(Dis)Appearables: A Concept and Method for Actuated Tangible UIs to Appear and Disappear based on Stages

Ken Nakagaki
MIT Media Lab
Cambridge, USA
ken_n@media.mit.edu

Jordan L Tappa
Massachusetts Institute of Technology
Cambridge, USA
jtapp@mit.edu

Yi Zheng
University of Southern California
Los Angeles, USA
yzheng63@usc.edu

Jack Forman
MIT Media Lab
Cambridge, USA
jackform@media.mit.edu

Joanne Leong
MIT Media Lab
Cambridge, USA
joaleong@media.mit.edu

Sven Koenig
University of Southern California
Los Angeles, USA
skoenig@usc.edu

Hiroshi Ishii
MIT Media Lab
Cambridge, USA
ishii@media.mit.edu

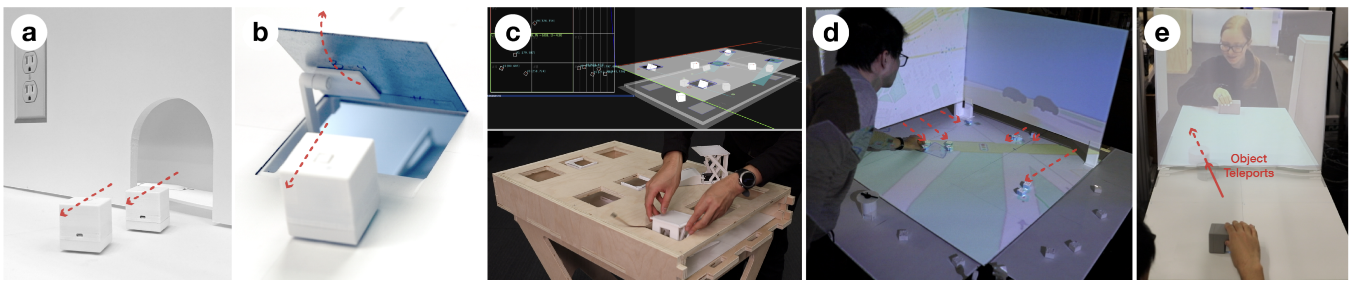

Figure 1: *(Dis)Appearables*: a. wall portal and b. floor portal that facilitate the actuated TUIs to appear and disappear for user experience, c. Stage Design + Control UIs [top] and Fabrication of *Stage* [bottom], d. Application for interactive mobility simulation, e. application for remote table hockey.)

ABSTRACT

(Dis)Appearables is an approach for actuated Tangible User Interfaces (TUIs) to appear and disappear. This technique is supported by *Stages*: physical platforms inspired by theatrical stages. Self-propelled TUI's autonomously move between front and back stage allowing them to dynamically appear and disappear from users' attention. This platform opens up a novel interaction design space for expressive displays with dynamic physical affordances.

We demonstrate and explore this approach based on a proof-of-concept implementation using two-wheeled robots, and multiple stage design examples. We have implemented a stage design pipeline which allows users to plan and design stages that are composed with front and back stages, and transition portals such as trap doors or lifts. The pipeline includes control of the robots, which guides them on and off stage. With this proof-of-concept prototype, we demonstrated a range of applications including interactive mobility simulation, self re-configuring desktops, remote hockey, and storytelling/gaming. Inspired by theatrical stage designs, this is a new take on 'controlling the existence of matter' for user experience design.

CCS CONCEPTS

• **Human-centered computing** → **Haptic devices**; *User interface design*; • **Hardware** → *Emerging interfaces.*

KEYWORDS

Actuated Tangible User Interfaces, Swarm User Interface, Dynamic Physical Affordance, Stage

ACM Reference Format:

Ken Nakagaki, Jordan L Tappa, Yi Zheng, Jack Forman, Joanne Leong, Sven Koenig, and Hiroshi Ishii. 2022. (Dis)Appearables: A Concept and Method for Actuated Tangible UIs to Appear and Disappear based on Stages. In *CHI Conference on Human Factors in Computing Systems (CHI '22), April 29-May 5, 2022, New Orleans, LA, USA.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3491102.3501906

CHI '22, April 29-May 5, 2022, New Orleans, LA, USA

ACM ISBN 978-1-4503-9157-3/22/04.
https://doi.org/10.1145/3491102.3501906

1 INTRODUCTION

"The ultimate display would, of course, be a room within which the computer can control the existence of matter."
(Sutherland, 1965) [48]

For years, researchers have been motivated by the malleability of Graphical User Interfaces (GUIs), where the existence of visual and graphical information are instantly and dynamically controlled within 2D flat displays or, more recently, in 3D Virtual Reality environments. As the quote from Sutherland in 'Ultimate Display' indicates, this power of graphical interfaces inspired researchers to envision ultimate technologies that can control the existence of any 'objects' in space [48]. Similarly, ultimate physical matter that is computationally controlled to conform and transform according to users' input and intention are envisioned with a focus on tangibility and physicality [13, 57].

Towards such visions, HCI researchers have explored ways to dynamically reconfigure physical interfaces through actuated and shape-changing Tangible User Interfaces (TUIs) [4, 35, 38]. Through such research streams, researchers have demonstrated the power of such approach for interaction design, including dynamic physical affordances [8] and data physicalization [15]. While GUIs have long relied on the spontaneous emergence of digital objects, how physical objects can appear and disappear (henceforth referred to as '(dis)appear' for brevity) has not been addressed in-depth. This is likely because the spontaneous (dis)appearance of physical matter violates the laws of physics. However, magicians and performers show have surprised and confused audiences by seemingly doing just that. They demonstrate that actual transportation is not needed to convince the audience that they just *perceived* transportation. Such illusions can play vital roles in a show's plot - such as the dramatic melting of the Wicked Witch in the musical Wicked. Even simple exits or entrances, such as walking on stage, are carefully choreographed by the director for continuity and dramatic impact. In HCI field, some research has explored ways for physical user interface elements to emerge and be rendered from interaction surfaces [8, 40, 52], but they reveal their underlying mechanisms and the back-stage rigging giving immediate explanation to such transitions. In this work, we investigate the design of tangible interfaces with (dis)appearances to/from a hidden back-stage, and how this affordance can engender new modes of human-computer interaction.

(Dis)Appearables is a general approach for actuated TUIs to (dis)appear from the user for physical interaction design. This approach is supported by self-propelled User Interfaces together with *Stage*, a physical platform consisting of front- and back-stages connected through transition portals.

In this paper, we define the general approach, design space, and interaction design benefits of *(Dis)Appearables* based on *Stages*. *Stages* enable actuated TUIs to (dis)appear, and facilitate the illusion of controlling the existence of matter. We demonstrate a *(Dis)Appearables* proof-of-concept with an existing actuated TUI system that uses two-wheeled self-propelled robotic devices [6, 34]; however, the approach is generalizable for other self-propelled TUIs. Our implementation introduces the design and fabrication pipeline of the *Stage* for the robotic devices to navigate on. Using the pipeline, we demonstrate applications that take advantage of the '(Dis)Appearing' effect, ranging from reconfigurable desktops to remote physical gaming. Inspired by theatrical stage designs, this is a new take on 'controlling the existence of matter' for user experience design of physical and tangible UIs.

The list of contributions includes:

- An introduction of a general interaction design approach, *(Dis)Appearables*, for actuated TUIs to (dis)appear from the user through *Stages* consisting of an interactive front-stage and hidden back-stage.
- The concept and Design Space of *(Dis)Appearables* including a range of physical effects, transition portal primitives, and a discussion on the overarching design implications.
- A proof-of-concept prototype based on off-the-shelf two-wheeled robotic hardware, and a stage design pipeline for designing stages and controlling the robots on-top of them.
- Applications to demonstrate the interaction design opportunity of *(Dis)Appearables* in multiple use cases.

2 RELATED WORK

2.1 Actuated TUIs and (Dis)appearing UIs

Often inspired by the flexibility and dynamism of GUI, researchers have been investigating physical interfaces that can actuate and shape change through novel mechanical hardware designs [4, 17, 35]. Various prior works have started to explore the affordance of a physical interface (dis)appearing. For example, with vertically actuated pin arrays, 2.5D physical UIs can appear by rising from the surface [8, 25]. Pneumatic technology has been employed to render volumetric shapes emerging from flat compressed shapes [54, 55, 63]. With *Dynablock*, Suzuki et al. extended the pin-display architecture to assemble blocks to compose 3D shapes rising from a surface [52].

Emergeables [40], on the other hand, proposed a top-down concept of physical interfaces that can emerge from a surface, for eyes-free interaction on a mobile device with 'emergeable' physical controllers with continuous inputs such as knobs and sliders emerging from smartphone screens. While we were inspired by the concept in this paper of emerging physical interfaces, their general implementation approaches were limited in the types of interfaces and shapes to emerge / render as well as tangible input flexibility.

In our paper, we introduce a novel approach for physical and tangible interfaces to dynamically (dis)appear from interaction surfaces through the combination of self-propelled robotic TUIs with physical platforms featuring front- and back-stage architectures, allowing the hardware to hide from the user's view and reach. Unlike the previous methods listed above, this approach leverages and focuses on the expression of (dis)appearing to design new user experiences.

Moreover, in our approach, by hiding tangible UIs in the back-stage from users, the self-propelled TUIs can physically reconfigure with additional modules (e.g. constructive assembly [65], or mechanical add-ons [34]) behind the walls and appear to users with the reconfigured shapes and functionality. Similar to how seeing an actor change costume on stage could disrupt storytelling and confuse the audience, such mechanical assembly and reconfiguration processes may disrupt the user experience if visible. For example, it can attract unnecessary attention from users or indicate unintended affordances [37]. The mechanical shells and unused self-propelled TUI's could clutter and crowd the interactive area. Our approach gives new opportunities for designers and researchers to selectively hide such transitions and modules that is not preferable to be seen by users. This helps to address one of the fundamental classic challenges in TUIs about 'physical clutter' [7, 44, 60]. In our applications, we demonstrate how leveraging this technique improves the quality and extends the use cases of self-propelled TUIs beyond previous works.

Fundamentally, *(Dis)Appearables* encourages and explores the holistic design of self-propelled user interfaces. That is, considering not just the robotic modules and their movements, but the larger infrastructure and environment in which they move. How that environment can be designed to support more seamless and novel modalities of interaction are questions this work probes.

2.2 Tabletop Actuated TUIs and Swarm UIs

Tabletop TUIs have explored the idea of giving actuation capabilities to passive tangible tokens on horizontal tabletop surfaces - one of the most classic style of TUIs [14, 59]. An early approach to enable tabletop actuated TUIs was implemented using electro-magnetic arrays that control the position of tangible pucks [35, 50, 51].

Instead of embedding the actuation capability directly into the tabletop surfaces for controlling passive tokens, another approach proposed for tabletop actuated TUIs is to employ self-propelled robotic devices. *Curlybot* was one of the first instances of such an interface that has embedded wheels to locomote across a flat plane while being able to detect users' movements [9]. While swarm robots have been technically demonstrated in the robotics field for collectively composing shapes and fulfilling tasks [41–43, 64], recently in HCI, Le et al. proposed the concept of *Swarm User Interface* (SUI). SUI is a new category of actuated TUIs [21], which is a type of interface using self-propelled elements that can collectively interact with users. This approach has displayed broad interaction opportunities within haptics [20], constructive assemblies [34, 65], gestural interaction [18], additional shape-changing capabilities [53], and furniture actuation [49]. As for extended locomotion capabilities, even levitating self-propelled TUIs have been proposed [3, 10].

In such a realm, there is a gap of in-depth discussion and exploration in designing the surrounding platform and space for which the interaction takes place. Additionally, the design space for robotic user interfaces that appear and disappear from users' attention has yet to be mapped out. We believe our generalizable technique can greatly contribute to designing the affordances and physical expressions of actuated TUIs by the introduction of *Stages*.

In Human Robot Interaction for education, Robert et al. proposed a system with a meter-sized wheeled robot that travels through a gate in a Mixed Reality environment. In this work, the robot appears from and disappears to the digital space through the gate [39]. This preliminary demonstration inspired us to expand this notion of (dis)appearing so as to be a platform approach for self-propelled TUIs. In this work, we explore the broader design implications and opportunities of such a platform interaction technique with a range of applications. Additionally, our approach takes into account the operation of one or more self-propelled robots, which can work collectively on functional stage designs for novel interactions, such as 'teleportation' or 'transfiguration'.

3 (DIS)APPEARABLES - CONCEPT AND DESIGN SPACE

In this section, we provide an overview of the general approach and design of *(Dis)Appearables*. While the prototypes and diagrams shown in this paper are developed based on tabletop two-wheeled robotic devices, the general concept should be applicable for broader self-propelled user interface hardware such as drones [10] and actuated curve interfaces [33].

3.1 Overall Design and Stage Composition

The overall design of *(Dis)Appearables* includes the self-propelled TUIs, *Stage*, and the computer (Figure 2a). *Stage* is the supporting physical platform, which allows self-propelled TUIs to dynamically and adaptively appear and disappear from the user's attention.

From the classic research in Tabletop TUIs, tangible pucks have been placed on top of a horizontal tabletop surface for graphical information associated with the pucks [36, 59]. *Stage* expands on the role of such tabletop surfaces as a physical platform for actuated TUIs. In our exploration, we define the composition of *Stage* with four basic primitive components: front-stage, back-stage, boundaries, and transition portals. The roles of each component for interaction design, as shown in Figure 2b, is described below.

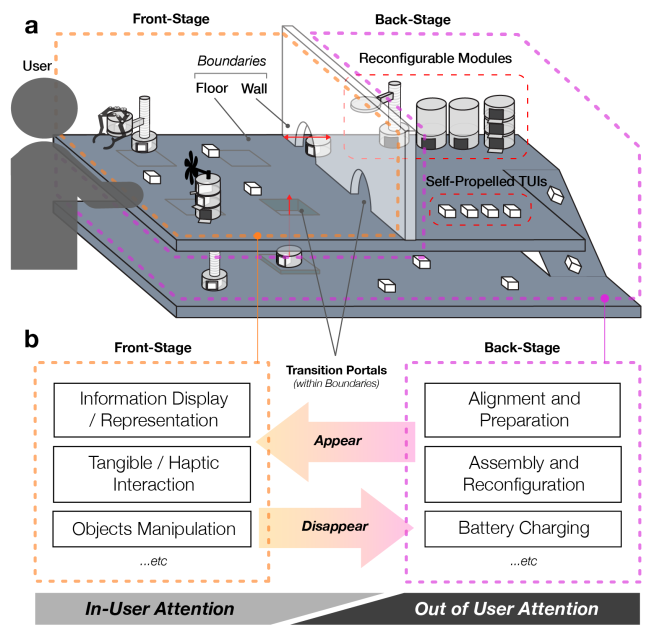

Figure 2: Overall Design of (Dis)Appeaerables (a. Physical Construction of the System, b. Interaction Modality that utilizes front-stage and back-stage to facilitate the appearing and disappearing effects.)

3.1.1 Front-Stage. Front-stage is where the self-propelled TUIs afford and signify interaction for users. The robotic devices can perform explicit interactions with users in the front stage as with classic Tabletop TUI systems. For example, these use cases include

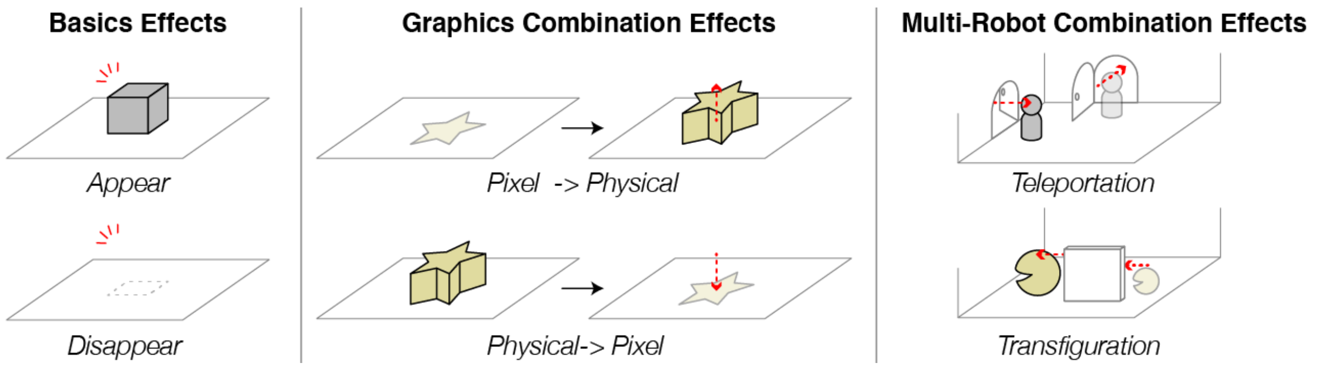

Figure 3: Expression and Illusions facilitated with Stage in *(Dis)Appearables.*

data representation, tangible interaction, and physical object manipulation.

3.1.2 Back-Stage. Back-stage is the space for actuated TUIs to perform background tasks outside of the users' attention. As shown in Figure 2a, the back-stage can be either *behind-wall* or *under-floor*. The back-stage allows UI systems to perform certain processes that are ideally hidden from a user's view. Such background tasks may include assembly and reconfiguration such as *Mechanical Shells* [34] and assembling blocks [65], or charging the actuated TUIs batteries. Furthermore, to appear to the front-stage effectively through the transition portal, the background task may include alignment and preparation for the entry onto the front stage.

3.1.3 Boundaries (Wall / Floor). Boundaries are physical barriers between the front-stage and back-stage. The boundaries are either walls (vertical) or floors (horizontal), which block users from seeing and reaching the actuated TUIs behind/underneath. Boundaries usually contain a type of transition portal that allow TUIs to transition between front- and back-stages.

3.1.4 Transition Portals. Transition portals bridge the front-stage and back-stage to allow self-propelled TUIs to move in-between. Thus, the portals are the key to the expression of (dis)appearance. Various types of portals could be used for different boundaries (wall/floor), intended interaction scenarios, and expressions.

3.2 Interaction Design Roles and Benefits of Stage

In this section, we summarize the roles and benefits of the *Stage* as two major points for the approach of *(Dis)Appearables*.

3.2.1 Directing Attention and Controlling Tangibility. By reconfiguring the position of actuated TUIs on the *Stage*, the user interfaces can be arranged so that they are intentionally hiding from users, and vice versa. With this, the stage provides novel opportunities for designers of an actuated TUI system to better focus and manage the users' attention in the foreground. Meanwhile, back-stage tasks can occur including battery charging, shell reconfiguration, or even the 3D printing of new shells.

3.2.2 Present Effects of Controlling Existence for Interactivity and Expression. With the design of *Stage* with front- and back-stages connected by portals, a new element of the interaction design is the way in which actuated TUIs enter and exit. *(Dis)Appearables* focuses on exploring this research space to demonstrate how *Stage* could support Self-propelled TUIs to control the existence on top of the *Stage* for user interaction. Furthermore, when the *Stage* is combined with *Mechanical Shells [31]*, to interchange the functionality and appearance of actuated TUIs, it becomes possible to design advanced expressions such as teleportation and transfiguration.

3.3 Physical Expressions and Effects Facilitated with Stage

The introduction of portals brings opportunity for robotic hardware to create novel physical expressions and illusions, which are described below (Figure 3).

3.3.1 Basic Effects: Appear and Disappear. Firstly, simple appearing and disappearing effects can be achieved by transitioning between the front- and back-stage. These are the basic effects presented and explored in *(Dis)Appearables*, and are used for other advanced effects as primitive effects.

3.3.2 Graphic Combination Effects. When the basic effect of appearing and disappearing is combined with graphical media (i.e. displays or projections), the stage allows the physical interface hardware to create effects that transition between pixel and physical. While such expression has been previously explored in stage performances [46], shape changing UIs [8, 40], or Human Robot Interaction [39], such technique combining with *Stage* in *(Dis)Appearables* for tabletop actuated TUIs brings great design oppurtunities which we later demonstrate.

3.3.3 Multi-Robot Combination Effects. When (dis)appearing effects are combined with multiple self-propelled device on *Stage*, other illusional effects can be created for user interaction. One example is an expression of *Teleportation* – enabled by the disappearance of one device from one portal, and the appearance of another device from another portal. By syncing the timing, it can convey the expression of the physical objects teleporting to other place.

By utilizing multiple devices, but with different shapes of actuated TUIs, it can create an expression of *Transfiguration*. By letting the two devices disappear and appear in co-located portal, it can

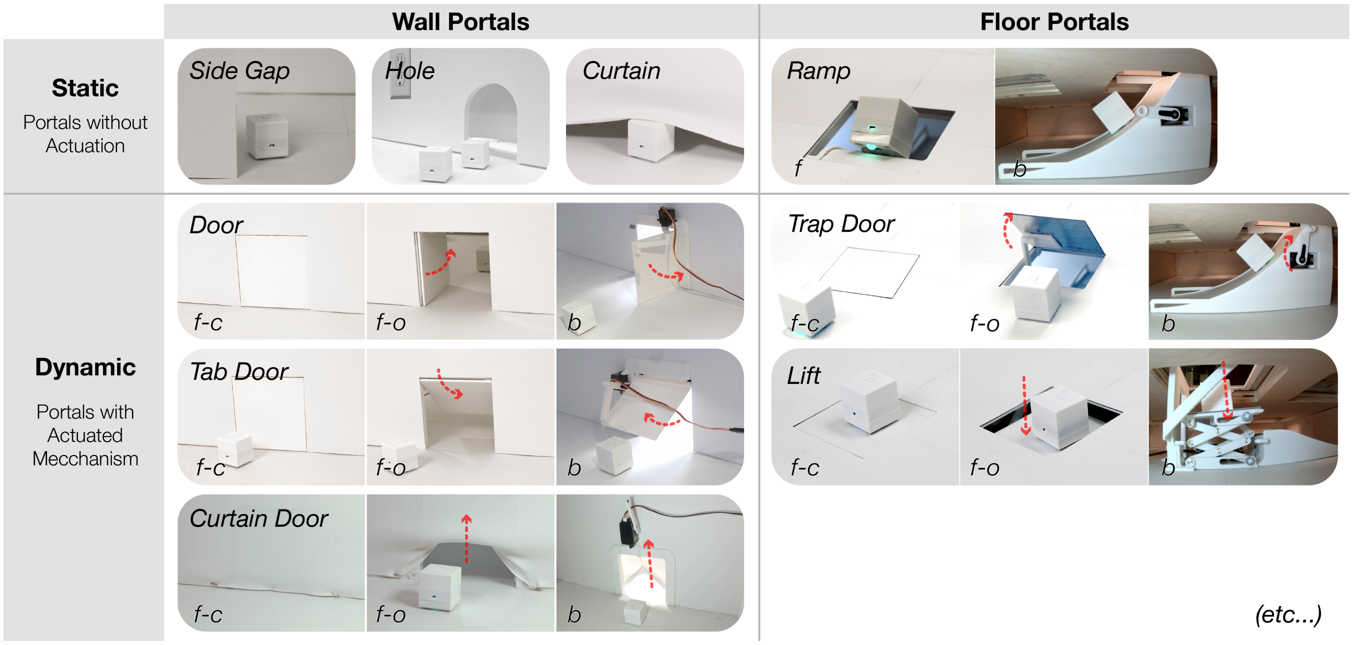

Figure 4: Transition Portals' Design Space in Four Quadrant based on Static / Dynamic, and Wall / Portal. (f-c: front-side closed, f-o: front-side opened, b: back-side)

convey an expression that a single device or character has changed its shape or size of its body. These techniques are often used in magic performances, and *(Dis)Appearables* intends to incorporate such techniques in the design of actuated TUI research [58].

4 TRANSITION PORTALS' DESIGN SPACE

Figure 4 shows the design space of the transition portals and example portals, which have been explored in our paper for tabletop wheeled TUIs. The design space is based on a four quadrants matrix with wall / floor portals, and static / dynamic portals. Static portals have no moving mechanism, while the dynamic portals contain actuation mechanisms (enabled by servo motors in our examples), so the portals can close and be concealed when they are not being used. The section below describes the example portals based on these categories. While this design space is based on the locomotion capability of wheeled robots on horizontal surfaces, other self-propelled devices would likely have different portal design spaces which would need to be further explored (as indicated in 7.6). Many of the transition portals are heavily inspired from theatrical stages [56], which effectively 'stage' a character's entrance and exit from the act to convey narratives to the audience depending on the portals.

4.1 Wall Portals

The simplest wall-portal, *Side Gap*, uses the edge of the wall. For this portal, the self-propelled TUI would simply go around the wall to transition between front- and back-stages. Other static wall-portals include, *Hole*, a simple tunnel on the wall (just like mouse holes seen in 'Tom and Jerry'), or *Curtains*. Unlike *Side Gap*, the other portals allow robots to transit in the middle of the wall.

As for the dynamic wall portals, we have implemented multiple types of doors using servomotors. The door hinge can be placed on the side or top. These hinged doors, with its explicit design, may be suitable for explicit character expression, such as entering/exiting buildings in storytelling. *Curtain Door* uses a fabric that can be actuated by a servo motor to vertically open and close. The advantage of employing the curtain portals for both static and dynamic uses is the concealing of the portals when not in use. These portals could be suitable for facilitating surprising user experiences by making physical objects appear and disappear without explicit anticipation.

4.2 Floor Portals

For the static floor portal, we prototyped *Ramp*. This portal allows the self-propelled TUIs to climb up the ramp to appear on the front stage through a hole in the floor. The dynamic portal, *Trap Door* adds an actuated hinged door to *Ramp* portal, so that when the portal is not in use, it can cover the hole in the floor for self-propelled TUIs to locomote over. Another dynamic floor portal is *Lift*, which vertically elevates the user interfaces up and down using a scissor mechanism.

Different types of portals can provide different ways for actuated TUIs to (dis)appear on the stage. Designers and researchers could effectively utilize these portal designs according to the intended applications. Some usages of these portals are demonstrated in the application section. Additionally, the portal prototypes presented in this paper are only examples, and there are a range of potential portal designs, especially when looking to theatrical stage design [56] as inspiration.

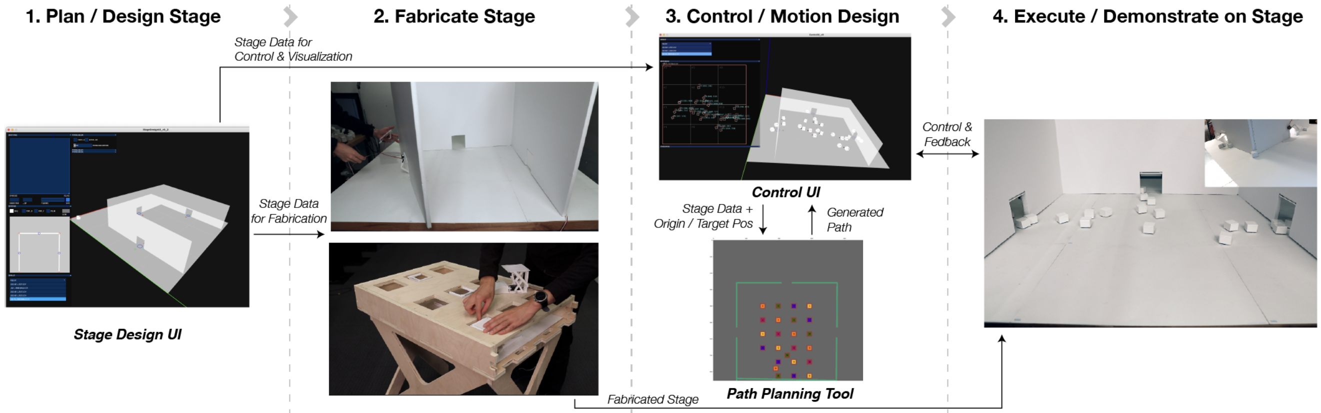

Figure 5: Four-phase Stage Design and Control pipeline.

5 STAGE DESIGN AND CONTROL PIPELINE

To explore the concept and design space of *(Dis)Appearables*, we implemented a system based on two-wheeled robots, *toio*™, used in HERMITS [34]. Building on top of previously implemented system, this paper focuses on a design and fabrication pipeline for *Stages*. We describe the four-phase design and control pipeline; 1) Design of Stage, 2) Fabrication of the Stage, 3) Control of self-propelled TUI on the Stage and 4) Control Execution (Figure 5) [1].

5.1 Plan and Design of the Stage with Design UI

First, to allow researchers and interaction designers to plan and design *Stage* as a platform for self-propelled TUIs, we have developed a GUI-based design tool for *Stages* using Processing (Figure 6). In this GUI-based software, users can freely design the size of the stage in a rectangular shape, the existence of an underground floor beneath the main stage, positioning of walls, and positioning of portals on the wall and on the floor. The maximum size of the stage was 1260 mm x 1188 mm based on available toio mats [5]. For the parameters of the portals, it is possible to adjust the types of portals (e.g. ramp or lift for floor portals) and other properties (e.g. direction of ramp).

With this software, users can plan the stage design and preview what the stage would look like with a real-time 3D visualization. After the design is completed, a design file can be exported to be used for the fabrication of *Stage*, as well as the control UI for the robotic hardware to navigate on the designed *Stage*.

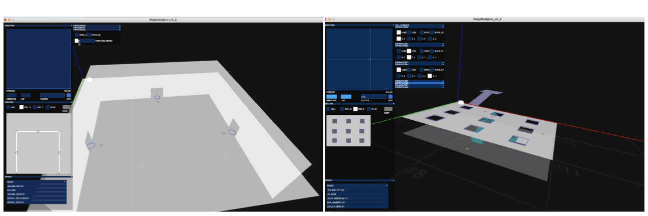

Figure 6: Example Views of Design UI of *Stage*.

[1]The source code for the implemented software (Design and Control UIs and Path Planning Tool) can be found on https://github.com/mitmedialab/disappearables_CHI2022.

5.2 Fabrication of the Stage

For the fabrication of *Stage*, the physical platform's basis has the similar composition as the one in HERMITS [34] - a multi-layer composition of ferromagnetic thin metal sheet and toio mats [5], placed on a rigid surface (e.g. wooden table). With the metal sheet on the interactive surface, the robots (embedded with magnets on their bottom) acquire a robust locomotion capability with stronger torque, which is suitable for haptic feedback [20] or mechanical shell control [34].

For the holes of floor portals, the metal sheet was cut using a water jet cutting machine. To compose a multi-layer stage with under floor, we have also utilized a CNC router (ShopBot), to cut out wooden boards to compose. The lower stage also has the multi-layer composition. For the wall, styrene board was cut and placed on *Stage* with extra holes for transition portals.

For transition portals with a dynamic mechanism, we have implemented a servomotor-activated mechanism. As shown in Figure 4, such servomotor-activated portals include doors, curtain doors, trap doors, and lifts. The portals on the floor (ramps, and lifts) were 3D printed with additional layers of toio mats and iron sheets to allow robots (embedded with magnets on the bottom as in [34]) to travel even a steep ramp (Figure 8b). These ramps and lifts have toio mats with fixed location pattern encoded, so that the control software can localize where robots are on the *stage* with corresponding portals. (Mats used for the portals and the base stage have different coordinate points so that the software can specifically identify the robots' positions.) As the toio mats [5] had a thickness of 0.1mm, we manually cut and overlayed them on the floor portal module. As shown in Figure 8a, we have designed these portal modules to be reconfigurable, so they can be reused in different stage designs.

5.3 Control System

Once the stage is fully fabricated, the robots can be placed on the *Stages* for control. To control the robots on the stage, we have developed two GUI software tools, a Control UI and a Path Planning Tool. The Control UI is an updated tool from the HERMITS' tool to handle the I/O data with robotic hardware via Raspberry Pi Bluetooth Modules [34]. The tool was developed in Processing as in the previous version. We have updated this software with

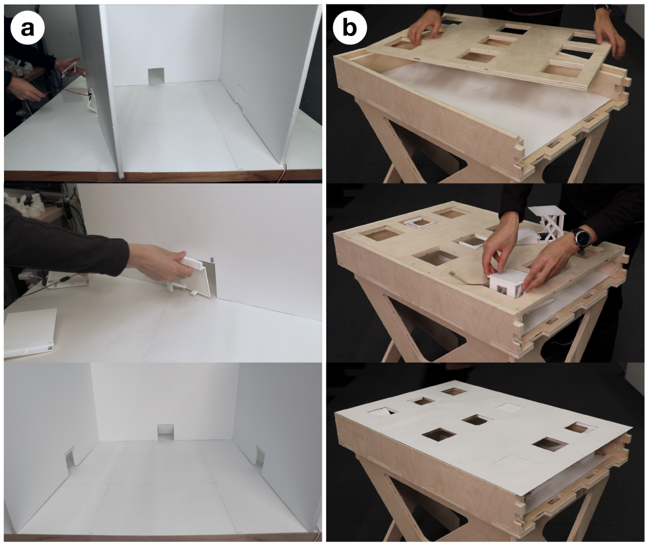

Figure 7: Example Fabrication of *Stage* (a. a stage with three walls and a hole on each wall, on a large stage surface, b. a stage with nine holes on a desk-shaped stage with under-floor design.)

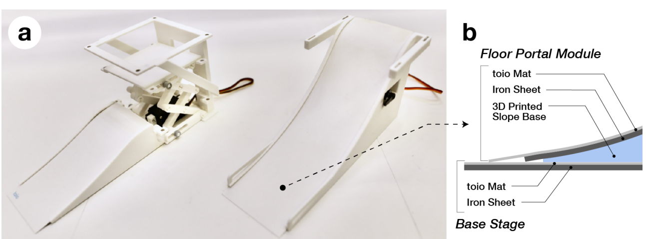

Figure 8: a: Modules of Floor Portals (Left-Lift, Right-Ramp), b: the section cut view of the Floor Portals with a layered design comprising toio mats and iron sheets.

added 3D visualizations which renders the detected robot positions in real-time on top of the model of *Stage* based on the exported design file from Design UI (Figure 9). With this control tool, users can control the behavior of robots using joysticks, keyboards and other primitive control interfaces. Additionally, the dynamic portals activated by servo motors were controlled with Arduino-based controller which is connected to the control computer with a wire. On this GUI, users can select origin and destination points on the stage for each robot using mouse cursors, which are passed to the path planning tool to generate paths between two points (Figure 9b).

5.4 Multi-Agent Path Planning Tool

The control UI communicates with a multi-agent path planning tool to automatically generate paths for multiple robots to navigate on *Stage* from designated origin points to target points.

To manipulate multiple robots on *Stages*, a C++-based path planning tool was developed based on a basic version of Conflict-Based

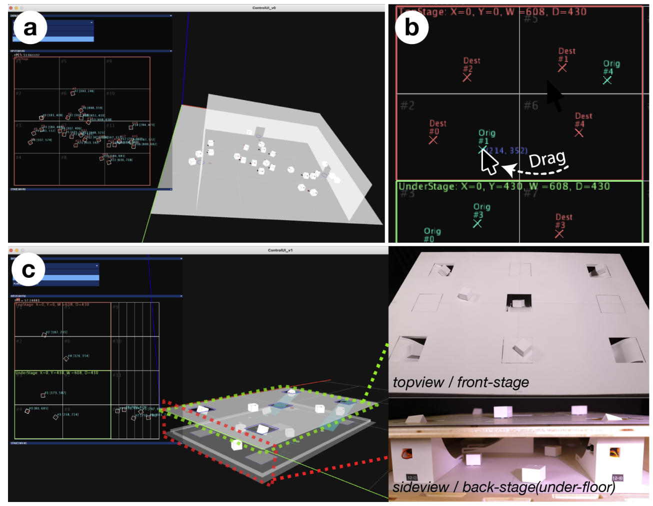

Figure 9: Control tool that visualize the real-time position of robots on the stages. (a: For the three wall stage in Figure 7a, b: Using a mouse cursor on the 2D view of GUI to specify the origin and destination points for each robot, c: Example of localization view with ramps and under-floor based on the design in Figure 7b. – the locations of robots on the software rendering and the actual setup are in sync.)

Search (CBS) [45], which is a popular optimal algorithm from Multi-agent Pathfinding (MAPF) literature. A survey by Stern et al. [47] gives a review of variants of MAPF problems and the algorithms developed for solving each variant. MAPF techniques have been popularly used in action planning for warehouse robots [30].

The path planner tool first reads a stage design file that is generated by the control UI and creates a grid map, where each grid size is equivalent to the toio robot's dimension – 32 x 32 mm. The tool also creates a set of blocked (impassable) cells on the grid map to represent the stage walls. It also reads the start and goal configurations (e.g., locations) of robots from a control file that is generated by the control UI. Based on the grid map and the robot configurations, the tool runs the CBS algorithm to find a set of collision-free paths for the robots. Each path contains a sequence of actions (move up, down, left, right, wait) that a robot needs to take at each time step. A set of paths is collision-free if robots do not collide with other robots or obstacles while moving on the map. A simple body collision checking is implemented based on the grid map. A circle shape safe zone, which uses the diagonal length of the robot as the circle's diameter, is defined for each robot. Two robots collide when their safe zones intersect. This circle design aims to avoid the potential collisions during rotations of robots.

Figure 10 shows, the generated path from origin to destination points for 24 robots on a stage, sized 1260 mm x 1188 mm, with three walls and three wall-portals. This multi-agent paths, for example, were generated approximately three seconds with our software.

Currently, the basic version of the control software does not take the reconfigurability of shells into account, as it would add to the complexity. However, it may be implemented in the future. For

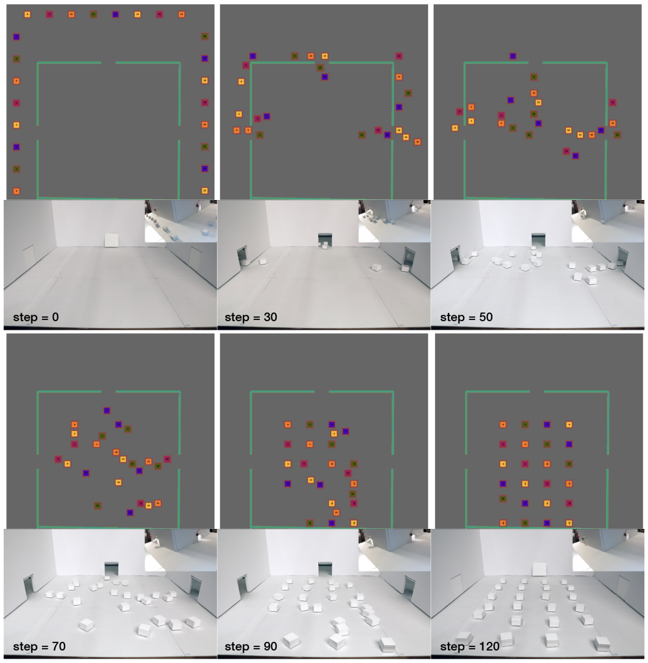

Figure 10: Sample Result of Path Planning. Total of 123 steps to navigate 24 robots on the stage (1260 x 1188 mm size)

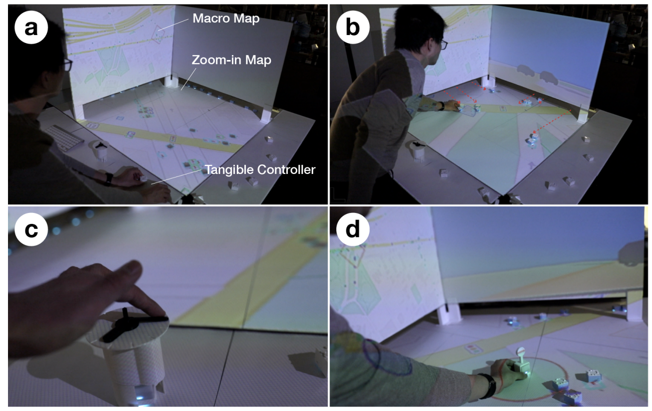

Figure 11: Urban Mobility Applications. (a. setup of interactive surfaces and projection, b. tangible interaction with robots, representing dynamic vehicle data, appeared from behind-wall, c. mechanical shells allowing time control, d. shells representing other tangible objects)

example, we could employ algorithms and planning techniques designed for MAPF for large agents (LA-MAPF), which takes different shapes of robots into account [26].

6 APPLICATIONS

Based on the approach and prototype presented, in this section we demonstrate the potential application space that utilizes the concept and technique of *(Dis)Appearables*. While each application demonstrates specific use cases, they indicate opportunities for novel design and research in interaction with digital data, computer devices, and physical environments via (dis)appearing effects.

6.1 Interactive Urban Mobility Simulation

The interactive mobility simulation demonstrates the use of *(Dis)-Appearables* for interaction with digital data. For the stage, the two walls placed on the stage hide the robots / actuated TUIs from the users' view, while the urban mobility simulation is projected on the front-stage (Figure 11a). Supplemental simulation information is projected on these walls for the users. As the users demand, the robots would appear from the backstage to snap to the graphical vehicle data point, so that users can directly grab the vehicle snapped to the data point (Figure 11c). By reconfiguring the modules in the back-stage, the robots can gain different shapes of module that provides different functionality in the mobility simulation (Figure 11c, d).

This application well-utilizes the capability of *(Dis)Appeaerables* by simulating a massive number of moving vehicles in a big city, but in a focused area, by circulating actuated TUIs with limited hardware resource in the background.

6.2 Affordance Design on Physical Desktop

(Dis)Appearables is a powerful approach to design affordances with TUIs as well as everyday physical objects – by radically controlling the existence of things. For example, an *(Dis)Appearable* embedded desk can make distracting devices disappear (Figure 12). Figure 12b shows a smartphone which, after being placed on a charger, is pulled out of users' reach behind a wall with robots. While the user is focused on an important task in the front-stage without the distraction, the phone can be charged in the back-stage. While researchers and industries work on features to disable notifications to help users' focus [1], physically disappearing the UI itself from users' reach and view may be more effective. Accordingly, on the physical work desk, appropriate UIs can (dis)appear according to what the users are supposed to focus on (e.g. a multi-functional mouse appear to support CAD operation as in Figure 12c). By utilizing *pixel to physical* transition effect, the CAD model would appear from behind the monitor as if the digital data was physicalized instantly (Figure 12d). Such functionality of on-demand shape appearance could be achieved by integrating the 3D print capability shown in Figure 15.

Applications using tabletop robots for physical desk reconfiguration has been explored in the past [19, 34, 53, 61], however, their tabletop surfaces are often cluttered with physical objects and (many) robots. Our example demonstrates how the interaction foreground space can be organized with the introduction of back-stage for.

6.3 Remote Table Hockey

The remote table hockey application demonstrates the teleportation and physical-digital transition effects of *(Dis)Appearables*. As shown in the Figure 13, the hockey pucks, which are self-propelled robots wearing shells, can transition in between one person's field to another person's field remotely. Following a hit by a person with a paddle, the puck can disappear from one person's field via a wide portal on the wall with a curtain where the real-time image

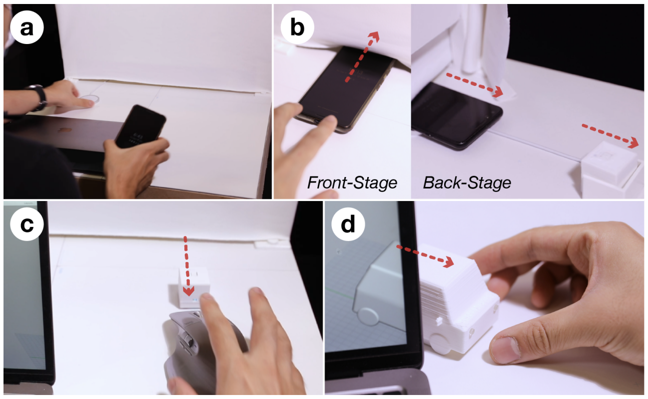

Figure 12: Desktop Reconfiguration Application (a. *(Dis)Appearables* embedded desk with minimal clutter on front-stage, b. A smartphone being pulled out from the front-stage (1) to the back-stage (2) after placed on the wireless charging pad – to help user focus on their task, c. A multi-functional mouse is pushed to the front for CAD task, d. Digital 3D CAD data in the display appearing into physical desktop with tangible shape.)

of the other person is projected (Figure 13 a ,b). The puck, then, can teleport to the another person's side of the field that is visible from the first person through the projected real-time image (Figure 13 c). Even if they are physically separated, they can have such a feeling of continuity through the virtually connected hockey fields via pucks which transition in-between. We believe the *perceived* transportation effect, together with physically replicating the force and speed of the opponents' hit, reinforces the physical presence of the another [24].

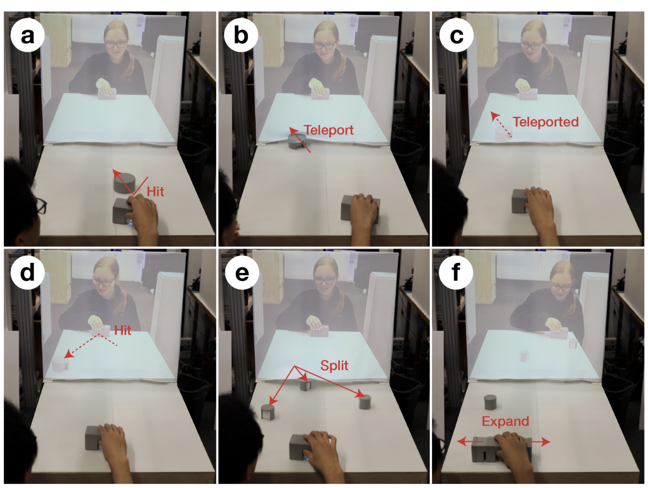

Figure 13: Remote Table Hockey Application that can create expression of teleportation for the hockey pucks to travel across remote desk surfaces.

Furthermore, by interchanging the shells in the back-stage, other special effects to make the game more dynamic and challenging can be achieved, such as splitting a puck into multiple pucks (Figure 13 d, e) – this sequence demonstrates a type of *transfiguration* effect for *(Dis)Appearables*, but with changing the *number* of objects, in combination with *transportation* effect. The paddle can also be composed with the *Mechanical Shell* with embedded robot, to dynamically shift the size of the paddle according to the game dynamics (Figure 13 f).

6.4 Gaming and Storytelling

The Pac-man application demonstrates multiple effects of *(Dis)-Appearables*. As shown in the Figure 14, just like the classic Pac-man, the characters can teleport from one end of the stage to the another to represent the continuous connection of the field (Figure 14 a1). The character can also transition to a larger body behind a wall, to chase and beat the ghosts (Figure 14 a2). As for interaction, this could be a physical representation of the original gaming controlled by other interfaces, or direct tangible gaming with rich shape representations that are, for example, studied to train and support motor skills for the elderly [11]. In a similar way, other character expressions, such as Mario traveling through pipes, can be expressed in a physical way which may advance existing tangible playing experiences [23] (Figure 14b).

Another gaming application is a card game. Many card games have employed the concept of 'summoning', to make monsters or characters appear on the field using their cards. While researchers have developed combining augmented reality to create such summoning expressions from cards with graphic overlays [2], the technique of *(Dis)Appearables* would enable the cards to summon physical props. As shown in Figure 14b, as the cards are placed on the floor portal by players (c1), the actuated TUI with an accompanying shell can appear from the floor surface, under the card, as if the physical characters emerged from the card (c2). The dragon shell prototype, as shown in (c3), is designed to transform it's shape from the compact box to a larger dragon-shape. While portal sizes in *(Dis)Appearables* restrict certain sizes and shapes of shells from moving between the front- and back-stages, such expansion transformation capabilities for shells could help to bypass such limitations.

6.5 On-demand Mechanical Shell 3D Printing in Back-Stage

To demonstrate how the dual stage design of *(Dis)Appearables* allows mechanical shells [31, 34] to be 3D printed on-demand in the back-stage, we have experimented to 3D Print shells directly on a 3D Printer bed, and control the toio robots to take the shells out from the 3D Print bed via a ramp. Figure 15 shows the example workflow. After taking the shells away from the 3D printer, the robots can bring the shell to the users in the front-stage for foreground interaction – in case of Figure 15, it is giving a heart-shaped small gift sent from a remote person via an opening box mechanism.

An ideal version of this application could automatically generate different design of mechanical shells according to the users' interaction happening in the front-stage (e.g. the 3D data physicalization seen in Figure 12d). To enable such interactions, instant 3D printing technology [52, 62] may be integrated into the *Stage* design in the

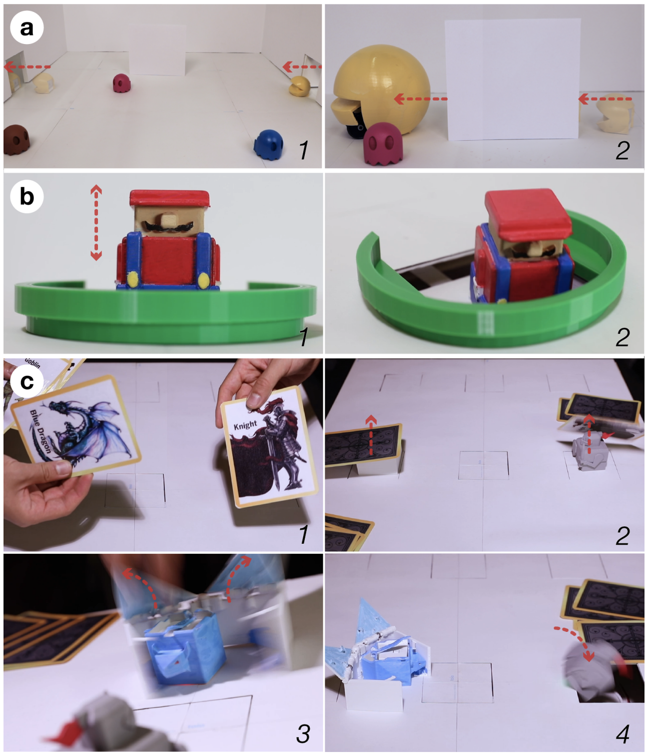

Figure 14: Gaming and Storytelling Applications (a. Pac-man application that demonstrates (1) teleportation, and (2) transfiguration effects, b. an expression of Mario traveling through pipes, c. card game application with (1, 2) an expression of summoning from cards, (3) shape expansion of the shell, and (4) character being defeated by pushed into a floor portal.)

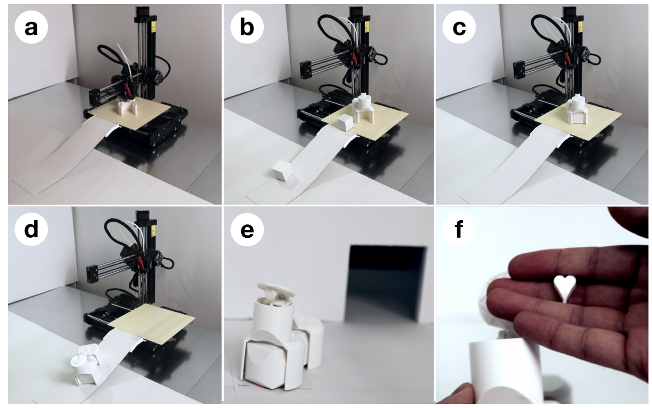

Figure 15: 3D Printing *Mechanical Shells* in the back-stage (a), robots picking up the shell from the print bed directly using a ramp (b, c, d), and the unit appearing on the front stage to provide foreground interaction (e). This example shows an application.

future. Shells with more complicated mechanism designs could be printed by employing recent 3D Printing technique [16].

7 DISCUSSION, LIMITATION AND FUTURE WORK

While we have introduced the general approach and concept of *(Dis)Appearables* with a proof-of-concept prototype, there is further research space and opportunities in this direction. In this section we discuss the limitations and potential future work below.

7.1 Multi-modal Stage Design Opportunity

While we have focused on the basic stage design primarily in the front- and back-stage designs inspired by theater designs, there are multi-modal design opportunities for the interactivity on *Stages*. For example, lighting is an important component to effectively guide users' attention to specific objects/characters during a stage performance, and such techniques could be applied to the *(Dis)Appearables* design with advanced light control using projectors, etc. Similarly, other theater techniques and methods including wire actions, smoke, silk screens, or sound effects could be incorporated into future *Stage* designs. These multi-modal interaction techniques and stage gimmicks should contribute to the effects and impact of appearing and disappearing, although the complicated technical implementation can be challenging to be resolved.

In HCI, there have been a broad set of studies explored to control the appearance of physical user interfaces and objects in combination with Augmented Reality system, and transparency changing materials [28, 29]. Combining those research modalities with our technique of physically obstructing the view/reach to TUIs can be an interesting direction to explore.

7.2 Optimizing Resource of Space for Back-Stage

As *(Dis)Appearables* step into designing the spatial environment for actuated and robotic UIs, a new design challenge is creating physical spaces with embedded back-stages. For example, when we implement the system in our living environment, it is important to optimize the back-stage space so that it does not take away space from the front-stage (or our foreground interaction space) while keeping enough space in the back for storing robotic hardware resource. Such optimization may better developed through a context dependant understanding of required applications and interactivity. In such a way, efficient space utilization would be a key criterion for *(Dis)Appearables* in the future, either by stacking unused hardware, or employing foldable mechanisms for certain modules, to maximize the use of the space.

7.3 Advanced Control of Robots on the Stage

Although our implemented control system of multiple robots are a basic tool that have successfully navigated multiple robots to reach their destination points on *Stages*, other advanced controls are needed for designing richer interaction and expression. For example, while our implementation took around three seconds to generate the path for 24 robots, dynamic and flexible path planning in response to the real-time interaction / intervention by users is crucial in the development of in situ interaction. Moreover, as

(Dis)Appearables introduces the incorporation of background tasks, including module assembly and battery charging, future control tools should automate these tasks. Such control tools also may be able to tune the way robots transition through the portal to create the appearing + disappearing effects more meaningful for user interaction – to achieve the variety of expression presented in our applications.

As for the methods to design the control of robots on the *Stage* for users, different ways to direct and orchestrate the motion can be improved. Rather than setting origin and destination coordinates in GUI tools, incorporating tangibility can be a more intuitive way to design motion, as explored in [9]. Physically grasping the robots on the stage, and moving them across the stage could be a way to direct the robots to make the path. With that, the automated path planning tool could intervene to help define the detailed path that avoids collisions between multiple robots. As two-handed control of multiple robots can be a difficult task for users, prior studies in gestural and spatial interaction with Swarm User Interfaces [18] could be applied as well.

7.4 Activating Stage and Portals with Robots

While the dynamic transition portals were actuated with servomotors in our prototype, these portals could be activated by the robots themselves as a means of driving the system with unified hardware. We have created a preliminary prototype to develop lift and door mechanisms activated by the toio-based robot hardware, but there were challenges the bulkiness of mechanism as well as torque limitations. These issues could be resolved with introducing improved mechanical shell designs with compact and torque optimized mechanisms. Control software to operate additional docking would be needed for such a tool as well.

7.5 User Study and Evaluation

The focus of this paper was in the development of an interaction concept and basic technology, and the future work could include a user study to explore and validate the effects introduced in *(Dis)Appearables*. While the COVID-19 pandemic made it difficult to allow people to experience the *(Dis)Appearables*, future user studies could elucidate which portal designs are best suited for specific types of application and storytelling expressions. One of the key questions here is to validate if users can believe physical objects to appear and disappear. How would users perceive and understand when physical objects appear and disappear, which would never happen in the physical world? How would such effects impact the affordance? What are the empirical differences from prior shape-changing and actuation approaches of (dis)appearing UIs [8, 40, 52]? Addressing these questions through user studies would be a meaningful contribution to the future design of actuated TUI and Stages.

Additionally, with the design and control tools developed in this paper, allowing designers, researchers, and children to prototype interactive applications could be another future work. Through such study, we would like to improve the design of our system and examine the broader application space created by users.

7.6 Designing Stages (or Space) for other Actuated TUIs to (Dis)Appear

Lastly, the project *(Dis)Appearable* with *Stages* using wheeled robots are one of the exploration and demonstration of our approach, and this can be applied to other self-propelled interactive hardware. For example, designing *Stages* for levitating TUIs (e.g. drones) [10, 22] would require 3D space deign (rather than 2D stage) considering more spatial locomotion capability for appearing and disappearing effects. In such way, other devices could include actuated curve interfaces [32, 33] (by extending locomotion capability as in snake robots [27]), room-scaled robots (vacuum robots) [12, 49], or even full-sized vehicles. Exploring the *Stage* and portal design that suits the locomotion capability of each interactive hardware is an open research agenda for future researchers.

8 CONCLUSION

In summary, we introduced *(Dis)Appearables*, an approach for self-propelled actuated TUIs to appear and disappear for user experience design using *Stages*. We have presented a design space for transition portals to enable versatile ways for the robotic entities to enter and exit from the users' attention, and introduced a design space for physical expression to be facilitated through the approach of *(Dis)Appearables*. We have developed a proof-of-concept prototype with two-wheeled robotic hardware and developed an implementation pipeline for *Stages* that includes hardware fabrication as well as design and control software. Multiple applications were presented to demonstrate the broad reconfigurable capabilities of the system. Through *(Dis)Appearables*, we demonstrate an approach for augmenting actuated TUI hardware with a *Stage* to serve as a conceptual and physical platform for a variety of robotic and actuated user interfaces. In doing this, we design a novel approach that expands the expressive range of future physical and tangible user interface designs. We believe designing peripheral environments for locomotive user interfaces is a rich opportunity for the engenderment of next generation interfaces.

ACKNOWLEDGMENTS

We thank Prof. Joseph Paradiso and Prof. Inami Masahiko, for their generous advice and feedback to the concept of this paper, as the first authors' thesis committee members.

We appreciate the toio team's voluntary members, including Akichika Tanaka (Sony Interactive Entertainment) and Dr. Alexis Andre (Sony CSL), for their feedback and support with the toio hardware.

We thank John Vincent and Dr. Pietro Buttoro from the Ford Motor Company for their insightful inputs specifically related to the mobility simulation application.

We thank Lauren Platt for the help with filming, Cedric Honnet for proofreading, and the members of the Tangible Media Group, Dr. Dan Novy, and Dr. Artem Dementyev for fruitful discussions for the project.

The research at the University of Southern California was supported by the National Science Foundation (NSF) under grant numbers 1409987, 1724392, 1817189, 1837779, and 1935712 as well as a gift from Amazon.

REFERENCES

[1] Apple. [n.d.]. *iOS 15 brings new ways to stay connected and powerful features that help users focus, explore, and do more with on-device intelligence.* https://www.apple.com/newsroom/2021/06/ios-15-brings-powerful-new-features-to-stay-connected-focus-explore-and-more/ - accessed January-6-2022.

[2] Santiago Bedoya-Rodriguez, Cristian Gomez-Urbano, Alvaro Uribe-Quevedoy, and Christian Quintero. 2014. Augmented reality RPG card-based game. In *2014 IEEE Games Media Entertainment*. IEEE, 1–4.

[3] Sean Braley, Calvin Rubens, Timothy Merritt, and Roel Vertegaal. 2018. GridDrones: A Self-Levitating Physical Voxel Lattice for Interactive 3D Surface Deformations.. In *UIST*. 87–98.

[4] Marcelo Coelho and Jamie Zigelbaum. 2011. Shape-changing interfaces. *Personal and Ubiquitous Computing* 15, 2 (2011), 161–173.

[5] Sony Interactive Entertainment. [n.d.]. *toio^{TM} Mat for Developers (in Japanese).* https://toio.io/news/2020/20200423-1.html - accessed January-6-2022.

[6] Sony Interactive Entertainment. [n.d.]. *Toy Platform toio^{TM}.* https://www.sony.net/SonyInfo/design/stories/toio/ - accessed January-6-2022.

[7] George W Fitzmaurice et al. 1997. *Graspable user interfaces.* University of Toronto.

[8] Sean Follmer, Daniel Leithinger, Alex Olwal, Akimitsu Hogge, and Hiroshi Ishii. 2013. inFORM: dynamic physical affordances and constraints through shape and object actuation.. In *Uist*, Vol. 13. 2501988–2502032.

[9] Phil Frei, Victor Su, Bakhtiar Mikhak, and Hiroshi Ishii. 2000. Curlybot: designing a new class of computational toys. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 129–136.

[10] Antonio Gomes, Calvin Rubens, Sean Braley, and Roel Vertegaal. 2016. Bitdrones: Towards using 3d nanocopter displays as interactive self-levitating programmable matter. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 770–780.

[11] Arzu Guneysu Ozgur, Maximilian J Wessel, Jennifer K Olsen, Wafa Johal, Ayberk Ozgur, Friedhelm C Hummel, and Pierre Dillenbourg. 2020. Gamified motor training with tangible robots in older adults: a feasibility study and comparison with the young. *Frontiers in aging neuroscience* 12 (2020), 59.

[12] iRobot. [n.d.]. *Roomba.* https://www.irobot.com/roomba - accessed January-6-2022.

[13] Hiroshi Ishii, Dávid Lakatos, Leonardo Bonanni, and Jean-Baptiste Labrune. 2012. Radical atoms: beyond tangible bits, toward transformable materials. *interactions* 19, 1 (2012), 38–51.

[14] Hiroshi Ishii and Brygg Ullmer. 1997. 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.

[15] Yvonne Jansen, Pierre Dragicevic, Petra Isenberg, Jason Alexander, Abhijit Karnik, Johan Kildal, Sriram Subramanian, and Kasper Hornbæk. 2015. Opportunities and challenges for data physicalization. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. 3227–3236.

[16] Shohei Katakura, Yuto Kuroki, and Keita Watanabe. 2019. A 3D printer head as a robotic manipulator. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*. 535–548.

[17] Hyunyoung Kim, Céline Coutrix, and Anne Roudaut. 2018. KnobSlider: Design of a Shape-Changing UI for Parameter Control. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–13.

[18] Lawrence H Kim, Daniel Drew, Veronika Domova, and Sean Follmer. 2020. User-defined Swarm Robot Control. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. xx–xx.

[19] Lawrence H Kim and Sean Follmer. 2017. Ubiswarm: Ubiquitous robotic interfaces and investigation of abstract motion as a display. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 1–20.

[20] Lawrence H Kim and Sean Follmer. 2019. SwarmHaptics: Haptic Display with Swarm Robots. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–13.

[21] Mathieu Le Goc, Lawrence H Kim, Ali Parsaei, Jean-Daniel Fekete, Pierre Dragicevic, and Sean Follmer. 2016. Zooids: Building blocks for swarm user interfaces. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. 97–109.

[22] Jinha Lee, Rehmi Post, and Hiroshi Ishii. 2011. ZeroN: mid-air tangible interaction enabled by computer controlled magnetic levitation. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*. 327–336.

[23] LEGO. [n.d.]. *LEGO®Mario™.* https://www.lego.com/en-us/themes/super-mario - accessed January-6-2022.

[24] Daniel Leithinger, Sean Follmer, Alex Olwal, and Hiroshi Ishii. 2014. Physical telepresence: shape capture and display for embodied, computer-mediated remote collaboration. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. 461–470.

[25] Daniel Leithinger, Sean Follmer, Alex Olwal, and Hiroshi Ishii. 2015. Shape displays: Spatial interaction with dynamic physical form. *IEEE computer graphics and applications* 35, 5 (2015), 5–11.

[26] Jiaoyang Li, Pavel Surynek, Ariel Felner, Hang Ma, T. K. Satish Kumar, and Sven Koenig. 2019. Multi-Agent Path Finding for Large Agents. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*. AAAI Press, 7627–7634. https://doi.org/10.1609/aaai.v33i01.33017627

[27] Pål Liljebäck, Kristin Ytterstad Pettersen, Øyvind Stavdahl, and Jan Tommy Gravdahl. 2012. A review on modelling, implementation, and control of snake robots. *Robotics and Autonomous systems* 60, 1 (2012), 29–40.

[28] David Lindlbauer, Jörg Mueller, and Marc Alexa. 2017. Changing the appearance of real-world objects by modifying their surroundings. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 3954–3965.

[29] David Lindlbauer, Jörg Müller, and Marc Alexa. 2016. Changing the appearance of physical interfaces through controlled transparency. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. 425–435.

[30] Hang Ma, Wolfgang Hönig, TK Satish Kumar, Nora Ayanian, and Sven Koenig. 2019. Lifelong path planning with kinematic constraints for multi-agent pickup and delivery. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 7651–7658.

[31] Ken Nakagaki. 2020. Mechanical Shells: Physical Add-ons for Extending and Reconfiguring the Interactivities of Actuated TUIs. In *Adjunct Publication of the 33rd Annual ACM Symposium on User Interface Software and Technology*. 151–156.

[32] Ken Nakagaki, Sean Follmer, Artem Dementyev, Joseph A Paradiso, and Hiroshi Ishii. 2017. Designing Line-Based Shape-Changing Interfaces. *IEEE Pervasive Computing* 16, 4 (2017), 36–46.

[33] Ken Nakagaki, Sean Follmer, and Hiroshi Ishii. 2015. Lineform: Actuated curve interfaces for display, interaction, and constraint. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. 333–339.

[34] Ken Nakagaki, Joanne Leong, Jordan L Tappa, João Wilbert, and Hiroshi Ishii. 2020. 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*.

[35] Gian Pangaro, Dan Maynes-Aminzade, and Hiroshi Ishii. 2002. The actuated workbench: computer-controlled actuation in tabletop tangible interfaces. In *Proceedings of the 15th annual ACM symposium on User interface software and technology*. 181–190.

[36] James Patten, Hiroshi Ishii, Jim Hines, and Gian Pangaro. 2001. Sensetable: a wireless object tracking platform for tangible user interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 253–260.

[37] Marianne Graves Petersen, Majken Kirkegaard Rasmussen, and Johan Trettvik. 2020. Affordances of Shape-Changing Interfaces: An Information Perspective on Transformability and Movement. In *Proceedings of the 2020 ACM Designing Interactive Systems Conference*. 1959–1971.

[38] Majken K Rasmussen, Esben W Pedersen, Marianne G Petersen, and Kasper Hornbæk. 2012. 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*. ACM, 735–744.

[39] David Robert and Cynthia Breazeal. 2012. Blended reality characters. In *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*. 359–366.

[40] Simon Robinson, Céline Coutrix, Jennifer Pearson, Juan Rosso, Matheus Fernandes Torquato, Laurence Nigay, and Matt Jones. 2016. Emergeables: Deformable displays for continuous eyes-free mobile interaction. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 3793–3805.

[41] John W Romanishin, Kyle Gilpin, and Daniela Rus. 2013. M-blocks: Momentum-driven, magnetic modular robots. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 4288–4295.

[42] Michael Rubenstein, Alejandro Cornejo, and Radhika Nagpal. 2014. Programmable self-assembly in a thousand-robot swarm. *Science* 345, 6198 (2014), 795–799.

[43] Daniela Rus. 1998. Self-reconfiguring robots. *IEEE Intelligent Systems and Their Applications* 13, 4 (1998), 2–4.

[44] Orit Shaer and Eva Hornecker. 2010. *Tangible user interfaces: past, present, and future directions.* Now Publishers Inc.

[45] Guni Sharon, Roni Stern, Ariel Felner, and Nathan R Sturtevant. 2015. Conflict-based search for optimal multi-agent pathfinding. *Artificial Intelligence* 219 (2015), 40–66.

[46] Shiro-A. [n.d.]. *Performance: Shiro-A at TEDxTokyo.* https://www.youtube.com/watch?v=PqjZFzcpUho - accessed January-6-2022.

[47] Roni Stern, Nathan R Sturtevant, Ariel Felner, Sven Koenig, Hang Ma, Thayne T Walker, Jiaoyang Li, Dor Atzmon, Liron Cohen, TK Satish Kumar, et al. 2019. Multi-agent pathfinding: Definitions, variants, and benchmarks. In *Twelfth Annual Symposium on Combinatorial Search*.

[48] Ivan E Sutherland. 1965. The ultimate display. *Multimedia: From Wagner to virtual reality* 1 (1965).

[49] Ryo Suzuki, Hooman Hedayati, Clement Zheng, James Bohn, Daniel Szafir, Ellen Yi-Luen Do, Mark D Gross, and Daniel Leithinger. 2020. 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*. xx–xx.

[50] Ryo Suzuki, Jun Kato, Mark D Gross, and Tom Yeh. 2018. Reactile: Programming Swarm User Interfaces through Direct Physical Manipulation. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–13.

[51] Ryo Suzuki, Abigale Stangl, Mark D Gross, and Tom Yeh. 2017. FluxMarker: Enhancing Tactile Graphics with Dynamic Tactile Markers. In *Proceedings of the 19th International ACM SIGACCESS Conference on Computers and Accessibility*. 190–199.

[52] Ryo Suzuki, Junichi Yamaoka, Daniel Leithinger, Tom Yeh, Mark D Gross, Yoshihiro Kawahara, and Yasuaki Kakehi. 2018. Dynablock: Dynamic 3D Printing for Instant and Reconstructable Shape Formation. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*. 99–111.

[53] Ryo Suzuki, Clement Zheng, Yasuaki Kakehi, Tom Yeh, Ellen Yi-Luen Do, Mark D Gross, and Daniel Leithinger. 2019. ShapeBots: Shape-changing Swarm Robots. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*. 493–505.

[54] Shan-Yuan Teng, Tzu-Sheng Kuo, Chi Wang, Chi-huan Chiang, Da-Yuan Huang, Liwei Chan, and Bing-Yu Chen. 2018. Pupop: Pop-up prop on palm for virtual reality. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*. 5–17.

[55] Shan-Yuan Teng, Cheng-Lung Lin, Chi-huan Chiang, Tzu-Sheng Kuo, Liwei Chan, Da-Yuan Huang, and Bing-Yu Chen. 2019. 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.

[56] Gary Thorne. 2011. *Stage design: A practical guide*. Crowood.

[57] Tommaso Toffoli and Norman Margolus. 1991. Programmable matter: concepts and realization. *Physica. D, Nonlinear phenomena* 47, 1-2 (1991), 263–272.

[58] Bruce Tognazzini. 1993. Principles, techniques, and ethics of stage magic and their application to human interface design. In *Proceedings of the INTERACT'93 and CHI'93 Conference on Human Factors in Computing Systems*. 355–362.

[59] Brygg Ullmer and Hiroshi Ishii. 1997. The metaDESK: models and prototypes for tangible user interfaces. In *Proceedings of the 10th annual ACM symposium on User interface software and technology*. 223–232.

[60] Brygg Ullmer and Hiroshi Ishii. 2001. Emerging frameworks for tangible user interfaces. *Human-Computer Interaction in the New Millenium* (2001), 579–601.

[61] Luke Vink, Viirj Kan, Ken Nakagaki, Daniel Leithinger, Sean Follmer, Philipp Schoessler, Amit Zoran, and Hiroshi Ishii. 2015. Transform as adaptive and dynamic furniture. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*. 183–183.

[62] Junichi Yamaoka and Yasuaki Kakehi. 2017. ProtoMold: An interactive vacuum forming system for rapid prototyping. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM, 2106–2115.

[63] Lining Yao, Ryuma Niiyama, Jifei Ou, Sean Follmer, Clark Della Silva, and Hiroshi Ishii. 2013. PneUI: 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.

[64] Mark Yim, David G Duff, and Kimon D Roufas. 2000. PolyBot: a modular reconfigurable robot. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, Vol. 1. IEEE, 514–520.

[65] Yiwei Zhao, Lawrence H Kim, Ye Wang, Mathieu Le Goc, and Sean Follmer. 2017. Robotic assembly of haptic proxy objects for tangible interaction and virtual reality. In *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces*. 82–91.