

Hyunyoung Kim University of Copenhagen Copenhagen, Denmark hyki@di.ku.dk Aluna Everitt University of Bristol Bristol, United Kingdom aluna.everitt@bristol.ac.uk Carlos E. Tejada University of Copenhagen Copenhagen, Denmark ct@di.ku.dk

Mengyu Zhong University of Copenhagen Copenhagen, Denmark fnh272@alumni.ku.dk Daniel Ashbrook University of Copenhagen Copenhagen, Denmark dan@di.ku.dk



Figure 1: MorpheesPlug is a toolkit to prototype shape-changing interfaces. (a) Users first use software to design a *widget* that can express shape-change, such as length or curvature change. (b) They then 3D print the widget with an off-the-shelf 3D printer. (c) They plug the widget into a control module that can adjust air pressure in the widget and actuate the widget. (d) An example application. The printed shape-changing interface holds the umbrella and gently pushes the umbrella towards user so that users can be reminded to take the umbrella.

# ABSTRACT

Toolkits for shape-changing interfaces (SCIs) enable designers and researchers to easily explore the broad design space of SCIs. However, despite their utility, existing approaches are often limited in the number of shape-change features they can express. This paper introduces MorpheesPlug, a toolkit for creating SCIs that covers seven of the eleven shape-change features identified in the literature. MorpheesPlug is comprised of (1) a set of six standardized widgets that express the shape-change features with user-definable parameters; (2) software for 3D-modeling the widgets to create 3Dprintable pneumatic SCIs; and (3) a hardware platform to control the widgets. To evaluate MorpheesPlug we carried out ten openended interviews with novice and expert designers who were asked to design a SCI using our software. Participants highlighted the ease of use and expressivity of the MorpheesPlug.

CHI '21, May 8–13, 2021, Yokohama, Japan

© 2021 Copyright held by the owner/author(s). Publication rights licensed to ACM. ACM ISBN 978-1-4503-8096-6/21/05...\$15.00 https://doi.org/10.1145/3411764.3445786

# **KEYWORDS**

Shape-changing Interfaces; Toolkit design

#### ACM Reference Format:

Hyunyoung Kim, Aluna Everitt, Carlos E. Tejada, Mengyu Zhong, and Daniel Ashbrook. 2021. MorpheesPlug: A Toolkit for Prototyping Shape-Changing Interfaces. In *CHI Conference on Human Factors in Computing Systems (CHI '21), May 8–13, 2021, Yokohama, Japan.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3411764.3445786

## **1 INTRODUCTION**

Shape-changing interfaces (SCIs) are emerging as a new generation of devices that can change their shapes to support dynamic affordances [9], leverage human dexterity [37], and support the personalization of physical interfaces [22]. The current design space of SCIs covers a wide range of features [20], including variable length [9], volume [22], curvature [50], and porosity [5]. The literature has featured numerous prototype systems exploring a huge variety of shapes, shape-changes, interactions, implementation techniques, and applications.

Despite the potential of SCIs to enhance the development of the next generation of interactive devices, there are still many challenges faced by the field [2]. One major barrier to the creation of SCIs is the lack of standardized toolkits for exploration and development [2]. Current approaches require substantial time, effort, domain-specific knowledge, and complex tools to create even simple SCIs. Unlike software-only user interfaces, physical UIs—including

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

Hyunyoung Kim, Aluna Everitt, Carlos E. Tejada, Mengyu Zhong, and Daniel Ashbrook

SCIs—interact with physical reality, requiring the addition of hardware components. Researchers have developed physical toolkits to simplify creating physical UIs, providing standardized hardware widget libraries [3, 11] and tools to ease the communication between the digital and physical worlds [17].

SCIs introduce new problems, because there are no standardized widget libraries, actuation methods, or design tools. It means that to experiment with or develop such UIs requires users need to design, fabricate, and implement all aspects of shape-changing systems. As a result, the literature illustrates many one-off application-specific SCIs [44]. Alexander et al. note that a primary necessary strand of the field is to create "a standard platform for hardware prototyping".

There are two primary challenges to creating such a standardized toolkit for prototyping SCIs. The first is *actuation*: given a desired shape change, how to choose a technical method to cause that transformation. Researchers have identified dozens of shape-change features [20] (e.g., length) and actuation methods [44] (e.g., servo motor), but there are no standards or guidelines for how a user can select a method to implement a desired feature.

Closely coupled with the issue of actuation is that of *fabrication*: how to physically instantiate an actuation method that causes the desired shape-change. Toolkits for physical UIs offer pre-made physical widgets that let users concentrate on applications [3, 11], but the bulk of the SCI literature focuses on novel techniques (e.g., [35]) or applications rather than broadly reusable widgets. The result is that SCIs tend to be one-offs, custom-made for a specific application, and require extensive technical prototyping skills.

As a first step towards addressing these challenges, we introduce MorpheesPlug, a toolkit aimed at simplifying the design, fabrication, and actuation of SCIs. MorpheesPlug does so by following in the footsteps of successful GUI and physical computing toolkits: providing physical widgets, control hardware and firmware, and a design environment. MorpheesPlug simultaneously addresses the *actuation* and *fabrication* challenges by providing six pneumatically powered shape-changing widgets which express a broad range of shape-change features from the Morphees+ framework [20]. Users can customize these widgets and incorporate them in their own SCI designs, eliminating the need to choose an actuation method for specific shape-change features and simplifing the design process. MorpheesPlug widgets are printable on commodity 3D printers with standard flexible filament, significantly lowering the barrier to prototyping SCIs.

With MorpheesPlug , we make the following contributions:

- We provide six customizable, 3D-printable widgets that express a wide range of shape changes via pneumatic actuation.
- (2) We characterize widget performance over a variety of printing parameters, illustrating the range of shape changes available.
- (3) We implement and publically share design software and control module for the widgets<sup>1</sup>.
- (4) We demonstrate the utility of our toolkit via five proof-ofconcept applications and a qualitative user study.

#### 2 RELATED WORK

MorpheesPlug is a toolkit that simplifies creating and exploring SCIs, using pneumatic widgets. As such, it is situated at the intersection of SCIs, physical UI toolkits, and pneumatically actuated soft UIs and robotics. In this section, we situate MorpheesPlug in the context of toolkit research, both for SCIs and physical UIs. Then we look into how pneumatic actuation was used for shape-changes.

#### 2.1 Toolkits for Shape-changing Interfaces

Alexander et al. [2] identified twelve grand challenges in SCIs research. Although many types of SCIs have been explored in the literature [44], most are custom-made, one-off projects developed to illustrate an interaction technique, actuator, or application; hence, Alexander et al. [2] call for the development of *toolkits for SCI* to "dramatically lower the barrier to implementation". They call for three advances in research: a hardware prototyping platform, a software application layer, and tools for end-user programming. To these three we add a fourth important need, adapted from Ledo et al. [25]: empowering new audiences, implying ease of acquisition or fabrication. A number of projects in the literature aim to overcome these barriers, either by explicitly presenting toolkits for SCIs or by addressing one or more of these challenges.

Ledo et al. [25] define toolkits as "present[ing] users with a programming or configuration environment consisting of many defined permutable building blocks, structures, or primitives, with a sequencing of logical or design flow affording a path of least resistance". While few papers in the SCI space explicitly identify their work as presenting toolkits, in this section we include research which addresses any aspects which could be useful as part of a toolkit.

Perhaps the most comprehensive example of a SCI toolkit is ShapeClip [14], a set of 1D linear actuators controlled by light emitted from standard computer screens. While it addresses Alexander et al. 's three research threads, the ShapeClip hardware consists of complex electromechanical components not readily accessible to casual users. The hardware also limits the types of shape-change to those that can be expressed via length feature.

Other systems, while not explicitly identified as toolkit research, present useful hardware building blocks for SCIs. One approach is to use electromechanical actuators as a driver of shape-change; for example, perhaps the earliest example approaching an SCI toolkit was Topobo [38], a system of passive and active (motorized) build-ing blocks that could record and re-play movements. LineFORM [32] and ChainFORM [31] are similarly collections of actuators which can record and re-play movements, but focus on rotational rather than linear motion. Each of these systems is constrained by its actuators: using motors limits the minimum size, dictates the kinds of shape-change transformations available, and leads to high-complexity hardware, requiring custom circuitry that is unavailable to a casual user.

Another type of SCI system uses shape-memory alloy (SMA) or nitinol wires to actuate shape changes. SMA-based actuation has the advantage of small size and flexibility, but at the expense of actuation speed. One early example, Bosu [36], offered a set of frames and fabric shapes on which the SMA wires could be fixed. While these components formed a small library of transformable

<sup>&</sup>lt;sup>1</sup>https://github.com/shape-changing-interfaces/MorpheesPlug

shapes, Bosu required users to assemble each component manually. NURBSforms [46] operated on the same principle as Bosu, but used flexible circuit boards, providing a standardized — and potentially mass-manufacturable — format. Both of these toolkits demonstrate shape changes based primarily on the curvature feature from Morphees+ [20], a result of the low-amplitude length change possible with SMA.

Some systems use pneumatic actuation to transform shapes. One of the earliest projects in this space was PneUI [50], which offered a technological framework for pneumatically actuated shape-change. Although the downside was that its shape-changing objects were all manually created, it illustrated the versatility of soft, pneumatic shape-change via multiple types of transformations, including curvature, volume, and texture. Other pneumatically actuated SCIs include Printflatables [41] and AeroMorph [35], both of which require custom-built equipment to create, and Siloseam [28], which presents a manual workflow for shape-changing silicone bladders.

Aside from ShapeClip, none of these examples present themselves as toolkit research. Instead, they focus more on novel actuation schemes and possibilities for expressing shape changes. One result of this limited focus is the lack of standardized widgets to express a wide range of shape-change features: most of these systems present at most one or two reusable transforming shapes and can express a fraction of the Morphees+ [20] feature space. Our goal with MorpheesPlug is to provide a diverse set of shape-change widgets that enable experimentation with much larger coverage of the feature space, while being easily fabricated by users with minimal required equipment and expertise.

#### 2.2 Physical UI Toolkits

Although few toolkits exist for SCIs, many of the same challenges are addressed by toolkits for physical user interfaces; in fact, SCIs can be viewed as a subset of physical UIs. In contrast to GUIs which take advantage of standardized hardware such as touchscreens or keyboards, physical UI toolkits aim to make novel input and output mechanisms accessible to non-expert users.

One of the earliest physical computing toolkits was Phidgets [11]. It applied the idea of GUI widgets to physical interaction controls, enabling a combination of function and interface in a reusable building-block component. Later physical UI toolkits expanded on this idea, adding novel connections between modules [3], more powerful widgets [49], or novel form-factors [18]. These examples illustrate a *prefabricated* approach, where the physical widgets are designed and manufactured by a third party, and end users assemble, but don't usually modify them. The advantage of this approach is less work for users, who can experiment with a set of validated widgets. The downside is that form-factors and capabilities are limited by the widget manufacturer's priorities.

A second approach to physical widgets is custom-fabrication. Toolkits in this category provide assistance to users in creating widgets (or widget-like components) tailored for a particular application. Midas [42], for example, provided tools to help users fabricate customized touch sensors that could wrap around objects of varying sizes; Pineal [24] added "remote widgets" to smartphones and watches via automated 3D modeling; and PaperPulse [39] fabricated predefined widgets with conductive inkjet printing. The advantage of this approach is much-greater flexibility: users can include different sizes and types of widgets in many configurations. However, customized widgets for each application can mean much greater time and effort for the user.

MorpheesPlug takes inspiration from both types of physical computing toolkit. We provide a set of predefined shape-change widgets which are customizable in the design stage, and then can be fabricated on unmodified commodity 3D printers. In this way we aim to support users with a set of pre-validated widgets that can be re-used if desired, but that have enough customizability to be tailored for a variety of applications.

## 2.3 Pneumatic Shape-Change

In order to grant MorpheesPlug widgets the broadest range of possible shape changes, while still being easily fabricatable by end users, we use air pressure as an actuation source. Many other projects in HCI and other fields have similarly used pneumatics for driving flexible interfaces and robotics.

Examples of pneumatically driven interfaces have mainly concentrated on exploring the diversity of interaction that such soft interfaces can offer. For example, Kim et al. 's Inflatable Mouse [23] illustrated multiple input and output behaviors, Harrison and Hudson's inflatable buttons provided dynamic haptics [16], and PneUI [50] demonstrated a wide variety of shape changes possible with elastic air bags. Despite the versatility of these interfaces, they are difficult to create, involving intensive manual assembly. Recent work by Moradi and Torres [29] underscores both the versatility and difficulty of working with flexible materials, demonstrating a wide range of shape change and investing considerable effort in laying out a workflow to lessen the effort of fabrication.

Some research has investigated 3D printing for pneumatically actuated SCIs. Although subject to the limitations of 3D printers, creating SCIs this way can—at least in theory—significantly lessen the effort required to create usable transforming objects. Vazquez et al. created a series of physical widgets using 3D printing [48], and Lee et al. developed a system of Lego-compatible pneumatic blocks for experimenting with soft robotics [26]. These projects relied on high-end multi-material inkjet-based 3D printers, which are not currently easily accessible to most end users; the materials available for these printers have low stretchability. Another possibility for 3D printing flexible objects is via FDM printing, using flexible filaments such as thermoplastic polyurethane (TPU). Thus far, most progress in TPU actuators has been made in the field of soft robotics, where the emphasis has been on locomotion and grasping [51].

MorpheesPlug 's pneumatic actuation is inspired by these previous efforts. Despite the versatility of these related approaches, their main shortcoming is ease of use, requiring complex fabrication, and actuation techniques. We directly tackle these challenges in two ways. First, we provide users with an easy-to-use design environment for creating SCIs. Second, MorpheesPlug uses inexpensive off-the-shelf fabrication equipment and material to create multiple widgets, enabling a wide range of shape-change possibilities.

Hyunyoung Kim, Aluna Everitt, Carlos E. Tejada, Mengyu Zhong, and Daniel Ashbrook

# **3 DESIGN RATIONALE**

Before building MorpheesPlug , a toolkit for prototyping SCIs, we discuss what kind of design goals we wanted to achieve in MorpheesPlug in terms of toolkit design. We looked into review literature that suggests design guidelines for toolkits [25]. Here we discuss how MorpheesPlug meets four of the five goals in toolkit research.

- (1) Reducing Authoring Time and Complexity. Fabricating SCIs is a challenging task. This process often entails the use of specialized equipment and requires engineering expertise. To address this challenge, we encapsulate the knowledge of the type of shape-change our six widgets will exhibit when pneumatically actuated. This, coupled with the analysis of how each widget implements features of SCIs taxonomies, allows designers to have an estimation of the expected shape-change the widgets will exhibit before fabricating them, reducing time, effort, and domain knowledge when building new SCIs.
- (2) Empowering New Audiences. Complex 3D modeling and electrical engineering can be a barrier for non-expert users who want to step in the area of SCIs. To simplify the process of designing the widgets [34], we provide a plug-in for CAD software that is widely available. Without a need for manually 3D modeling the widgets, users can choose the widget type and alter the parameters of it to to create 3D models with the plug-in. To evaluate if MorpheesPlug can be used by new audience than researchers in SCI field, we conduct a user study with hobby makers.
- (3) Integrating with Current Practices and Infrastructures. While pneumatically actuating SCIs allows designers to create a wide range of shape-change with a single actuation method, the fabrication of these artifacts is not always a straightforward process, often requiring manual assembly [50], or special machinery [35, 41]. Our work aims to use existing, consumer-level tools (e.g., off-the-shelf 3D-printers, materials, and design tools) to fabricate SCIs.
- (4) Enabling Replication and Creative Exploration. Ideal toolkits should support easy replication of previous work [10] and exploring design spaces that has not examined before [34]. To show that MorpheesPlug has such properties, we replicate one of the SCIs that had a huge impact in the field [9] as well as suggest novel interfaces with MorpheesPlug.

Based on these design goals, we designed MorpheesPlug . We aimed to build MorpheesPlug to be easy to use for researchers as well as engineering novices to significantly reduce their iteration time and effort. One goal suggested for toolkit design that we did not aim was *Creating Paths of Least Resistance*, which means that toolkits should guide users to design good interfaces rather than bad ones [2, 30]. SCI field is still at the early stage, and we believe that there are too few design guidelines to be generalized (e.g., [13, 21, 47]) comparing to the vast design space of SCIs. Therefore, we planned not to guide users what kind of SCIs they should design at this stage of the research. Future studies can contribute to the design guidelines for SCIs using MorpheesPlug , as it would allow quickly implementing a wide range of SCIs.

# 4 MORPHEESPLUG WIDGETS

MorpheesPlug is comprised of three basic components: (1) a set of shape-changing widgets; (2) a design environment; (3) a control module. A widget is the minimum unit in MorpheesPlug that creates shape-change when 3D-printed and then pneumatically actuated. Widgets are the core of MorpheesPlug . The design interface is a plug-in for CAD software that users can create 3D models of the widgets and customize them on the software. A module is a physical interface that users can control air pressure in a widget. This section shows how we designed the widgets and how they can express shape-change features.

We designed the widgets primarily based on the features and also literature from HCI, soft robotics, and material science. Note that we excluded the speed, feature, because the feature relies on the actuation method, not the design of the widgets. Also, we did not include stretchability, granularity, and strength, because we first wanted to focus on features that involve clear visual shape-changes in the scope of the paper. Figure 2 shows the widgets we designed, and Figure 3 shows how the widgets can express shape-change features. Below, we describe how we designed each widget and how they can express the shape-change features.



Figure 2: The six widgets that MorpheesPlug provide. The widgets can express different shape-change features such as length, curvature, etc.

# 4.1 Fold widget

The Fold widget is a widget that is primarily designed to implement length change (Figure 3). It is consists of a single layer of thin chamber that is folded in 90 degrees several times. The structure was originally used in material science [4] as a dielectric elastomer actuator. When inflated, the fold slightly opens, and the whole structure elongates.

We found that the widget can implement all of the shape-change features we aimed for. For example, when length-changing widgets are connected to make a rectangular shape or cube, they can also implement area and volume features under the size feature. To change modularity feature, it can be attached a a static object and elongate in a slot. It would lock the static object and slot together. To express porosity, there can be several Fold widgets and a solid surface on top of them. When the widgets elongate, they close the space between the widgets and the surface. When they shorten, they open the space and increase porosity. We considered that the widget can express amplitude and curvature features at the same

CHI '21, May 8-13, 2021, Yokohama, Japan



Figure 3: Top: Widgets provided by MorpheesPlug. Left: shape-change features from literature [20, 40]. Middle: Illustrations of how the widgets can express the shape-change features.

time. When there are multiple Fold widgets on the same flat surface and some of them elongate, the surface would look like a curved surface. It would express amplitude and curvature features. In the same sense, when the widgets elongate while some in between them do not, they can express zero-crossing. Lastly, when the widget is placed aligned to a surface and elongates, it would change closure feature between an end of the widget and an end of the surface.

# 4.2 Spiral widget

Spiral widget is a widget that has a curved thin chamber. When looked from the top of the widget, it resembles a spiral shape. This widget is design to express changes in the curvature feature. When inflated, the curved surface unbends and changes the angle between the central point and the end points of the surface. This widget can also represent changes in the length feature. When inflated, the distance between two diametrically opposed point

Hyunyoung Kim, Aluna Everitt, Carlos E. Tejada, Mengyu Zhong, and Daniel Ashbrook

increases. Additionally, if the widget is designed to have multiple arcs, its enveloping area increases when inflated. While doing so, the porosity of the widget also increases as the space between the arcs increases. Similarly to Fold widgets, a Spiral widget can be put in a slot and inflated to lock itself in the slot. In this way, the two objects that contain the slot and the widget can combine into one and change modularity feature. When a Spiral widget changes curvature, it also changes amplitude. The widget can also express zero-crossing. When there are multiple Spiral widgets placed next to each other and when only some of them are deflated, the deflated widgets would create bumpy surfaces therefore change zero-crossing. When a Spiral widget has a single spiral and is inflated, the distance between the two end points increases, changing closure.

## 4.3 Teeth widget

Similarly to the Spiral widget, Teeth widget is designed to express curvature and amplitude. However, unlikely to Spiral, a Teeth widget has a straight shape when deflated and bends when inflated. The length between two end points of a Teeth widget would be decreased when the widget is inflated. Users can put a Teeth widget on a flat surface and increase porosity between the widget and the surface by inflating the widget. By connecting multiple Teeth widgets and inflating one of every second of them, users can express zero-crossing. When it is inflated the two ends of the widget get closer, expressing closure feature.

## 4.4 Bump widget

Bump widget is designed to have it on a flat surface and express a bumpy surface on it. Users can have several of them connected to each other. When one Bump widget is inflated, it can express the length feature. When it is inflated in a slot, it can lock an object attached it and the slot, expressing modularity. When there are multiple Bump widgets and there are static objects on and under them, inflation of the widgets would change porosity between the widgets and the objects. Similarly, when there are multiple Bump widgets and only some of them are inflated, they change amplitude, curvature, and zero-crossing.

# 4.5 Accordion widget

Accordion widget is designed to take advantages of both Fold and Bump widgets. Like the Fold widget, it can express length feature when elongated. Thanks to it, it can express all the features that Fold widget can express. Like Bump widget, it can have several chambers on the surface like tiles. Because the chambers are connected, users can express curvature, amplitude, and zero-crossing features on a connected smooth surface, similarly to PolySurface [6]. Thanks to the grooved surfaces on the four sides, it can have more length change than Bump widget.

# 4.6 Auxetic widget

We designed our auxetic widget to display porosity feature. I got inspired from the literature [12]. When inflated, the widget opens up width-wise, enlarging a central area and thus increasing its porosity. In addition, once actuated, the width of this widget increases, also displaying shape-change in the area feature. Further, when it has reach its maximum shape-change, the outer shapes of separate from each other, exhibiting the closure feature. Lastly, it is able display the modularity feature once expanded by attaching to near objects.

# **5** IMPLEMENTATION

MorpheesPlug is comprised of three main components: (1) a set of 3D-printable, inflatable widgets that can represent seven out of eleven of the Morphees+ features [20]; (2) a design environment for makers to model SCIs; (3) a control module responsible for actuating SCIs widgets. All of the resulting output (hardware, firmware, software, and designs) will be made available online.

#### 5.1 Design Software

Our design environment is built on top of Autodesk Fusion 360 (Figure 4) using its Python API for scripting remote command execution. In order to design a shape-changing widget using our tool, the user selects the desired widget from a drop-down list, and proceeds to modify the controlling parameters. The design automatically updates to match the user's inputs.



Figure 4: Detail view of our design environment, based on Autodesk Fusion 360. The user is presented with a window where they can specify the sizes for the different parameters that compose our widgets.

# 5.2 Fabrication

All MorpheesPlug widget designs are printed as a single structure using consumer-grade FDM<sup>2</sup> 3D-printers with elastic filament (NinjaFlex, shore 85A). To test our designs, we fabricated dozens of our widgets using three different 3D-printers (Lulzbot Taz 6, Ultimaker 2, and Crealitly Ender 3 Pro). During our initial tests, we found that the default print settings for these printers would produce non-airtight objects, causing the resulting objects to exhibit very little shape-change, if at all. To address this, we explored different printing settings for each of our printers, and got best results by over-extruding our designs, lowering the print speed, and increasing the numbers of top and bottom layers to 10 and 7, respectively.

<sup>&</sup>lt;sup>2</sup>Fused Deposition Modeling

When the overhang surface too large, we allowed support. An in detail view of the parameters can be found in Table 1.

Our explorations uncovered a trade-off between the wall thickness of our widgets, and their subsequent airtightness, and their respective shape-change capabilities: thicker walls provide better seals, but restrict the shape-changing capabilities of the widgets. We opted for maximizing the shape-change capabilities of our widgets, by using only two layers of perimeter shells throughout our designs. This decision, however, meant that on occasion our widgets would print with small imperfections on their outer walls, causing air to leak. It could be addressed by dipping the widget on flexible resin (Formlab Elastic 50A Resin), and cured it.

 Table 1: List of modified printing parameters with their respective values used to fabricate our widgets

Parameter	Value	Parameter	Value
(TAZ 6, Ender 3)	1200	(Overhang < 4 cm)	0 %
Printing Speed		Interior Fill	
(mm/min)			
(Ultimaker 2) Printing	400	(Overhang > 4 cm)	10 %
Speed (mm/min)		Interior Fill	
Extrusion Multiplier	1.3	Combine Infill Every	2 layers
Top Solid Layers	10	Outline Overlap	25 %
Bottom Solid Layers	7	Outline/Perimeter	2
		Shells	

# 5.3 Characterization

We developed six 3D-printable, inflatable, shape-changing widget designs. For MorpheesPlug to be of the most practical use, we wish to quantify the shape-changing capabilities of our widget designs. To do so, we explored the effects of the constructing parameters for our designs by fabricating numerous instances of our widgets, systematically varying these parameters. We constrained our explorations in two ways. First, our preliminary experiments revealed that widgets with heights of less than 2 cm display very little shape-change. Second, we were unable to print airtight overhang surface wider than 4 cm without support. With support, widgets showed less flexibility and shape-change in general. To keep the print setting the same over the widgets show the maximum possible shape-change, we decided to fabricate widget designs with sections less than or equal to 4 cm (e.g., Thickness in Figure 4). We printed the Bump widget vertically. With these constrains in place, we set to print various iterations of our designs while changing each of the constructing parameters, one at a time. Once printed, we connected each widget to our control module, and actuated it setting our compressor at 100 kPa (kiloPascals). We proceeded to record the difference in size from each of our widgets, as seen in Figure 5, repeating each measure ten times.

Figure 6 presents the results of our explorations. We learned that the parameters that influence the area sections of the widgets that are perpendicular to the direction of the shape-change affect the most the behavior of the widgets, while the parameters that affect area sections parallel to the direction of the shape-change negate the shape-changing capabilities of our widgets. We believe this is because parameters that are parallel to the direction of shape-change



Figure 5: Our characterization setup. To measure the length change of the Fold widget, we placed a printed widget next to a ruler and measured the length of both deflated and inflated states.

restrict our widgets' movement when inflated, but parameters that are perpendicular to this movement do not, while at the same time increasing the structure's inflatable volume.

# 5.4 Control Module

The final part of our toolkit is an electronic control module to allow designers to easily control the actuation of our widgets (Figure 7). This module, measuring 4 cm x 4 cm x 4.5 cm, is made up of five components: (1) two electronic solenoid valves to control airflow to, and from the widgets; (2) a barometric pressure sensor; (3) a custom circuit board used to interconnect all the components from our module; (4) an LED to display to the designer the status of the valves; (5) and a micro-controller to drive all the components. Once the widgets have finished printing, the designer proceeds to connect them to our control module.



Figure 7: An exploded view of the control module. The module has two valves to let air in and out of the widgets.

## 6 **DEMONSTRATION**

We present five applications to demonstrate the capability of MorpheesPlug to express various shape-change features in different scales. These applications were created using our design environment, widgets, and modules, but were manually actuated by the authors.

#### 6.1 Umbrella pusher

The umbrella pusher is to demonstrate the spiral widget's ability to hold an object. It also uses the fact that widget's character that when it unbends less when is has a lower height (Figure 8 left). To create the umbrella pusher, we first created a spiral widget with 2cm

Hyunyoung Kim, Aluna Everitt, Carlos E. Tejada, Mengyu Zhong, and Daniel Ashbrook



Figure 6: Results of the characterization. These plots show magnitude of the modified features versus the change of shape the widgets expressed. The numbers on the widgets show the baseline size of the features. For example, Fold widget had a baseline parameters of gap 10 mm, length 20 mm, height 20 mm, and width 10 mm. We then changed each parameter one by one, e.g., changed length from 10 mm to 80 mm (red line in the plot).

height and three arcs with our plug-in. We then manually lowered the height of the central part to make the part hold an umbrella even when the widget is actuated. We then 3D printed the model with zero in-fill.

The umbrella pusher can installed at an apartment entrance and hold an umbrella. On rainy days, the spiral widget gets inflated and pushes the umbrella towards users when they approach to it (Figure 8 middle, right).

#### 6.2 Anti-rain phone case

The anti-rain phone case is to show that MorpheesPlug supports heterogeneous shape-changes and integration with existing 3D models. We combined three Teeth widgets and one Fold widget to create a shape that bends from back of a phone can elongate. We then combined them with a 3D model of a phone case form the Internet (Figure 9 left). When the phone case is inflated, it bend over the phone screen and block rain, strong sun-light on the phone



Figure 8: Left: The 3D model of an umbrella pusher created by our plug-in and edited in CAD software. Middle: The umbrella pusher is holding an umbrella. Right: On a rainy day, the umbrella pusher slightly unbends and pushes the umbrella towards on the way of users to remind to take the umbrella. The central part of it unbends less and still holds the umbrella.

(Figure 9 middle, right). It can also prevent someone from looking at private information similarly to a shape-changing phone [40].



Figure 9: Left: The 3D model of anti-rain phone case. Middle: A user holding a 3D printed anti-rain phone case. Right: When it rains, the phone case can inflate and block rain drops over the phone.

#### 6.3 Posture-correcting cushion

The posture-correcting cushion (Figure 10) is to show that MorpheesPlug can handle high pressure and human weight. We used the same 3x3 Accordion widget from the characterization section.

When users sit on the cushion in an incorrect posture, it can push them to remind them to sit correctly. Unfortunately, the pressure of the cushion was higher than the sensing range of the pressure sensors in the current version of our module.





Figure 10: Left: The 3D model of posture-correcting cushion. Middle: A user sits on the cushion learning forward. The cushion recognizes higher pressure at the front. Right: The front cushion inflates and correct the user's posture.

#### 6.4 Physical bar chart

The physical bar chart (Figure 11) was specifically designed to replicate pin-based SCIs (e.g., [7, 9, 15, 33, 45]). We wondered if MorpheesPlug could easily replicate existing SCIs, and pin-based SCIs have been widely used to introduce novel interactions and understandings of SCIs. Our 3D printer took 21 hours to print nine Fold widgets (2.3 hr per widget), which would take longer

than using off-the-shelf parts. However, the widgets would allow less assembly time because users just need to plug tubes to the widgets and connect them to modules. Although it allowed quick prototyping, the final form is different from existing pin-based SCIs. First, they do not have the "pin" shape. To improve the form factor, users would need to print a static pin on top of each widget or assemble and hide the widgets. Second, the resolution is lower than high-resolution ones (e.g., a pin in inFORM [9] has a 9.525 mm<sup>2</sup> footprint). Currently a widget has a footprint of 625 mm<sup>2</sup> (2.5 cm x 2.5 cm) and need spaces between them. If we reduce the footprint of the widget, the widget would have less length change with the same number of folds. Additionally, we noticed the bar chart tilted slightly to one side when actuated. As a possible remedy, we could insert plastic separators between each of the fold widgets to prevent tilting on actuation.



Figure 11: Left: A 3D model of Fold widget. Middle: We 3D printed nine copies of the 3D model and put them in a grid. Right: Some of the widgets are actuated.

# 6.5 Window blind

The window blind is designed for an aesthetic purpose. We created seven Auxetic widgets in three different sizes, and then manually combined with added connecting space between the neighboring widgets (Figure 12 left). When inflated, the widgets expand and increase porosity between and within the widgets. User can adjust the porosity by changing the air pressure in the widgets.



Figure 12: Left: The 3D model of the window blind. Middle: Deflated window blind. Right: Inflated window blind.

#### 7 USER STUDY

We conducted a user study to evaluate MorpheesPlug plug-in. We were particularly interested if the plug-in meets the design rational we aimed. As we aimed that our toolkit enables novices to create SCIs and also integrates with current practices, we invited both novice and expert users in our study, in terms of 3D modeling skills.

Hyunyoung Kim, Aluna Everitt, Carlos E. Tejada, Mengyu Zhong, and Daniel Ashbrook

#### 7.1 Participants

Participants were recruited through an advert on a social media page used by a local maker community and the word-of-mouth. Participants approached the researcher via email and we recruited 10 participants (age 25-64, female 2).

We got participants with both expert and novice skill levels in 3D modeling. With novice users (P6-8,10, background in Computer Science, Communication, or Public Health. No or little 3D modeling experiences), we wanted to see if they can understand the toolkit and implement their ideas using our plug-in. With experts (P1-7,9, background in Architecture, Industrial Design, or Robotics. Advanced skill in 3D modeling or using 3D modeling software at work), we wanted to see if the plug-in can integrate with their experience of 3D modeling and help them save time. Four out of six expert participants had experience in Fusion 360 (P2,3,5,9). Regarding experiences related to SCIs, P9 was both a hobbyist and also founded a small-sized enterprises in soft robotics. P4 had experiences in fabricating inflated tin foils, and P2 had experiences in compliant mechanisms [19]. We compensated each participant with a \$20 worth local product.

#### 7.2 Procedure and Tasks

The studies were performed in person for all participants aside from P9, where the study was done via video conference. Each study took around one hour and was recorded via audio, video, and screen recording with consent.

Each study consisted of three parts. First, the participants signed a consent form and answered to biographical questionnaire (10 min). Second, we showed examples of SCIs [9, 50], and asked the participants to brainstorm ideas for new shape-changing interfaces they want (20 min). To help them brainstorm, we asked them to think about their work and daily life. We also provided them a few ideas from other participants when they wanted [43]. They were then asked to choose one of the ideas to design for the rest of the user study. Lastly, we demonstrated our printed widgets and two example applications (umbrella pusher and anti-rain phone case) and asked the participants to 3D model their ideas using MorpheesPlug plug-in in a think-aloud manner (30 min). We showed them how to use the plug-in and supported them when they do not know how to use other Fusion 360 functions. After the 3D modeling, we asked them questions about strengths and weaknesses of the MorpheesPlug plug-in. Note that the Auxetic widget was not implemented in the plug-in at the moment of the study.

#### 7.3 Results

Three of the authors analyzed the transcribed interviews from all the participates. We specifically focused on the design rationale we discussed earlier in the paper. Additionally, we wanted to understand the usability of MorpheesPlug plug-in and the potential direction for enhancing the plug-in.

*7.3.1 General response.* Overall, all participants were excited when seeing our widgets and example applications, showing both the potential novelty and usability of MorpheesPlug for designing SCIs. P10 stated: "It was relatively straight forward and intuitive...".

P4 was enthusiastic in seeing all the widgets we developed and exploring their actuation.

7.3.2 Reducing authoring time and complexity. All participants agreed that the authoring time and complexity of designing from scratch is reduced by our plug-in. All participants except P8 could closely design what they sketched using the plug-in. This demonstrates the potential for fast adoption of our toolkit plug-in for designing novel SCIs. P3 emphasised that our plug-in "... only required a little bit of modification to execute my idea." Similarly, P6 stated that they were "positively surprised at how easy it was to implement a version of my sketch using the basic shapes available." This positive response is likely due to our plug-in providing ready to use widgets, where users do not need to design from scratch.

In terms of customisation, participants also stated that they would like to directly edit the widgets by dragging arrows (e.g., for stretching etc) and not only use numeric input for parameters when creating the widgets. Two participants also commented that they wanted to edit more parameters. For example, P3 wanted one and a half layers in Accordion and P4 also wanted to change the length of gap in Accordion, which is currently fixed to 1cm. Furthermore, some participants brought out the need of automatically generating the widget parameters by determining the dimension of shape-changing (e.g., P7 need to have *X* amount of length change).

7.3.3 Empowering new audiences. Complex 3D modeling can be a barrier for non-expert users who want to step in the area of SCIs. The four novice participants we recruited appreciated that our plugin enabled them to create 3D models without expert skills required. The 3 out of 4 novices (P6, P7, P10) were able to understand the concept of SCIs, the actuation capabilities of our widgets, and design their own 3D model from their sketches.

7.3.4 Integrating with current practices and infrastructures. MorpheesPlug aimed to use existing, consumer-level tools to fabricate SCIs. To achieve this, we built the widget design software as a plugin for Fusion 360. Participants who had experiences with Fusion 360 showed efficiency of creating their intended designs. For example, P2 was able to create a relatively complex shape in less than 30 minutes (see figure 13). They created Bump widgets and added rigid parts around them as a handle to hold a pen by squeezing one bump widget.

On the other hand, users who were not used to 3D modeling or Fusion 360 struggled with using functions other than the MorpheesPlug plug-in. They had to tell us what they want to do, and we had to tell them where the related functions are and how to use them. It hindered them from exploring and editing the 3D models. To make MorpheesPlug widespread, we need to create plug-ins for other CAD software or develop an independent software.

7.3.5 *Enabling creative exploration.* The plug-in enabled creative exploration by letting users explore the widgets and their parameters. As P3 explained: "Another advantage would be to see that. You know, not all the time we can imagine all the possible shapes. When you have a plug-in, you see an idea where to start with, 'Okay, this may be possible' ... Probably I can also do something with that...just by looking at the module I can learn what are the possibilities".

However, the difficulty of editing the widgets caused difficulties for users (especially novices) to freely explore the widgets. More



Figure 13: Idea sketch and its 3D model from P2. It is a penpressor to help people who lack fine motor skills. The two Bump widgets inside of the rigid body has an air channel between them. Users can press the rigid body to compress one of the widgets, and the air in the widget would travel to the other widget to press the top of the pen.

than one participant revealed the desire for characterization of widgets as well as real time simulation of shape-changes for better understanding how the widgets would behave. Another suggestion for improving the creativity was having random or irregular shapes automatically generated for users to explore the properties of shape changing widgets (P3).

# 8 DISCUSSION AND FUTURE WORK

MorpheesPlug 's widgets are able to express seven of the eleven shape-changing features detailed in the Morphees+ taxonomy [20]. We intentionally focused on features of this space that resulted in significant physical change, what meant that features like granularity, stretchability, strength, and speed are not in the scope of this work. As mentioned earlier, we use a constant pressure of 100kPa to power our module and widgets, which causes smaller widgets to be actuated quicker than larger ones. Future work can explore how to employ pneumatic actuation to represent these features by using techniques such as jamming [8], or by dynamically regulating pressure to vary actuation speed.

Future work can also continue to explore the effects of different fabrication parameters on the shape-change potential for MorpheesPlug widgets. For example, our Fold widget not only changes length when actuated, but also curvature as the surface becomes uneven and round. This effect is caused by the homogeneous thickness of the outer walls of the widgets. There is opportunity to explore the effects of such parameters to more precisely control how each widget expresses its respective shape-changing features.

Conversely, while we were able to successfully fabricate shapechanging widgets using consumer-level elastic filament (Ninjaflex, shore 85A) on off-the-shelf hardware, the limited elasticity of this CHI '21, May 8-13, 2021, Yokohama, Japan

material reduced the shape-changing capability of our designs. For example, when we fabricated our Fold widget with 1 cm height, its shape varied very little when inflated. More elastic materials, such as silicone, could allow larger shape change on smaller objects, at the expense of ease of fabrication.

Continuing, although our control module only presents a single air output to actuate the widgets, designers can actuate multiple widgets in tandem by making use of Y-splitters to connect multiple widgets to a single module. Additionally, while a single computer can control multiple modules, these must be connected via USB. We plan to explore different ways to control our modules (e.g., via Bluetooth, or WiFi), and alternatives for controlling various widgets with a single module. These improvements could benefit the portability of our work.

Once printed, our widgets are airtight, and capable of holding their shape after actuation. While in our experiments we used a dedicated air compressor to power our module and widgets, in the future we wish to explore more-accessible options by testing the efficacy of miniature air pumps. A further benefit could be miniaturization by embedding pumps into the control module.

Finally, we plan to evaluate MorpheesPlug in terms of the quality of interaction. It would be interesting to compare MorpheesPlug to SCIs that have other mechanism other than pneumatic actuation, such as mechanical [1] or manual [27].

## 9 CONCLUSION

In this paper, we presented MorpheesPlug, a toolkit for prototyping shape-changing interfaces (SCIs). By providing six widgets and using pneumatic actuation, the toolkit expresses seven shape-change features. To make the widgets accessible to users, we implemented a plug-in for CAD software where users can change the parameters of the widgets. We presented three applications using MorpheesPlug and conducted user studies to illustrate MorpheesPlug's ability to express shape-change features and easily prototype SCIs. We envision that MorpheesPlug can be a first step towards building a standardized toolkit for prototyping SCIs.

#### ACKNOWLEDGMENTS

We thank Jason Alexander, Irina Shklovski, Kasper Hornbæk, and the HCC Section at the University of Copenhagen for reviewing the paper. This work is supported by the University of Copenhagen and EPSRC (EP/M021882/2).

#### REFERENCES

- Parastoo Abtahi and Sean Follmer. 2018. Visuo-Haptic Illusions for Improving the Perceived Performance of Shape Displays. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (Montreal QC, Canada) (CHI '18). Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3173574.3173724
- [2] Jason Alexander, Anne Roudaut, Jürgen Steimle, Kasper Hornbæk, Miguel Bruns Alonso, Sean Follmer, and Timothy Merritt. 2018. Grand Challenges in Shape-Changing Interface Research. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (Montreal QC, Canada) (CHI '18). Association for Computing Machinery, New York, NY, USA, Article 299, 14 pages. https://doi.org/10.1145/3173574.3173873
- [3] Ayah Bdeir. 2009. Electronics as Material: littleBits. In Proceedings of the Third International Conference on Tangible and Embedded Interaction (TEI'09). ACM, New York, New York, USA, 397–400. https://doi.org/10.1145/1517664.1517743
- [4] Federico Carpi, Claudio Salaris, and Danilo De Rossi. 2007. Folded dielectric elastomer actuators. *Smart Materials and Structures* 16, 2 (mar 2007), S300–S305. https://doi.org/10.1088/0964-1726/16/2/s15

Hyunyoung Kim, Aluna Everitt, Carlos E. Tejada, Mengyu Zhong, and Daniel Ashbrook

- [5] Marcelo Coelho and Pattie Maes. 2009. Shutters: A Permeable Surface for Environmental Control and Communication. In Proceedings of the 3rd International Conference on Tangible and Embedded Interaction (Cambridge, United Kingdom) (TEI '09). Association for Computing Machinery, New York, NY, USA, 13–18. https://doi.org/10.1145/1517664.1517671
- [6] Aluna Everitt and Jason Alexander. 2017. PolySurface: a design approach for rapid prototyping of shape-changing displays using semi-solid surfaces. In Proceedings of the 2017 Conference on Designing Interactive Systems. 1283–1294.
- [7] Åluna Everitt, Faisal Taher, and Jason Alexander. 2016. ShapeCanvas: An Exploration of Shape-Changing Content Generation by Members of the Public. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (San Jose, California, USA) (CHI '16). Association for Computing Machinery, New York, NY, USA, 2778–2782. https://doi.org/10.1145/2858036.2858316
- [8] Sean Follmer, Daniel Leithinger, Alex Olwal, Nadia Cheng, and Hiroshi Ishii. 2012. Jamming User Interfaces: Programmable Particle Stiffness and Sensing for Malleable and Shape-Changing Devices. In UIST '12: Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology. ACM Press, New York, New York, USA, 519–528. https://doi.org/10.1145/2380116.2380181
- [9] Sean Follmer, Daniel Leithinger, Alex Olwal, Akimitsu Hogge, and Hiroshi Ishii. 2013. InFORM: Dynamic Physical Affordances and Constraints through Shape and Object Actuation. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology* (St. Andrews, Scotland, United Kingdom) (*UIST '13*). Association for Computing Machinery, New York, NY, USA, 417–426. https://doi.org/10.1145/2501988.2502032
- [10] Saul Greenberg. 2007. Toolkits and interface creativity. Multimedia Tools and Applications 32, 2 (01 Feb 2007), 139–159. https://doi.org/10.1007/s11042-006-0062-y
- [11] Saul Greenberg and Chester Fitchett. 2001. Phidgets: Easy Development of Physical Interfaces through Physical Widgets. In Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology - UIST '01. ACM Press, Orlando, Florida, 209. https://doi.org/10.1145/502348.502388
- [12] Joseph N Grima and Kenneth E Evans. 2006. Auxetic behavior from rotating triangles. Journal of materials science 41, 10 (2006), 3193–3196.
- [13] Erik Grönvall, Sofie Kinch, Marianne Graves Petersen, and Majken K. Rasmussen. 2014. 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 (Toronto, Ontario, Canada) (CHI '14). Association for Computing Machinery, New York, NY, USA, 2559–2568. https://doi.org/10. 1145/2556288.2557360
- [14] John Hardy, Christian Weichel, Faisal Taher, John Vidler, and Jason Alexander. 2015. ShapeClip: Towards Rapid Prototyping with Shape-Changing Displays for Designers. In CHI '15: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems. ACM Press, New York, New York, USA, 19–28. https://doi.org/10.1145/2702123.2702599
- [15] John Hardy, Christian Weichel, Faisal Taher, John Vidler, and Jason Alexander. 2015. ShapeClip: Towards Rapid Prototyping with Shape-Changing Displays for Designers. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (Seoul, Republic of Korea) (CHI '15). Association for Computing Machinery, New York, NY, USA, 19–28. https://doi.org/10.1145/ 2702123.2702599
- [16] Chris Harrison and Scott E. Hudson. 2009. Providing Dynamically Changeable Physical Buttons on a Visual Display. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09). Association for Computing Machinery, Boston, MA, USA, 299–308. https://doi.org/10.1145/1518701.1518749
- [17] Björn Hartmann, Leith Abdulla, and Scott Klemmer. 2007. Authoring Sensor-Based Interactions by Demonstration with Direct Manipulation and Pattern Recognition. In CHI '07: Proceeding of the Twenty-Fifth Annual SIGCHI Conference on Human Factors in Computing Systems. 1–10.
- [18] Steve Hodges, Nicolas Villar, Nicholas Chen, Tushar Chugh, Jie Qi, Diana Nowacka, and Yoshihiro Kawahara. 2014. Circuit Stickers: Peel-and-Stick Construction of Interactive Electronic Prototypes. In Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems - CHI '14. ACM Press, Toronto, Ontario, Canada, 1743–1746. https://doi.org/10.1145/2556288.2557150
- [19] Larry L Howell. 2013. Compliant mechanisms. In 21st century kinematics. Springer, 189–216.
- [20] Hyunyoung Kim, Celine Coutrix, and Anne Roudaut. 2018. Morphees+: Studying Everyday Reconfigurable Objects for the Design and Taxonomy of Reconfigurable UIs. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (Montreal QC, Canada) (CHI '18). Association for Computing Machinery, New York, NY, USA, Article 619, 14 pages. https://doi.org/10.1145/3173574. 3174193
- [21] Hyunyoung Kim, Céline Coutrix, and Anne Roudaut. 2019. KnobSlider: Design of a Shape-Changing Parameter Control UI and Study of User Preferences on Its Speed and Tangibility. *Frontiers in Robotics and AI* 6 (2019), 79. https: //doi.org/10.3389/frobt.2019.00079
- [22] Seoktae Kim, Hyunjung Kim, Boram Lee, Tek-Jin Nam, and Woohun Lee. 2008. Inflatable Mouse: Volume-Adjustable Mouse with Air-Pressure-Sensitive Input

and Haptic Feedback. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (Florence, Italy) (CHI '08). Association for Computing Machinery, New York, NY, USA, 211–224. https://doi.org/10.1145/1357054.1357090

- [23] Seoktae Kim, Hyunjung Kim, Boram Lee, Tek-Jin Nam, and Woohun Lee. 2008. Inflatable Mouse: Volume-Adjustable Mouse with Air-Pressure-Sensitive Input and Haptic Feedback. In Proceeding of the Twenty-Sixth Annual CHI Conference Extended Abstracts. ACM, New York, New York, USA, 211–224. https://doi.org/ 10.1145/1357054.1357090
- [24] David Ledo, Fraser Anderson, Ryan Schmidt, Lora Oehlberg, Saul Greenberg, and Tovi Grossman. 2017. Pineal: Bringing Passive Objects to Life with Embedded Mobile Devices. In CHI '17: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems. ACM Press, New York, New York, USA, 2583–2593. https://doi.org/10.1145/3025453.3025652
- [25] David Ledo, Steven Houben, Jo Vermeulen, Nicolai Marquardt, Lora Oehlberg, and Saul Greenberg. 2018. Evaluation Strategies for HCI Toolkit Research. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (Montreal QC, Canada) (CHI '18). Association for Computing Machinery, New York, NY, USA, Article 36, 17 pages. https://doi.org/10.1145/3173574.3173610
- [26] Jun-Young Lee, Jaemin Eom, Woo-Young Choi, and Kyu-Jin Cho. 2018. Soft LEGO: Bottom-Up Design Platform for Soft Robotics. In 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). 7513–7520. https://doi.org/ 10.1109/IROS.2018.8593546
- [27] John C. McClelland, Robert J. Teather, and Audrey Girouard. 2017. Haptobend: Shape-Changing Passive Haptic Feedback in Virtual Reality. In Proceedings of the 5th Symposium on Spatial User Interaction (Brighton, United Kingdom) (SUI '17). Association for Computing Machinery, New York, NY, USA, 82–90. https: //doi.org/10.1145/3131277.3132179
- [28] Hedieh Moradi and César Torres. 2020. Siloseam: A Morphogenetic Workflow for the Design and Fabrication of Inflatable Silicone Bladders. In *Proceedings of the* 2020 ACM on Designing Interactive Systems Conference (Eindhoven, Netherlands) (DIS '20). Association for Computing Machinery, New York, NY, USA, 1995–2006. https://doi.org/10.1145/3357236.3395473
- [29] Hedieh Moradi and César Torres. 2020. Siloseam: A Morphogenetic Workflow for the Design and Fabrication of Inflatable Silicone Bladders. In Proceedings of the 2020 ACM Designing Interactive Systems Conference (DIS '20). Association for Computing Machinery, Eindhoven, Netherlands, 1995–2006. https://doi.org/10. 1145/3357236.3395473
- [30] Brad A Myers, Scott E Hudson, and Randy Pausch. 2000. Past, Present, and Future of User Interface Software Tools. ACM Transactions on Computer-Human Interaction (TOCHI 7, 1 (March 2000), 3–28. https://doi.org/10.1145/344949.344959
- [31] Ken Nakagaki, Artem Dementyev, Sean Follmer, Joseph A Paradiso, and Hiroshi Ishii. 2016. 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.
- [32] 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 and Technology (Charlotte, NC, USA) (UIST '15). Association for Computing Machinery, New York, NY, USA, 333–339. https://doi.org/10.1145/2807442.2807452
- [33] Ken Nakagaki, Luke Vink, Jared Counts, Daniel Windham, Daniel Leithinger, Sean Follmer, and Hiroshi Ishii. 2016. Materiable: Rendering Dynamic Material Properties in Response to Direct Physical Touch with Shape Changing Interfaces. In CHI '16: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems. ACM, New York, New York, USA, 2764–2772. https://doi.org/10.1145/ 2858036.2858104
- [34] Dan R Olsen. 2007. Evaluating User Interface Systems Research. In UIST '07: Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology. New York, New York, USA, 251–258. https://doi.org/10.1145/1294211. 1294256
- [35] Jifei Ou, Mélina Skouras, Nikolaos Vlavianos, Felix Heibeck, Chin-Yi Cheng, Jannik Peters, and Hiroshi Ishii. 2016. AeroMorph - Heat-Sealing Inflatable Shape-Change Materials for Interaction Design. In Proceedings of the 29th Annual Symposium on User Interface Software and Technology (Tokyo, Japan) (UIST '16). Association for Computing Machinery, New York, NY, USA, 121–132. https: //doi.org/10.1145/2984551.2984520
- [36] Amanda Parkes and Hiroshi Ishii. 2010. Bosu: A Physical Programmable Design Tool for Transformability with Soft Mechanics. In Proceedings of the 8th ACM Conference on Designing Interactive Systems (Aarhus, Denmark) (DIS '10). Association for Computing Machinery, New York, NY, USA, 189–198. https: //doi.org/10.1145/1858171.1858205
- [37] D. Pratichizzo, M. Malvezzi, I. Hussain, and G. Salvietti. 2014. The Sixth-Finger: A modular extra-finger to enhance human hand capabilities. In *The 23rd IEEE International Symposium on Robot and Human Interactive Communication*. 993–998.
- [38] Hayes Solos Raffle, Amanda J Parkes, and Hiroshi Ishii. 2004. Topobo: A Constructive Assembly System with Kinetic Memory. In CHI '04: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, New York, New York, USA, 647–654. https://doi.org/10.1145/985692.985774

- [39] Raf Ramakers, Kashyap Todi, and Kris Luyten. 2015. PaperPulse: An Integrated Approach for Embedding Electronics in Paper Designs. In CHI '15: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems. ACM Press, New York, New York, USA, 2457–2466. https://doi.org/10.1145/2702123. 2702487
- [40] Anne Roudaut, Abhijit Karnik, Markus Löchtefeld, and Sriram Subramanian. 2013. Morphees: Toward High "Shape Resolution" in Self-Actuated Flexible Mobile Devices. In CHI '13: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM Press, New York, New York, USA, 593–602. https://doi.org/10.1145/2470654.2470738
- [41] Harpreet Sareen, Udayan Umapathi, Patrick Shin, Yasuaki Kakehi, Jifei Ou, Hiroshi Ishii, and Pattie Maes. 2017. Printflatables: Printing Human-Scale, Functional and Dynamic Inflatable Objects. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (Denver, Colorado, USA) (CHI '17). Association for Computing Machinery, New York, NY, USA, 3669–3680. https://doi.org/10.1145/3025453.3025898
- [42] Valkyrie Savage, Xiaohan Zhang, and Björn Hartmann. 2012. Midas: Fabricating Custom Capacitive Touch Sensors to Prototype Interactive Objects. In UIST '12: Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology. ACM Press, New York, New York, USA, 579. https://doi.org/10.1145/ 2380116.2380189
- [43] Pao Siangliulue, Kenneth C. Arnold, Krzysztof Z. Gajos, and Steven P. Dow. 2015. Toward Collaborative Ideation at Scale: Leveraging Ideas from Others to Generate More Creative and Diverse Ideas. In Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing (Vancouver, BC, Canada) (CSCW '15). Association for Computing Machinery, New York, NY, USA, 937–945. https://doi.org/10.1145/2675133.2675239
- [44] Miriam Sturdee and Jason Alexander. 2018. Analysis and Classification of Shape-Changing Interfaces for Design and Application-Based Research. ACM Computing Surveys (CSUR) 51, 1 (April 2018), 2–32. https://doi.org/10.1145/3143559

- [45] F. Taher, Y. Jansen, J. Woodruff, J. Hardy, K. Hornbæk, and J. Alexander. 2017. Investigating the Use of a Dynamic Physical Bar Chart for Data Exploration and Presentation. *IEEE Transactions on Visualization and Computer Graphics* 23, 1 (2017), 451–460.
- [46] Yasaman Tahouni, Isabel P. S. Qamar, and Stefanie Mueller. 2020. NURBSforms: A Modular Shape-Changing Interface for Prototyping Curved Surfaces. In Proceedings of the Fourteenth International Conference on Tangible, Embedded, and Embodied Interaction (Sydney NSW, Australia) (TEI '20). Association for Computing Machinery, New York, NY, USA, 403–409. https://doi.org/10.1145/3374920.3374927
- [47] John Tiab and Kasper Hornbæk. 2016. Understanding Affordance, System State, and Feedback in Shape-Changing Buttons. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (San Jose, California, USA) (CHI '16). Association for Computing Machinery, New York, NY, USA, 2752–2763. https://doi.org/10.1145/2858036.2858350
- [48] Marynel Vázquez, Eric Brockmeyer, Ruta Desai, Chris Harrison, and Scott E Hudson. 2015. 3D Printing Pneumatic Device Controls with Variable Activation Force Capabilities. In CHI '15: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems. ACM Press, New York, New York, USA, 1295–1304. https://doi.org/10.1145/2702123.2702569
- [49] N Villar, J Scott, S Hodges, K Hammil, and C Miller. 2012. .NET Gadgeteer: A Platform for Custom Devices. In *Pervasive Computing*. Springer Berlin Heidelberg, 216–233. https://doi.org/10.1007/978-3-642-31205-2\_14
- [50] 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 UIST '13 Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology. ACM, New York, New York, USA, 13–22. https://doi.org/10.1145/2501988.2502037
- [51] Hong Kai Yap, Hui Yong Ng, and Chen-Hua Yeow. 2016. High-Force Soft Printable Pneumatics for Soft Robotic Applications. *Soft Robotics* 3, 3 (Sept. 2016), 144–158. https://doi.org/10.1089/soro.2016.0030