the dynamism of software into the physical world^{131,220}. This is not a straightforward task — physical objects are much harder to reconfigure than digital ones. A survey of this work can be found in Chapter 2 of this document — in this section, I focus particularly on implementations that leverage “snake-like” structures.

Serpentine and Continuum Robots

Many researchers look to the natural world for insights, and in the realm of shape-change, the snake is a frequent source of inspiration. The resulting serpentine robots have earned a good deal of attention, due to their ability to bend and wiggle their way through complex environments⁷⁹. Typically, these robots are built with a motor at each joint, for maximum configurability^{76,77}.

HCI researchers have adopted some of these designs for use as “one-dimensional” shape-changing interfaces such as LineFORM⁸¹ and ChainFORM⁸². But there are challenges associated with directly driving each joint — the structure becomes heavy, and difficult to actuate at greater lengths. As a result, these interfaces are usually designed to sit flat on a table or other surface (as opposed to bending out-of-plane).

Some of these issues are addressed by continuum robots. These designs feature tendons, threaded through a tubular structure, that are pulled to alter the overall curvature of the robot. The powerful actuation (motors that pull the tendons) is moved *outside* of the main structure.

In traditional implementations, multiple curves require multiple sets of tendons. However, some roboticists instead approach this by using one set of tendons, but then altering the stiffness of the robot (i.e. constraining it) at various points along their primary axis^{86,87,89}. Once these continuum robots are appropriately stiffened, the actuating tendon then pulls the robot into a newly-defined shape. This constraint-driven technique is what we adopt and expand upon in this chapter.

4. Building Robotic Surfaces at Human-Scale

It is rare to see a slim robotic arm that can contend with human-scale forces. In most designs, since the end effector is relatively free to move (connected only to the last link of the kinematic chain), power-hungry motors are required at each joint to hold the assembly in position. At the number of junctions increase, the torque demands become greater and greater, adding weight and expense to the machine. Though this is often acceptable in an industrial setting, it is impractical to build a shape-

changing surface in this manner — particularly if we'd like such surfaces to become ubiquitous in the built environment.

A more lightweight and scalable option is to take advantage of joint constraints (often seen in flexible, continuum robots). Under this paradigm, the heavy, powerful actuators are brought to the edges of the surface, leaving only lightweight "constraining" actuators at the joints. This is one of the key ideas in our implementation, which allows us to create robotic materials that can be integrated into the surfaces around us.



Figure 7. Our robotic surface consists of four fully-actuated columns, flanked by two decorative columns (a). Each column contains seventeen double-hinged cells, which can be constrained in either a valley (l), rigid (m), or mountain (n) orientation. These constraints are achieved by rotating a plastic clamp with a low-cost servo motor (e,f). For each cell pair, the clamp prevents either one or both sets of hinges from opening. After configuration, a belt-drive system (g) squeezes the column endpoints together, resulting in a rendered shape (c). This process is illustrated in (h-k), which models unconstrained hinges as fixed-length tendons, and models constrained hinges as ultra-stiff springs. The first and last cells are attached to servo motors (b,d), used for deployment and retraction procedures. All electronics within the column draw power from a shared 6V bus, running on flexible cables that are integrated into the cell structure. However, the servo motors attached to the topmost cell (b) are powered by a 7.4V rechargeable battery, to compensate for a small voltage drop across the column.

System Overview

Our prototype surface is a semi-modular material — it consists of multiple independent columns (in our prototype we provide four, but additional copies can be added), each comprising seventeen double-jointed cells. By rotating a rigid clamp with a low force servo motor, each cell pair can be constrained in one of three orientations: a mountain position, a valley

position, and a coplanar (flat) position. When the endpoints of each column are squeezed together, the structure buckles out of plane, and the column forms a curve that is governed by the cell constraints (Figure 7). Each column functions as a "cross-section" of a larger three-dimensional surface.

The cells themselves are designed with a planar manufacturing process in mind, initially intended to be folded and assembled from a single flat sheet (Figure 8). This sheet is a laminated, FR4-PET composite: the rigid outer layers serve as lightweight skeleton, and the flexible inner layer works as an integrated hinge. (Similar rigid-flex constructions can be seen in Karras et al.²²¹ and Rus et al.¹⁴). For ease of prototyping, we opted to replace the "pillars" of the cell with 3D-printed PLA components, allowing for faster disassembly and reassembly during development (Figure 7f). (These "hybrid" cells are the ones that we feature in all our application scenarios, mechanical tests, and supplementary video.) A single cell is 65 mm on each side, and weighs 140 grams (including all internal components).

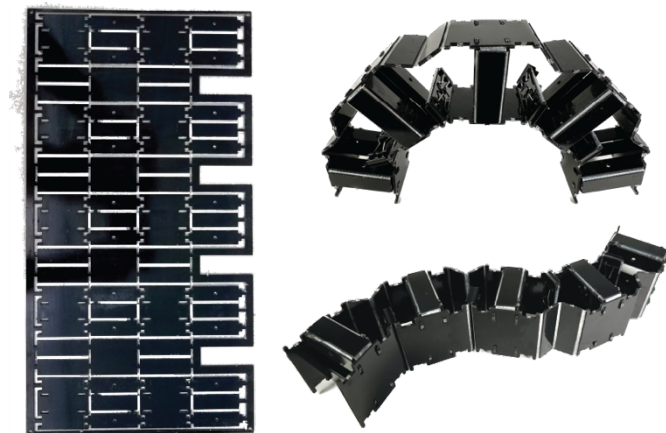


Figure 8. A flat, pre-cut sheet (left) can be folded to form a column of double-jointed cells (right). While ultimately, we used fasteners and 3D-printed pillars in our prototype columns (for quick repairability during the research phase), our design is intended to support this style of foldable construction.

Cell Fabrication and Assembly

Because we are interested in shape-changing surfaces at architectural scales, mass-manufacturability was front-of-mind from the earliest stages of this research. Planar fabrication processes, such as those used in circuit board production (i.e. laser cutting, layup/bonding) are highly scalable, and our procedure is designed to be achievable within the context of this already-mature manufacturing pipeline. Both FR4 and PET are common

materials in this process, which led us to choose an FR4-PET composite as our base material. (The high strength-to-weight ratio is also desirable.)

Cells were fabricated in groups of three, as shown in Figure 9. To begin, two sheets of a heat-activated adhesive, each with a paper backing, were bonded to the front and back sides of a metalized PET film (0.05 mm, McMaster-Carr 7538T11). Bonding was achieved with an off-the-shelf heat press (CREWORKS 15"x15" Heat Press), at 120 C for 3 seconds. After cooling, the paper backing was removed, and the cell pattern was laser cut (Rabbit Laser RL-80-1290) into the PET-adhesive composite. This layer was then sandwiched between two externally-machined FR4 sheets (0.8mm, JLCPCB), along with six flexible flat (FFC) cables (JUSHUO JS05A-30P-100-4-8), and bonded at 180 C for 20 seconds. (To ensure consistent alignment, we found it helpful to add a small amount of masking tape around the edges of the still-to-be-laminated layers.)

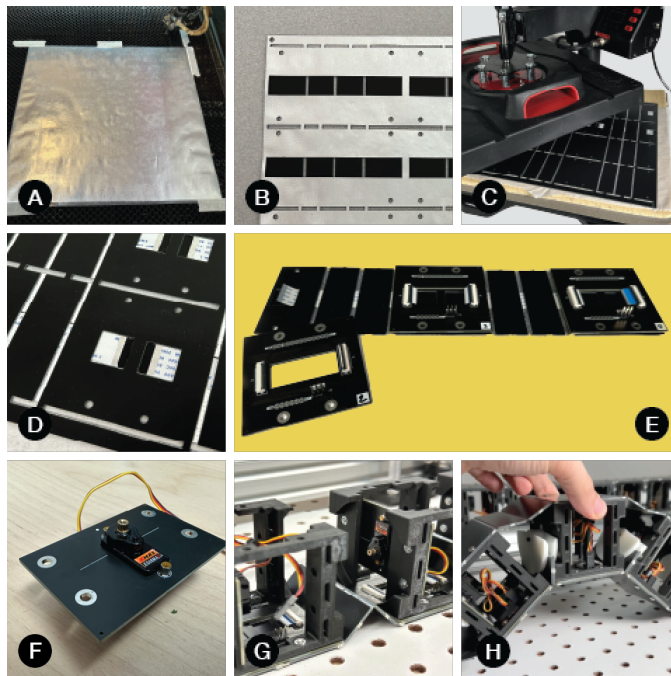


Figure 9. Fabrication process for our double-jointed, configurable columns. PET-adhesive composites are laminated then and laser-cut (a). The patterned sheet is then sandwiched between two FR4 plates (b), alongside several FFC cables, and laminated (C). The completed FR4-PET composite is shown in (d). The exposed FFC cables are attached to a connecting PCB (e). A small servo motor is attached to a mounting plate (f). This mounting plate is attached to two PLA 3D-printed frames, which are fastened to the composite (g). SLA 3D-printed clamps are added, and the structure is finalized (h).

Following the bonding process, cell segments were manually cut from their parent sheets, and the exposed FFC cables were attached to individual circuit boards, joining the cells electrically (Figure 9d,e). Note that each group of three cells contains at least one "half-piece" at its end (Figure 9e), which is connected via a "jumper" board to another group of three cells (mating with a corresponding "half-piece"). At scale, these

extra jumper boards may not be necessary, as a rigid-flex PCB manufacturing process could ensure connectivity without the use of external components²²². Cells in a column all share power connections for the motor-driven clamps, and each cell has a unique data line for receiving servo commands. (We discuss a more extensible, daisy-chaining scheme in Section 7.)

By rotating a passive, 3D-printed clamp (Figure 7e,f), we are able to constrain each cell pair in a either a mountain, valley, or rigid orientation. This clamp is driven by low-cost servo motor (EMAX ES08MA II), which we mount on a small FR4 plate (Figure 9f). This motor plate is attached to two 3D-printed frames, and the entire assembly is then fastened to the PET-FR4 laminates (Figure 9g,h).

Endpoint Actuation

The endpoints of each column (the first and last cell) can be moved axially, squeezing and stretching the chain of cells that sit in-between. A pair of belt-drive assemblies control the endpoint positions, pulling them along a two meter aluminum rail. These belt drives are where our more powerful gear motors (E-S Motor 5840WG-555PM-17-EN) appear. An integral worm gear at the output shaft prevents back driving, so after the endpoints are moved into position, the device consumes no power. (This is important for surfaces that may be used as furniture, and will be static for long periods of time.)

A standard PID controller has some difficulty steering the column endpoints along the rail. Because the shape of the column is constantly changing, there are many non-linearities in the system — the same force from the gearmotor can have different effects on the endpoint movement, depending on both the column configuration and the deployment status. For instance, in Figure 11b, the column starts in a "stiffer" state, closer to the vertical than in Figure 11c. As a consequence, the endpoint in (b) does not move as far as the endpoint in (c), even when they are subjected to the same control signal. To compensate for this, our lower-level controller runs two feedback loops: an inner loop that ramps a PWM signal to maintain a constant velocity, and an outer loop that generates a target velocity as a function of position error. This allows for smooth-running control in any column configuration. (We also get the added benefit of

speed control, which can allow for more "animated" or expressive movements.)

In addition to axial movement, we can also control the angular position of the endpoint cells. Servo motors (DS5160, 60 kg-cm) at the ends of each column are sufficient to raise the structure out of plane in a controlled fashion (Figure 7), and are used primarily for the deployment and retraction procedures described in Section 5.

Power, Modularity, and Mounting

Each column of our surface can be constructed and controlled independently, which allows us to extend the surface by adding additional identical copies. We built two dedicated driver boards for each column — one to control the smaller motors that configure the column hinges (Figure 10a), and one to control the larger gearmotors that move the column endpoints (Figure 10b).

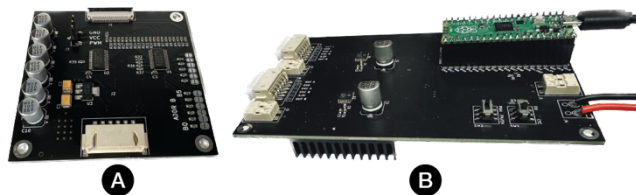


Figure 10. Circuit boards for driving small servo motors (a) and larger gearmotors (b), one pair per column. The gearmotor control firmware runs on a Raspberry Pi Pico.

The driver board for the servo motors is attached to the bottom cell of the column as shown in Figure 7d, it and travels with the cell as it moves along the linear rail. It receives instructions over I2C from a master computer, and sends PWM signals to each of servo motors in the column. These servo motors run off a 6V supply that is routed the integrated flex cables in the FR4-PET composite.

The driver board for the gearmotors is stationary, located at the bottom of the rail near the gearmotors themselves (Figure 7e). An onboard Raspberry Pi Pico runs a low-level position controller, and receives high-level position commands from a master computer over a USB connection.

The linear rail is a two-meter v-slot aluminum extrusion, with a 20mm x 40mm cross section. Belt drives for both the top and bottom cell run on

this rail. Two 3D-printed brackets, at the ends of each rail, are used to bolt the assemblies to a free-standing wall or horizontal base.

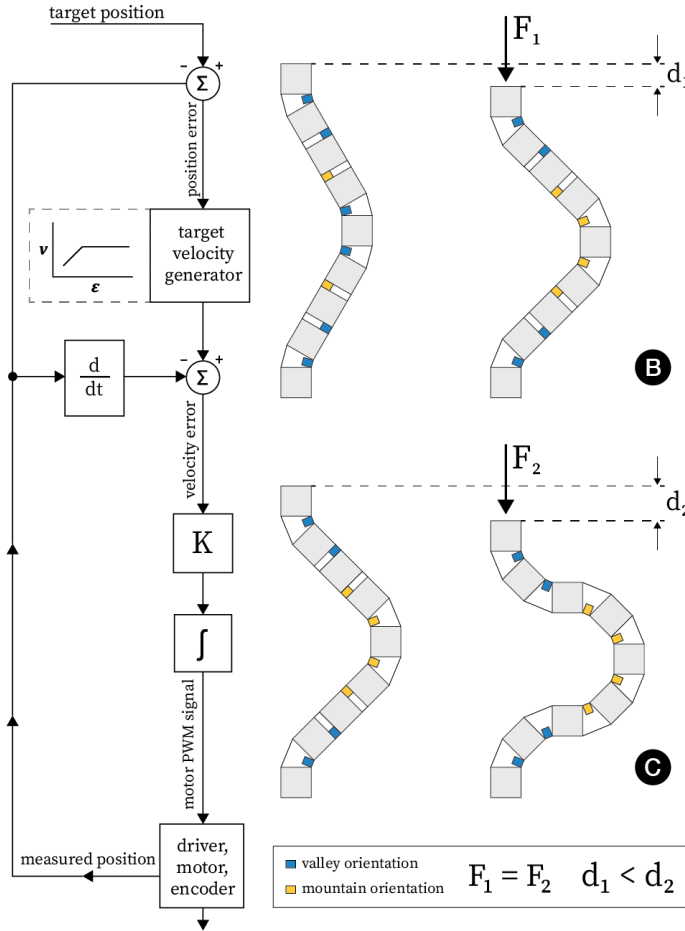


Figure 11. Since a column can be configured in many different ways, and because additional shape-change happens during the deployment process, there are many nonlinearities in our system response. The same force, applied at the endpoints of a column, may result in a different displacement depending on the initial configuration (b,c). We solve this by adding an inner loop to our position controller (left), that drives the column endpoints at a consistent velocity irrespective of the configuration. The target velocity is function of the real-time position error (it is a linear function, with a configurable limit). The proportional and integral gains transform the real-time velocity error into a PWM signal that maintains the target velocity..

Software Pipeline

To demonstrate the actuation capabilities of our robotic surface, we built a lightweight software system, used to reconfigure columns, pre-program shapes and initiate transformations. It consists of (1) a custom, high-level Python API, running on a Raspberry Pi 3B, and (2) a custom, low-level motor control API, running on each of the four Raspberry Pi Picos shown in Figure 10b (one microcontroller per column).

Through a remote Python REPL, we can use the high-level API to alter individual joint constraints (mountain/valley/rigid); control the position, velocity, and angle of column endpoints; and trigger scripted movement sequences. These scripted movement sequences (such as our

deployment/retraction procedures, or pre-defined joint configurations for common shapes) are themselves written with the high-level API. During development, we would often construct shapes by SSH-ing into the Pi 3B and using the Python REPL to adjust joints and move endpoints.

Our low-level API is used for driving the column endpoints with a DRV8874 IC (see Figure 11 for the nested feedback loops that make up this motor controller). Our firmware accepts simple, "G-code-like" commands over a USB serial port, and can act as either a velocity or a position controller. (During normal operation, we do not send these commands directly; rather, this is done via the Python API).

5. Heuristics for Control

By design, the mechanics of our shape-change procedure are relatively straightforward (simply configure a column, then squeeze the endpoints together). Still, there are some subtleties that bear discussing, as the nonlinear behavior of our robotic columns is not always intuitive.

When considering how to control a novel structure, it's helpful to construct a rough model of the system dynamics. For our surface, we can describe each column as a collection of square-shaped rigid bodies, connected by pairs of ultra-stiff springs. (See Figure 7, where the blue and yellow markers represent clamping forces at the valley and mountain hinges respectively.) This small amount of compliance is explicitly used in our deployment and retraction procedures; so representing the hinges as simpler, revolute joints would not be entirely sufficient.

Conceptually, configuring a column is akin to removing one (or zero) clamping springs from each cell pair. Each newly unclamped hinge can be represented as a fixed-length cable, which will be pulled taut when the corresponding cell pair is bent at 45 degrees (Figure 7).

It's also useful to draw a distinction between our system of winged, double-jointed cells, and a more "textbook" kinematic chain. Importantly, while the latter chain remains at a constant length regardless of the joint configuration, our robotic column *grows* in length as the tucked-in wings unfold from between the cells (Figure 12). This has implications for our

control strategies (endpoint positioning) as well as the space of shapes that we can render.

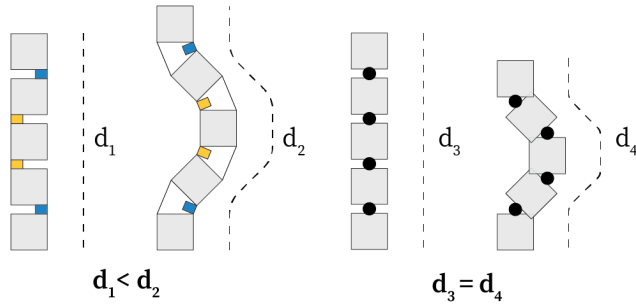


Figure 12. When the joints of a typical snake-like robot are articulated (right), the overall length of the robot does not change. However, in our double-hinged construction (left), a column will grow length as its joints rotate. This change in length has implications for our control techniques, as well as the space of shapes we can achieve.

Deploying Shapes

When a column is in its fully-retracted (flat) state, squeezing it from the edges will not trigger any transformation. The sequence of mountain and valley constraints has no effect here — the cells will simply push against each other until the axial motors stall. Similarly, if the “crest” of the column lies below the line-of-action imposed by the cell endpoints (Figure 13), then a squeezing motion will only bring the column back to a neutral position (as opposed to popping it out-of-plane).

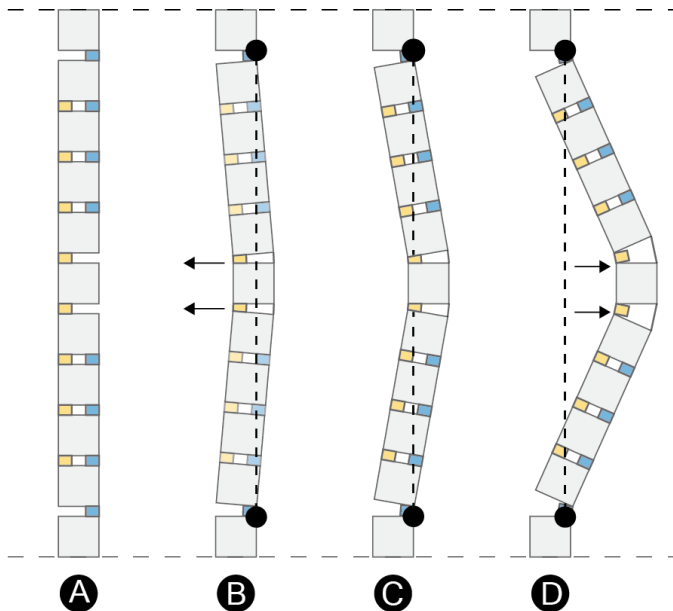


Figure 13. A configured column (a) may exist in a bistable state. The mechanical singularity is depicted in (c) — though the endpoints are fixed, the column has slightly grown in length (see Figure 12), and the folded hinges act as internal springs, pushing the cells apart. From this state, the column will snap to either (b) or (d), depending the external disturbance. In state (b), the crest of the column is below the line of action imposed by the endpoints, so squeezing the column will return it to state (a). In state (d), the crest of the column is above the line of action, so squeezing the column will deploy the shape.

To successfully deploy a column, we must first move it from a “retractable” state (crest below the line-of-action) to a “deployable” state (crest above the line-of-action). However, since the length of the column *increases* as the unclamped wings unfold, there is a region between these two states where cells are squeezed together – this forms a small mechanical energy barrier that we must overcome (Figure 13c). Put another way, the column exhibits a degree of bistability. Near this energy barrier, spring forces in the hinges will push the column into either the deployable or retractable state. (Anecdotally, colleagues who have physically interacted with the wall have described this feeling as “magnetic”.)

While a user can transverse this energy barrier by physically pushing or pulling on a column, an automated control strategy is often more desirable. To allow for this, the first and last cells in our columns are attached to servo motors (Figure 7), which can tilt these endpoints out-of-plane. Importantly, since the pivot is located on the side of the cell closest to the wall (the opposite of a typical valley fold), this rotation also creates additional space for the newly-lengthened column to pass through. Once the crest is sufficiently out of plane, we can return the endpoint to its original flat state.

From here, deployment becomes more intuitive. Moving the column endpoints closer together will cause the configured structure to bend, and the desired shape is achieved.

Retracting Shapes

The bistability that we encounter during deployment adds some complexity to the retraction process as well. To retract a shape, we begin by driving the endpoints of the deployed column back to their original position. At this point, however, we are still on the “wrong side” of the energy barrier – and unlike the deployment procedure, we can not rely on the endpoint servos to push the column over this boundary. (Flush with the rest of the wall, these first and last cells are already at or near their minimum angular position.) Instead, we drive the endpoints even further apart, stretching the column as far as the configuration allows. This quick motion triggers an oscillation within the column, and the crest bobs above and below the endpoint line-of-action. With precise timing, we can

“catch” the crest at the right moment, squeeze the endpoints together, and return to the neutral (flat) state. The entire sequence occurs within a one-second timeframe – the squeeze is initiated approximately 300 ms after the stretch. (Once empirically determined, we used this timing for all application examples without issue).

Even though this is an open-loop routine, we are able to reliably retract the columns using this “stretch-oscillate-squeeze” approach. Still, for a more robust procedure, it may be beneficial to add an accelerometer to our FR4 cells, and leverage orientation data from the structure (see section 6 for a brief discussion).

Axial Movement

In the procedures so far, we’ve only considered manipulating the position of *one* endpoint; the other has remained fixed. However, in our columns, both the first *and* last cells can be driven independently along a linear rail. This allows us to alter the deployment trajectory as well as the resultant shape. In Figure 6 for example, we squeeze our columns from the bottom rather than the top, arriving at an alcove-like structure that a person can sit underneath. In Figure 7, we squeeze both top and bottom endpoints together simultaneously, so that during deployment, the crest of the column travels perpendicular to the surface (as opposed to an upward or downward angle). This latter type of motion is particularly relevant for robotic manipulation tasks, for which we require position control of an end effector at multiple points along a trajectory.

In addition, we can axially reposition the deployed columns, by moving both endpoints in the same direction. (The position control loop described in Section 4.3 helps keep these endpoints in sync – without it, we are at risk of damaging the column by over-stretching or over-squeezing.) This is how we achieve the adjustable-height desk in Figure 5, and the “wave” greeting in Figure 2.

6. Mechanical Characterization

One of the main contributions of our work is a lightweight robotic material capable of withstanding human-scale forces. During the development of our application scenarios, we found that the surface could handle common household objects (full pitchers of water, laptop computers, cast-iron pans) without any visible trouble. Specifically, we tested the two-column shelf configuration (shown in Figure 5d) at loads up to 5 kg (11 lb), and our system was able to move these loads vertically at 12 cm/s. This load capacity is shape-dependent. A surface configuration made with "triangular supports" (see the right-most column in Figure 6a) is most robust, and when constructed with two columns, this "reinforced shelf" was able to handle loads of 7 kg before noticeable fatigue. (Note that for the above examples, we did not test to failure; see the next section for these characteristics).

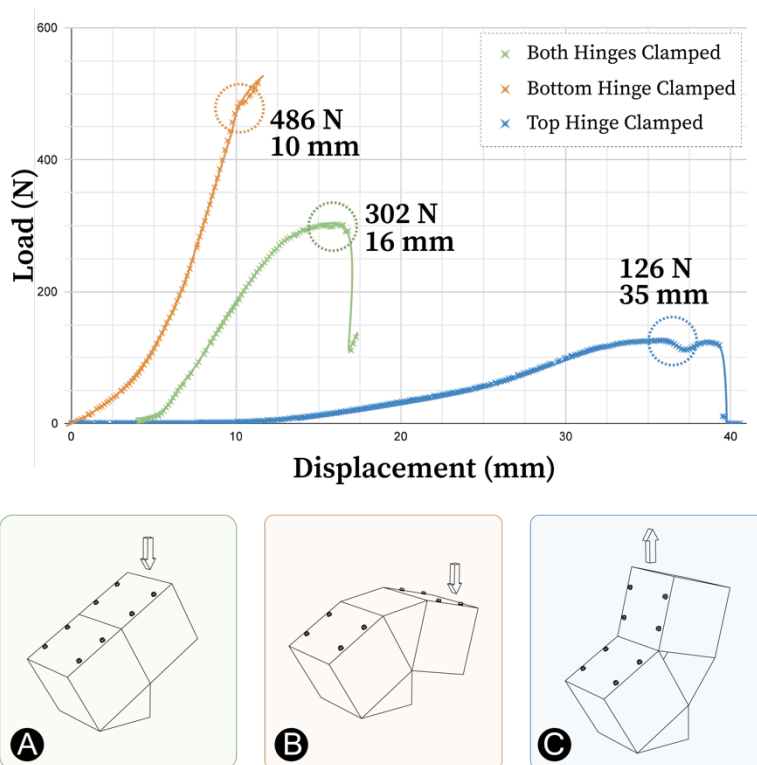


Figure 14. Mechanical tests for cell pairs in multiple configurations. (a) Compression test on cells with both hinges clamped mounted in a 45° degree angle. They can withstand 302 N of force. (b) Compression test with only the bottom hinge clamped yields 486 N of strength. (c) Tensile test on cells with the top hinge clamped shows 126 N of strength.

For a more precise characterization, we measured the forces on our cells in the three configurations of clamping and under tension and compression. We performed three tests, as outlined in the following subsections. We used a Mark-10 ESM303 Motorized Tension/Compression Test Stand with a 1.5~kN load cell for all tests.

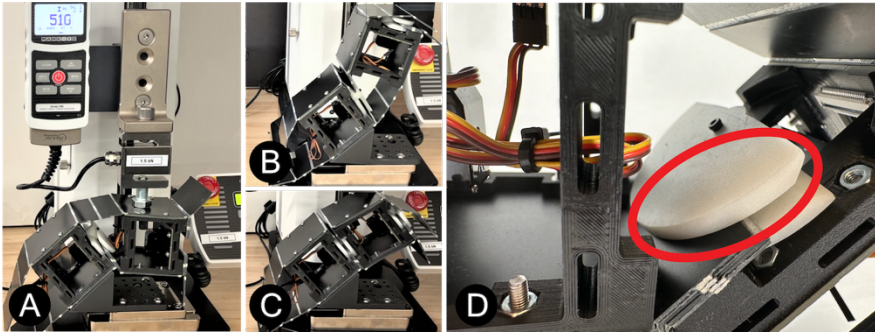


Figure 15. Test configuration for cell pairs (a-c). Failure occurs at the SLA 3D-printed clamp (d) circled in red.

Compression — Both Hinges Clamped:

Bending forces are generally of most concern to our robotic column. To evaluate these, we mounted two cells onto a custom-made jig that positions them in a 45° angle. In this setup, we clamped both hinges to produce our rigid connection and performed a compression test, i.e., pushed the load cell down onto the cell. We show the test setup and the results in Figure 14a. Our test shows that the cells can withstand up to 302 N.

Compression — Bottom Hinges Clamped:

Next, we were interested in how much force the cells can withstand if the only the bottom hinge is clamped, while the top hinge can extend. We kept the same test setup and only altered the clamping configuration, as shown in Figure 14b. The force that it can withstand is higher at 486 N, where it started to fail. The increase in strength is contributed by the top hinge being in tension, which is more favorable for our construction.

Tension — Top Hinge Clamped:

Lastly, we tested how our cells perform under tension. As the test setup in Figure 14c shows, we clamp the top hinge and pull on the cell. This is the weakest configuration of our robotic column and withstands 126 N before the SLA-printed clamp breaks.

When we test the cell pairs to failure, the failure reliably occurs at the SLA 3D-printed clamp, as shown in Figure 15d. A different material choice for the clamp could postpone this breakage — but we still prefer for the clamp to be the first component to fail, since it is the easiest component to replace.

7. Discussion, Future Work, and Conclusion

Typically, programmable surfaces make use of a separate actuator for each configurable bit, with the actuators themselves responsible for driving the shape change in their own local areas. This technique is difficult to implement at large scales, because a high-powered actuator is required for each individual cell. Our constraint-driven approach circumvents this problem. By decoupling the “per-cell” actuation from the “driving” actuation (moving powerful motors to the surface boundaries) we are able to achieve human-scale shape change without the use of large motors for each cell.

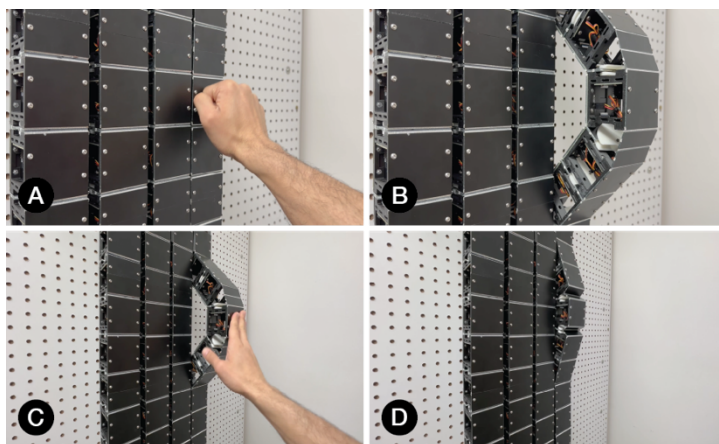


Figure 16. A mock interaction with our robotic surface. A user knocks on a section of the wall (a) and the corresponding piece pops out of plane (b). A user could potentially use this as a handle to pull on the column, and shape the surface into a desirable object. To return the handle to a neutral state, the user might nudge the protrusion (c), and the surface would retract this piece (d).

Constraints in our system can be quickly added and removed, and the deployment process takes a few seconds or less. This speedy reconfiguration means that our surface can support real-time user interactions, which we plan to explore more extensively in future work. In Figure 16, we show a mock example — a user knocks on a portion of the wall, and the wall “responds” by popping that small portion out-of-plane. The user could then use this newly emerged section as a “handle” to pull and sculpt the column into a desired shape. If, on the other hand, they decide that they do not need this handle, they could give it a nudge (Figure 16c), and watch it retract back into the wall. (While we consider this to be a relatively safe interaction, users would have to be mindful of a pinching force that can occur between cells, particularly as a column folds into the flat state. A fully-featured, user-facing robotic surface might operate similarly to an automatic doorway, reopening upon detection of an obstruction).

The interactions that we describe require input sensing, to perceive both intentional and unintentional user contact. Such sensors (i.e. for detecting acceleration, orientation, or capacitance) could be added to our FR4 column with little or no change to the overall structure (the integrated flex cables already provide power and data connections to each cell).

Beyond detecting human touch, adding these sensors would also allow for a tighter control loop. Although we already have closed-loop control over the column endpoints and clamp position, the column can still fall into an “unknown” state if it is in an underconstrained configuration (multiple angles under 45 degrees). While it’s good to know that we can successfully deploy and retract shapes with an open-loop approach, additional orientation data would allow for robust, in-situ reconfigurations.

Going forward, we may also wish to alter our cell-to-cell communication scheme. Currently, each cell in a column receives a PWM signal from a per-column control board (Figure 10a), which rotates the associated clamp. However, a separate data line for each cell will not scale to longer columns — for better extensibility, we may wish to adopt a daisy-chained control scheme²²³.

In our current system, cells are 65mm on each side. If we wish to increase our resolution (decrease the cell size), we may also want to explore alternate clamping mechanisms. Some possibilities include stiffness control through shape-memory polymers⁸⁹, pneumatically-actuated pressurized joints²²⁴, and electrostatics²²⁵.

Finally, future work should go towards making these surface more "expressive". Especially for non-anthropomorphic robots, exaggerated or animated movements can be useful for communicating intent, and are generally more engaging for human users^{226,227}. Some users also enjoy the perceived companionship they get from objects such as these, particularly as an add-on to an otherwise functional device²²⁸. Subtle gestures from these robotic objects can even mediate human-human communications, while still existing on the periphery of our attention²²⁹. Architecture can shape human interactions, and *robotic* architecture expands this space even further.

Our robotic surface provides a platform for exploring topics such as these. By adopting a constraint-based configuration method (as opposed to

directly driving the joints), we are able to build shape-changing surfaces at human-scale, allowing for rich new interactions between people and the built environment. Our fabrication technique, which makes use of rigid-flex composites, is not only accessible to researchers, but can be readily industrialized using planar manufacturing processes. Our application examples demonstrate the validity of this approach, as we are able to render multiple architectural objects (shelves, desks, and alcoves), all from a single surface.

VI. Fluent Interaction with Cyber-Physical Architecture

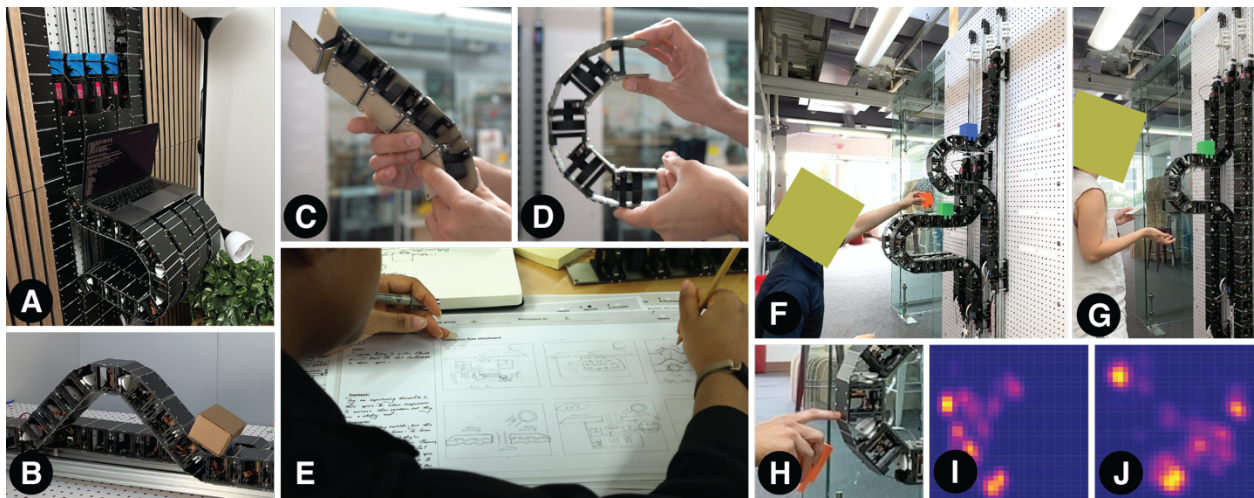


Figure 1. Shape-changing walls and floors (a, b) afford exciting new interactions with the built environment. To uncover the most compelling contexts, and the human values that underlie them, we conducted a series of speculative design workshops (c-e) where architects and designers received hands-on experience observing and ideating upon these technologies. We followed this with an elicitation study (f, g), analyzing the multimodal interaction strategies (h-j) that arose during collaborative tasks with our shape-changing robotic wall.

1. Introduction

Since the construction of the very first shelter, we have largely demanded *rigidity* from the built environment — we expect it to be static. But if we instead presumed that the walls around us could adapt and transform (without, of course, compromising stability), then what would we desire from these new, shape-changing surroundings? What capabilities would they afford? And how would we, as “users”, communicate our intentions to these entities?

As robotic technologies begin to transcend the industrial sector — entering homes^{230–232}, offices^{233–235}, and public spaces^{236–239} — these questions become more salient. While one strain of futurists envisions a world that is brimming with roving automatons, a smaller community champions a more “unobtrusive” vision of ubiquitous robotics^{240,241}. In this second vision, users would reap the benefits of automation by leveraging robotic technology that is embedded within the built environment itself. Shape-changing surfaces^{242,243}, actuated furniture^{29,244}, and kinetic facades²⁴⁵ promise spaces that can physically adapt to human needs — transforming when needed, and disappearing into the background when not in use.

Unlike smartphones or laptops that we explicitly engage with and then put away, architectural-scale robotic systems are omnipresent — we exist *within* them rather than merely *using* them²⁴⁶. They must navigate a delicate balance: remaining inconspicuous when at rest, yet being immediately responsive when called upon. They must interpret ambiguous human signals (the placement of an object, a muttered complaint about clutter) and determine whether and how to act. Most critically, they must do this while preserving human autonomy, building trust through repeated interactions in both public and private spaces.

This chapter addresses these challenges through two complementary studies, examining both the aspirational values and practical realities of interacting with cyber-physical architecture. First, we conduct a series of speculative design workshops to uncover what merit users might find in a shape-changing environment (Figure 1e). Then, we ground these visions in a task-based elicitation study, revealing how users naturally interact with — and recover from errors made by — these systems (Figure 1f).

Our platform for this exploration is a shape-changing robotic wall (Figure 1a). This structure, which we have adapted from Chapter 5, consists of three independent columns, which can move vertically on a set of two-meter rails. By remotely configuring the mechanical constraints that exist within each column, the structure can transition from a flat, rigid state into a curved, protruding shape (i.e. a shelf or an awning). These resulting shapes are stable, and when used as shelves, can hold common household objects such as laptops, backpacks, cookware, and books.

Chapter 5 of this dissertation introduced methods for fabricating and actuating such surfaces. However, user interaction techniques — and

design considerations for cyber-physical architecture more broadly — were beyond that initial scope. Our work augments that research, providing a framework for understanding how users conceptualize and communicate with robotic environments.

Specifically, we contribute:

1. Eleven design values for shape-changing environments, derived from speculative workshops with designers and architects. These reveal fundamental tensions between proactive assistance and user autonomy, and between personalization and public ownership.
2. An empirical characterization of multimodal interaction with robotic surfaces. This includes the development of a modal entropy metric, which quantifies the diversity and evolution of a user's interaction strategies as they discover system capabilities.
3. A set of research directions for cyber-physical architecture, synthesizing results from both studies.

Our work concludes with a proposal for reframing interaction with cyber-physical architecture: rather than mapping discrete user actions to system responses, our findings make the case for a dynamic model of user intent, that continuously integrates multimodal evidence. This approach, which we argue could alleviate some of the tensions unveiled by our design workshops, may be an essential step in achieving fluent interaction²⁴⁷ with robotic environments.

2. Related Work

A shape-changing environment is, in many ways, a robot that you can walk inside of²⁴⁶. To develop effective techniques for interacting with these structures, we draw on perspectives from tangible computing, room-scale actuation, and human-robot collaboration.

Foundations

Our research builds on work by Hong et. al²⁰ who conducted one of the first (to our knowledge) ideation and elicitation studies on the subject of large-scale shape-changing interfaces. In their initial study, the authors hosted a 30-minute brainstorming session, where small groups of participants worked together to generate application ideas for a hypothetical, wall-sized shape display (resembling inFORM⁵⁵, but turned on its side). A recurring theme was the concept of reconfigurable spaces — participants imagined situations in which seating and tabletops could be constructed when needed, and removed when not in use.

To further explore this theme, the authors then conducted an elicitation study that focused first and foremost on the concept of explicit reconfiguration. Using a virtual shape display (a series of images, projected onto a wall), the authors asked users to perform actions that corresponded to various geometric changes, such as scaling or rotating a pattern of actuated cells. By design, the methodology mimicked that of Wobbrock et al.^{248,249} — users were presented with a “before” and “after” state, and were instructed to come up with an action that could have triggered that change.

Most often, the users enacted gestures (using one or both hands), although this varied depending upon proximity. Users that were at least one meter away from the wall, for instance, would occasionally abandon gestures in favor of tool use (e.g. drawing on a phone screen) or voice commands. When standing up close, however, they opted for more hands-on techniques — a majority (16 out of 20) said that they would want to touch the wall when tasked with transforming it from one fine-grained state to another.

Interestingly, we found that touch was *not* the dominant mode in our own task-based elicitation study — we saw instead a diverse set of multimodal interaction techniques, even when the wall was well-within reach. This suggests that while direct manipulation is a powerful paradigm for explicit state transformation, the interaction strategies for more goal-oriented, collaborative tasks are more complex. Our work seeks to understand how users navigate these scenarios, and expands on the role of agency in shape-changing environments.

Intelligent Environments

A long-standing goal in computing (specifically, within HCI) has been the development of environments that are imbued with ambient intelligence^{250,251}. These are physical spaces which can sense and respond to human activity.

Initially born out of research in ubiquitous computing²⁵², these environments have appeared in various forms throughout the last three decades. Early work explored interactive tabletops^{253,254}, augmented whiteboards²⁵⁵, and ambient ceiling displays²⁴ — connecting digital information to specific physical objects, often with tangible controls. This later expanded to room-scale projection mapping, with influential projects like IllumiRoom²⁵⁶ demonstrating highly immersive visual experiences. More recently, systems like Dynamicland^{257,258} have fully embraced the “room-as-a-computer” paradigm, with a particular emphasis on the human-to-human collaborative opportunities that these environments afford.

Most of these systems use cameras as the primary input device, though researchers have also explored additional modes for room-scale interaction. Often, these leverage sensors that are embedded into architectural elements: electromagnetic grids⁴⁸, optical fibers⁴⁷, load cells⁴⁹, or conductive paint²⁵⁹. Coupled with machine learning algorithms, these methods enable off-body activity recognition²⁶⁰, allowing environments to understand both where users are, and what users are doing.

While there has been great progress in sensors and displays, a recent (and ambitious) goal is to enable the environment to actuate — to change its physical form^{244,261}. In HCI, this has been explored through

reconfigurable walls^{243,262,263}, floors^{56,60}, and furniture^{36,94}. In galleries, building lobbies, and public spaces, these installations often appear under the name “kinetic walls”^{264–266}. On an even larger scale, actuated building facades like the Al Bahr Towers²⁶⁷ and reactive pavilions^{268–270} change state in response to environmental data or human presence — for instance, providing reactive sunshading.

Multimodal Interaction

As environments become more capable, the channels of communication must become richer. Multimodal interaction, which combines inputs like voice, gesture, and touch, offers a more natural and expressive way for humans to communicate with machines. The seminal example is “Put-That-There”^{271,272}, which demonstrated how a deictic gesture (e.g. pointing) could resolve the ambiguity of a spoken command, and vice-versa.

This is especially important in shared physical environments, where achieving a joint understanding between humans and robots is a central goal²⁷³. Robots often use a combination of gaze, gesture, and speech to establish common ground and communicate intent²⁷⁴, allowing for the kind of fluid, back-and-forth exchange that Hoffman and Breazeal describe as “fluent” collaboration³⁸. For instance, when picking items from a shelf, a warehouse robot could concurrently parse vocal and gestural cues from a human partner, helping to retrieve specific objects in cluttered environments²⁷⁵. Recently, advances in transformer architectures have greatly enhanced the perceptual capabilities of such robotic systems²⁷⁶. However, even state-of-the-art models can still struggle over longer time horizons²⁷⁷.

In addition to perceiving human input, embodied agents can also use their physicality to enhance communication²⁷⁸. Compared to voice-only interfaces, these agents can be more engaging, and users have demonstrated higher situational awareness when interacting with them²⁷⁹. These developments point towards richer possibilities that speculative design can help us explore.

Speculative Design

Speculative design is a research practice that uses sketches, objects, and fictions to explore possible futures (as opposed to near-term products). Its primary purpose is to challenge assumptions and provoke debate about the social, cultural, and ethical implications of emerging technologies²⁸⁰. The result is a shift in thinking from what is *probable* to what is *preferable*, which can help guide the design of human-interfacing systems²⁸¹.

The field makes a distinction between affirmative design, which reinforces prevailing market logics, and critical design, which uses a form of “social dreaming” to prompt reflection²⁸². A complementary approach is design fiction²⁸³, which employs diegetic prototypes — imaginary products that exist within a film or narrative — to test and explore ideas before they are actually built.

Increasingly, these methods have been adopted within HCI — which has in turn spurred the development of hybrid techniques that combine speculation and elicitation. Through “speed dating”, for instance, groups of participants are exposed to a rapid showcase of future scenarios, prompting reactions that are noted by the investigators²⁸⁴. Other methods involve theatrical performance, immersing users in potential scenarios while observing their responses²⁸⁵.

This is particularly relevant for human-robot interaction, aspects of which are, arguably, inherently speculative²⁸⁶. Certain robotic form factors may be “maladapted” to non-industrial contexts²⁸⁷, and require reconceptualization for domestic settings. Speculative methods are one way to attack this, engaging stakeholders through role-playing²⁸⁸, brainstorming²⁸⁹, and focused discussions²⁹⁰. For shape-changing interfaces in particular, design fictions can be used to address the central question of practical utility²⁹¹, visualizing how shape-change might manifest in everyday life²⁹². Participants in our workshops explored this through sketching and scenario-building, which can be a powerful method of inquiry²⁹³.

3. Design Workshops

A central step for any new technology is articulating its value. When we introduce cyber-physical architecture, what actual problems do we solve, and what new experiences do we enable? To answer this, we conducted four speculative design workshops, aimed at uncovering characteristics that could pull shape-changing environments towards real-world adoption. Beyond just cataloging applications (the *what*), our study attempts to reveal the core values that underlie them (the *why*).

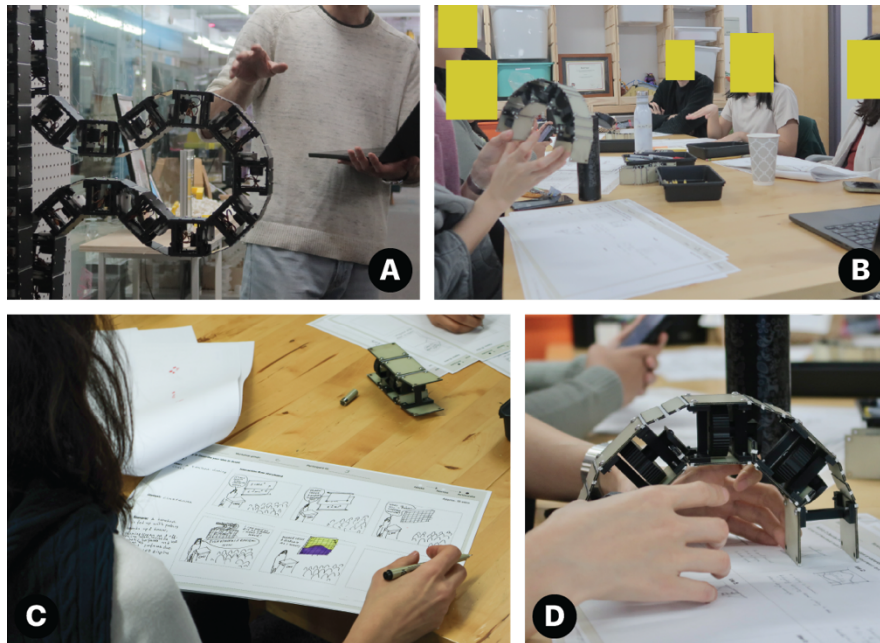


Figure 2. Participants in our design workshops were able to observe our robotic wall (a), and could request shape-changing demonstrations from our facilitator. They were also given a toy model to explore (d), which could be manually reconfigured. Throughout the workshop, participants sketched scenarios (c) and discussed hopes and concerns (d) related to cyber-physical architecture.

Participants and Procedures

We recruited 20 participants, explicitly filtering for those with experience in either architecture or design. Selecting one specific demographic (e.g. older adults, wheelchair users, etc.) would have been arbitrary and premature. Instead, we opted for a more exploratory approach, requesting input from people trained to think critically about spatial relationships and human needs.

Our participants had varying degrees of industry experience (ranging from 2 to 11 years), and were between 24 and 34 years old (median age: 27). At the time of the workshop, all were enrolled in either a Masters or PhD program.

We split these participants into four groups (4–6 participants each), and held a separate, 3-hour workshop for each group. This split was based primarily on participant availability, though we also endeavored to keep an even distribution of ages and backgrounds across the four sessions.

Each session was conducted in the same manner. We began by introducing the participants to our robotic wall, demonstrating various transformations and briefly explaining the principle of operation (Figure 2a). We also gave each of them a small, non-actuated toy model to experiment with (Figure 2d). We did not discuss any example use cases, as we didn't want to bias the participants towards one particular context or set of applications. This introduction lasted 20 minutes.

Immediately afterwards, we led participants through a set of three design activities. Though each activity was to be completed independently, we did not prohibit participants from talking with each other, and some short, spontaneous conversations did occur.

The three activities were:

1. **IDEATION**. For 25 minutes, participants were instructed to list, sketch, or doodle any concepts that involved a robotic surface. There were no explicit constraints, though we did ask them to consider the questions: “Where do you see these surfaces in the future?”, and “How might these surfaces become integrated into everyday life?”.
2. **NARROWING**. After brainstorming, participants had 15 minutes to choose their favorite ideas and add additional detail, outlining specific applications or scenarios. We asked them to describe six ideas, with the stipulation that at least four of them should feature our specific style of robotic wall (as opposed to a pin display⁵⁵ or similar).
3. **ARTICULATION**. Finally, we gave participants 60 minutes to construct interaction flows (in the form of storyboards) for four of their ideas. We also asked them to clearly specify the context and scenario associated with each idea.

Following these activities, we moderated an hour-long group discussion where participants presented their interaction flows (Fig 2b). We steered the conversation by asking a few key questions: How should agency be shared between users and walls? What tangible qualities of the robotic surface led the participants to their ideas? How would they extend their ideas if there were no technological constraints? And finally, what were the underlying values and concerns that motivated these scenarios?

Analysis Method

The technique of content analysis, popularly defined by Krippendorff²⁹⁴, allows for the systematic examination of communication and interaction. We found this method appropriate for our study, as we were exploring design-oriented data rather than lived experiences.

We began with data familiarization, carefully examining all storyboards and discussion transcripts to gain a comprehensive understanding of the material. From this initial review, we identified seven overarching categories that pervaded the workshops: *form, function, values, contexts, users, ownership, and interaction techniques*.

We then proceeded to code the transcripts and storyboards: identifying individual sketches, scenarios, and conversations that belonged to each of the seven categories. Within these categories, we identified more detailed codes. For example, the “context” category included specific settings such as “home”, “outdoors”, “café”, and “office”. The “ownership” category included subcodes such as “private”, “public”, and “personal”.

To move beyond a simple catalog of applications, we extracted all transcript quotations that mentioned or alluded to an underlying value. Next, we engaged in a focused coding process to interpret the values expressed in these quotations. After progressive refinement of the codebook, we emerged with a set of eleven values that underlay the storyboarded scenarios.

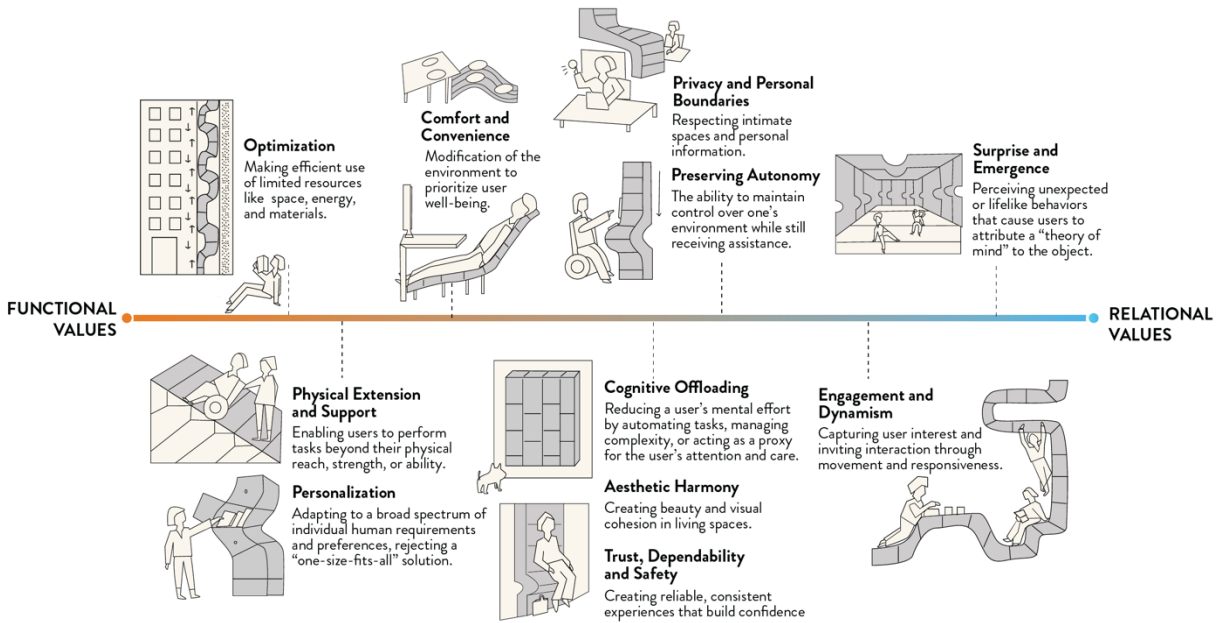


Figure 3. Analyzing the data from our design workshops, we identified eleven values which underpinned the scenarios our participants created. Some values are best categorized as "functional" (associated with tangible and pragmatic concerns), while others are best categorized as "relational" (describing psychological qualities of human-machine interaction). During our workshop discussions, tensions emerged between many of these values.

Findings and Framing

Is there real value in a shape-changing environment? These design workshops were born out of an initial concern — that despite the technological allure of “robotic walls”, the actual applications proposed in prior work can sometimes appear a bit contrived. We hypothesized that this could be due to (1) insufficient exploration of key contexts, and (2) limitations of the technology in its current instantiation (i.e. perhaps there exists a more appropriate form factor or interaction mode). Through these workshops, we aimed to identify values that could guide interested researchers towards building “the right thing”²⁹⁵.

Eleven values emerged during our workshop analysis, which we arranged on a spectrum from “most functional” to “most relational”. Functional values encompass the pragmatic and performance-oriented qualities of shape-changing technology. Relational values capture interactions and dependencies between people and a robotic environment. We encourage the reader to examine Figure 3 for an overview of the values, before continuing to the more detailed narrative in the following subsections.

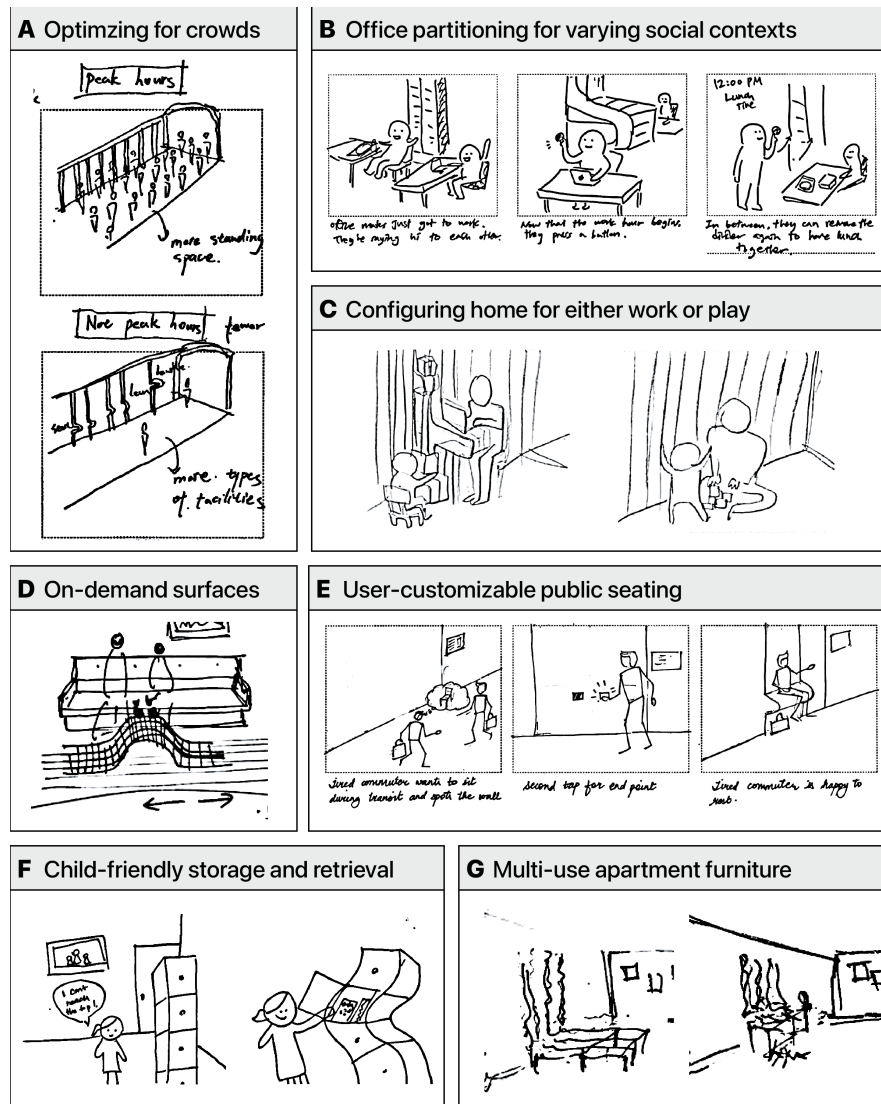


Figure 4. Participants developed storyboard scenarios illustrating the value of space optimization. The examples include (a) managing peak and off-peak crowding, (b) partitioning spaces for different social interactions, (c) configuring homes for work and leisure, (d) creating on-demand surfaces, (e) customizing public seating, (f) enabling child-friendly storage, and (g) designing multi-use apartment furniture.

Resource Optimization

Noting the inherent reconfigurability of the wall, many participants suggested applications that prioritized multifunctionality and reuse. In dense urban environments, where living space is limited, they imagined dynamic room partitions, expandable storage, and on-demand seating in small apartments and offices (Figure 4). “[People] are looking for solutions where they can sort of optimize their space,” mentioned one participant, drawing from their experience as an architect. Another agreed: “While working, while interacting with clients, the most important thing [they] say is more space and less cost. That’s how the ratio goes.”

This mode of space optimization was proposed for public areas as well: namely, train cars, subway stations, and airports (Figure 4a, 4e). “You don’t necessarily need permanent furniture,” shared one participant, “but if it’s a long transit, I just want to sit and rest for a bit.”

At larger scales, many participants envisioned the technology as an architectural element for energy savings, serving as a temperature-sensitive facade that could adjust to block solar heat and reduce cooling demands (Figure 5). This transformation was most often described as occurring automatically. However, others preferred a higher level of human agency, sketching wall-mounted sliders or describing mobile applications for remote control.

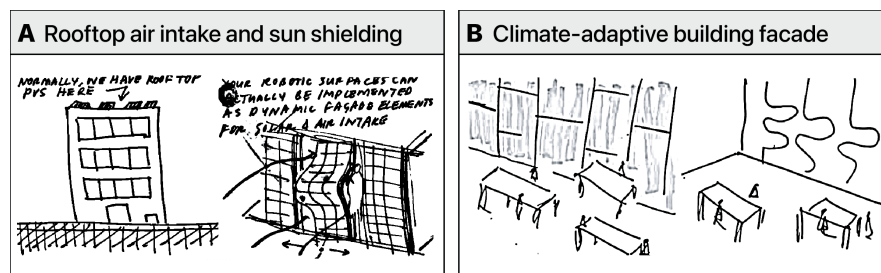


Figure 5. Storyboard scenarios illustrate architectural-scale applications for energy efficiency: (a) rooftop air-intake and sun-shielding structures, and (b) climate-adaptive facades that adjust to block solar heat and reduce cooling demand.

In addition to conserving space and energy, shape-changing structures were seen as a way to conserve and reuse materials. Participants described desks and chairs that could “grow up together with [a] child”, as well as modular elements that could shift between functions over time. “After two months, [maybe] he doesn’t want his books, he wants a [music] collection,” said one participant, describing an imagined user. “It is multi-use... it doesn’t necessarily have to be treated as a bookshelf.”

But there was also a tension among participants, centered around the large upfront ecological cost of the proposed “resource-saving” applications. How many pieces of furniture does the wall need to replace, for instance, in order to justify the initial investment? Participants wrestled with the question, conceding that while small living spaces may be a promising context, reconfigurability alone may not provide sufficient value. “Even then, people get very inventive,” one participant argued. “So... for it to have to be a robot wall, I feel like there’s [got to be] some sort of automation.”

Cognitive Offloading

Many participants saw value in offloading mental tasks to the environment — automating minor functions, or permitting the wall to move proactively. “The usefulness of it being a robot is [that] I don’t have to do anything,” declared one participant. “It just does it.” In the minds of some participants, these more agentic behaviors seemed to warrant the complexity of a technological solution.

Several scenarios involved “just-in-time” support, in which the wall would handle finer details so that users could remain focused on a primary activity. One participant sketched a kitchen backsplash that served as an assistant chef: delivering ingredients, removing clutter, and holding a mobile phone to display recipe steps (Figure 6c). Another described a chair-like form that offered posture support over long work sessions, and could prompt the user to take a break and stretch.

Other scenarios ranged from simple caregiving, such as playing with pets or adjusting sunlight for plants, to more critical support, such as detecting a fall for an elderly user. There was, of course, a caveat: the automation had to be competent. Participants warned that a slow or clumsy robot would create more frustration than it solved. If a command took too long to execute, one participant said that they would simply give up: “okay, whatever, I will do it.” As is the case in other robotic systems, successful offloading relies heavily on established trust²⁹⁶, which participants noted can be fragile.

Preserving Autonomy

Sometimes at odds with the desire for proactive assistance, participants emphasized that any automation must still respect user autonomy. Multiple scenarios reflected a preference for explicit, user-initiated interactions. “For me, it’s mostly on the human to start,” said one participant. “You hit a button and it does something.” (Figure 6b).

This desire for control was often rooted in a practical concern about the risks of misinterpretation. “I do not want it to predict what I want, because what if it predicts something wrong?” asked one participant. “I live on the third floor,” joked another, “What if it throws me out the window?”. To give users reliable agency, participants proposed graphical

user interfaces for precise control, and simple voice commands for more atomized interaction. Automation was more readily accepted in cases where the functions were temporary and reversible (i.e. the costs of misinterpretation were lower).

A related discussion emerged around the degree of “surveillance” that would be necessary to facilitate proactive robotic behavior. A few participants indicated that they would be more amenable to sensing mechanisms if the corresponding behavior resembled a physical reflex, as opposed to a more calculated decision. “It’s not making inferences about you as a person,” said one participant, “It’s only making an inference about where your body is going.”

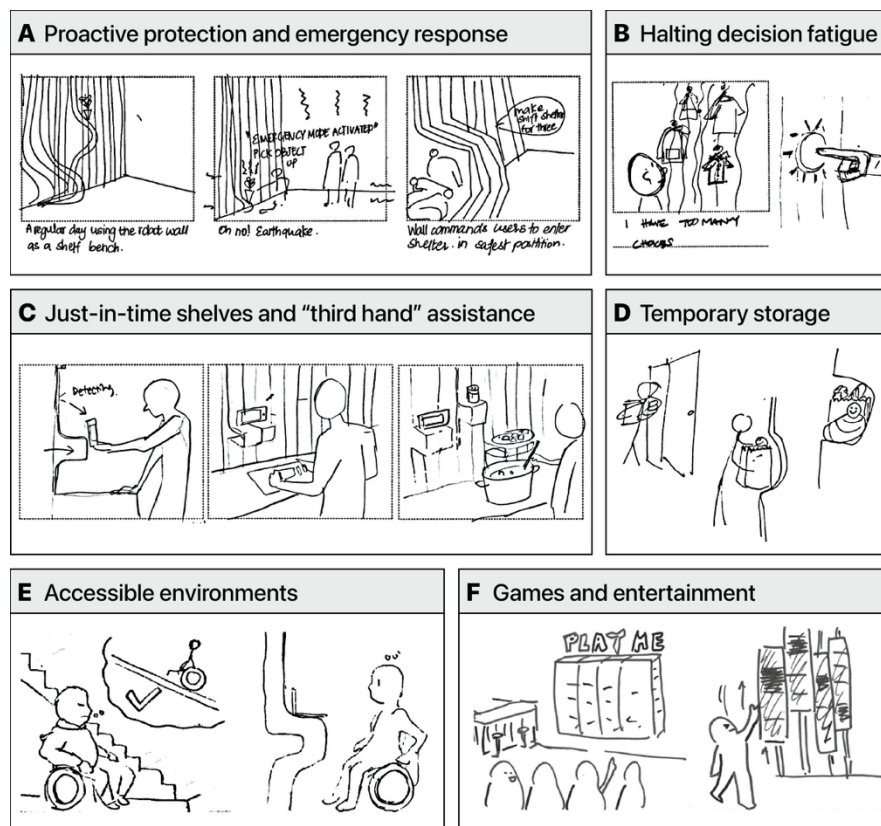


Figure 6. Storyboard scenarios depict protective and assistive applications of responsive surfaces. Examples include (a) transforming into emergency shelters, (b) reducing decision fatigue by simplifying choices, (c) providing just-in-time shelving and assistance in the kitchen, (d) enabling temporary storage, (e) supporting accessible environments, and (f) applications with games and entertainment.

Privacy and Personal Boundaries

Conversations about agency and proactivity led participants to consider the privacy implications that come along with a robotic environment. Interestingly, this was discussed in two modes: one in which the wall was portrayed as a prying, unwanted intruder; and another in which the wall

took an active, protective role in establishing boundaries between a user and other people.

In the more protective case, participants described office cubicles that could promote either focused work or social interaction; and apartment partitions that could move to obscure sensitive areas (Figure 5b). “If you’re staying in a studio apartment and your friend comes to visit you, you might not want him to see your sleeping space,” said one participant, “because that’s more personal to you.”

Participants sketched “secret” spaces: camouflaged doors with fingerprint access, and concealed storage nooks in bathrooms and bedrooms. “You can hide little objects,” suggested a participant. One amusing scenario involved a couple using a shape-changing bed as a more “active” enforcer of personal boundaries: “If your husband’s irritating you, just press a button... you’re gone off the bed.”

The second discussion mode encompassed perceptual and data privacy. We noticed mixed views on sensing and predictive features. Some participants were more accepting: “If it really gives enough value for me, then I would [give] out my right to be seen.” Others, however, preferred manual control over predictive actions, worried about misinterpretation and unwanted outcomes. There was some inherent vulnerability in sharing space with an intelligent environment.

Trust, Dependability, and Safety

As participants crafted increasingly sophisticated robotic behaviors, they also raised their expectations for reliability. Trust, dependability, and safety emerged as values that would determine whether users could accept robotic environments in their daily lives.

Participants envisioned trust building gradually, almost like training a pet. “Anything new is always jarring,” reflected one participant, “it takes time to get used to [it].” Starting with low expectations seemed important, allowing new capabilities to be introduced bit by bit. Legible behavior was also crucial. “If it’s approaching me and it naturally shows me that it’s slowing down, [then] I can trust it,” explained a participant.

Yet the margin for error was small. “If it ever knocks something out of place, even the tiniest thing,” warned one participant, “[then] I can't trust it after that point, right?” This became especially critical for vulnerable populations. For some scenarios that the participants described — wheelchair transfer assistance, rehabilitation support — a system failure could cause serious personal harm.

Physical form also influenced trust perceptions in ways that transcended pure functionality. “The softness of it... reduced my worry,” shared one participant, referring to the “rounded” edges of our shelf configuration (Figure 1a). Tactile qualities and visual cues could be important in establishing user confidence.

Participants also imagined these surfaces in “protective” roles: transforming into emergency shelters during earthquakes, lifting precious objects during floods, and serving as fireproof partitions (Figure 6a). While these applications highlighted the demanding standards for trustworthiness, they also revealed participants’ excitement about the wall's physical capabilities.

Physical Extension and Support

Robotic surfaces were often proposed as tools for physical augmentation — extending human reach, strength, and precision beyond natural limitations.

Many participants gravitated toward industrial applications, where the shape-changing capabilities could enhance worker safety. “In building construction, sometimes you need to carry things [between] floors before the lift is made,” explained one participant, who sketched large surfaces that could elevate objects, or create kinks near unfinished exteriors for workers to latch onto. Others described dynamic scaffolding that could be adapted for different functions (e.g. window washing), and snake themselves around buildings.

Examining these “heavy-duty” tasks pointed to a well-known challenge in robotics: the strength needed for useful augmentation also carries greater risk of harm. This pushed participants toward form factors quite different from those on factory floors. There was a large emphasis on collapsible structures, lightweight panels, and hand-configurable modules.

Many drew clear inspiration from our prototype, though in practice, these large-scale applications would likely necessitate a more robust constraint mechanism.

Participants also envisioned scenarios where the environment could act as a “helping hand” during cognitively demanding tasks. In a hospital, shape-changing lights in an operating room could reposition themselves as a surgery progressed. In an office, a shape-changing table could stabilize objects that were accidentally knocked over, preventing spills or damage during delicate work (Figure 7e).

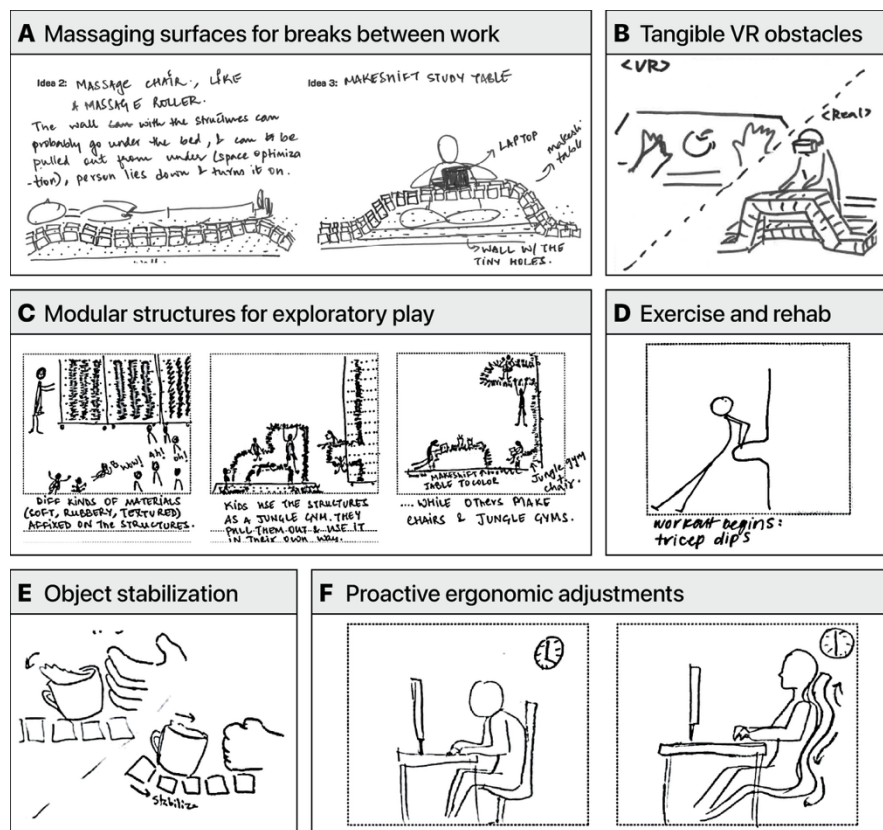


Figure 7. Storyboard scenarios illustrate diverse applications of responsive surfaces. Examples include a) massaging surfaces for breaks during work, b) tangible VR obstacles, c) modular structures for exploratory play, d) exercise and rehabilitation support, e) stabilizing objects to prevent spills, and f) ergonomic adjustments in office settings.

Some of the most compelling applications emerged in therapeutic contexts, where the wall could provide both physical support and procedural guidance. “Consider a physical therapy approach,” said one participant. “If someone doesn’t remember how to do a particular exercise, they can lean up against the wall, and the wall comes out to meet them in the place where they should be.” Some participants even imagined the wall verbally speaking to users, to guide them through weight training and rehabilitative activities (Figure 7d).

Accessibility emerged as a particularly rich area for exploration, with participants describing surfaces that could adapt to different physical needs. They sketched walls that could bring objects down from high places, actuated handrails that could help people stand up, and load-bearing surfaces that could transform from stairs into ramps. Rather than designing for an “average” user, participants envisioned environments that could meet people where they are. “I see that as an opportunity,” said one.

Personalization

Personalization seemed straightforward to participants: it would be beneficial to have surfaces that could tailor themselves to individual bodies, preferences, and needs. But as discussions deepened, it became apparent that this value was inextricably tied to the question of “who controls the space”.

In personally-owned spaces, participants embraced elaborate customization scenarios. They described walls that could shift from “wavy” to “straight lines” based on mood, surfaces that responded to musical beats, and environments that adapted as children grew or lifestyles changed. “Different individuals have different body types,” reflected one participant, “and [we] can customize it to the needs of everyone.”

This was especially important for accessibility, particularly within family households. Rather than requiring permanent modifications that might stigmatize some users or inconvenience others, participants envisioned surfaces that could temporarily accommodate wheelchairs, then reconfigure for walking family members. The same kitchen counter could lower for a child learning to cook, adjust for a parent using mobility aids, then return to standard height for daily use.

In privately-controlled spaces — offices, stores, gyms, theaters — personalization introduced competing interests. Though participants were enthusiastic about adjustable counters in coffee shops, and walls in climbing gyms that adapted to different skill levels, they also recognized that businesses must balance individual customization with broader operational needs.

The most provocative tensions surfaced around public spaces. While participants initially described benches that could adjust for different users,

the implications quickly became uncomfortable. “I think it gives a lot of power over public spaces to people who can afford this kind of technology,” warned one participant. “You can very easily make that anti-homeless architecture, right? By just having every public bench be [made of] this, and then at night they go down.”

This concern extended to the visual character of shared environments. Shape-changing surfaces promised to beautify public spaces, but also threatened to disrupt visual continuity. When every surface can transform at will, who determines what the neighborhood should look like?

Aesthetic Harmony

For many participants, the most compelling aesthetic quality was the technology’s ability to disappear. “The motion is the [unique selling proposition],” explained one participant, “its ability to move and create forms... and then go back to nothing.” This connected to discussions around privacy as well. “I thought of a concealed door,” shared one participant. “[This] part kind of curves into a push bar... [then] the door shuts and it again flattens out.”

Participants also described scenarios where instead of adding mechanical agents to a space (e.g. a robotic humanoid maid), the technology could become one with the space itself. They sketched undulating surfaces that could shuffle objects from room to room, eliminating visual clutter by moving stray items out of sight).

Material choices reinforced this philosophy of integration. Participants preferred soft, curved forms over angular mechanical aesthetics, and envisioned context-specific finishes — wood for homes, stainless steel for kitchens, concrete for public spaces. Scale became another aesthetic dimension. Participants sketched small interior panels as well as entire building facades — which could create dynamic patterns and draw attention from passers-by on the street.

Engagement and Dynamism

Participants were drawn to the wall’s dynamic qualities, envisioning new modes of interaction that ranged from direct, tactile manipulation to the passive appreciation of ambient motion.

The most hands-on scenarios involved a form of co-physical interaction, where users could manually shape their environment. One of the most vivid concepts was a large, child-friendly play structure that children could literally pull apart and reshape. “I was imagining this to be, like, a techie version of [Cas Holman's construction sets],” one participant explained, referencing a collection of large wooden toys for unstructured play (Figure 7c). This style of interaction would likely require some means of inferring intent. The robotic structure must be intelligent enough to know when to yield control to a child's push or pull, and when to stiffen to securely hold a new form.

Participants also envisioned these surfaces facilitating new forms of entertainment. Several imagined the wall giving physical form to virtual reality experiences, creating tangible obstacles that mirrored the digital world (Figure 7b). Others proposed more playful, arcade-style applications, such as a large-scale version of “whack-a-mole” (Figure 6f).

The surface's capacity for more “freeform” motion proved equally compelling, and was further heightened by the concept of a concealed actuation mechanism. “If I'm not seeing what's inside, it's a mystical road for me,” explained one participant. “I don't know how this is happening.”

At a public scale, this dynamism translated into a powerful tool for civic and commercial appeal. Participants imagined building facades that could become local landmarks through their eye-catching transformations, creating a spectacle for passers-by. “If you see something very interesting going on on a facade,” said one participant, “it automatically [brings] business.” This hinted at a deeper interest in more spontaneous, natural behavior.

Surprise and Emergence

Although dependability is critical for developing trust in a system, a compelling counterargument unfolded around the value of unpredictability. Participants explored how emergent, surprising behaviors could encourage a user to attribute a theory of mind to the wall, viewing it less as a tool and more as a “living organism” with its own agency. Zoomorphic traits, like subtle breathing motions or responsive “nods” could give the sense of “being seen”²²⁸ by the environment. This was not

limited to ambient actions. One participant suggested that if a user's command was followed *slightly* imprecisely, it could be interpreted as the wall balancing the user's instruction with its own internal priorities. They compared the relationship to having a pet dog.

A few participants associated surprise with intuition, describing scenarios in which uncertain user intentions might lead the wall to improvise a response. One participant claimed that user desires are often under-specified, and said that under those conditions, they would be more forgiving of rogue actions by the wall. “My mom gives me very vague instructions,” they empathized, “I need cues too.”

Perceived agency could also spur and reward curiosity. For instance, if a user observes the wall reacting to changes in musical tempo, they might reason that “it could react to [rhythmic] motion as well.” A surprising interaction might reveal one function, which in turn encourages the organic discovery of related capabilities.

Comfort and Convenience

Participants also saw value in an environment capable of providing proactive care. This often took a somatic form, with surfaces that could physically soothe the body (Figure 7a). Participants described massaging furniture that could sense and respond to strain — one envisioned an ergonomic chair that would shift forms after a period of prolonged use (Figure 7f). These scenarios extended to affective support as well, with proposals for environments that could regulate mood: a “calming room” with wave-like walls, and an energizing one with rapid motions and lighting.

Other participants described a form of logistical care, aimed at reducing the friction of daily life. In public, this included on-demand convenience, like temporary seating in transit hubs or retail shelves that could lower items on command. In the home, they pictured a domestic assistant: a wall that could “get the carrot for me” while a user cooks, or help carry meals from the kitchen to the dining room.

Preliminary Implications

Our workshops revealed three key design tensions for shape-changing environments.

1. A conflict exists between proactive assistance and user autonomy. Participants wanted environments that could anticipate their needs, but they were also wary of ceding too much control.
2. The value of personalization is highly context-dependent. While celebrated in private settings, customizing shared spaces raised significant social and ethical concerns.
3. Predictable behavior is largely desirable, but can come at a cost. Participants expected dependability for critical tasks, but were also intrigued by surprising, dynamic actions that fostered a sense of “aliveness” and partnership.

Participants did not envision the robotic wall through a single value lens, but across a wide spectrum of expectations. At times these expectations converged, while at other times they diverged. This plurality suggests that the design of such systems must be guided by specific contexts, environments, and use patterns.

These workshops articulate applications and values — the *what* and the *why*. To understand *how* users might actually interact with such systems to achieve these goals, we turn to the next section.

4. Elicitation Study

Prior to this work, we, the authors, had our own set of assumptions for how users ought to control a robotic surface. The designers and architects in our speculative workshop had opinions on this as well — some of which ran counter to our own. To gather more concrete evidence regarding natural interaction techniques, we conducted an elicitation study using our prototype robotic wall (Figure 8). Our goal was not only to observe users’ initial actions, but also to understand how they react and adapt when the system makes mistakes.

This requires a slightly different approach to elicitation. Typically, HCI elicitation studies are conducted by presenting users with a set of “referents” — images, mock-ups, or animations that show an object or interface transitioning between states²⁴⁹. Participants are then asked to perform the action that they would have used in order to trigger that state transition.

Our approach draws instead from the “live” Wizard-of-Oz techniques more commonly seen in Human-Robot Interaction literature²⁹⁷. Rather than showing participants a referent, we gave them a physical task to perform with the robotic wall. As they attempt the task, a human operator observes the scene and remotely generates responses. Though it requires more upfront setup, this methodology offers two key advantages. First, it allows us to naturally introduce system errors, and observe how the participants recover. Second, by focusing on a functional goal (as opposed to a singular state transition), we can elicit multi-step interaction sequences, which may unveil latent behavioral patterns.



Figure 8. In our Wizard-of-Oz elicitation study, participants interacted with our shape-changing robotic wall. They were given a series of cards (i) and tasked with placing colored blocks on shelves at varying heights. Participants had to solicit shelves from the wall, both while standing (a, b) and sitting (c). After placing all blocks, and adjusting the heights of the shelves, they had to return the wall to the flat state (d-g). The robotic wall was controlled by human operator, using a graphical interface (h).

Participants and Procedures

We recruited 12 participants for our elicitation study (6 male, 6 female). They ranged in age from 18 to 38, with a median age of 25. Each one visited our lab independently, and each session lasted one hour.

When the participants entered, we first introduced them to our robotic wall (Figure 8). This was a modified version of the robotic surface presented in Chapter 5, now composed of three shape-changing columns that could move and reconfigure themselves in parallel. We showed participants a video of the wall in motion, but importantly, did not demonstrate any potential interaction techniques.

Each participant was given the same task, which was repeated throughout the study: to place a series of colored blocks at targeted positions on the wall (Figure 8a-c). To achieve this, participants would have to solicit a

“shelf” from one or more columns, and adjust the height upwards or downwards if necessary. After all blocks were placed, participants would retrieve them, and then attempt to return the wall to the flat state (Figure 8d-g). For each trial, participants were given a randomized card which showed where the blocks should be placed (Figure 8i).

Participants were told that the wall could see them, hear them, and feel them. In reality, an experimenter was both overseeing and controlling the wall’s actions, using a remote graphical interface. At random intervals, the interface would suggest “errors” (i.e. response delays, incorrect heights), which the experimenter would then incorporate into their performance. Otherwise, the experimenter attempted to respond to each participant’s actions in good faith.

The study consisted of four stages, each with twelve block placements and retrievals. In the first and fourth stages, participants were free to use any interaction modality. In the second and third stages, we introduced a forced choice: a “voice only” mode and a “hands only” mode. Specifically, participants were told that certain sensing capabilities (i.e. audio, or vision/touch) had been deactivated. The ordering of these forced choice stages was randomized for each participant. Within each stage, half of the trials were conducted while the participant was standing up, and the other half were conducted while the participant was sitting in a chair (Figure 8c).

After all stages were completed, we conducted a ten-minute interview. We asked each participant to describe their preferences, frustrations, and recommendations; and then revealed the Wizard-of-Oz component of the study.

Analysis Method

We recorded all user sessions on video. We began our analysis by reviewing these recordings — developing a preliminary codebook by taking coarse notes on the voice, gesture, and touch actions that occurred. Next, we conducted a second, fine-grained pass through the videos. In this pass, we logged timestamps and codes for each action, adding contextual details where necessary. Throughout the analysis, these codes were progressively refined.

This process resulted in over 2,000 identified actions, tagged with a set of roughly 200 individual codes. Afterwards, we grouped these codes hierarchically, ending with 50 primary codes and 194 total subcodes (across all three modalities).

Overview of Findings

There is no “one-size-fits-all” modality for interacting with robotic surfaces. Participants had varying preferences — some of which depended on the physical context (standing vs. sitting), the type of action (gross vs. fine-grained control), and the social environment (public vs. private). But perhaps most notably, many opted for a multimodal approach, fluidly mixing voice, gesture, and touch in order to accomplish the study tasks.

We can begin to visualize this data by plotting participants’ actions on a timeline, as shown in Figures 9 through 11. The plot at the bottom of Figure 9, for instance, shows the sequence of actions taken by Participant 12 throughout the course of the final stage. Note that for some points on the timeline, multiple actions were observed. For instance, a participant may have issued a vocal command (“make a shelf here”) along with an accompanying gesture (pointing at a desired location).

For each participant, we can also calculate a *modal entropy* — a measure of the diversity of actions for each participant. Participant 2, for instance, who primarily interacted with the wall via a small set of touch actions, will have a lower modal entropy than Participant 5, who alternated between voice, gesture, and touch. These calculations are explained in Section 4.6, and are derived from the lower-dimensional action representations introduced in Sections 4.4 and 4.5.

By examining the modal entropies for the first and final stages of the study, we can create four qualitative groupings of participants. The groupings capture the participants’ preferred interaction style, as well as describe how these preferences changed over the course of the experiment (before and after exposure to the two “forced-choice” stages).

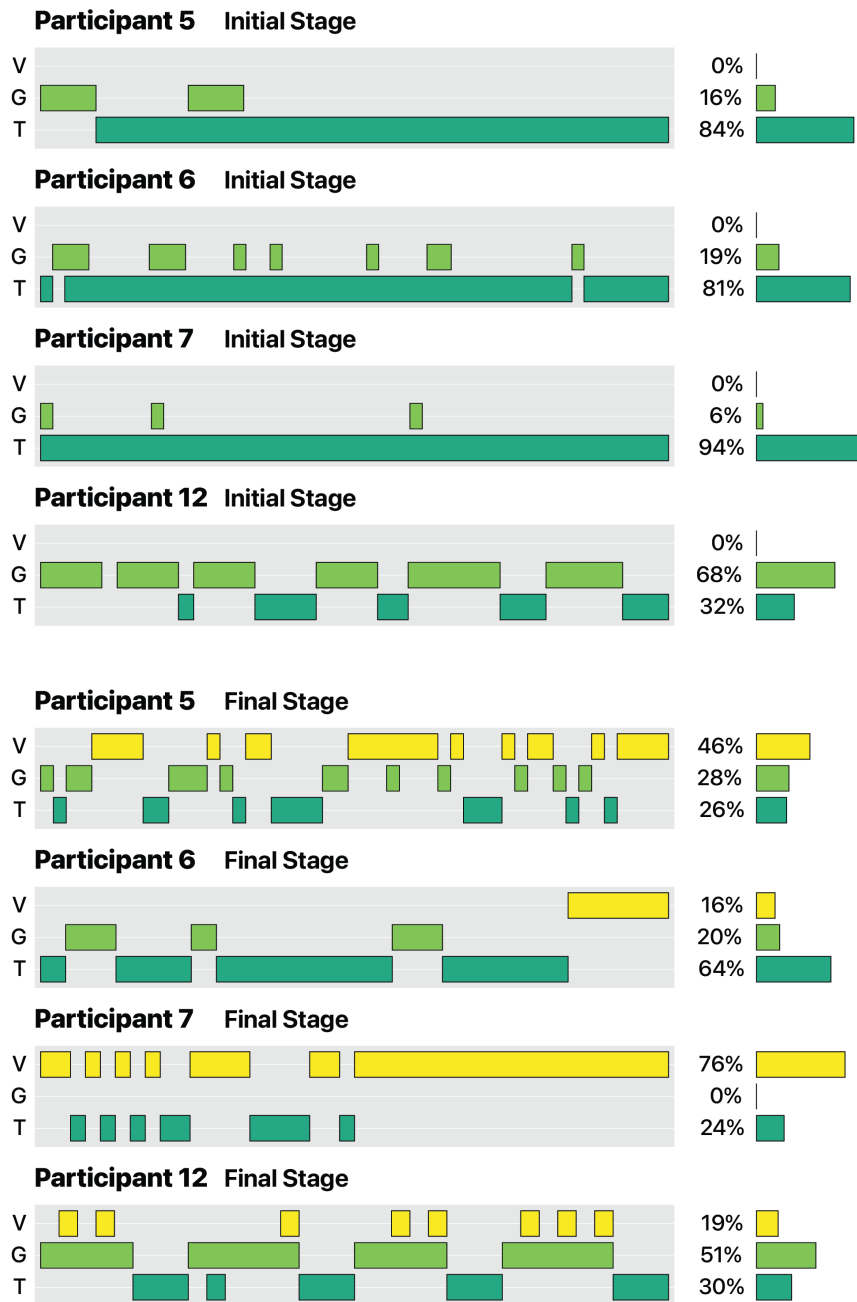


Figure 9. Timeline of vocalizations (V), gestures (G), and touch actions (T) for four participants (P5, P6, P7, P12), during both the initial and final stages of the study. These participants began the study favoring physical interactions, but by the final stage, had adopted more multimodal strategies. The right side of the figure displays the distribution of actions for each participant and stage.

Group 1: Adopted Multi-Modal Preference. The four participants in Figure 9 (P5, P6, P7, P12) began the study by using only physical interaction modes — gesture and touch. However, by the end of the study, they had incorporated voice commands into their repertoire. In the first stage, for instance, Participant 12 attempted to adjust the height of the shelves by holding an object at a desired location, waiting for the shelf to “meet” the object at the target spot. By the end of the study, they chose to first place the objects on the shelves (at any location), and then make

smaller height adjustments with short voice commands (“go up”, “go down”). Similarly, Participant 5 initially treated the wall as something to be physically “molded” — interacting almost exclusively through touch. But in the final stage, they adopted a more diverse style, adding pointing gestures, verbal instructions (“move all the curves to the first marker”), and verbal confirmations (“okay”, “no”). These shifts imply that there may be interaction techniques which, while not always readily apparent, become preferable once users “discover” them.

(Continued on next page.)

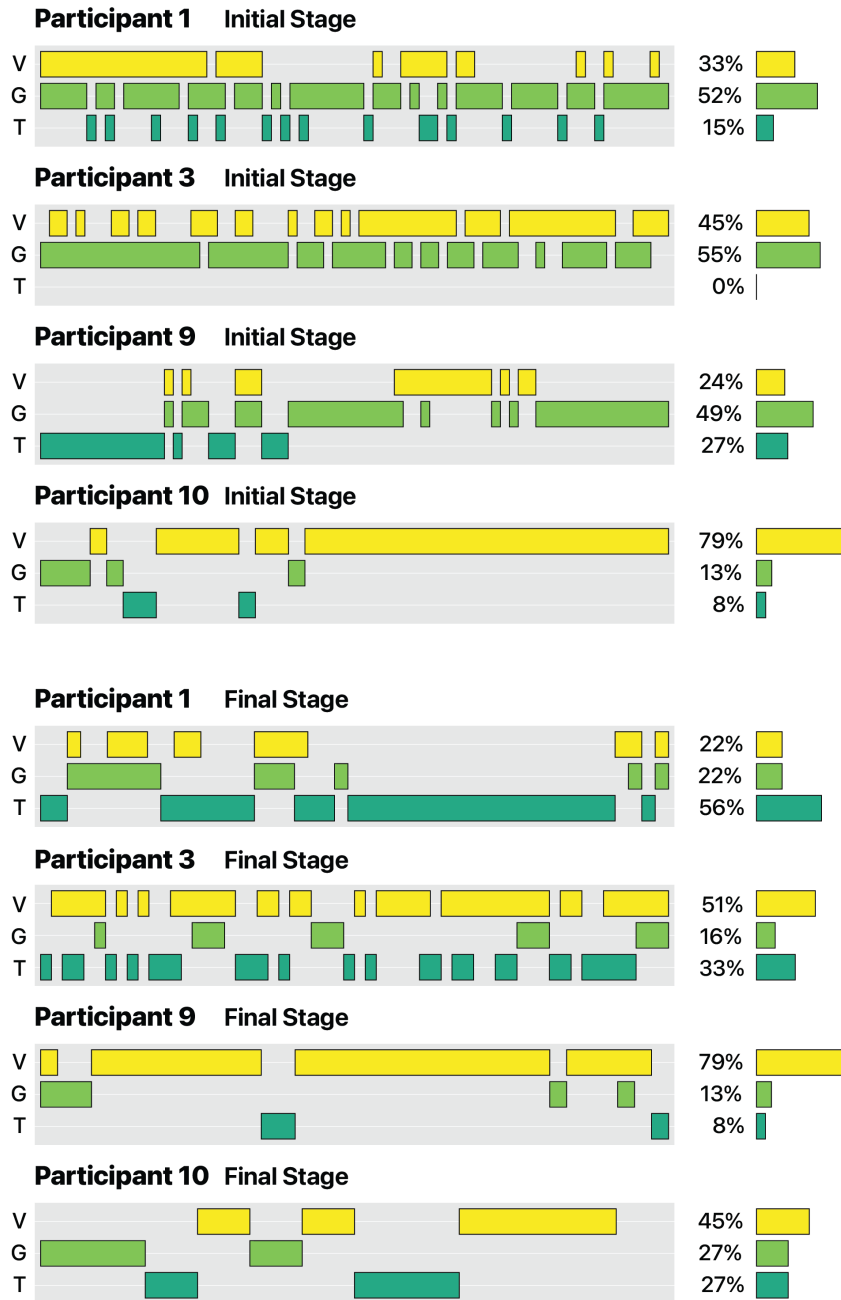


Figure 10. Timeline of vocalizations (V), gestures (G), and touch actions (T) for four participants (P1, P3, P9, P10), during both the initial and final stages of the study. These participants started and finished the study using multimodal strategies. The right side of the figure displays the distribution of actions for each participant and stage.

Group 2: Stable Multi-Modal Preference. Another third of the participants (P1, P3, P9, P10) used both vocal and physical interaction modes from the onset of the study (Figure 10). All retained this multimodal preference, though many altered their distribution of actions as the study progressed. Participant 1, for example, increased their use of touch, particularly when soliciting new shelves from the flat wall. “I could just point to the [shelf] that I wanted to come out,” they said, “I feel it’s more direct for me.” For the same subtask, Participant 9 was drawn in the

opposite direction, using voice commands to communicate intentions while concurrently reaching for objects: “If I have my hands tied... it’s really helpful if I can just verbally let it know which [shelf] I want out.” A system with flexible interaction modes may be able to better accommodate these variable priorities.

(Continued on next page.)

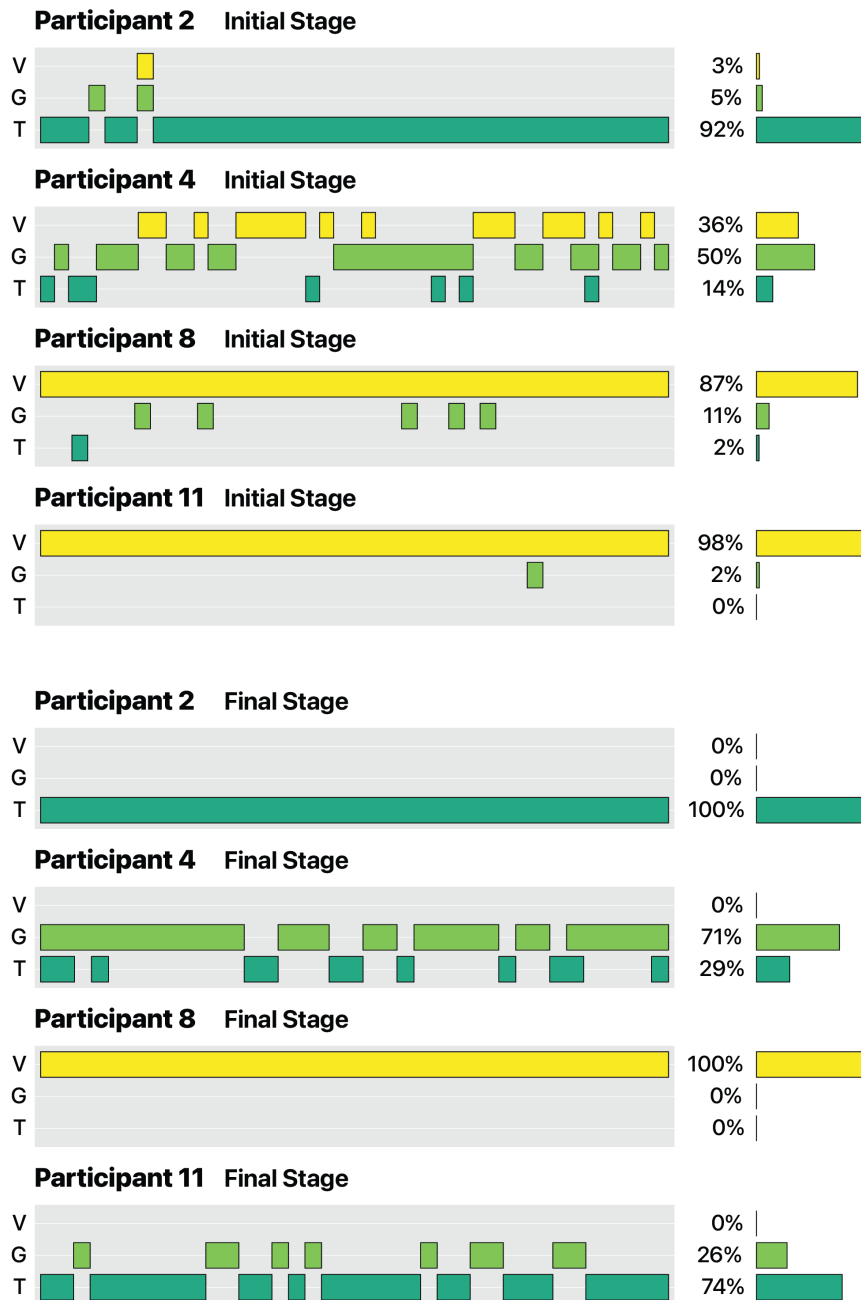


Figure 11. Timeline of vocalizations (V), gestures (G), and touch actions (T) for four participants (P2, P4, P8, P11), during both the initial and final stages of the study. These participants ended the study favoring either vocal actions or physical actions exclusively. The right side of the figure displays the distribution of actions for each participant and stage.

Group 3: Single-Modal Preference. The final group of participants (P2, P4, P8, P11) ended the study favoring one interaction mode — either purely vocal or purely physical (Figure 11). Within this group, only Participant 4 began the study using multiple modes. In the final stage, this participant eliminated vocalizations, explaining that these commands presented a larger cognitive load: “I have to say how many steps [the shelf] has to go down, and it involves some counting.” Participant 11, likewise, switched their primary mode from vocal to physical. The remaining

participants maintained a consistent modal preference, rooted in technical assumptions about the capabilities of the system. “From the start, I already didn’t want to touch it,” said Participant 8, referencing concerns about mechanical stability. Participant 2 assumed the opposite: “I didn’t think about motion [sensing] as a possible factor,” they said, explaining why they opted for purely touch-based interactions instead of gestures. Notably, our forced-choice stages did not distinguish between gesture and touch (they were grouped into one “physical” category), which may explain this reasoning.

Overall, the majority of participants (8 out of 12) ended the study with a clear multimodal preference — demonstrated through recorded actions as well as debriefing interviews. It is worth noting that this multimodality was observed in two forms. The first is in the form of complementary actions — for instance, saying “make a shelf here” while touching a spot on the wall. The second is in the form of sequential actions — for instance, pointing at an out-of-reach object (gesture), then saying “move down” (voice), then retrieving the object, and then double-tapping the shelf in order to dismiss it (touch). As we will see in the following subsections, instead of deriving an explicit mapping from user actions to robotic responses, it can be helpful to interpret these actions as “evidence” towards a particular user intention, upon which a robotic system can act accordingly.

Voice

We recorded and transcribed a total of 848 vocalizations, over the course of the study. Some of these were simple, atomic commands (“higher”, “lower”), while others were more complex (“Column Three, I need a block placement at the same height as Column One”).

To better visualize and understand this dataset, we can embed the transcribed vocalizations in a two-dimensional feature space (Figures 12 and 13). Specifically, we use a pre-trained encoder model, based on MPNet²⁹⁸, to compute a vector that captures the semantic meaning for each vocalization. Then, we apply t-SNE²⁹⁹ to reduce the dimensionality (from 768-D to 2-D), using a Euclidean distance metric.

Vocalization Embeddings

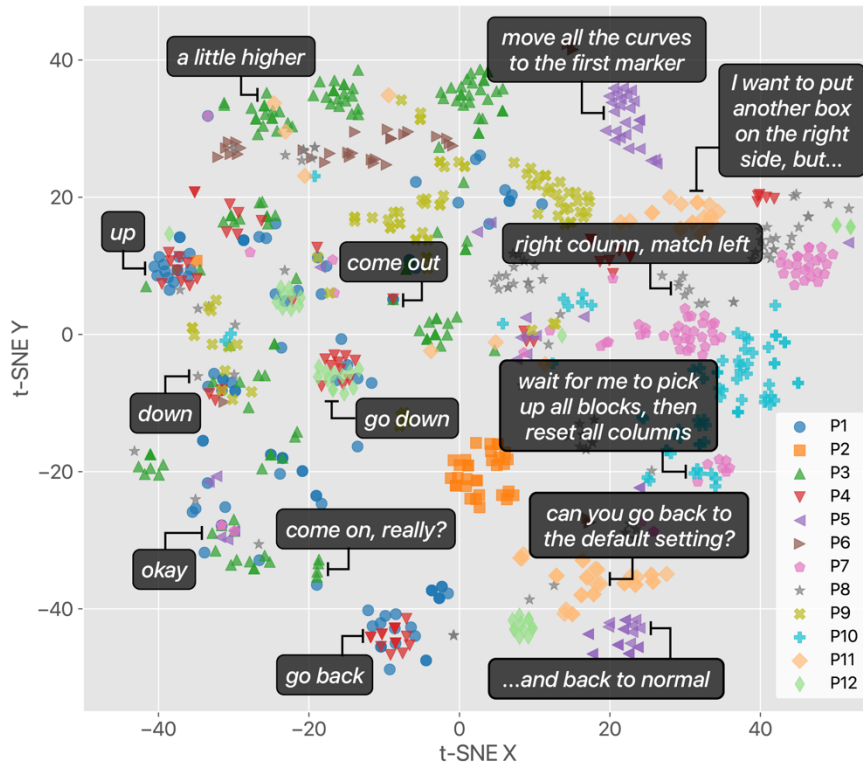


Figure 12. Semantic embeddings for participants' vocalizations during the elicitation study, visualized in a two-dimensional latent space. Vocalizations on the left of the plot tend to be shorter, and have higher overall agreement between participants. Vocalizations on the right of the plot tend to be longer, and reflect more individualized speaking styles.

Vocalization Embeddings, with Superclusters

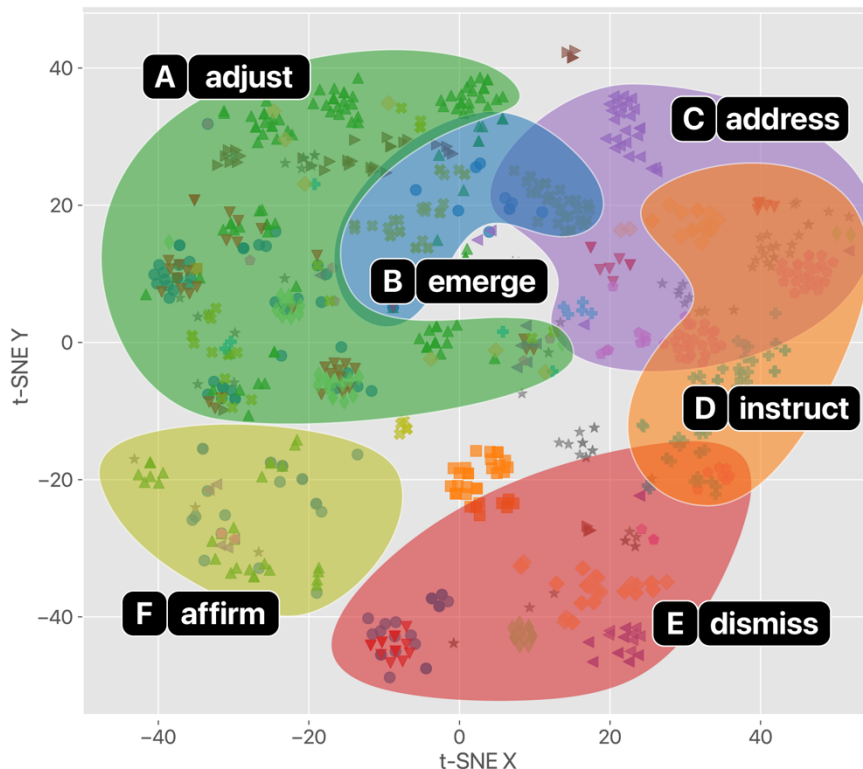


Figure 13. Once participants' vocalizations are plotted, we can observe several superclusters, some of which overlap. Categories include simple instructions for vertical movement (a), commands for shelves to emerge (b) and retract (e), short anthropomorphic reactions to establish common ground (f), and longer, multi-step instructions (d), many of which were addressed (c) to specific columns on the wall.

Plotting these vocalizations, we can begin to examine the broad structure of the dataset. Vocalizations on the left side of the plot tend to be shorter — one to four words, usually describing a location (“here”, “there”), a short command (“go back”, “go up a bit”), or an anthropomorphic response³⁰⁰ (“there you go”, “good job”, “that’s fine”). Vocalizations on the right side of the plot tend to be longer — they include requests (“can you go back to the straight status?”), detailed imperatives (“give me a curve on the second marker for the right column”), grounding statements related to the task (“I am a very short person, so...”), and multi-step instructions.

Notably, the clusters on the left of the plot, which represent shorter utterances, tend to contain vocalizations from a wider range of participants. This suggests a high degree of consensus for simple, atomic commands. Towards the right, as vocalizations become more complex, the clusters show less participant diversity — reflecting more individualized speech patterns. However, even within these areas, there is still a degree of semantic overlap, implying some shared strategies for articulating goals. We examine some of these patterns in the subsections below.

Manipulation Analogs.

Many of the shorter voice commands can be interpreted as coarse substitutions for direct manipulation techniques. Most often, these were used for adjusting the height of a shelf: participants would give instructions such as “a little higher up” or “move all the way down”. Some participants expressed frustration at the imprecision of vocal manipulation, especially compared to more direct, physical methods. Others, however, were generally unbothered by the lack of fine control. These participants resolved uncertainties by referencing spatial landmarks (“go to the second notch from the bottom”) or building off of previous forms (“right column, match left”). To adequately respond to these commands, a system will have to incorporate knowledge of its physical surroundings as well as its physical state.

Addressing and Implicit Context.

Our robotic wall consisted of three columns, and participants could create shelves from any one of the three. To clarify their intentions, many participants chose to address these columns by name (i.e. “left column, make the same shelf as the middle column, please”). However, this was

not intuitive for all. In the voice-only stage, 38% of “new shelf” commands were unaddressed, requiring our human facilitator to interpret the implicit context around them (i.e. gaze, body posture) in order to respond. Context was even more critical for “adjustment” commands, 70% of which were unaddressed. In this latter case, the context was largely temporal — a more ambiguous “adjustment” command was typically preceded by a less ambiguous “new shelf” command.

We should also note that while this addressing scheme is manageable for users in a laboratory setting, it would likely become cumbersome when applied to wider surfaces. A vocally-controlled room-scale system will presumably need to leverage implicit human cues, both spatial and temporal, just as our facilitator did.

Emergence, Dismissal, and Goals.

There were two main approaches to creating new shelves. The first approach is analogous to “sculpting” — participants would request a shelf without a specified location, and then use the manipulation analogs described earlier to refine the form. The second approach was a “one-shot” technique — participants would specify both the shelf and its target location in a single utterance (“middle column, shelf at the topmost position”).

There was a similar dichotomy among dismissal commands. Most participants alternated between action-oriented directives (“retract”, “reset”) and state-based descriptions (“back to normal”, “default position”). However, some ventured further, chaining multiple operations into procedural recipes: “Bring all columns to the bottom, let me pick up the blocks, and then reset them all, once you detect that there's nothing on your wall”.

A few participants chose to articulate abstract goals rather than explicit commands (e.g. “I want to put another box on the rightmost side”). These participants would also reference prior states, saying things like “do the same thing as before” or “return to the previous position.” Such commands assume the system maintains a working memory of past configurations.

Establishing Common Ground.

Throughout the study, participants actively worked to establish mutual understanding with the robotic wall. Confirmatory utterances like “okay, good” and “there you go” served as real-time feedback, signaling when the wall had correctly interpreted their intent. Conversely, when errors occurred, participants would verbally chastise the wall (“no, wrong”) or express frustration (“come on, really?”). This behavior was moderated by social context: one participant admitted they would be “less polite” to the wall in private settings, while others felt self-conscious speaking to an inanimate surface in a public setting.

Several participants attempted to construct a shared vocabulary, assigning names to specific heights (e.g. “alpha”, “bravo”) or numbering shelf locations for future reference. One participant even tried to establish persistent behavioral rules (“If there is nothing on the wall, reset the column”), seeking to offload cognitive burden onto the system itself. During debriefing interviews, participants emphasized the importance of common ground — questioning whether “left” referred to their perspective or the wall’s, and expressing a desire for explicit confirmation that the system was actively listening.

(Continued on next page.)

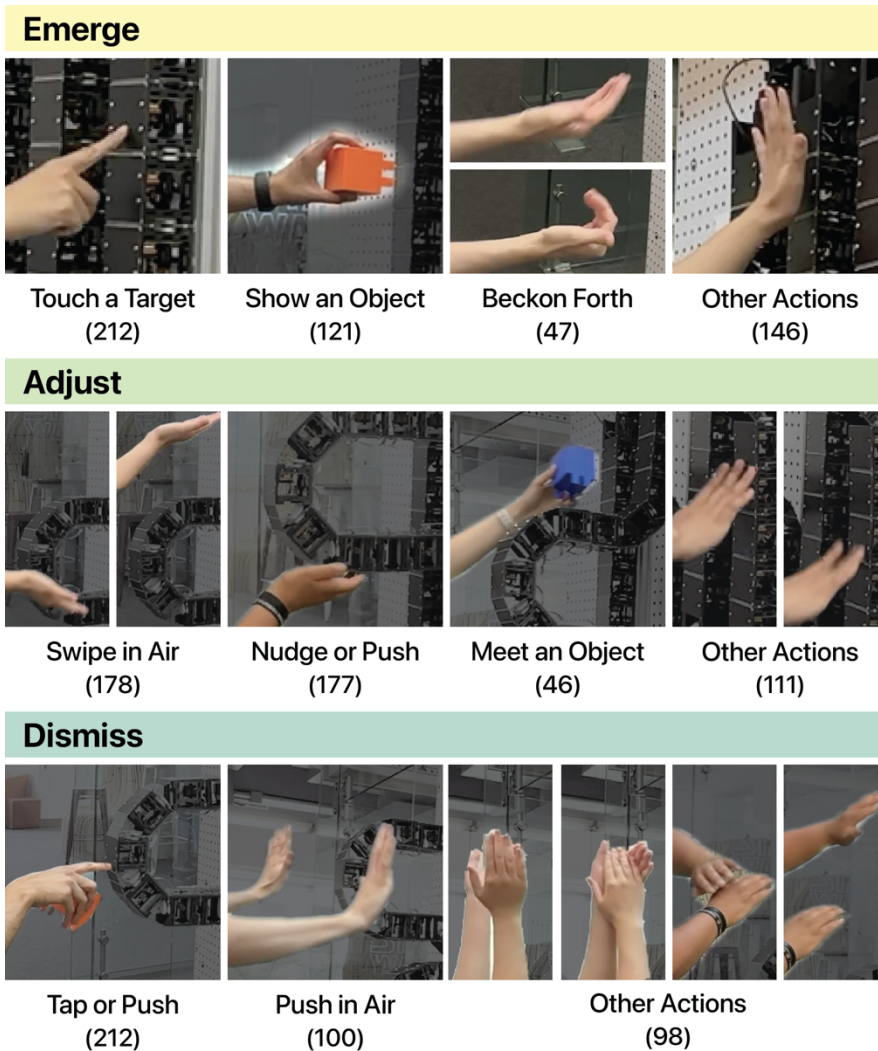


Figure 14. Examples of the most common gesture and touch actions that appeared in our elicitation studies. Raw counts are in parentheses, below each image.

Gesture and Touch

In our dataset, we identified 755 instances of gestures, and 724 touch actions. We define touch actions as hand or arm movements that involve physical contact with the robot wall (tapping, grabbing, pushing, etc.). All other communicative upper body movements were categorized as gestures.

The most common gesture and touch actions are summarized in Figure 14. Broadly speaking, touch actions were more often used for requesting and dismissing shelves, whereas for height adjustments, there was a more even split between modes. Beyond these functional differences, gestures were also more prevalent in multimodal interactions. In the first and final stages (i.e. free choice), 210 vocalizations were accompanied by gestures, whereas only 41 vocalizations were accompanied by touch actions.

We can expand this analysis by plotting these actions in a lower-dimensional feature space, similar to the one described in Section 4.4. Recall that each unique gesture and touch action is tagged with an appropriate subcode. These subcodes, however, can be expanded into descriptive sentences (as shown in Table 1, below), which we can use to perform a semantic analysis (Figures 15 and 16). This allows us to, at a glance, observe the agreement between participants for individual actions, as well as reason about larger behavioral patterns. It is also useful for calculating a modal entropy, described in Section 4.6.

Code	Description
g_point_location	"the user pointed to a panel on the wall"
g_meet_object	"the user held an object above or below a shelf on the wall"
t_push_exterior_low	"the user pushed the bottom of the shelf upward"
t_tap_face_repeated	"the user tapped a panel on the flat part of the wall, repeatedly"
...	...

Table 1. Example codes and corresponding natural-language descriptions (gesture and touch).

As we move up the vertical axis of Figure 15, actions become more “tactile”. At the bottom, we find mid-air sweeping motions and representational gestures performed at a distance from the wall. The middle zone contains gestures that approach but don't quite contact the surface — hands hovering near panels, objects held against shelves, and mimetic pushing motions. The top third consists almost entirely of direct touch actions, with the left cluster focusing on shelf manipulation and the right on interactions with the flat wall surface. We examine some of these clusters in the following pages.

Gesture and Touch Embeddings

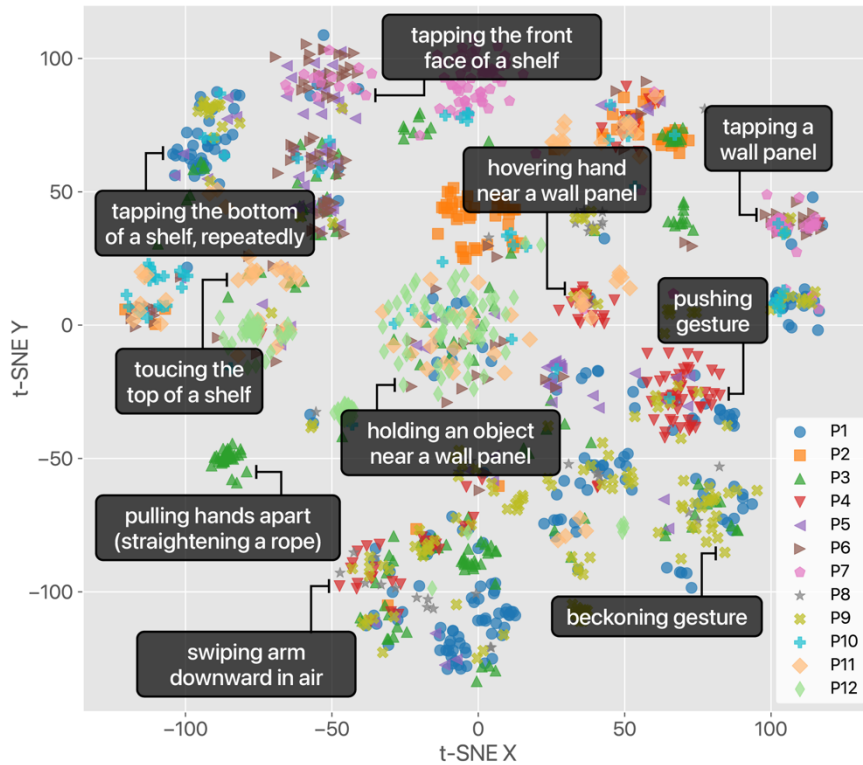


Figure 15. By attaching a text description to each gesture and touch action, we can compute semantic embeddings and visualize them in a two-dimensional latent space. Actions at the top of the plot are more tactile, involving direct manipulation with the wall. Actions near the bottom are gestures performed without contacting the robotic surface. In between are gestures that approach the surface, but don't quite touch it (a pushing gesture beside a deployed shelf, or a hand hovering over a flat panel).

Gesture and Touch Embeddings, with Superclusters

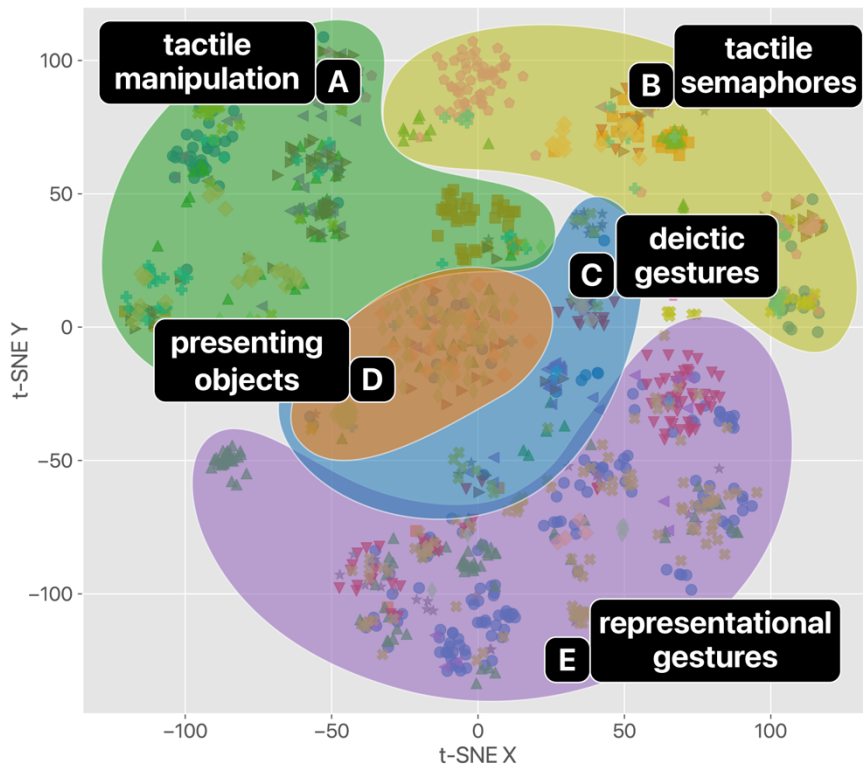


Figure 16. When gesture and touch actions are plotted, several superclusters appear. Categories include tactile "sculpting" actions (a), semaphoric tactile actions like taps and knocks (b), a variety of iconic and metaphoric gestures (e), and deictic gestures such as pointing (c). A subset of deictic gestures are interactions in which users which present objects to the wall (d) — holding blocks near certain locations, where they desired shelves to appear.

Object Recognition and Response.

Several participants expected the wall to recognize and respond to objects held in space. They would bring objects towards the flat surface, or suspend blocks at desired shelf heights, waiting for the wall to “meet” the object with an appropriately positioned shelf. Participant 12 mentioned this explicitly: “The wall should basically follow me,” they said, “start slowly moving, so I know it’s following.”

Direct Manipulation.

Touch interactions clustered into two behavioral patterns. The first involved direct physical manipulation of shelves and columns — participants often pushed, nudged, and tapped shelf edges to adjust heights and dismiss unwanted forms. Some attempted to “pull” shelves out of the wall in the flat state (though due to mechanical constraints, our facilitator had to discourage this). These actions leveraged the physical affordances of the robotic wall. A small number of participants instead borrowed actions from touchscreen vocabularies — “scrolling” flat areas of the wall to alter shelf heights, or pinching the flat surface to solicit new forms.

The most common method of soliciting new shelves was to tap or touch a location — in the physical-only stage, 7 out of 12 participants used this as their primary technique. The remaining 5 participants used a combination of deictic and representational gestures, which we describe below.

Representational Gestures.

Some participants used iconic and metaphoric gestures³⁰¹ to communicate with the wall. The most frequent choice was a “beckoning” gesture — to make shelves emerge, these participants would curl their fingers inward, or sweep their hands towards their body. One participant also attempted to command the wall through hand signs — holding up a number of fingers (one, two, or three) to signify that they were addressing a particular column.

A small number of participants also developed gestures for confirmation and control. Many of these followed a swiping or pointing gesture — for instance, a participant would motion for a shelf to move upward or downward, then hold a fist or put their palms outward when they wanted

the wall to stop. When the wall behaved appropriately, some would give a “thumbs-up” sign. One participant also ventured into mimetic actions – pulling their arms apart (as if straightening a rope) to tell certain columns to “straighten out” and retract.

Deictic Gestures.

In the free-choice stages, deictic gestures (specifically pointing) were often used in conjunction with voice commands. For soliciting shelves, 19% of vocalizations were accompanied by a gesture, and for adjusting shelves, 44% of vocalizations were accompanied by a gesture. Participants would gesture towards spots on the wall where they wanted a shelf to move or appear, or point up or down to indicate a direction.

While sitting, a few participants performed grabbing or reaching gestures to indicate that they wanted to retrieve an object that was located on a high shelf. Instead of directly telling the wall to move down, they expected the wall to interpret this action.

(Continued on next page.)

Modal Entropy

The two-dimensional embeddings in Sections 4.4 and 4.5 provide a helpful way to visualize and reason about the diverse interaction styles that appeared in our study. To further quantify this, we can interpret these plots as latent state spaces, and estimate the distribution of a user’s actions across both vocal and physical modalities. We introduce two metrics: an *across-mode entropy* that captures the balance of user activity between different modalities, and a *within-mode entropy* that describes the variety of user behaviors for any single interaction mode.

To begin, we compute two probability density functions for each participant. We achieve this through Kernel Density Estimation — within each latent space, we apply a Gaussian kernel to the participant’s two-dimensional action embeddings (Figure 17). The kernel bandwidth is unique for each participant, and is calculated via Scott’s Rule³⁰², scaled by a factor of 0.4 to better capture the local structure of the clusters.

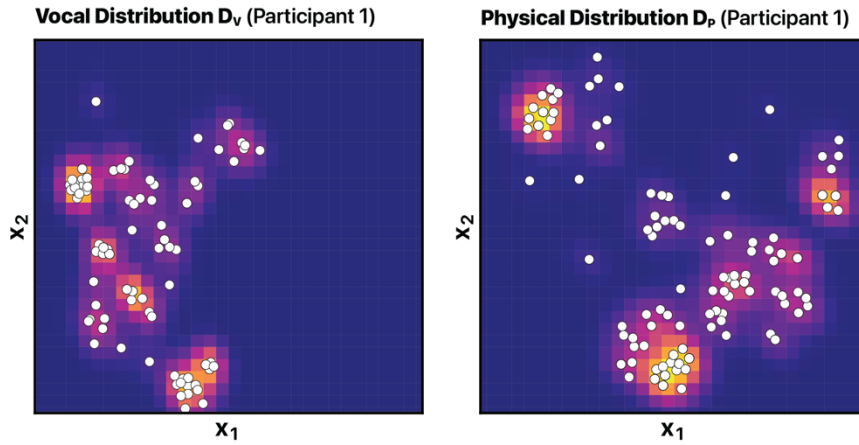


Figure 17. We can discretize the latent spaces in Figures X and Y, and interpret them as two non-overlapping state spaces. After applying a Gaussian kernel to participant actions within these spaces, we obtain probability mass functions for each participant’s vocal (a) and physical (b) actions. The plots above show distributions for one participant (P1), across all stages. White circles represent actual, observed actions.

This process gives us one distribution for a participant’s vocal actions, D_v , and one for their physical actions, D_p . Formally, we can express this as:

$$D(x) = \frac{1}{n} \sum_{i=1}^n K_h(x - x_i)$$

In the expression above, x refers to a coordinate in the two-dimensional latent space, n is the total number of actions for a participant within that space, and K_h is the Gaussian kernel, with bandwidth h .

After obtaining the distributions D_V and D_P , we can discretize them, evaluating each function on a uniform 32-by-32 grid. For each grid cell k , we compute $D(x_k)$ (where x_k is the coordinate at the center of cell k) and multiply this value by the cell area:

$$p_k = D(x_k) \cdot \Delta A$$

After normalizing across the 1024 cells, we can then calculate the Shannon entropy for each modality:

$$H = - \sum_{k=1}^m p_k \log_2(p_k) , m = 1024$$

This gives us an entropy H_V for vocal actions, and an entropy H_P for physical actions. Such metrics can help us distinguish between interaction styles that, at a coarse view, may appear to be similar. For instance, in the final stage of the study, Participants 1 and 12 both used physical actions around 80% of the time (78% for P1, 81% for P12). However, the physical action entropy for P1 is nearly double that of P12 — reflecting their wide use of representational gestures, tactile manipulation, deictics, and so on.

We can further characterize a participant's interaction style by estimating their overall probability of performing a vocal or physical action. For a given number of vocal actions n_V and physical actions n_P in a free-choice stage, we obtain:

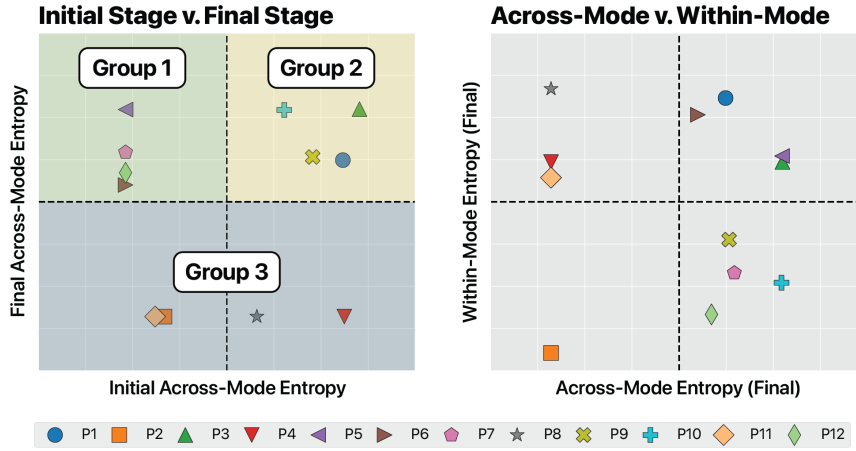
$$\hat{P}_V = \frac{n_V}{n_V + n_P} , \hat{P}_P = \frac{n_P}{n_V + n_P}$$

Then, we can use these estimations to calculate an *across-mode entropy*, capturing the uncertainty surrounding a user's modality choice:

$$H_{\text{across}} = -\hat{P}_V \log_2(\hat{P}_V) - \hat{P}_P \log_2(\hat{P}_P)$$

If a participant only uses vocal actions, or only uses physical actions, then their across-mode entropy will be zero. If they frequently alter between modes, then this entropy will increase.

Figure 18.



Left: Each participant's across-mode entropies for the initial and final stages are plotted against each other. Participants in Group 1 began the study using only one mode, but migrated to multimodal strategies by the end. Participants in Group 2 both began and ended the study using multimodal strategies. Participants in Group 3 began the study with varied styles, but all settled on a single interaction mode by the end.

Right: Each participant's final across-mode entropy is plotted against their final within-mode entropy. A participant could have a lower across-mode entropy (left half of plot), but still have used a wide variety of actions within that mode (top half of plot).

Figure 18-left plots the across-mode entropy for participants in both the initial and final stages of the study. Three clusters appear, which map directly to the qualitative groupings described in Section 4.3. Participants in the top left quadrant, for instance, began the study with a low across-mode entropy (using only physical actions), and ended the study with a high across-mode entropy (mixing speech, gesture, and touch).

It can also be informative to derive a singular *within-mode entropy* that captures both the vocal H_V and physical H_P entropies computed in Equation 3. For this, we use a simple weighted sum:

$$H_{\text{within}} = \hat{P}_V \cdot H_V + \hat{P}_P \cdot H_P$$

Figure 18-right compares the across-mode and within-mode entropies for participants in the final stage of the study, yielding additional insights. Participants in the top-left quadrant stuck to one modality, but had a rich repertoire of actions within that modality (P8, for instance, was a "vocal specialist"). Participants in the bottom-right quadrant were the opposite —

while they frequently switched between vocal and physical modes, their library of actions within each mode was more limited.

These metrics are most meaningful when used to compare participants within a population. To facilitate these comparisons, the entropy values plotted in Figure 18 are standardized (converted to z-scores). For a participant j , we can calculate the standardized across-mode entropy $z(H_a^j)$ and within-mode entropy $z(H_w^j)$, using the mean and standard deviation of the participant entropies in the population:

$$z(H_a^j) = \frac{H_a^j - \mu_{H_a}}{\sigma_{H_a}}, \quad z(H_w^j) = \frac{H_w^j - \mu_{H_w}}{\sigma_{H_w}}$$

It is important to note that these measures are sensitive to the dimensionality reduction technique. In particular, the t-SNE embeddings that we use are dependent on hyperparameters, which must be tuned in order to balance the preservation of local and global structures in the data. While we found that we were able to obtain descriptive embeddings with this method, future researchers may wish to consider TriMap³⁰³ or PaCMAP³⁰⁴, which have been proposed as more robust to changes in parameters³⁰⁵.

Despite these caveats, we see the metrics as useful diagnostic tools for identifying unpredictability within the context of a well-defined task. A low-entropy user is stylistically consistent, while a high-entropy user employs a more varied, multimodal repertoire. An intelligent environment could perhaps recognize these distinct styles and adjust its own collaborative strategy.

5. So, What Should You Build?

A key aim of our work was to uncover well-defined research directions for advancing the state-of-the-art of shape-changing environments. Drawing on evidence from our design workshops and elicitation study, we propose five opportunities for future investigators.

“Raise” a Robotic Environment

Shape-changing environments are a bit of an alien concept, and while they excited the designers and architects in our workshops, the participants also recognized that such structures might initially be jarring. In addition to an unfamiliar physical presence, a newly installed robotic wall might feel invasive if it “knows too much” about a user right out-of-the-box. To build trust gradually, participants suggested a more developmental approach — one likened it to “training a pet”. The system would begin with deliberately limited capabilities, but progressively learn skills and user preferences through repeated interactions.

Behavioral patterns from our elicitation study suggest that users might find this trajectory intuitive. Many participants expanded their interaction repertoires over time, attempting to establish shared vocabularies and teach persistent rules. A “raisable” environment would formalize this co-evolution. While slower to get up and running, users might feel greater ownership over the final robotic behavior, as it would be something that they taught the system themselves²⁹⁶.

Researchers that pursue this direction will have to tackle challenges in both technical feasibility (i.e. persistent memory, one-shot learning), and interaction design. Neural models for continual learning, while growing increasingly advanced³⁰⁶, have still been shown to be unreliable in HRI contexts, and trust drops significantly when robots are forgetful²⁷⁷. Users preferred teaching methods are also unexplored within this context — in particular, the question of how users could force the system to “un-learn” misaligned behaviors.

Directly Malleable Surfaces

It is important for users to maintain authority over robotic environments. Workshop participants explicitly valued a form of tactile control, imagining scenarios where they could manually lift office dividers or physically reprimand misbehaving surfaces: “You could just jerk it back to where you want it to be and be like: ‘No!’”. Users embraced this instinct in our elicitation study as well: during the final stage, the majority of participants (9 out of 12) used touch actions at least a quarter of the time.

They pushed shelf edges to adjust heights, tapped surfaces to dismiss forms, and attempted to physically mold the wall.

A promising research direction involves surfaces that can be manually sculpted, such as the mesh presented in Chapter 4. More sophisticated variants of these structures might transition between malleable states for user control, and rigid states for structural stability. Such systems will need to distinguish between deliberate reshaping and incidental contact. They may also need to infer a user’s global-state intentions from their localized manipulations, in order to more fluently accommodate user behavior.

Crafting Legible Intent

When we debriefed participants after our elicitation study, many brought up the importance of establishing common ground. In particular, because there would often be a small delay while our robotic wall reconfigured its internal constraints, participants suggested that subtle changes in lighting might help convey that the wall was still “paying attention” to vocal or gestural commands. Concerns over legibility extended to physical motion as well. Early in the study, one participant flinched at an abrupt shelf deployment, remarking that “it felt like I was being attacked.”

While motion legibility is well-studied for robotic manipulators³⁰⁷ and mobile platforms³⁰⁸, architectural-scale systems face a different set of challenges. They are non-anthropomorphic (and can not establish joint attention through gaze²⁷⁴) they must telegraph intent at multiple locations in space, and it is unclear how users would respond to more naturalistic means of communication (artificial speech may not always be preferred²⁷⁹). Findings from our design workshops echo the known trust-building benefits of transparent behavior³⁰⁹, but for cyber-physical architecture, the preferred means of transparency must still be uncovered.

Co-Design for Accessibility

While our design workshops surfaced compelling accessibility scenarios (adjustable counters for wheelchair users, guidance systems for physical therapy), these concepts emerged from participants without lived experience of disability, or formal training in the area. To engage with

these ideas more seriously, researchers should consider centering disabled users as co-designers of shape-changing environments.

Aspects of our elicitation study hint at some of the complexity ahead. Seated participants often preferred voice and gesture over touch, suggesting that proximity and physical capability reshape one's interaction preferences. Still, these preliminary observations are not substitutes for direct engagement with disability communities, who bring essential expertise about their own needs, and the social dynamics of assistive technology^{310,311}.

Future work could employ participatory design methods to explore how shape-changing environments might serve households with multiple, intersecting access needs — for instance, a family where one member uses a wheelchair, another has low vision, and a child has sensory processing differences. Such co-design efforts must navigate the friction that our workshops revealed between personalization and shared ownership, ensuring that adaptations don't stigmatize or inconvenience other users. Instead, this research direction should embrace “universal design”³¹² — an environment that is built to be flexible from the ground up is often preferable to one that is dotted with retrofitted accommodations.

Large-Scale, Load-Bearing Structures

Deployable and configurable seating emerged as one of the most common applications across our design workshops. Participants envisioned on-demand benches in transit hubs, adaptive chairs for growing children, and collapsible furniture in small apartments. Yet our prototype, like most shape-changing surfaces, cannot support human body weight. Those that do, often trade structural stability for geometric resolution³¹³ or configuration speed⁶⁰. Creating structures that are simultaneously dynamic, load-bearing, and scalable remains an open engineering challenge.

6. Conclusion: Implicit Context Necessitates a Model of User Intent

Cyber-physical architecture occupies a distinct niche within the broader robotic ecosystem. Unlike field or industrial robots, designers do not often imagine shape-changing walls as *autonomous* — executing a set of goals, largely independent of the humans around them. However, they do envision them as *agentic* — able to respond intelligently, or even proactively, to human needs and environmental triggers. Even in situations that are purely user-initiated, these robotic surfaces are expected to react in ways that leverage implicit context and overcome unstated instructions. Since these robots are one with the room, after all, designers perhaps assume that the robotic room “already knows” what’s happening within it.

This notion was reinforced by the results of our elicitation study. When debriefed (and sometimes during the study itself), users often mentioned that they would like the robotic wall to infer their intentions from the current context. “If I’m sitting,” said one participant, “it should always [create a shelf] at the lowest level.” Importantly, this inference goes beyond scene understanding — interpreting a user’s moment-to-moment interactions also requires knowledge of the temporal context, physical state, and co-occurrent actions (both vocal and physical). A simple mapping of human cues to robotic responses is likely insufficient for these tasks.

What, then, does this imply for the design of room-scale robotic interactions? When asked, users tend to *say* that they desire a high level of control over these systems. However, in practice, they omit critical context and resist the cognitive burden of explicit specification. We argue, therefore, that these user interactions should be understood not as direct commands, but instead, as evidence that feeds into a dynamic, probabilistic model of user intent.

Such intent models are not new to the field of HCI. Early work by Horvitz^{314,315} demonstrated Bayesian networks for inferring user goals, and gesture recognition systems have long employed Hidden Markov Models for temporal disambiguation³¹⁶. Probabilistic methods have proven effective for handling input uncertainties³¹⁷. However, many systems either process modalities through separate pipelines (before late-state

fusion), or employ end-to-end models that map sensor data directly to actions without maintaining belief states.

These approaches may miss the temporal dependencies that our elicitation study revealed. For instance, the vast majority (70%) of “shelf adjustment” commands were unaddressed, relying entirely on context established by previous “new shelf” commands. Users also punctuated interactions with utterances that served as checkpoints (e.g. “good job” or “no, wrong”), establishing common ground that shaped how subsequent actions should be interpreted. These patterns suggest that user actions build meaning *cumulatively*. Each action (both implicit and explicit) provides evidence that refines the context for interpreting the next.

Cyber-physical architecture is uniquely positioned to leverage these models, in that it is persistent and omnipresent — unlike a traditional interface, it can begin reasoning about user intentions before a user approaches it or calls it to action. A shape-changing wall, for instance, might notice a person repeatedly glancing at a high shelf while carrying a bag, building confidence in the latent intent to “retrieve something” before any direct interaction occurs. A well-calibrated system could offer help when confidence is high, while deferring to explicit commands when intentions remain ambiguous.

Moving forward, the design of robotic environments must embrace this complexity. The eleven values extracted from our workshops — from functional to relational — cannot all be achieved through simple stimulus-response mappings. Neither can the multimodal fluency that participants sought in our elicitation study. Instead, these findings point towards the adoption of modality-agnostic frameworks, treating every user action as probabilistic evidence within a broader temporal and spatial context. This would enable robotic environments to build understanding from the full spectrum of human signals (spoken, gestural, tactile), and the wider surroundings in which they appear.

Acknowledgements

Thank you to my advisors, Scott Hudson and Alexandra Ion, for their unwavering support throughout my time at Carnegie Mellon. I can't think of any other lab(s) that I'd rather have been a part of!

Thank you to my committee members, Nikolas Martelaro and Nayda Peek, for their advice and assistance during the writing of this document.

Thank you to my many colleagues and collaborators, without whom the research in this document would not be possible!

Thank you to my friends — from Pittsburgh, from Brooklyn, and from Maryland (the day one crew!).

Thank you to Diana (and Niko), for turning the final stretch of the PhD into the best one!

Thank you to my parents, for everything! And thank you to my sister, Sara (who still beat me to a PhD, even though I arguably had a head start)

References

1. Weiser, M. The computer for the 21st century. *SIGMOBILE Mob. Comput. Commun. Rev.* **3**, 3–11 (1999).
2. Weiser, M. & Brown, J. S. Designing calm technology. *PowerGrid Journal* **1**, 75–85 (1996).
3. Abowd, G. D. The Internet of Materials: A Vision for Computational Materials. *IEEE Pervasive Computing* **19**, 56–62 (2020).
4. McEvoy, M. A. & Correll, N. Materials that couple sensing, actuation, computation, and communication. *Science* **347**, (2015).
5. Norman, D. *The Design of Everyday Things: Revised and Expanded Edition*. (Basic books, 2013).
6. Gaver, W. W. Technology affordances. in *Proceedings of the SIGCHI conference on Human factors in computing systems Reaching through technology - CHI '91* 79–84 (ACM Press, New Orleans, Louisiana, United States, 1991). doi:10.1145/108844.108856.
7. Breazeal, C., Dautenhahn, K. & Kanda, T. Social Robotics. in *Springer Handbook of Robotics* (eds Siciliano, B. & Khatib, O.) 1935–1972 (Springer International Publishing, Cham, 2016). doi:10.1007/978-3-319-32552-1_72.
8. Hoffman, G. & Ju, W. Designing robots with movement in mind. *J. Hum.-Robot Interact.* **3**, 91–122 (2014).
9. Paepcke, S. & Takayama, L. Judging a bot by its cover: An experiment on expectation setting for personal robots. in *2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* 45–52 (2010). doi:10.1109/HRI.2010.5453268.
10. Kwon, M., Jung, M. F. & Knepper, R. A. Human expectations of social robots. in *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* 463–464 (2016). doi:10.1109/HRI.2016.7451807.
11. Washburn, A., Adeleye, A., An, T. & Riek, L. D. Robot Errors in Proximate HRI: How Functionality Framing Affects Perceived Reliability and Trust. *J. Hum.-Robot Interact.* **9**, 1–21 (2020).
12. Yao, L. *et al.* PneuUI: pneumatically actuated soft composite materials for shape changing interfaces. in *Proceedings of the 26th annual ACM symposium on User interface software and Technology* 13–22 (2013).
13. Alexander, J. *et al.* Grand challenges in shape-changing interface research. in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* 1–14 (2018).
14. Rus, D. & Tolley, M. T. Design, fabrication and control of origami robots. *Nature Reviews Materials* **3**, 101–112 (2018).
15. Kim, S.-J., Lee, D.-Y., Jung, G.-P. & Cho, K.-J. An origami-inspired, self-locking robotic arm that can be folded flat. *Sci. Robot.* **3**, eaar2915 (2018).
16. Son, H., Park, Y., Na, Y. & Yoon, C. 4D Multiscale Origami Soft Robots: A Review. *Polymers* **14**, 4235 (2022).

17. Kalantari, S., Pourjabar, S., Xu, T. B. & Kan, J. Developing and user-testing a “Digital Twins” prototyping tool for architectural design. *Automation in Construction* 135, 104140 (2022).
18. Gulay, E. & Lucero, A. Understanding the Role of Physical and Digital Techniques in the Initial Design Processes of Architecture. in *IFIP Conference on Human-Computer Interaction* 312–329 (Springer, 2021).
19. Kinetic Brick Wall | 2019. *HYPERSONIC* <https://www.hypersonic.cc/brick-wall>.
20. Hong, J., Wang, X., Everitt, A. & Roudaut, A. Interacting with actuated walls: Exploring applications and input types. *International Journal of Human-Computer Studies* 172, 102986 (2023).
21. Albarrak, L., Metatla, O. & Roudaut, A. Using Virtual Reality and Co-Design to Study the Design of Large-scale Shape-Changing Interfaces. in *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* 1–17 (Association for Computing Machinery, New York, NY, USA, 2023). doi:10.1145/3544548.3581144.
22. Ishii, H. & Ullmer, B. Tangible bits: towards seamless interfaces between people, bits and atoms. in *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems* 234–241 (Association for Computing Machinery, New York, NY, USA, 1997). doi:10.1145/258549.258715.
23. Ishii, H. The tangible user interface and its evolution. *Commun. ACM* 51, 32–36 (2008).
24. Ishii, H. *et al.* ambientROOM: integrating ambient media with architectural space. in *CHI 98 Conference Summary on Human Factors in Computing Systems* 173–174 (ACM, Los Angeles California USA, 1998). doi:10.1145/286498.286652.
25. Gershenfeld, N. A. *When Things Start to Think*. (Macmillan, 1999).
26. Gershenfeld, N., Krikorian, R. & Cohen, D. The Internet of Things. *Scientific American* 291, 76–81 (2004).
27. Dourish, P. *Where the Action Is*. (MIT press Cambridge, 2001).
28. Gross, M. D. & Green, K. E. Architectural robotics, inevitably. *Interactions* 19, 28–33 (2012).
29. Grönvall, E., Kinch, S., Petersen, M. G. & Rasmussen, M. K. Causing commotion with a shape-changing bench: experiencing shape-changing interfaces in use. in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* 2559–2568 (2014).
30. Grønbæk, J. E., Korsgaard, H., Petersen, M. G., Birk, M. H. & Krogh, P. G. Proxemic transitions: designing shape-changing furniture for informal meetings. in *Proceedings of the 2017 CHI conference on human factors in computing systems* 7029–7041 (2017).
31. Grønbæk, J. E., Rasmussen, M. K., Halskov, K. & Petersen, M. G. KirigamiTable: Designing for proxemic transitions with a shape-changing tabletop. in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* 1–15 (2020).
32. Fujita, K., Suzuki, A., Takashima, K., Ikematsu, K. & Kitamura, Y. TiltChair: manipulative posture guidance by actively inclining the seat of an office chair. in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* 1–14 (2021).
33. Nguyen, B. V. D., Han, J. & MOERE, A. V. Towards Responsive Architecture that Mediates Place: Recommendations on How and When an Autonomously Moving Robotic Wall Should Adapt a Spatial Layout. *Proceedings of the ACM on Human-Computer Interaction* 6, 1–27 (2022).

34. Takashima, K. *et al.* Study and Design of a Shape-Shifting Wall Display. in *Proceedings of the 2016 ACM Conference on Designing Interactive Systems* 796–806 (ACM, Brisbane QLD Australia, 2016). doi:10.1145/2901790.2901892.
35. Onishi, Y., Takashima, K., Higashiyama, S., Fujita, K. & Kitamura, Y. WaddleWalls: Room-scale Interactive Partitioning System using a Swarm of Robotic Partitions. in *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology* 1–15 (2022).
36. Suzuki, R. *et al.* Roomshift: Room-scale dynamic haptics for vr with furniture-moving swarm robots. in *Proceedings of the 2020 CHI conference on human factors in computing systems* 1–11 (2020).
37. Yixian, Y. *et al.* Zoomwalls: Dynamic walls that simulate haptic infrastructure for room-scale vr world. in *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* 223–235 (2020).
38. Hoffman, G. & Breazeal, C. Achieving fluency through perceptual-symbol practice in human-robot collaboration. in *Proceedings of the 3rd ACM/IEEE international conference on Human robot interaction* 1–8 (Association for Computing Machinery, New York, NY, USA, 2008). doi:10.1145/1349822.1349824.
39. Mok, B. K.-J., Yang, S., Sirkin, D. & Ju, W. A place for every tool and every tool in its place: Performing collaborative tasks with interactive robotic drawers. in *2015 24th IEEE international symposium on robot and human interactive communication (RO-MAN)* 700–706 (IEEE, 2015).
40. Mok, B. K., Yang, S., Sirkin, D. & Ju, W. Empathy: interactions with emotive robotic drawers. in *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction* 250–251 (Association for Computing Machinery, New York, NY, USA, 2014). doi:10.1145/2559636.2563720.
41. Vázquez, M., Steinfeld, A., Hudson, S. E. & Forlizzi, J. Spatial and other social engagement cues in a child-robot interaction: effects of a sidekick. in *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction* 391–398 (ACM, Bielefeld Germany, 2014). doi:10.1145/2559636.2559684.
42. Forlizzi, J., Disalvo, C., Zimmerman, J., Mutlu, B. & Hurst, A. The SenseChair: The Lounge Chair as an Intelligent Assistive Device for Elders. *Gain: Journal of Business and Design* (2006).
43. Knight, H., Lee, T., Hallawell, B. & Ju, W. I get it already! the influence of ChairBot motion gestures on bystander response. in *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)* 443–448 (2017). doi:10.1109/ROMAN.2017.8172340.
44. Agnihotri, A. & Knight, H. Persuasive ChairBots: A (Mostly) Robot-Recruited Experiment. in *2019 28th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)* 1–7 (2019). doi:10.1109/RO-MAN46459.2019.8956262.
45. Yang, S. *et al.* Experiences developing socially acceptable interactions for a robotic trash barrel. in *2015 24th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)* 277–284 (IEEE, 2015).
46. Bu, F., Mandel, I., Lee, W.-Y. & Ju, W. Trash Barrel Robots in the City. in *Companion of the 2023 ACM/IEEE International Conference on Human-Robot Interaction* 875–877 (ACM, Stockholm Sweden, 2023). doi:10.1145/3568294.3580206.
47. Swaminathan, S. *et al.* OptiStructures: Fabrication of Room-Scale Interactive Structures with Embedded Fiber Bragg Grating Optical Sensors and Displays. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 1–21 (2020).
48. Zhang, Y., Yang, C. (Jack), Hudson, S. E., Harrison, C. & Sample, A. Wall++: Room-Scale Interactive and Context-Aware Sensing. in *Proceedings of the 2018*

- CHI Conference on Human Factors in Computing Systems* 1–15 (ACM, Montreal QC Canada, 2018). doi:10.1145/3173574.3173847.
49. Yoshida, T. *et al.* Flexel: A Modular Floor Interface for Room-Scale Tactile Sensing. in *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology* 1–12 (ACM, Bend OR USA, 2022). doi:10.1145/3526113.3545699.
 50. Hansen, K. L. *et al.* FeetBack: Providing Haptic Directional Cues Through a Shape-changing Floor. in *Nordic Human-Computer Interaction Conference* 1–10 (Association for Computing Machinery, New York, NY, USA, 2022). doi:10.1145/3546155.3546653.
 51. Swaminathan, S., Yim, Y., Hudson, S. E., Bennett, C. L. & Carrington, P. From Tactile to NavTile: Opportunities and Challenges with Multi-Modal Feedback for Guiding Surfaces during Non-Visual Navigation. in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* 1–13 (ACM, Yokohama Japan, 2021). doi:10.1145/3411764.3445716.
 52. Zou, H., Zhou, Y., Yang, J. & Spanos, C. J. Device-free occupancy detection and crowd counting in smart buildings with WiFi-enabled IoT. *Energy and Buildings* 174, 309–322 (2018).
 53. Swaminathan, S. *et al.* Input, Output and Construction Methods for Custom Fabrication of Room-Scale Deployable Pneumatic Structures. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 3, 1–17 (2019).
 54. Leithinger, D., Follmer, S., Olwal, A. & Ishii, H. Shape displays: Spatial interaction with dynamic physical form. *IEEE computer graphics and applications* 35, 5–11 (2015).
 55. Follmer, S., Leithinger, D., Olwal, A., Hogge, A. & Ishii, H. inFORM: dynamic physical affordances and constraints through shape and object actuation. in *Uist* vol. 13 2501–988 (Citeseer, 2013).
 56. Teng, S.-Y. *et al.* Tilepop: Tile-type pop-up prop for virtual reality. in *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* 639–649 (2019).
 57. Yoshida, S., Sun, Y. & Kuzuoka, H. PoCoPo: Handheld pin-based shape display for haptic rendering in virtual reality. in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* 1–13 (2020).
 58. Siu, A. F., Gonzalez, E. J., Yuan, S., Ginsberg, J. B. & Follmer, S. Shapeshift: 2D spatial manipulation and self-actuation of tabletop shape displays for tangible and haptic interaction. in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* 1–13 (2018).
 59. Suzuki, R. *et al.* LiftTiles: constructive building blocks for prototyping room-scale shape-changing interfaces. in *Proceedings of the Fourteenth International Conference on Tangible, Embedded, and Embodied Interaction* 143–151 (2020).
 60. Je, S. *et al.* Elevate: A walkable pin-array for large shape-changing terrains. in *Proceedings of the 2021 CHI Conference on human Factors in Computing Systems* 1–11 (2021).
 61. Zarate, J. J., Gudozhnik, O., Ruch, A. S. & Shea, H. Keep in touch: portable haptic display with 192 high speed taxels. in *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems* 349–352 (2017).
 62. Pelrine, R. *et al.* Diamagnetically levitated robots: An approach to massively parallel robotic systems with unusual motion properties. in *2012 IEEE International Conference on Robotics and Automation* 739–744 (IEEE, 2012).
 63. Suzuki, R., Kato, J., Gross, M. D. & Yeh, T. Reactile: Programming swarm user interfaces through direct physical manipulation. in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* 1–13 (2018).

64. Weiss, M., Schwarz, F., Jakubowski, S. & Borchers, J. Madgets: actuating widgets on interactive tabletops. in *Proceedings of the 23rd annual ACM symposium on User interface software and technology* 293–302 (2010).
65. Patten, J. & Ishii, H. Mechanical constraints as computational constraints in tabletop tangible interfaces. in *Proceedings of the SIGCHI conference on Human factors in computing systems* 809–818 (2007).
66. Tibbits, S. 4D Printing: Multi-Material Shape Change. *Architectural Design* 84, 116–121 (2014).
67. Wang, G. *et al.* 4DMesh: 4D printing morphing non-developable mesh surfaces. in *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* 623–635 (2018).
68. Wang, G., Tao, Y., Capunaman, O. B., Yang, H. & Yao, L. A-line: 4D printing morphing linear composite structures. in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* 1–12 (2019).
69. Luo, D., Gu, J., Qin, F., Wang, G. & Yao, L. E-seed: Shape-Changing Interfaces that Self Drill. in *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* 45–57 (ACM, Virtual Event USA, 2020). doi:10.1145/3379337.3415855.
70. Qiu, Y., Lu, Z. & Pei, Q. Refreshable tactile display based on a bistable electroactive polymer and a stretchable serpentine Joule heating electrode. *ACS applied materials & interfaces* 10, 24807–24815 (2018).
71. Besse, N., Rosset, S., Zarate, J. J. & Shea, H. Flexible active skin: large reconfigurable arrays of individually addressed shape memory polymer actuators. *Advanced Materials Technologies* 2, 1700102 (2017).
72. Besse, N. *et al.* Understanding graphics on a scalable latching assistive haptic display using a shape memory polymer membrane. *IEEE transactions on haptics* 11, 30–38 (2017).
73. Rauf, A. M., Bernardo, J. S. & Follmer, S. Electroadhesive Auxetics as Programmable Layer Jamming Skins for Formable Crust Shape Displays. in *2023 IEEE International Conference on Robotics and Automation (ICRA)* 2591–2597 (IEEE, London, United Kingdom, 2023). doi:10.1109/ICRA48891.2023.10161500.
74. Stoy, K., Brandt, D. & Christensen, D. J. *Self-Reconfigurable Robots: An Introduction*. (Mit Press, 2010).
75. Hirose, S. & Yamada, H. Snake-like robots [Tutorial]. *IEEE Robotics & Automation Magazine* 16, 88–98 (2009).
76. Wright, C. *et al.* Design and architecture of the unified modular snake robot. in *2012 IEEE international conference on robotics and automation* 4347–4354 (IEEE, 2012).
77. Wright, C. *et al.* Design of a modular snake robot. in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems* 2609–2614 (IEEE, 2007).
78. Vaquero, T. S. *et al.* EELS: Autonomous snake-like robot with task and motion planning capabilities for ice world exploration. *Science Robotics* 9, eadh8332 (2024).
79. Liu, J., Tong, Y. & Liu, J. Review of snake robots in constrained environments. *Robotics and Autonomous Systems* 141, 103785 (2021).
80. Kelasidi, E., Pettersen, K. Y. & Gravidahl, J. T. Energy efficiency of underwater snake robot locomotion. in *2015 23rd Mediterranean Conference on Control and Automation (MED)* 1124–1131 (2015). doi:10.1109/MED.2015.7158907.
81. Nakagaki, K., Follmer, S. & Ishii, H. Lineform: Actuated curve interfaces for display, interaction, and constraint. in *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* 333–339 (2015).

82. Nakagaki, K., Dementyev, A., Follmer, S., Paradiso, J. A. & Ishii, H. ChainFORM: a linear integrated modular hardware system for shape changing interfaces. in *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* 87–96 (2016).
83. Russo, M. *et al.* Continuum Robots: An Overview. *Advanced Intelligent Systems* 5, 2200367 (2023).
84. Robertson, M. A. & Paik, J. New soft robots really suck: Vacuum-powered systems empower diverse capabilities. *Science Robotics* 2, ean6357 (2017).
85. Walker, I. D. *et al.* Continuum robot arms inspired by cephalopods. 5804, 303–314 (2005).
86. Kaufmann, J., Bhovad, P. & Li, S. Harnessing the multistability of kresling origami for reconfigurable articulation in soft robotic arms. *Soft Robotics* 9, 212–223 (2022).
87. Yang, Y., Chen, Y., Li, Y. & Chen, M. Z. 3D printing of variable stiffness hyper-redundant robotic arm. in *2016 IEEE International Conference on Robotics and Automation (ICRA)* 3871–3877 (IEEE, 2016).
88. Robertson, M. A. & Paik, J. New soft robots really suck: Vacuum-powered systems empower diverse capabilities. *Science Robotics* 2, (2017).
89. Firouzeh, A., Salerno, M. & Paik, J. Stiffness control with shape memory polymer in underactuated robotic origamis. *IEEE Transactions on Robotics* 33, 765–777 (2017).
90. Yim, M. *et al.* Modular Self-Reconfigurable Robot Systems [Grand Challenges of Robotics]. *IEEE Robotics & Automation Magazine* 14, 43–52 (2007).
91. Davey, J., Kwok, N. & Yim, M. Emulating self-reconfigurable robots – design of the SMORES system. in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems* 4464–4469 (2012). doi:10.1109/IROS.2012.6385845.
92. Zykov, V., Chan, A. & Lipson, H. Molecubes: An open-source modular robotics kit. in *IROS-2007 Self-Reconfigurable Robotics Workshop* 3–6 (Citeseer, 2007).
93. Spröwitz, A., Moeckel, R., Vespignani, M., Bonardi, S. & Ijspeert, A. J. Roombots: A hardware perspective on 3D self-reconfiguration and locomotion with a homogeneous modular robot. *Robotics and Autonomous Systems* 62, 1016–1033 (2014).
94. Hauser, S. *et al.* Roombots extended: Challenges in the next generation of self-reconfigurable modular robots and their application in adaptive and assistive furniture. *Robotics and Autonomous Systems* 127, 103467 (2020).
95. Sproewitz, A., Billard, A., Dillenbourg, P. & Ijspeert, A. J. Roombots-mechanical design of self-reconfiguring modular robots for adaptive furniture. in *2009 IEEE international conference on robotics and automation* 4259–4264 (IEEE, 2009).
96. Romanishin, J. W., Gilpin, K. & Rus, D. M-blocks: Momentum-driven, magnetic modular robots. in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems* 4288–4295 (IEEE, 2013).
97. Romanishin, J. W., Gilpin, K., Claici, S. & Rus, D. 3D M-Blocks: Self-reconfiguring robots capable of locomotion via pivoting in three dimensions. in *2015 IEEE International Conference on Robotics and Automation (ICRA)* 1925–1932 (IEEE, 2015).
98. Jorgensen, M. W., Ostergaard, E. H. & Lund, H. H. Modular ATRON: Modules for a self-reconfigurable robot. in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*(IEEE Cat. No. 04CH37566) vol. 2 2068–2073 (Ieee, 2004).
99. Murata, S. *et al.* M-TRAN: Self-reconfigurable modular robotic system. *IEEE/ASME transactions on mechatronics* 7, 431–441 (2002).

100. Yi, S., Sycara, K. & Temel, Z. Reconfigurable Robot Control Using Flexible Coupling Mechanisms. in *Robotics: Science and Systems XIX* (Robotics: Science and Systems Foundation, 2023). doi:10.15607/RSS.2023.XIX.095.
101. Nisser, M., Cheng, L., Makaram, Y., Suzuki, R. & Mueller, S. ElectroVoxel: Electromagnetically Actuated Pivoting for Scalable Modular Self-Reconfigurable Robots. in *2022 International Conference on Robotics and Automation (ICRA)* 4254–4260 (2022). doi:10.1109/ICRA46639.2022.9811746.
102. Abdel-Rahman, A., Cameron, C., Jenett, B., Smith, M. & Gershenfeld, N. Self-replicating hierarchical modular robotic swarms. *Commun Eng* 1, 1–10 (2022).
103. Le Goc, M. *et al.* Zooids: Building Blocks for Swarm User Interfaces. in *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* 97–109 (ACM, Tokyo Japan, 2016). doi:10.1145/2984511.2984547.
104. Nakagaki, K., Leong, J., Tappa, J. L., Wilbert, J. & Ishii, H. HERMITS: Dynamically Reconfiguring the Interactivity of Self-propelled TUIs with Mechanical Shell Add-ons. in *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* 882–896 (ACM, Virtual Event USA, 2020). doi:10.1145/3379337.3415831.
105. Suzuki, R. *et al.* ShapeBots: Shape-changing Swarm Robots. in *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* 493–505 (ACM, New Orleans LA USA, 2019). doi:10.1145/3332165.3347911.
106. Sturdee, M. & Alexander, J. Analysis and classification of shape-changing interfaces for design and application-based research. *ACM Computing Surveys (CSUR)* 51, 1–32 (2018).
107. Callens, S. J. & Zadpoor, A. A. From flat sheets to curved geometries: Origami and kirigami approaches. *Materials Today* 21, 241–264 (2018).
108. Belke, C. H. & Paik, J. Mori: a modular origami robot. *IEEE/ASME Transactions on Mechatronics* 22, 2153–2164 (2017).
109. Yan, Z. *et al.* Controlled mechanical buckling for origami-inspired construction of 3D microstructures in advanced materials. *Advanced functional materials* 26, 2629–2639 (2016).
110. Chen, S., Chen, J., Zhang, X., Li, Z.-Y. & Li, J. Kirigami/origami: unfolding the new regime of advanced 3D microfabrication/nanofabrication with “folding”. *Light: Science & Applications* 9, 75 (2020).
111. Gu, J. *et al.* PneuMesh: Pneumatic-driven truss-based shape changing system. in *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* 1–12 (2022).
112. Kovacs, R. *et al.* TrussFormer: 3D printing large kinetic structures. in *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* 113–125 (2018).
113. Usevitch, N. S. *et al.* An untethered isoperimetric soft robot. *Sci. Robot.* 5, eaaz0492 (2020).
114. Stuart, A. D., Hammond, Z. M. & Follmer, S. Balloon Animal Robots: Reconfigurable Isoperimetric Inflated Soft Robots. in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* 6941–6947 (2021). doi:10.1109/IROS51168.2021.9635842.
115. Liu, Y. *et al.* A review on tensegrity structures-based robots. *Mechanism and Machine Theory* 168, 104571 (2022).
116. Agogino, A. K., SunSpiral, V. & Atkinson, D. Super Ball Bot-Structures for Planetary Landing and Exploration. <https://ntrs.nasa.gov/citations/20190001164> (2018).
117. Sabelhaus, A. P. *et al.* System design and locomotion of SUPERball, an untethered tensegrity robot. in *2015 IEEE international conference on robotics and automation (ICRA)* 2867–2873 (IEEE, 2015).

118. Rieffel, J. & Mouret, J.-B. Adaptive and Resilient Soft Tensegrity Robots. Preprint at <http://arxiv.org/abs/1702.03258> (2018).
119. Fenci, G. E. & Currie, N. G. Deployable structures classification: A review. *International Journal of Space Structures* 32, 112–130 (2017).
120. Puig, L., Barton, A. & Rando, N. A review on large deployable structures for astrophysics missions. *Acta astronautica* 67, 12–26 (2010).
121. Multistable inflatable origami structures at the metre scale | Nature. <https://www.nature.com/articles/s41586-021-03407-4>.
122. Mira: Deployable scissor arch for transitional shelters - Google Scholar. https://scholar.google.com/scholar_lookup?&title=Deployable%20scissor%20arch%20for%20transitional%20shelters&journal=Autom.%20Constr.&doi=10.1016%2Fj.autcon.2014.03.014&volume=43&pages=123-131&publication_year=2014&author=Mira%2CLA&author=Thrall%2CAP&author=Temmerman%2CN.
123. Siéfert: Programming stiff inflatable shells from... - Google Scholar. https://scholar.google.com/scholar_lookup?&title=Programming%20stiff%20inflatable%20shells%20from%20planar%20patterned%20fabrics&journal=Soft%20Matter&doi=10.1039%2FD0SM01041C&volume=16&pages=7898-7903&publication_year=2020&author=Sifert%2CE&author=Reyssat%2CE&author=Bico%2CJ&author=Roman%2CB.
124. Kovacs, R. *et al.* TrussFab: Fabricating Sturdy Large-Scale Structures on Desktop 3D Printers. in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* 2606–2616 (ACM, Denver Colorado USA, 2017). doi:10.1145/3025453.3026016.
125. Rambold, L. *et al.* AirTied: Automatic Personal Fabrication of Truss Structures. in *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* 1–10 (ACM, San Francisco CA USA, 2023). doi:10.1145/3586183.3606820.
126. Liu, J., Lee, Y.-C. & Cardoso Llach, D. Computational design and fabrication of highly customizable architectural space frames: Making a flat-cut Weaire-Phelan structure. *International Journal of Architectural Computing* 19, 37–49 (2021).
127. Lyu, Y. *et al.* TensionFab: Fabrication of Room-scale Surface Structures From the Tension-Active Form of Planar Modules. in *Proceedings of the CHI Conference on Human Factors in Computing Systems* 1–16 (ACM, Honolulu HI USA, 2024). doi:10.1145/3613904.3641958.
128. Agrawal, H. *et al.* Protopiper: Physically Sketching Room-Sized Objects at Actual Scale. in *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* 427–436 (ACM, Charlotte NC USA, 2015). doi:10.1145/2807442.2807505.
129. Campos Zamora, D., He, L. & Froehlich, J. E. MobiPrint: A Mobile 3D Printer for Environment-Scale Design and Fabrication. in *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology* 1–10 (ACM, Pittsburgh PA USA, 2024). doi:10.1145/3654777.3676459.
130. Zhang, X. *et al.* Large-scale 3D printing by a team of mobile robots. *Automation in Construction* 95, 98–106 (2018).
131. Sturdee, M. *Sketching as a Support Mechanism for the Design and Development of Shape-Changing Interfaces*. (Lancaster University (United Kingdom), 2018).
132. Hernandez, H., Preza, E. & Velazquez, R. Characterization of a Piezoelectric Ultrasonic Linear Motor for Braille Displays. in *2009 Electronics, Robotics and Automotive Mechanics Conference (CERMA)* 402–407 (2009). doi:10.1109/CERMA.2009.82.
133. Kim, J. *et al.* Braille display for portable device using flip-latch structured electromagnetic actuator. *IEEE transactions on haptics* 13, 59–65 (2020).

134. Polygerinos, P. *et al.* Soft robotics: Review of fluid-driven intrinsically soft devices; manufacturing, sensing, control, and applications in human-robot interaction. *Advanced Engineering Materials* 19, 1700016 (2017).
135. Fujii, J., Matsunobu, T. & Kakehi, Y. Colorise: Shape-and color-changing pixels with inflatable elastomers and interactions. in *Proceedings of the Twelfth International Conference on Tangible, Embedded, and Embodied Interaction* 199–204 (2018).
136. Nojiri, F. & Kakehi, Y. BelliesWave: color and shape changing pixels using bilayer rubber membranes. in *ACM SIGGRAPH 2014 Posters* 1–1 (2014).
137. Amiri Moghadam, A. A. *et al.* Laser cutting as a rapid method for fabricating thin soft pneumatic actuators and robots. *Soft robotics* 5, 443–451 (2018).
138. Ou, J., Heibeck, F. & Ishii, H. TEI 2016 Studio: Inflated Curiosity. in *Proceedings of the TEI'16: Tenth International Conference on Tangible, Embedded, and Embodied Interaction* 766–769 (2016).
139. Kopic, C. & Gohlke, K. InflatiBits: A Modular Soft Robotic Construction Kit for Children. in *Proceedings of the TEI'16: Tenth International Conference on Tangible, Embedded, and Embodied Interaction* 723–728 (2016).
140. Shtarbanov, A. FlowIO Development Platform—the Pneumatic “Raspberry Pi” for Soft Robotics. in *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems* 1–6 (2021).
141. Sreetharan, P. S., Whitney, J. P., Strauss, M. D. & Wood, R. J. Monolithic fabrication of millimeter-scale machines. *Journal of Micromechanics and Microengineering* 22, 055027 (2012).
142. Wu, Y. *et al.* Insect-scale fast moving and ultrarobust soft robot. *Science Robotics* 4, (2019).
143. Balakrishnan, B., Nacev, A. & Smela, E. Design of bending multi-layer electroactive polymer actuators. *Smart Materials and Structures* 24, 045032 (2015).
144. Marchese, A. D., Katzschnmann, R. K. & Rus, D. A recipe for soft fluidic elastomer robots. *Soft robotics* 2, 7–25 (2015).
145. Mintchev, S., Salerno, M., Cherpillod, A., Scaduto, S. & Paik, J. A portable three-degrees-of-freedom force feedback origami robot for human–robot interactions. *Nature Machine Intelligence* 1, 584–593 (2019).
146. McClintock, H., Temel, F. Z., Doshi, N., Koh, J. & Wood, R. J. The milliDelta: A high-bandwidth, high-precision, millimeter-scale Delta robot. *Science Robotics* 3, (2018).
147. Tang, Y., Li, Y., Hong, Y., Yang, S. & Yin, J. Programmable active kirigami metasheets with more freedom of actuation. *Proceedings of the National Academy of Sciences* 116, 26407–26413 (2019).
148. Kellaris, N., Gopaluni Venkata, V., Smith, G. M., Mitchell, S. K. & Keplinger, C. Peano-HASEL actuators: Muscle-mimetic, electrohydraulic transducers that linearly contract on activation. *Science Robotics* 3, eaar3276 (2018).
149. Mitchell, S. K. *et al.* An Easy-to-Implement Toolkit to Create Versatile and High-Performance HASEL Actuators for Untethered Soft Robots. *Advanced Science* 6, 1900178 (2019).
150. Leroy, E., Hinchet, R. & Shea, H. Multimode hydraulically amplified electrostatic actuators for wearable haptics. *Advanced Materials* 32, 2002564 (2020).
151. Zhang, W.-M., Yan, H., Peng, Z.-K. & Meng, G. Electrostatic pull-in instability in MEMS/NEMS: A review. *Sensors and Actuators A: Physical* 214, 187–218 (2014).
152. Diller, S., Majidi, C. & Collins, S. H. A lightweight, low-power electroadhesive clutch and spring for exoskeleton actuation. in *2016 IEEE International Conference on Robotics and Automation (ICRA)* 682–689 (IEEE, 2016).

153. Hinchet, R., Vechev, V., Shea, H. & Hilliges, O. Dextres: Wearable haptic feedback for grasping in vr via a thin form-factor electrostatic brake. in *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* 901–912 (2018).
154. Qian, J.-Y., Hou, C.-W., Li, X.-J. & Jin, Z.-J. Actuation Mechanism of Microvalves: A Review. *Micromachines* 11, 172 (2020).
155. Biegelsen, D. K. *et al.* AirJet paper mover: An example of mesoscale MEMS. in *Micromachined Devices and Components VI* vol. 4176 122–129 (International Society for Optics and Photonics, 2000).
156. Anjewierden, D., Liddiard, G. A. & Gale, B. K. An electrostatic microvalve for pneumatic control of microfluidic systems. *Journal of Micromechanics and Microengineering* 22, 025019 (2012).
157. Tice, J. D., Desai, A. V., Bassett, T. A., Apblett, C. A. & Kenis, P. J. Control of pressure-driven components in integrated microfluidic devices using an on-chip electrostatic microvalve. *RSC Advances* 4, 51593–51602 (2014).
158. Yoshida, K., Tanaka, S., Hagihara, Y., Tomonari, S. & Esashi, M. Normally closed electrostatic microvalve with pressure balance mechanism for portable fuel cell application. *Sensors and Actuators A: Physical* 157, 290–298 (2010).
159. Bae, B., Han, J., Masel, R. I. & Shannon, M. A. A bidirectional electrostatic microvalve with microsecond switching performance. *Journal of Microelectromechanical Systems* 16, 1461–1471 (2007).
160. Tilli, M. & others. *Handbook of Silicon Based MEMS Materials and Technologies*. (William Andrew Publishing, Boston, 2015).
doi:<https://doi.org/10.1016/B978-0-323-29965-7.00046-4>.
161. Jr, C. F. C. & Holden, H. T. *Printed Circuits Handbook, Seventh Edition*. (McGraw-Hill Professional, 2016).
162. Solberg, V. Embedding passive and active components: PCB design and fabrication process variations. *IPC APEX EXPO* (2015).
163. Karagozler, M. E. *et al.* Electrostatic latching for inter-module adhesion, power transfer, and communication in modular robots. in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems* 2779–2786 (IEEE, 2007).
164. Sato, K. & Shikida, M. An electrostatically actuated gas valve with an S-shaped film element. *Journal of micromechanics and microengineering* 4, 205 (1994).
165. Márquez Segura, E. *et al.* Designing Future Social Wearables with Live Action Role Play (Larp) Designers. in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* 1–14 (2018).
166. Russomanno, A., O’Modhrain, S., Gillespie, R. B. & Rodger, M. W. Refreshing refreshable braille displays. *IEEE transactions on haptics* 8, 287–297 (2015).
167. Rothmund, P. *et al.* A soft, bistable valve for autonomous control of soft actuators. *Science Robotics* 3, (2018).
168. Lederman, S. J. & Klatzky, R. L. Haptic perception: A tutorial. *Attention, Perception, & Psychophysics* 71, 1439–1459 (2009).
169. Drotman, D., Ishida, M., Jadhav, S. & Tolley, M. T. Application-driven design of soft, 3-D printed, pneumatic actuators with bellows. *IEEE/ASME Transactions on Mechatronics* 24, 78–87 (2018).
170. Kalisky, T. *et al.* Differential pressure control of 3D printed soft fluidic actuators. in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* 6207–6213 (2017). doi:10.1109/IROS.2017.8206523.
171. Kawamoto, J. & Asano, F. Active viscoelastic-legged rimless wheel with upper body and its adaptability to irregular terrain. in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems* 157–162 (IEEE, 2012).

172. Sanchez, S. & Bhounsule, P. A. Design, Modeling, and Control of a Differential Drive Rimless Wheel That Can Move Straight and Turn. *Automation* 2, 98–115 (2021).
173. Laney, D. & Hong, D. Kinematic analysis of a novel rimless wheel with independently actuated spokes. in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* vol. 47446 609–614 (2005).
174. Jeans, J. B. & Hong, D. IMPASS: Intelligent mobility platform with active spoke system. in *2009 IEEE International Conference on Robotics and Automation* 1605–1606 (IEEE, 2009).
175. Koutsourelis, M., Michalas, L. & Papaioannou, G. Advanced dielectric charging characterization in capacitive MEMS. in *2012 IEEE International Conference on Industrial Technology* 545–550 (IEEE, 2012).
176. Gulay, E. & Lucero, A. Understanding the Role of Physical and Digital Techniques in the Initial Design Processes of Architecture. in *IFIP Conference on Human-Computer Interaction* 312–329 (Springer, 2021).
177. Singh, K. Industrial motivation for interactive shape modeling: a case study in conceptual automotive design. in *Acm siggraph 2006 courses* 3–9 (2006).
178. Leen, D., Ramakers, R. & Luyten, K. StrutModeling: A low-fidelity construction kit to iteratively model, test, and adapt 3D objects. in *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology* 471–479 (2017).
179. Tahouni, Y., Qamar, I. P. & Mueller, S. NURBSforms: a modular shape-changing interface for prototyping curved surfaces. in *Proceedings of the Fourteenth International Conference on Tangible, Embedded, and Embodied Interaction* 403–409 (2020).
180. Weller, M. P., Do, E. Y.-L. & Gross, M. D. Posey: instrumenting a poseable hub and strut construction toy. in *Proceedings of the 2nd international conference on Tangible and embedded interaction* 39–46 (2008).
181. Sanders, E. B.-N. & Stappers, P. J. Probes, toolkits and prototypes: three approaches to making in codesigning. *CoDesign* 10, 5–14 (2014).
182. Buchenau, M. & Suri, J. F. Experience prototyping. in *Proceedings of the 3rd conference on Designing interactive systems: processes, practices, methods, and techniques* 424–433 (2000).
183. Lim, Y.-K., Stolterman, E. & Tenenberg, J. The anatomy of prototypes: Prototypes as filters, prototypes as manifestations of design ideas. *ACM Transactions on Computer-Human Interaction (TOCHI)* 15, 1–27 (2008).
184. Buxton, B. *Sketching User Experiences: Getting the Design Right and the Right Design*. (Morgan kaufmann, 2010).
185. Pallasmaa, J. *The Eyes of the Skin: Architecture and the Senses*. (Wiley-Academy ; John Wiley & Sons, Chichester : Hoboken, NJ, 2012).
186. Greenberg, E. The value of the full-scale prototype in architectural education. *Charrette* 4, 27–39 (2017).
187. Schon, D. A. Designing as reflective conversation with the materials of a design situation. *Research in engineering design* 3, 131–147 (1992).
188. Binder, T., De Michelis, G., Ehn, P., Jacucci, G. & Linde, P. *Design Things*. (MIT press, 2011).
189. Evans, M. A. Rapid prototyping and industrial design practice: can haptic feedback modelling provide the missing tactile link? *Rapid Prototyping Journal* 11, 153–159 (2005).
190. Jones, D., Snider, C., Nassehi, A., Yon, J. & Hicks, B. Characterising the Digital Twin: A systematic literature review. *CIRP journal of manufacturing science and technology* 29, 36–52 (2020).

191. Khajavi, S. H., Motlagh, N. H., Jaribion, A., Werner, L. C. & Holmström, J. Digital twin: vision, benefits, boundaries, and creation for buildings. *IEEE access* 7, 147406–147419 (2019).
192. Glaessgen, E. & Stargel, D. The digital twin paradigm for future NASA and US Air Force vehicles. in *53rd AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference 20th AIAA/ASME/AHS adaptive structures conference 14th AIAA* 1818 (2012).
193. Wessely, M., Tsandilas, T. & Mackay, W. E. Shape-aware material: Interactive fabrication with shapeme. in *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* 127–139 (2018).
194. Piya, C. & Ramani, K. Proto-tai: Quick design prototyping using tangible assisted interfaces. in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* vol. 46285 V01AT02A097 (American Society of Mechanical Engineers, 2014).
195. Feick, M., Bateman, S., Tang, A., Miede, A. & Marquardt, N. Tangi: Tangible proxies for embodied object exploration and manipulation in virtual reality. in *2020 IEEE international symposium on mixed and augmented reality (ISMAR)* 195–206 (IEEE, 2020).
196. Sheng, J., Balakrishnan, R. & Singh, K. An interface for virtual 3D sculpting via physical proxy. *GRAPHITE* 6, 213–220 (2006).
197. Chien, C., Liang, R.-H., Lin, L.-F., Chan, L. & Chen, B.-Y. Flexibend: Enabling interactivity of multi-part, deformable fabrications using single shape-sensing strip. in *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* 659–663 (2015).
198. Bächer, M. *et al.* Defsense: Computational design of customized deformable input devices. in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* 3806–3816 (2016).
199. Willis, K. D., Xu, C., Wu, K.-J., Levin, G. & Gross, M. D. Interactive fabrication: new interfaces for digital fabrication. in *Proceedings of the fifth international conference on Tangible, embedded, and embodied interaction* 69–72 (2010).
200. Balakrishnan, R., Fitzmaurice, G., Kurtenbach, G. & Singh, K. Exploring interactive curve and surface manipulation using a bend and twist sensitive input strip. in *Proceedings of the 1999 symposium on Interactive 3D graphics* 111–118 (1999).
201. Grossman, T., Balakrishnan, R. & Singh, K. An interface for creating and manipulating curves using a high degree-of-freedom curve input device. in *Proceedings of the SIGCHI conference on Human factors in computing systems* 185–192 (2003).
202. Dementyev, A., Kao, H.-L. & Paradiso, J. A. Sensortape: Modular and programmable 3d-aware dense sensor network on a tape. in *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* 649–658 (2015).
203. Shahmiri, F. & Dietz, P. H. Sharc: A geometric technique for multi-bend/shape sensing. in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* 1–12 (2020).
204. Lee, J., Su, V., Ren, S. & Ishii, H. HandSCAPE: a vectorizing tape measure for on-site measuring applications. in *Proceedings of the SIGCHI conference on Human Factors in Computing Systems* 137–144 (2000).
205. Weichel, C., Alexander, J., Karnik, A. & Gellersen, H. SPATA: Spatio-tangible tools for fabrication-aware design. in *Proceedings of the ninth international conference on tangible, embedded, and embodied interaction* 189–196 (2015).
206. Sanoff, H. *Community Participation Methods in Design and Planning*. (John Wiley & Sons, 1999).

207. Luck, R. Participatory design in architectural practice: Changing practices in future making in uncertain times. *Design Studies* 59, 139–157 (2018).
208. Sanoff, H. Schools Designed with Community Participation. (2002).
209. Abrams, R. Byker revisited. *Built Environment* (1978-) 117–131 (2003).
210. Sanoff, H. Community participation in riverfront development. *CoDesign* 1, 61–78 (2005).
211. Brandt, E. How tangible mock-ups support design collaboration. *Knowledge, Technology & Policy* 20, 179–192 (2007).
212. Stappers, P. J., van Rijn, H., Kistemaker, S., Hennink, A. & Visser, F. S. Designing for other people's strengths and motivations: Three cases using context, visions, and experiential prototypes. *Advanced engineering informatics* 23, 174–183 (2009).
213. Baumgart, B. G. A polyhedron representation for computer vision. in *Proceedings of the May 19-22, 1975, national computer conference and exposition* 589–596 (1975).
214. Piker, D. Kangaroo: form finding with computational physics. *Architectural Design* 83, 136–137 (2013).
215. Adriaenssens, S. & Barnes, M. R. Tensegrity spline beam and grid shell structures. *Engineering structures* 23, 29–36 (2001).
216. Siu, A. F., Kim, S., Miele, J. A. & Follmer, S. shapeCAD: An Accessible 3D Modelling Workflow for the Blind and Visually-Impaired Via 2.5D Shape Displays. in *Proceedings of the 21st International ACM SIGACCESS Conference on Computers and Accessibility* 342–354 (Association for Computing Machinery, New York, NY, USA, 2019). doi:10.1145/3308561.3353782.
217. Qian, W., Gao, C., Sathya, A., Suzuki, R. & Nakagaki, K. SHAPE-IT: Exploring Text-to-Shape-Display for Generative Shape-Changing Behaviors with LLMs. in *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology* 1–29 (Association for Computing Machinery, New York, NY, USA, 2024). doi:10.1145/3654777.3676348.
218. Steed, A., Ofek, E., Sinclair, M. & Gonzalez-Franco, M. A mechatronic shape display based on auxetic materials. *Nat Commun* 12, 4758 (2021).
219. Jansen, Y. et al. Opportunities and Challenges for Data Physicalization. in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* 3227–3236 (Association for Computing Machinery, New York, NY, USA, 2015). doi:10.1145/2702123.2702180.
220. Boem, A. & Troiano, G. M. Non-rigid HCI: A review of deformable interfaces and input. in *Proceedings of the 2019 on Designing Interactive Systems Conference* 885–906 (2019).
221. Karras, J. T. et al. Pop-up mars rover with textile-enhanced rigid-flex PCB body. in *2017 IEEE International Conference on Robotics and Automation (ICRA)* 5459–5466 (IEEE, 2017).
222. Isaac, J. Rigid-flex technology: mainstream use but more complex designs. *CIRCUITREE-CAMPBELL* 20, 30 (2007).
223. Gonzalez, J. T. & Hudson, S. E. Layer by Layer, Patterned Valves Enable Programmable Soft Surfaces. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 1–25 (2022).
224. Sozer, C., Paternò, L., Tortora, G. & Menciassi, A. A novel pressure-controlled revolute joint with variable stiffness. *Soft Robotics* 9, 723–733 (2022).
225. Levine, D. J., Turner, K. T. & Pikul, J. H. Materials with electroprogrammable stiffness. *Advanced Materials* 33, 2007952 (2021).
226. Anderson-Bashan, L. et al. The greeting machine: an abstract robotic object for opening encounters. in *2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)* 595–602 (IEEE, 2018).

227. Whittaker, S., Rogers, Y., Petrovskaya, E. & Zhuang, H. Designing personas for expressive robots: Personality in the new breed of moving, speaking, and colorful social home robots. *ACM Transactions on Human-Robot Interaction (THRI)* 10, 1–25 (2021).
228. Zuckerman, O. *et al.* Companionship is not a function: The effect of a novel robotic object on healthy older adults' feelings of "being-seen". in *Proceedings of the 2020 CHI conference on human factors in computing systems* 1–14 (2020).
229. Erel, H. *et al.* Enhancing emotional support: The effect of a robotic object on human–human support quality. *International Journal of Social Robotics* 14, 257–276 (2022).
230. Shafiullah, N. M. M. *et al.* On bringing robots home. *arXiv preprint arXiv:2311.16098* (2023).
231. Forlizzi, J. How robotic products become social products: an ethnographic study of cleaning in the home. in *Proceedings of the ACM/IEEE international conference on Human-robot interaction* 129–136 (2007).
232. Metz, C. Invasion of the Home Humanoid Robots. *The New York Times* <https://www.nytimes.com/2025/04/04/technology/humanoid-robots-1x.html> (2025).
233. Stock-Homburg, R. & Heitlinger, L. One size does not fit all: Developing robot user types for office robots. in *International Conference on Human-Computer Interaction* 204–214 (Springer, 2023).
234. Mutlu, B. & Forlizzi, J. Robots in organizations: the role of workflow, social, and environmental factors in human-robot interaction. in *Proceedings of the 3rd ACM/IEEE international conference on Human robot interaction* 287–294 (2008).
235. Yoon, J. & Wakabayashi, D. Meet Your New Corporate Office Mate: A 'Brainless' Robot. *The New York Times* <https://www.nytimes.com/2022/11/17/business/south-korea-office-robots-naver.html> (2022).
236. Weinberg, D., Dwyer, H., Fox, S. E. & Martelaro, N. Sharing the sidewalk: Observing delivery robot interactions with pedestrians during a pilot in Pittsburgh, PA. *Multimodal Technologies and Interaction* 7, 53 (2023).
237. Brown, B., Bu, F., Mandel, I. & Ju, W. Trash in motion: Emergent interactions with a robotic trashcan. in *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* 1–17 (2024).
238. Seyitoğlu, F. & Ivanov, S. Understanding the robotic restaurant experience: a multiple case study. *Journal of Tourism Futures* 8, 55–72 (2022).
239. Kamino, W. & Sabanovic, S. Coffee, tea, robots? the performative staging of service robots in 'robot cafes' in japan. in *Proceedings of the 2023 ACM/IEEE international conference on human-robot interaction* 183–191 (2023).
240. Han, V. Y. *et al.* Towards Unobtrusive Physical AI: Augmenting Everyday Objects with Intelligence and Robotic Movement for Proactive Assistance. in *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology (UIST '25)* (Association for Computing Machinery, Busan, Republic of Korea, 2025).
241. Wang, Y. & Green, K. E. "Space Agency": A "Strong Concept" for Designing Socially Interactive, Robotic Environments. in *International Conference on Social Robotics* 295–307 (Springer, 2021).
242. Houayek, H., Green, K. E., Gugerty, L., Walker, I. D. & Witte, J. AWE: an animated work environment for working with physical and digital tools and artifacts. *Personal and Ubiquitous Computing* 18, 1227–1241 (2014).
243. Gonzalez, J. T. *et al.* Constraint-Driven Robotic Surfaces, At Human-Scale. in *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* 1–12 (2023).

244. Vazquez, E., Randall, C. & Duarte, J. P. Shape-changing architectural skins: A review on materials, design and fabrication strategies and performance analysis. *Journal of Facade Design and Engineering* 7, 93–114 (2019).
245. Sharaidin, K. Kinetic facades: towards design for environmental performance. (RMIT University, 2024).
246. Kumar, N. *et al.* Design of Two Morphing Robot Surfaces and Results from a User Study On What People Want and Expect of Them, Towards a " Robot-Room". in *2024 IEEE International Conference on Robotics and Automation (ICRA)* 11239–11244 (IEEE, 2024).
247. Hoffman, G. Evaluating fluency in human–robot collaboration. *IEEE Transactions on Human-Machine Systems* 49, 209–218 (2019).
248. Wobbrock, J. O., Aung, H. H., Rothrock, B. & Myers, B. A. Maximizing the guessability of symbolic input. in *CHI'05 extended abstracts on Human Factors in Computing Systems* 1869–1872 (2005).
249. Wobbrock, J. O., Morris, M. R. & Wilson, A. D. User-defined gestures for surface computing. in *Proceedings of the SIGCHI conference on human factors in computing systems* 1083–1092 (2009).
250. Cook, D. J., Augusto, J. C. & Jakkula, V. R. Ambient intelligence: Technologies, applications, and opportunities. *Pervasive and mobile computing* 5, 277–298 (2009).
251. Dunne, R., Morris, T. & Harper, S. A survey of ambient intelligence. *ACM Computing Surveys (CSUR)* 54, 1–27 (2021).
252. Weiser, M. The Computer for the 21 st Century. *Scientific american* 265, 94–105 (1991).
253. Wellner, P. The DigitalDesk calculator: tangible manipulation on a desk top display. in *Proceedings of the 4th annual ACM symposium on User interface software and technology* 27–33 (1991).
254. Wellner, P. Interacting with paper on the DigitalDesk. *Communications of the ACM* 36, 87–96 (1993).
255. Stafford-Fraser, Q. & Robinson, P. BrightBoard: A video-augmented environment. in *Proceedings of the SIGCHI conference on Human factors in computing systems* 134–141 (1996).
256. Jones, B. R., Benko, H., Ofek, E. & Wilson, A. D. IllumiRoom: peripheral projected illusions for interactive experiences. in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* 869–878 (2013).
257. Victor, B. Dynamicland intro. (2024).
258. Bachmann, G. The Promise of the Promise—The Dynamic Medium Group in Oakland, California. in *(Re-) Imagining New Media* 15–30 (Springer, 2021).
259. Wessely, M. *et al.* Sprayable user interfaces: Prototyping large-scale interactive surfaces with sensors and displays. in *Proceedings of the 2020 CHI conference on human factors in computing systems* 1–12 (2020).
260. Das, D. *et al.* Explainable activity recognition for smart home systems. *ACM Transactions on Interactive Intelligent Systems* 13, 1–39 (2023).
261. Green, K. E. *Architectural Robotics: Ecosystems of Bits, Bytes, and Biology*. (MIT Press, 2016).
262. Balci, O., Poncelet, S., Nguyen, A. B. V. D. & Vande Moere, A. Manifesting Architectural Subspaces with Two Mobile Robotic Partitions to Facilitate Spontaneous Office Meetings. in *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems* 1–16 (2025).
263. Larrea-Tamayo, H. ARkits: architectural robotics kits. (Massachusetts Institute of Technology, 2015).
264. Washabaugh, B., Johnstone, G., Gonzalez, J., Altman, A. & Team, the H. Kinetic Brick Wall. (2019).
265. Mok, J., Park, E., Yeo, J., Yoo, S. & Yang, S. Hyper-Matrix (2012). (2012).

266. Rozin, D. Wooden mirror. *IEEE Spectrum* 38, 69 (2001).
267. Attia, S. Evaluation of adaptive facades: The case study of Al Bahr Towers in the UAE. *QScience Connect* 2017, 6 (2017).
268. Strobel, K. M., Binkley, C. B. & Sabin, J. E. Lumen tensegrity towers. in *Proceedings of IASS Annual Symposia* vol. 2018 1–8 (International Association for Shell and Spatial Structures (IASS), 2018).
269. Sung, D. *et al.* Bloom. (2011).
270. Wicaksono, I. Living Knitwork Pavilion. (2023).
271. Bolt, R. A. “Put-that-there” Voice and gesture at the graphics interface. in *Proceedings of the 7th annual conference on Computer graphics and interactive techniques* 262–270 (1980).
272. Schmandt, C. & Hulteen, E. A. The intelligent voice-interactive interface. in *Proceedings of the 1982 conference on Human factors in computing systems* 363–366 (1982).
273. Chai, J. Y. *et al.* Collaborative effort towards common ground in situated human-robot dialogue. in *Proceedings of the 2014 ACM/IEEE International Conference on Human-Robot Interaction* 33–40 (Association for Computing Machinery, New York, NY, USA, 2014). doi:10.1145/2559636.2559677.
274. Huang, C.-M. & Thomaz, A. L. Joint Attention in Human-Robot Interaction. in *AAAI fall symposium: dialog with robots* (2010).
275. Pathak, A., Venkatesan, K., Taha, T. & Muthusamy, R. A Multi-Modal Interaction Framework for Efficient Human-Robot Collaborative Shelf Picking. *arXiv preprint arXiv:2504.06593* (2025).
276. Team, G. R. *et al.* Gemini robotics: Bringing ai into the physical world. *arXiv preprint arXiv:2503.20020* (2025).
277. Ayub, A., De Francesco, Z., Holthaus, P., Nehaniv, C. L. & Dautenhahn, K. Continual learning through human-robot interaction: Human perceptions of a continual learning robot in repeated interactions. *International Journal of Social Robotics* 17, 277–296 (2025).
278. Saupé, A. & Mutlu, B. Robot deictics: how gesture and context shape referential communication. in *Proceedings of the 2014 ACM/IEEE International Conference on Human-Robot Interaction* 342–349 (Association for Computing Machinery, New York, NY, USA, 2014). doi:10.1145/2559636.2559657.
279. Luria, M., Hoffman, G. & Zuckerman, O. Comparing social robot, screen and voice interfaces for smart-home control. in *Proceedings of the 2017 CHI conference on human factors in computing systems* 580–628 (2017).
280. Malpass, M. Between wit and reason: Defining associative, speculative, and critical design in practice. *Design and Culture* 5, 333–356 (2013).
281. Candy, S. The futures of everyday life: Politics and the design of experiential scenarios. *University of* (2010).
282. Dunne, A. & Raby, F. *Speculative Everything, With a New Preface by the Authors: Design, Fiction, and Social Dreaming*. (MIT press, 2024).
283. Sterling, B. Shaping things. (*No Title*) (2005).
284. Zimmerman, J. & Forlizzi, J. Speed dating: Providing a menu of possible futures. *She Ji: The Journal of Design, Economics, and Innovation* 3, 30–50 (2017).
285. Lindley, J. & Coulton, P. Back to the future: 10 years of design fiction. in *Proceedings of the 2015 British HCI conference* 210–211 (2015).
286. Winkle, K. Robots from Nowhere: A Case Study in Speculative Sociotechnical Design and Design Fiction for Human-Robot Interaction. in *2025 20th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* 1152–1165 (IEEE, 2025).
287. Auger, J. Living with robots: a speculative design approach. *J. Hum.-Robot Interact.* 3, 20–42 (2014).

288. Brunnmayr, K. & Weiss, A. Approaching Future Robot Technologies via Speculative Role-Playing. in *Companion of the 2024 ACM/IEEE International Conference on Human-Robot Interaction* 278–282 (Association for Computing Machinery, New York, NY, USA, 2024). doi:10.1145/3610978.3640570.
289. Sturdee, M., Hardy, J., Dunn, N. & Alexander, J. A Public Ideation of Shape-Changing Applications. in *Proceedings of the 2015 International Conference on Interactive Tabletops & Surfaces* 219–228 (Association for Computing Machinery, New York, NY, USA, 2015). doi:10.1145/2817721.2817734.
290. Grafström, A., Holmgren, M., Linge, S., Lagerberg, T. & Obaid, M. A Speculative Design Approach to Investigate Interactions for an Assistant Robot Cleaner in Food Plants. in *Adjunct Proceedings of the 2022 Nordic Human-Computer Interaction Conference* (Association for Computing Machinery, New York, NY, USA, 2022). doi:10.1145/3547522.3547682.
291. Rasmussen, M. K., Pedersen, E. W., Petersen, M. G. & Hornbæk, K. Shape-changing interfaces: a review of the design space and open research questions. in *Proceedings of the SIGCHI conference on human factors in computing systems* 735–744 (2012).
292. Sturdee, M., Coulton, P. & Alexander, J. Using design fiction to inform shape-changing interface design and use. *The Design Journal* 20, S4146–S4157 (2017).
293. Sturdee, M. & Lindley, J. Sketching & drawing as future inquiry in HCI. in *Proceedings of the Halfway to the Future Symposium 2019* 1–10 (2019).
294. Krippendorff, K. *Content Analysis: An Introduction to Its Methodology*. (Sage publications, 2018).
295. Jung, J.-Y. et al. Making the Right Thing: Bridging HCI and Responsible AI in Early-Stage AI Concept Selection. in *Proceedings of the 2025 ACM Designing Interactive Systems Conference* 2992–3012 (2025).
296. Chi, V. B. & Malle, B. F. People dynamically update trust when interactively teaching robots. in *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction* 554–564 (2023).
297. Riek, L. D. Wizard of oz studies in hri: a systematic review and new reporting guidelines. *Journal of Human-Robot Interaction* 1, 119–136 (2012).
298. Song, K., Tan, X., Qin, T., Lu, J. & Liu, T.-Y. Mpnet: Masked and permuted pre-training for language understanding. *Advances in neural information processing systems* 33, 16857–16867 (2020).
299. Maaten, L. van der & Hinton, G. Visualizing data using t-SNE. *Journal of machine learning research* 9, 2579–2605 (2008).
300. Kim, J. & Im, I. Anthropomorphic response: Understanding interactions between humans and artificial intelligence agents. *Computers in Human Behavior* 139, 107512 (2023).
301. Studdert-Kennedy, M. Hand and Mind: What Gestures Reveal About Thought. *Language and Speech* 37, 203–209 (1994).
302. Scott, D. W. *Multivariate Density Estimation: Theory, Practice, and Visualization*. (John Wiley & Sons, 2015).
303. Amid, E. & Warmuth, M. K. TriMap: Large-scale dimensionality reduction using triplets. *arXiv preprint arXiv:1910.00204* (2019).
304. Wang, Y., Huang, H., Rudin, C. & Shaposhnik, Y. Understanding how dimension reduction tools work: an empirical approach to deciphering t-SNE, UMAP, TriMAP, and PaCMAP for data visualization. *Journal of Machine Learning Research* 22, 1–73 (2021).
305. Huang, H., Wang, Y., Rudin, C. & Browne, E. P. Towards a comprehensive evaluation of dimension reduction methods for transcriptomic data visualization. *Communications biology* 5, 719 (2022).

306. Wang, L., Zhang, X., Su, H. & Zhu, J. A comprehensive survey of continual learning: Theory, method and application. *IEEE transactions on pattern analysis and machine intelligence* 46, 5362–5383 (2024).
307. Dragan, A. D., Lee, K. C. & Srinivasa, S. S. Legibility and predictability of robot motion. in *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* 301–308 (IEEE, 2013).
308. Lichtenthäler, C., Lorenzy, T. & Kirsch, A. Influence of legibility on perceived safety in a virtual human-robot path crossing task. in *2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication* 676–681 (IEEE, 2012).
309. Bhaskara, A., Skinner, M. & Loft, S. Agent transparency: A review of current theory and evidence. *IEEE Transactions on Human-Machine Systems* 50, 215–224 (2020).
310. Nanavati, A., Ranganeni, V. & Cakmak, M. Physically assistive robots: A systematic review of mobile and manipulator robots that physically assist people with disabilities. *Annual Review of Control, Robotics, and Autonomous Systems* 7, (2023).
311. Threatt, A. L. *et al.* An assistive robotic table for older and post-stroke adults: results from participatory design and evaluation activities with clinical staff. in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* 673–682 (2014).
312. Hamraie, A. *Building Access: Universal Design and the Politics of Disability*. (U of Minnesota Press, 2017).
313. Kinch, S., Grönvall, E., Petersen, M. G. & Rasmussen, M. K. Encounters on a shape-changing bench: exploring atmospheres and social behaviour in situ. in *Proceedings of the 8th International Conference on Tangible, Embedded and Embodied Interaction* 233–240 (2014).
314. Horvitz, E. J., Breese, J. S. & Henrion, M. Decision theory in expert systems and artificial intelligence. *International journal of approximate reasoning* 2, 247–302 (1988).
315. Horvitz, E. Lumiere project: Bayesian reasoning for automated assistance. *Decision Theory & Adaptive Systems Group, Microsoft Research. Microsoft Corp. Redmond, WA* (1998).
316. Wilson, A. D. & Bobick, A. F. Parametric hidden markov models for gesture recognition. *IEEE transactions on pattern analysis and machine intelligence* 21, 884–900 (2002).
317. Schwarz, J., Hudson, S., Mankoff, J. & Wilson, A. D. A framework for robust and flexible handling of inputs with uncertainty. in *Proceedings of the 23rd annual ACM symposium on User interface software and technology* 47–56 (2010).