CARLOS E. TEJADA

PRINT-AND-PLAY FABRICATION

Thesis submitted to the PhD School of the Faculty of Science
University of Copenhagen

COLOPHON

Author

Carlos E. Tejada
Human-Centered Computing (HCC)
Department of Computer Science (DIKU)
University of Copenhagen

Title of PhD Thesis

Print-and-Play Fabrication

Advisor

Daniel L. Ashbrook
Associate Professor
University of Copenhagen

Assessment Committee

Irina Shklovski, University of Copenhagen (chair)
Eve Hoggan, Aarhus University
Anne Roudaut, University of Bristol

ABSTRACT

In recent years, it has become increasingly accessible to create interactive applications on screen-based devices. Contrary to this ease, and despite their numerous benefits, creating tangible interactive devices is a task reserved for experts, requiring extensive knowledge on electronics, and manual assemblies. While digital fabrication equipment holds promise to alleviate this situation, the majority of research exploring this avenue still present significant barriers for non-experts, and other-domain experts to construct tangible devices, often requiring assembly of electronic circuits and printed parts, prohibitive fabrication pipelines, or intricate calibration of machine learning models.

This thesis introduces *Print-and-Play Fabrication*: a digital fabrication paradigm where tangible interactive devices are printed, rather than assembled. By embedding interior structures inside three-dimensional models that leverage distinct properties of fluid behavior, this thesis presents a variety of techniques to construct tangible devices that can sense, process, and respond to user's interactions without requiring assembly of parts, circuits, or calibration of machine learning models.

Chapter 2 provides an overview of the fabrication of tangible devices literature through the lens of Print-and-Play Fabrication. This chapter highlights the post-print activities required to enable each of the efforts in the literature, and reflects on the status of the field.

Chapters 3 and 4 introduce two novel techniques for constructing tangible devices that can sense user's interactions. AirTouch uses basic principles of fluid behavior to enable the construction of touch-sensing devices, capable of detecting interactions in up to 12 locations, with an accuracy of up to 98%. Blowhole builds on this concept by employing principles of acoustic resonance to construct tangible devices that can detect where they are gently blown on. Blowhole-enabled devices can enable up to seven interactive locations, with an accuracy of up to 98%.

Conversely, in Chapter 6 I introduce a technique to encapsulate logic computation into 3D-printed objects. Inspired by concepts from the Cold War era, I embed structures capable of representing basic logic operations using interacting jets of air into three-dimensional models. AirLogic takes the form of a toolkit, enabling non-expert designers to add a variety of input, logic processing, and output mechanisms to three-dimensional models.

Continuing, Chapter 5 describes a toolkit for fabricating objects capable of changing their physical shape using pneumatic actuation. MorpheesPlug introduces a design environment, a set of pneumatically actuated widgets, and a control module that, in tandem, enable non-experts to construct devices capable of changing their physical shape in order to provide output.

Last, I conclude with reflections on the status of Print-and-Play Fabrication, and possible directions for future work.

DANSK RESUMÉ

I de senere år er det blevet mere og mere tilgængeligt at skabe interaktive applikationer på skærmbaserede apparater. I modsætning til denne tilgængelighed, og til trods for deres mange fordele, er skabelsen af såkaldte "tangible devices" en opgave som kun kan udføres af eksperter, da det kræver omfattende viden om elektronik og manual samling. Selvom digital fabrikation er lovende for at afhjælpe denne situation, indeholder størstedelen af forskningen i dette felt stadigvæk store barrierer for ikke-eksperter og for eksperter fra andre domæner, når det kommer til skabelsen af tangible devices, da det ofte kræver samling af elektroniske kredsløb og printede dele, uoverkommelige fabrikationsprocesser, og indviklet kalibrering af maskinlæringsmodeller.

Denne afhandling introducerer *Print-and-Play Fabrication*: et digitalt fabrikationsparadigme hvor interaktive tangible devices bliver printet i stedet for samlet. Ved at indsætte indre strukturer i tredimensionelle modeller som udnytter egenskaber fra væskedynamik præsenterer afhandlingen en vifte af teknikker til at konstruere tangible devices som kan føle, beregne, og besvare brugeres interaktioner uden brug af dele, kredsløb, eller kalibrering af maskinlæringsmodeller.

Kapitlet 2 giver et overblik over fabrikationen af litteraturen om tangible devices gennem et perspektiv som tager udgangspunkt i *Print-and-Play Fabrication*. Dette kapitel fremhæver de aktiviteter, man skal udføre efter at have printet, som er påkrævet for at opnå, og reflekterer på feltets status.

Kapitlerne 3 og 4 introducerer to nye teknikker til at konstruere tangible devices som kan føle brugerens interaktioner. AirTouch udnytter grundlæggende væskedynamiksprincipper for at skabe apparater som kan måle berøring ved op til 12 lokationer med en nøjagtighed på op til 98%. Blowhole bygger på dette koncept ved at bruge principper fra akustisk resonans til at konstruere tangible devices som kan måle når brugeren puster mildt på dem. Apparater som anvender Blowhole muliggør op til syv interaktive lokationer, med en nøjagtighed på op til 98%.

Omvendt introducerer jeg i kapitel 6, en teknik som indkapsler logikberegning I 3D-printede objekter. Inspireret af koncepter fra koldkrigstiden indsætter jeg strukturer som kan repræsentere grundlæggende logikoperationer ved hjælp af luftstråler i tredimensionelle modeller. AirLogic tager form af et toolkit, der tillader designere som ikke er eksperter at tilføje en vifte af inputmekanismer, logikoperationer, og outputmekanismer til tredimensionelle modeller.

Videre beskriver kapitel 5 et toolkit for fabrikerede objekter som kan ændre sin fysiske form gennem pneumatisk aktivering. MorpheesPlug introducerer et designmiljø, et sæt af pneumatisk aktiverede widgets, og et kontrolmodul som samlet tillader ikke-eksperter at konstruere apparater som kan ændre sin fysiske form for at give output.

Slutteligt konkluderer jeg med refleksioner om status på Print-and-Play Fabrication, og mulige retninger for det fremtidige arbejde.

Para Mamami.

Ahora que el tiempo ha pasado,
doy gracias por haber llegado hasta aquí.
— *Enrique Bunbury,* Ahora.

ACKNOWLEDGMENTS

This thesis would not exist without the help, patience, and kindness of many people during the lengthy PhD process. I extend my deepest gratitude to all of them.

Primero, y más importante, debo agradecer a mi familia. A mis padres (Reyna y Juan José Tejada), y hermanita (Katherine Tejada). Gracias por todo el apoyo, amor, y paciencia durante este largo y arduo proceso. No tengo palabras para agradecerles por todo lo que hacen. Al resto de mi querida familia Tejada–Modesto en la República Dominicana, gracias por siempre estar atentos a mi bienestar. Finalmente a mi segunda familia, los Alonzo, gracias por todo el apoyo, consejos, risas, y comidas.

This thesis would not have been completed without the advice, guidance, and patience from my supervisor Dan Ashbrook. Thank you for taking a chance with an grad student who clearly had no idea what he was doing.

Continuing, I owe a huge debt of gratitude to all my coauthors: Jess McIntosh, Sebastian Boring, Raf Ramakers, Mengyu Zhong, Hyunyoung Kim, Osamu Fujimoto, Zhiyuan Li, Aluna Everitt, and Azier Marzo. Thank you for your hard work, and thoughtful comments. Naturally, this thesis would not exist without them.

Last, but certainly not least, I am indebted to my wonderful colleagues at the Human-Centered Computing Section from the Department of Computer Science in the University of Copenhagen. Kasper Hornbæk, Pernille Bjørn, Irina Shklovski, Joanna Bergström, Henning Pohl, Hyunyoung Kim, Jess McIntosh, Jonas Schjerlund, Andreea Muresan, Tor-Salve Dalsgaard, Klemen Lilija, Xiaoyi Wang, and Aske Mottelson. Thank you for your the thoughtful scientific (and also for the non-scientific) discussions, laughs, beers, and for making the conference crunching time a lot more bearable.

Carlos E. Tejada
Copenhagen, 2021

PREFACE

This thesis represents the culmination of four and a half years of independent research carried out in two universities. The main purpose of this thesis is to demonstrate that I was able to carry out international-level research during this period.

STRUCTURE OF THESIS

The main contribution of this thesis is the concept of Print-and-Play Fabrication, which I present the three parts that make up this document. Part I aims to prepare the reader by introducing the concept of Print-and-Play Fabrication, and providing background on the current position of the state-of-the-art for the fabrication of tangible, interactive devices. In Part II, I present four novel techniques for fabricating tangible, interactive devices using consumer-grade 3D printers. Last, Part III offers high-level reflections about the projects that comprise this document, and outlines future steps for this avenue of research.

PAPER SELECTION

The core content of this thesis comes from four papers I have contributed to, of which three are published [55, 121, 123], and one is in manuscript [122]. Each of these papers introduces a novel technique for construction of tangible interactive devices requiring minimal post-fabrication activities, while focusing on a specific aspect of interactivity (sensing interactions, processing inputs, and output display). These papers all share the Print-and-Play Fabrication ideal that tangible devices should be fabricated instead of assembled. Below, find abstracts for each of the papers that comprise this document.

Paper Abstracts

AirTouch: 3D-printed Touch-sensitive Objects Using Pneumatic Sensing

3D printing technology can be used to rapidly prototype the look and feel of 3D objects. However, the objects produced are passive. There has been increasing interest in making these objects interactive, yet they often require assembling components or complex calibration. In this paper, we contribute AirTouch, a technique that enables designers to fabricate touch-sensitive objects with minimal assembly and calibration using pneumatic sensing. AirTouch-enabled objects are 3D printed as a single structure using a consumer-level 3D printer. AirTouch uses pre-trained machine learning models to identify interactions with fabricated objects, meaning that there is no calibration required once the object

has completed printing. We evaluate our technique using fabricated objects with various geometries and touch sensitive locations, obtaining accuracies of at least 90% with 12 interactive locations.

Blowhole: Blowing-activated Tags for Interactive 3D-printed Models

Interactive 3D models have the potential to enhance accessibility and education, but can be complex and time-consuming to produce. We present Blowhole, a technique for embedding blowing-activated tags into 3D-printed models to add interactivity. Requiring no special printing techniques, components, or assembly and working on consumer-level 3D printers, Blowhole adds acoustically resonant cavities to the interior of a model with unobtrusive openings at the surface of the object. A gentle blow into a hole produces a unique sound that identifies the hole, allowing a computer to provide associated content. We describe the theory behind Blowhole, characterize the performance of different cavity parameters, and describe our implementation, including easy-to-use software to automatically embed blowholes into preexisting models. We illustrate Blowhole's potential with multiple working examples.

AirLogic: A Toolkit for 3D-printing Stand-Alone, Interactive Objects

The promise of on-demand fabrication of custom, interactive devices is closer to reality thanks to recent developments in 3D-printing of interactive devices. While recent work has presented novel ways to 3D-print artifacts such as speakers, electromagnetic actuators, and hydraulic robots; these efforts are non-trivial to instantiate, requiring assembly of circuits or mechanical parts. The present work introduces AirLogic: a toolkit for the creation of stand-alone, interactive objects using pneumatic widgets. Objects constructed using AirLogic, require no electronic circuits, and little to no assembly of physical components. AirLogic is comprised of a set of 12 pneumatic widgets, and a design environment, which designers can use to embed input, logic processing, and output capabilities to existing 3D models. We present an evaluation of the performance of our widgets, and a four applications that illustrate AirLogic's potential.

MorpheesPlug: A Toolkit for Prototyping Shape-Changing Interfaces

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 out of the eleven shape-change features. 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 3D-printable pneumatic SCIs; and (3) a hardware platform for

controlling the widgets. To evaluate MorpheesPlug we carried out ten open-ended 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.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Part I

INTRODUCTION & BACKGROUND

1 INTRODUCTION

1.1 THE INEQUALITIES OF INTERACTIVITY

We are surrounded by products with dedicated physical interfaces, like steering wheels, musical instruments, and game controllers. While the advent of screen-based devices has led to a rise in touch, and mouse based applications, there are some benefits of physicality that simply do not translate to screens [58]. It is because of these benefits (e.g., hands-free interaction, speed, virtuosity, control) that most experts use dedicated physical interfaces for their crafts, rather than their screen counterparts.

Because the increased presence of screen-based devices, significant advances has been made to streamline the creation of screen-based applications. In 60 years, the state-of-the-art has gone from programming computers using punched cards, to creating immersive virtual environments by dragging virtual elements and dropping them in their respective places [1]. Currently, you can create an online store without knowing anything about web programming [2], or creating body-based augmented reality applications using visual programming [88].

Contrary to this ease of crafting interactive digital experiences, constructing interactive physical objects remains a challenging undertaking. While the construction, and usage of interactive digital interactive experiences has been researched for over forty years [124, 125], no long-term research has explored ways to construct interactive experiences in the physical world. Because of this, constructing physical interactive experiences still require significant domain knowledge, and remains out of reach for other-domain experts. Although existing research has explored electronic toolkits to lower the threshold of making physical, interactive devices [6, 33], these require the assembly of electronic sensors, actuators and physical parts to construct an interactive device.

While the process of assembling circuits and printed parts may seem straightforward to some, this is an intricate process, and remains out of the reach of other domain experts (e.g., doctors, architects). This stands in contrast to the easiness of creating interactive applications for screen-based devices, where little to no domain expertise is needed.

1.2 THE PERSONAL FABRICATION REVOLUTION

A promising remedy that could tip the making scales to balance is personal fabrication. While the technologies that enable personal fabrication devices are almost fifty years old, these advances are now accessible

1 Unity
2 Squarespace

and reliable enough to be used by consumers and hobbyists. In their respective books Gershenfeld and Anderson describe a future where we all have a fabrication machine in our homes [5, 29]. Homologous to our current use of 2D paper printers, the authors describe how in this potential future we would print our objects, rather than buy them, dawning an age of on-demand fabrication.

1.3 BROKEN PROMISES OF THE FABRICATION REVOLUTION

While fabrication machines have become accessible, and reliable enough to make the jump from industry and research laboratories to the hands of consumers and hobbyists, we are still far away from the fabrication revolution Gershenfeld and Anderson professed. Despite recent market analyses placing 3D printer market penetration at almost 35%, 77% are used in the industry, rather than by individuals [1]. This same analysis reveals that the main use for 3D printers is prototyping future products, rather than creation of usable things.

I believe that the lackluster adoption of personal fabrication machines is due to two main factors. First, current fabrication equipment is limited to what types of objects it can produce. Current fabrication devices can construct a variety of shapes using different types of plastics and metals, which, while interesting and useful, is far from the replicators shown in science fiction, capable of creating everything from food to medicaments, or the machines capable of building other machines presented by Gershenfeld [29].

(a) 3D-printed Jewelry (b) 3D-printed Prohstetics

Figure 1.1: Although current digital fabrication devices can be used to create a variety of objects, like jewelry (Figure 1.1a) or prosthetics (Figure 1.1b) designers require domain expertise to create this objects.

The second issue relates to the expertise required to construct usable objects. While current personal fabrication equipment is able to produce a variety of interesting objects from prosthetic arms for children (Figure 1.1b) to intricate jewelry pieces (Figure 1.1a), the construction of these objects require ample domain knowledge that consumers might not possess. This leaves very limited options for low-entry uses for fabrication devices.

1.4 TOWARDS ON-DEMAND FABRICATION OF INTERACTIVE DEVICES

Despite the utility of custom-made three-dimensional shapes, recent work has highlighted that "everyday makers" are most interested in fabricating objects they can interact with [111]. Inspired by these findings, and the personal fabrication ideal described in Section 1.2, this thesis explores a potential future where digital fabrication equipment not only enables the construction of custom three-dimensional shapes, but of tangible devices as well.

The creation of these tangible devices should be as natural and automatic as possible. Similarly to today's document creation pipeline, where once the document is constructed in a word processing software, and sent to the desktop printer there are no further actions to carried out, so should be the creation process of tangible devices. Designers should be able to model their desired device using a CAD software suite, send this design to a personal fabrication machine, and once printed, no post-print activities should be necessary to materialize the device.

1.4.1 *Challenges*

The concept of constructing interactive devices using digital fabrication equipment is not new. A multitude of research endeavors in the Human-Computer Interaction (HCI) community have explored techniques to fabricate interactive devices using 3D-printers or other digital fabrication equipment [8]. However, despite these advances, the introduced techniques have yet to escape the research laboratories and move on into the masses. This is due to two main factors:

Convoluted Fabrication Pipelines

A paramount research challenge researchers need to resolve is the democratization of techniques. Revisiting the paper document metaphor, when the same document is printed in different types of 2D printers, the outcome is the same. So it should be when constructing interactive devices: the method of construction should not affect the capability of the device. Additionally, we should strive to automate the construction process as much as possible, opting to use accessible fabrication machines and devices over manual labor. We should also avoid using non-accessible materials.

Complex Post-Print Activities

An additional challenge is the removal of complex post-print activities. Current fabrication pipelines for the construction of interactive devices rely on convoluted post-print activities in order to fabricate these devices. These post-print activities can take the shape of assembly of circuits, printed parts, complex removal of support material, or calibration of per-object, or per-user machine learning models. This means that in order

for non-experts to construct tangible devices they must be well-versed in electronics, programming, and mechanics—limiting the adoption of the proposed techniques.

1.4.2 *Print-and-Play Fabrication*

As a way towards the potential future of on-demand fabrication of interactive devices described in Section 1.4, this thesis introduces Print-and-Play Fabrication: a digital fabrication paradigm where interactive objects are printed instead of assembled. Print-and-Play fabrication techniques construct devices that are immediately usable after fabrication, meaning there is no assembly of parts, circuits, or calibration of machine learning models needed. Additionally, Print-and-Play techniques, are constructable using off-the-shelf fabrication equipment and materials.

The concept of Print-and-Play Fabrication directly addresses the shortcomings of the personal fabrication revolution described in Section 1.3. First in increases the versatility of off-the-shelf fabrication materials and printers. All Print-and-Play Fabrication should be able to be instantiated using consumer-grade fabrication equipment and materials. Second, Print-and-Play techniques should not require any post-print activities or domain-specific knowledge (e.g., programming, electronics) to be enabled. Additionally, they must provide a design environment to streamline the modeling of tangible devices by non-expert designers. These two factors increase the target audience of these techniques.

Contributions of this Thesis

This thesis focuses on the easy construction of tangible devices without requiring significant post-fabrication activities (e.g., assembly of parts, circuits, calibration of machine learning models), or specialized fabrication pipelines. The work encompassed in this document presents techniques for constructing tangible devices that can sense, process, and provide output to user's interactions. I achieve this by embedding custom geometries in the interior of three-dimensional models that leverage well-studied physical phenomena such as fluid dynamics, or acoustic resonance.

An overarching theme of this work is *air*. All of the work presented in this thesis is powered by air, being from a constant, pressurized air source (Chapters 3, 5, and 6), or from the user's lungs (Chapter 4). The use of air as a medium to enable the construction of interactive devices was motivated by two factors. First, air and fluid behaviors have been extensively explored in physics, which gives us a ample understanding how air behaves in a variety of situations, and which ones of these lend themselves to digital fabrication pipelines. Second, thanks to the broad understanding of fluid behavior, we can employ these concepts to construct pre-trained machine learning models, or use their respective mathematical equations, to identify user's interactions.

The papers that make up this thesis explicitly address the challenges described above. First, the interactive devices constructed using the

proposed techniques are printed in commodity 3D printers as a single structure, requiring no post-print assemblies of parts. Second, because they embed custom interior structures in the interior of the augmented designs, and leverage well-studied physical phenomena to enable interactions, they require no electronics to sense, process, or provide output to user's interactions. Third, as an additional benefit of leveraging well-studied physical phenomena to enable interactions, these devices employ mathematical models, or pre-trained machine learning models, requiring no per-object, or per-user calibration. A summary of the individual contributions of the papers that comprise this document can be found below:

AirTouch extends the possibilities of pneumatic sensing on fabricated objects by enabling sensing users interactions not only on soft objects, but in rigid ones as well. *AirTouch* is based on basic principles of fluid dynamics that relate pressure to discharge area: when there is a change in the discharge area of a system, the pressure will vary in return. Objects augmented with *AirTouch* can sense touch interactions in up to 12 individual locations, and are printed as a single structure. These objects are fabricated as a single structure without requiring any assembly of parts or circuits. Additionally, they make use of pre-trained machine learning models, so no per-object or per-user calibration is necessary.

Blowhole continues to explore techniques for enabling sensing of user's interactions on 3D-printed objects by addressing one of *AirTouch's* principals shortcomings: the need for a constant air source and barometric pressure sensors. Based on the principle of acoustic resonance, *Blowhole* embeds resonant cavities in the interior of the model that, when gently blown on, emit an identifiable sound. Interactive devices constructed using *Blowhole* are fabricated as a single structure using commercially-available printers and materials, and make use of a mathematical model to identify blow locations based on the frequency of the generated sound.

Switching gears to computing, AirLogic aims to answer the question "how can we 3D-print devices that can think?" Inspired by research from the Cold War era, I developed a series of widgets that can not only sense user's interactions, and provide output to them, but also compute simple logic operations (e.g., `AND, OR, NOT`, and `XOR`).

Last, with MorpheesPlug I explore the construction of devices that can provide physical output. This work was inspired by recent research on shape-changing interfaces, where physical interfaces change their shape depending on their intended use. We approached this concept from a fabrication standpoint, creating a set of pneumatically-actuated widgets that can represent a variety of changes in shape. These widgets are fabricated as a single structure, using off-the-shelf printers and materials.

2 FABRICATION OF INTERACTIVE OBJECTS

The construction of interactive objects using digital fabrication equipment has been a popular topic in the Human-Computer Interaction (HCI) literature in recent years. Researchers have proposed a variety of techniques for constructing objects that can sense, provide output to, and process user's interactions. Ballagas et al. present a comprehensive review of this design space [8], grouping previous endeavors by the type of mechanisms used to 3D-print interactive objects. This chapter, in contrast, aims to identify how this literature is progressing towards the Print-and-Play Fabrication ideal.

2.1 PRINT-AND-PLAY FABRICATION LITERATURE REVIEW

The process of interacting with a device, tangible or screen based, is comprised of three main steps. First, the device senses how it is being interacted with (e.g., mouse clicks, capacitive touch screen sensor). Next, the device processes the provided interaction input. Last, the device provides an output to the user (e.g., visualizing on screen, change of physical shape). Mirroring this process, the structure of this section is divided into three main parts. First, I explore the literature relevant for constructing tangible devices that can sense user's interactions. Second, I review previous work that has tackled embedding processing capabilities to tangible devices. Last, I highlight endeavors that construct devices for output display.

2.1.1 *Sensing User's Interactions*

A variety of efforts have explored how to construct devices that can sense user's interactions using 3D-printers. Some early research on prototyping interactive objects focused on adding interactive functionality to the objects rather than on simple methods for fabrication. For example, some systems require assembling electronics and other components inside a printed shell [45, 70, 99, 100] and others require casting silicone [42, 96].

Other approaches require less assembly. Some research has detected changes in acoustical signals caused by user manipulation of geometry [61, 65, 102]; however, the requirement for complex or movable geometry can mean considerable post-print effort for cleaning, assembling, and gluing.

Some recent work has come much closer to the Print-and-Play ideal, enabling interactivity with significantly less or no post-print manipulation. One approach is to use multi-material printers to enable capacitive touch sensing [37, 107, 109] or optical sensing [133]; however, these

approaches require attachment of multiple points of circuitry or optic sensors to operate, and the size of object is limited. Another optical approach is to use computer vision to detect user interaction [112]; however, cameras are prone to problems with occlusion (i.e., touches on the back of an object cannot be detected, nor can touches hidden by the hand itself), and it is difficult to differentiate touching from merely being close to the object.

Several projects require nearly no post-print manipulation. Touch & Activate [82] used an affixed microphone and speaker to detect how acoustic sweeps were changed by user touch; this technique worked with many objects, including off-the-shelf ones, but required a new machine-learning model to be trained for every object. Tickers and Talker [113] used centimeter-scale physical markers which made unique sounds when plucked, but significantly impact the geometry of the object. INTACT [47] uses a 3D model of an object placed on a 6-axis force sensor to mathematically determine where the user is touching. While it can sense touch with high precision, objects are limited in size to around 20 cm, and require recalibration after moving.

2.1.2 *Computing*

Previous approaches for digitally fabricating interactive objects have often used existing computing devices (e.g., computers, phones) connected to the fabricated objects in order to drive the interactions. Researchers have explored the use of acoustic [102, 113, 121], pneumatic [84, 123, 131], electronic [103, 107, 109], and optic [99, 133] techniques to fabricate objects that can respond to user's interactions. While the computation required to drive most of these approaches could theoretically be implemented using smaller computing devices and embedded inside the final 3D-printed object, this would require significant engineering expertise from the designer to assemble circuits or printed parts.

Other approaches "store" the results of the interactions for later processing. For example, in Off-line Sensing, Schmitz et al. introduce an approach to develop one-time 3D-printed sensors using liquids to memorize the results of interactions. These sensors can then be read using a smartphone [106]. Similarly, Iyer et al. developed 3D-printable structures capable of storing linear, and rotational interactions using a coil-like structure. These devices can transmit the stored interactions wirelessly once in range [50]. While successful, these efforts require manual intervention during deployment. They require designers to assemble multiple 3D-printed parts [50], or carefully pour liquid into the constructed objects [106].

Last, are the endeavors that embed computation inside the constructed object. While this has been traditionally achieved by assembling custom electronics and other components inside a 3D-printed shell [70], other efforts aim to simplify this process by leveraging existing computing devices and embed them inside the fabricated object. Acoustruments, for example, makes use of a smartphone embedded inside the object

to enable rich input modalities [61]. Pineal builds on this concept and makes use of smartwatches in addition to smartphones in order to augment objects of various dimensions while also providing rich output to the user [62]. Other endeavors make use of mechanical computing to augment digitally fabricated objects with simple logic processing. Ion et al. developed logic cells that can be embedded inside 3D-printed objects, enabling them to compute simple logic operations [48], while Song et al. devised a technique for manufacturing micro-mechanical logic gates using digital fabrication equipment [116]. Despite their success, these approaches share a common limitation: they rely on complex or lengthy assembly processes in order to be realized.

2.1.3 *Providing Output*

A large body of work in the HCI and material science communities have explored techniques for creating objects, and materials that can provide output of computations either acoustic, visual, or haptic. This segment of devices stray the most from being Print-and-Play. Some efforts require complex circuitry to be enabled [35, 115], assembly of parts [48], prohivitive fabrication pipelines [12, 59, 86], or a combination of these [66, 77, 131]. An effort that approaches to being Print-and-Play is Acoustic Voxels [65]. In it, the authors propose a set of acoustic filters that modify incoming sounds depending on their configuration. The objects augmented using these technique can be printed as a single structure, and do not require electronics or calibration of machine learning models.

2.2 DISCUSSIONS & OBSERVATIONS

Despite the strides made by previous explorations on streamlining the construction of tangible devices, the literature nonetheless strays from the Print-and-Play Fabrication ideal. This section discusses the advances made by previous work through the lens of Print-and-Play Fabrication.

Section 2.1.1 shows how previous work has progressed from requiring complex assemblies and fabrication pipelines to enable sensing user's interactions on fabricated objects, to almost seamless approaches that require very little intervention post-fabrication. However, despite these progresses, there most streamlined processes for constructing objects that can sense user's interactions still present significant friction. For example, efforts like Touch & Activate [82], and Tickers and Talkers [113], while not requiring any complex assemblies or prohivitive fabrication pipelines, they require per-object calibration of machine learning models–a task arguably more complex for non-expert users.

Contrary to techniques that enable sensing of interactions on tangible devices, efforts that enable computation in 3D-printed objects diverge significantly from the Print-and-Play Fabrication ideal. Efforts that aim to mechanically embed computation into tangible devices, such as [48], require lengthy assemblies of numerous printed parts, while

endeavors that "store" the results of interactions require the careful pouring of liquids into constructed objects [106], both tasks demanding significant domain knowledge from the designer. The technique that approaches closer to the Print-and-Play Fabrication ideal is Pineal [62]. While requiring manual assembly post-print, this technique provides designers guidelines to designers how to assemble the printed parts and mobile devices—an interesting way of decreasing the complexity for constructing tangible devices.

Last are the techniques that construct tangible devices capable of displaying the output of computations to their users. When compared to sensing approaches, techniques that construct tangible devices for displaying the output of computations to their users remains unexplored. In their review [8], Ballagas et al. identified only 18 such efforts, compared to the 79 that enable the construction of tangible devices to sense user's interactions. The techniques explored, however, stray the farthest from the Print-and-Play Fabrication ideal, requiring lengthy assembly of parts [48, 133], and electronics [70].

In closing, the literature exploring the construction of tangible devices using 3D printers trends towards the Print-and-Play Fabrication deal, forgoing elaborate assemblies for a more accessible construction process. This accentuates the importance of Print-and-Play Fabrication. Despite the literature trending towards more Print-and-Play-friendly techniques, the literature does does not yet provide a technique that fulfills the requirements for Print-and-Play Fabrication, previously discussed. This thesis introduces four distinct approaches for constructing immediately usable tangible devices. These techniques aim to inch closer to the ideal future of on-demand fabrication of tangible devices.

Part II

PRINT-AND-PLAY FABRICATION

3

AIRTOUCH

This chapter is very similar to the paper I published with the same name [123]. In this chapter, I introduce AirTouch, a technique for fabricating tangible devices that respond to user's touch.

In this chapter I introduce a pneumatic sensing technique for fabricating tangible devices that are able to sense user's touch using 3D printers. Before AirTouch, techniques using pneumatic sensing were present predominantly to enable sensing in soft objects: the barometric pressure inside an object would vary as it is deformed by pressing or squeezing. AirTouch is the first technique to enable sensing of user's interactions on *hard* objects, using principles of fluid behavior.

In this chapter, I introduce the operating principle of AirTouch, present the chosen internal structure design, and justify the design decisions made regarding this interior structure. I detail the evaluation procedure, and the obtained accuracies, for this approach, and close with a series of inspiring applications, and discussions on the future of pneumatic sensing techniques.

The most challenging aspect of developing AirTouch was to produce a technique that would not require per-object calibrations post-print. However, using strategically placed tube reductions, and standardized cavity sizes, enabled AirTouch objects of the same outlet configuration reuse the same machine learning model to identify interactions, regardless of their exterior geometry.

This chapter is based on the collaborative effort described below.

Title
AirTouch: 3D-printed Touch-Sensitive Objects Using Pneumatic Sensing

Authors
Carlos E. Tejada, Raf Ramakers, Sebastian Boring, Daniel Ashbrook

DOI
https://doi.org/10.1145/3313831.3376136

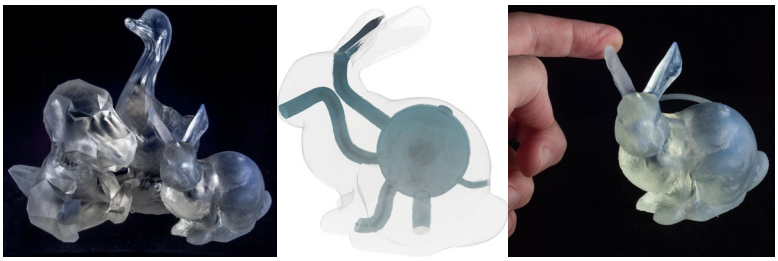
Venue
Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems

What was the role of the PhD student in designing the study?
The PhD student was the first author of the paper, and responsible of the described studies.

How did the PhD student participate in data collection and/or development of theory?
The PhD student was responisble for study implementation, execution, data collection, and theory development.

Which part of the manuscript did the PhD student write or contribute to?
The PhD student contributed to all parts of the manuscript.

Did the PhD student read and comment on the final manuscript?
Yes.



(a) (b) (c)

Figure 3.1: AirTouch augments 3D-printed objects to enable touch-sensitivity. It works by detecting the pressure change resulting from users blocking tiny air outlets fabricated into the objects. Figure 3.1a Three active animal objects: each has touch points on its ear, nose, foot, and back. The same machine learning model works interchangeably with all three. Figure 3.1b Structure inside the bunny, illustrated via computer rendering. The external solid object is shown as clear, while the internal hollow tubes are rendered in translucent blue. Figure 3.1c The AirTouch-enabled bunny with interactive locations on the ear, nose, back and feet. When the user touches any of these locations, the respective label is displayed.

Abstract

3D printing technology can be used to rapidly prototype the look and feel of 3D objects. However, the objects produced are passive. There has been increasing interest in making these objects *interactive*, yet they often require assembling components or complex calibration. In this paper, we contribute *AirTouch*, a technique that enables designers to fabricate touch-sensitive objects with minimal assembly and calibration using pneumatic sensing. AirTouch-enabled objects are 3D printed as a single structure using a consumer-level 3D printer. AirTouch uses pre-trained machine learning models to identify interactions with fabricated objects, meaning that there is no calibration required once the object has completed printing. We evaluate our technique using fabricated objects with various geometries and touch sensitive locations, obtaining accuracies of at least 90% with 12 interactive locations.

3.1 INTRODUCTION

Over the past decade, additive manufacturing has moved from industry to desktop-sized 3D printers that empower makers to produce intricate three-dimensional shapes. In contrast to this new ease of producing forms, making interactive and responsive objects usually requires inserting electronic circuitry [92, 101] and thus requires engineering expertise and assembly effort. Similarly to Willis et al. [133], we envision what we call *Print-and-Play*: a future where interactive objects are ready to use immediately after fabrication, without extra effort on the maker's part. Such a capability will empower makers, designers, researchers, and educators to instantly turn passive fabricated forms into interactive

artifacts. While recent research has made some progress towards this goal [82, 109, 113, 121], these projects still fall short of being purely Print-and-Play: they require per-user [113] or per-object [82, 121] training, are susceptible to environmental interference [113, 121], or significantly alter the object's external form [113].

In this paper, we present *AirTouch*, a novel technique for fabricating touch-sensitive objects that are instantly responsive after 3D printing without the need for assembly or calibration. Our technique minimally modifies the form of the object, adding only a tiny hole to each touch location and inlets for an external source of air and a barometric pressure sensor; when a user touches a hole, the air pressure inside the object changes in a predictable way. AirTouch-enabled objects can be fabricated as a single structure on consumer-level resin-based 3D-printers, without the need for any post-hoc assembly, and can enable up to 12 different interactive locations on fabricated objects.

To summarize, this work contributes:

- *AirTouch*, a novel technique for fabricating objects that are instantly touch-sensitive after printing by measuring differences in air pressure inside the object;
- a characterization of the number of interactive locations that can be enabled with our technique, and its performance; and
- a set of applications that demonstrate AirTouch's potential.

3.2 RELATED WORK

AirTouch builds on prior research on prototyping and fabricating interactive objects, and research around pneumatic interfaces. Earlier in this thesis I have presented a comprehensive review of the literature regarding the fabrication of interactive objects (see Chapter 2). This section thus presents an overview of previous work on pneumatic interfaces that inspired AirTouch.

3.2.1 *Pneumatic Input for Interaction*

AirTouch's use of air pressure as its sensing mechanism is inspired by other pneumatic interaction work. A number of projects have used pneumatic techniques to augment the digital fabrication process, aiming to reduce fabrication time by inflating objects to their final form [84, 98, 134], or to add movement to otherwise static objects [78, 79, 135].

Various researchers have investigated the use of pressure as an input mechanism. Most use sensing of pressure inside flexible enclosures, including buttons [31, 39, 131], computer mice [57], robots [115], and balloons [75]. Due to measuring the pressure of air trapped inside a flexible structure, these systems are limited to sensing one interaction location, and can suffer from ambiguity if the user presses with different levels of force. In contrast, AirTouch's use of rigid object enables

sensing up to 12 different interaction points, and it can operate reliably, independent of the user's pressing force on those points.

3.3 AIRTOUCH OVERVIEW

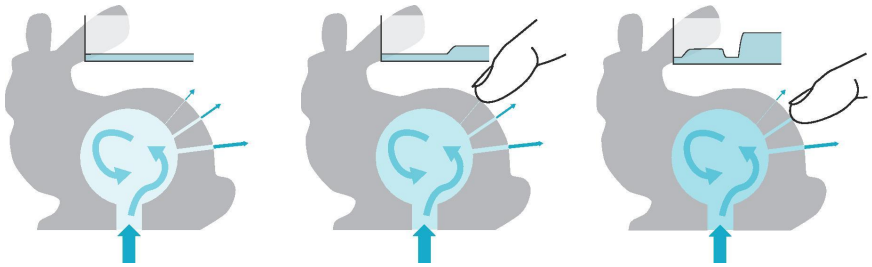


Figure 3.2: Representation of AirTouch's working principle. When an outlet is covered, the barometric pressure inside the fabricated object rises to an identifiable level. As each outlet is unique in size, covering different outlets yields a different barometric pressure response.

AirTouch makes use of some of the basic principles of fluid behavior. In particular, we use the *principle of continuity* [95]—which indicates that the total flow of air exiting an object must equal the flow of air entering the object—and Bernoulli's principle [11], which relates flow and pressure. In combination, these two principles predict that when there is a change in the size of an opening through which fluid is passing, the pressure will vary in response. Our objects are therefore comprised of a series of pressurized tubes with uniquely sized outlets. Covering an outlet changes the amount of area through which air can escape, and thereby changes the pressure in the object. In the following, we sketch some of the mathematical theory of operation that enables AirTouch to function.

3.3.1 *Theory of Operation*

The behavior of AirTouch can be approximated with principles from fluid dynamics. By the continuity principle—that the total output from a system must be equal to its input—we know that Q_I, the flow of air entering the object, is equal to $\sum Q_i$, the total flow coming out of all openings. The relationship of flow to the cross-sectional area A of an opening is given by Bernoulli's principle [95], allowing flow to be expressed as

$$Q = CA\sqrt{\Delta P} \tag{3.1}$$

where C is a constant incorporating an adjustment for the shape of the orifice, the density of air, and other unknowns, and ΔP is the difference in pressure before and after the opening. We can now use Equation 3.1 to express and simplify the continuity equation in terms

of the sum of the areas of each outlet i and the difference in pressure between the inside of the object and the atmosphere:

$$\begin{aligned} Q_I &= C \sum Q_i \\ &= C \sum A_i \sqrt{\Delta P} \end{aligned} \tag{3.2}$$

Now consider the situation where we cover one outlet. Because of continuity, the input flow (held constant by the air compressor and regulator valve) and output flow are identical to when no outlets are blocked. Therefore, the pressure inside the object must increase to compensate for the decreased total discharge area. Assuming we have blocked outlet x, we now have

$$C \sum A_i \sqrt{\Delta P} = C \left(\sum A_i - A_x \right) \sqrt{\Delta P_x} \tag{3.3}$$

illustrating that with a smaller total area we must have a new, larger pressure ΔP_x. Solving Equation 3.3 for ΔP_x allows us to predict the new pressure that will result from covering an outlet of cross-sectional area A_x:

$$\Delta P_x = \frac{(\sum A_i)^2 \Delta P}{(\sum A_i - A_x)^2} \tag{3.4}$$

or, equivalently, given a pressure change of ΔP_x, the outlet area which resulted in that pressure change:

$$A_x = \sum A_i \left(1 - \sqrt{\frac{\Delta P}{\Delta P_x}} \right) \tag{3.5}$$

As presented here, these fluid dynamics equations work with incompressible, steady flows—that is, liquids flowing in steady state—and perfectly shaped outlets of known geometry. Our system does not hold to these constraints: 3D-printed objects, and thereby our outlets, are not perfect; air is compressible; and our objects are subject to internal turbulence due to their complex geometry (see Figure 3.1b). Due to these factors, the above equations do not perfectly match our observed data, but provide guidelines for understanding and predicting the general behavior of the system.

3.3.2 *Internal Structure*

AirTouch adds an internal structure to 3D models that distribute incoming flow from an air compressor to outlets on the object's surface (Figure 3.1b). This internal structure consists of several components: a central flow-distribution chamber which supplies all tubes with air; an inlet via which the air source provides pressurized air flow to the flow-distribution

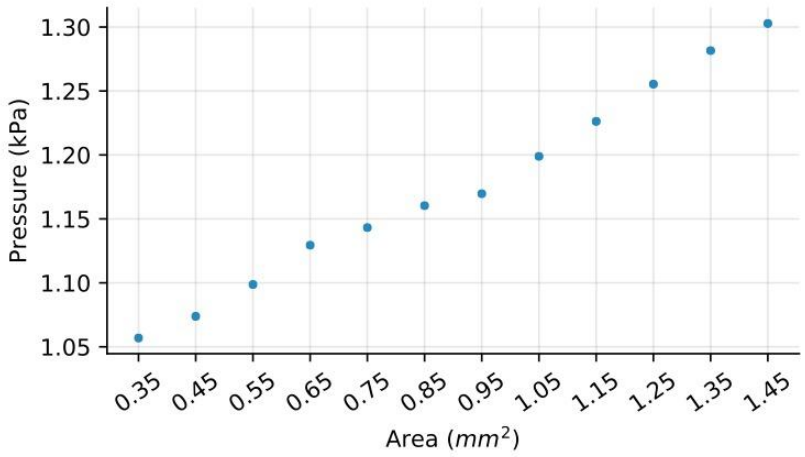


Figure 3.3: Pressure in kilopascals (kPa) for single touches on twelve differently sized outlets (ranging in cross-sectional area from .35 mm^2 to 1.45 mm^2 in 0.1 mm^2 increments).

chamber; a connection point for the pressure sensor; a series of tubes that connect the flow-distribution chamber to touch locations on the object's surface. By using this structure in all AirTouch-augmented objects, we ensure that the pressure increases when touching the same outlet are comparable, regardless to the outer geometry of the augmented object.

3.3.3 *Sensing User Interactions*

The basic user interaction with an AirTouch-enabled object is via touch. When a user touches one of the outlets on the object's surface, the airflow through that outlet is blocked. As each outlet has a different size, the airflow through each outlet is unique and is proportional to the outlet area (Equation 3.2). Blocking the flow from an outlet causes a identifiable rise in the barometric pressure inside the object. Our system records these changes in air pressure using a barometric sensor, and translates them to an outlet ID, and subsequent position on the object's surface. Figure 3.2 shows an abstract representation of the unique barometric increases sensed when the user covers different outlets, and Figure 3.3 shows actual sensed touch data.

3.4 PARAMETER EXPLORATION

Every AirTouch object contains a flow-distribution chamber, tubes, and outlets. In this section, we report on the design and fabrication of this internal structure based on literature and additional testing. Our design decisions were guided by the following three requirements: (1) the object, including the internal structure, must be fabricatable with a consumer-level 3D printers; (2) the outside geometry of the object cannot be modified; (3) per-object calibration is not allowed.

3.4.1 *Experimental Setup*

We constructed a test setup consisting of a JunAir 2000-40PD air compressor, Festo MS4-LR-1/4-D5-AS valve, and an analog Panasonic PS-A (ADP5) barometric sensor. The sensor is connected to an Arduino Uno, which samples the air pressure at 5 kHz. We connect standard air compressor tubing (polyethylene, 6 mm diameter) to the valve, and the barometric sensor to the object. This sensor is responsible for detecting the subtle pressure changes in the system caused by user interactions, and can sense pressures relative to atmospheric pressure from 0 to 6 kiloPascals (kPa). The sensor's limited operating range requires that we empirically set the valve's value for objects having a different number of outlets to avoid saturation. To reduce the noise from the sensor readings, we filter the incoming signal using the 1€ Filter [17] using β and cutoff values set empirically, dependent on the base pressure output of the compressor.

We print our objects using the FormLabs Form 2 3D printer, a consumer-level resin-based stereolithography (STL) printer capable of high resolutions. We use STL technology because in our initial experiments, it became clear that current FDM-based printers are not capable of precisely printing tiny holes at arbitrary orientations. We print AirTouch-enabled objects as single structures with no assembly needed. The only addition to the standard post-processing required of all STL print processes is a 30-second flushing stage with the air compressor immediately after printing to prevent residual resin causing blockages during curing.

3.4.2 *Flow-Distribution Chamber*

AirTouch objects embed a spherical flow distribution chamber to distribute incoming flow between between tubes; 3D printing small-size spherical shapes does not require support material. Instead of using a flow distribution chamber, we also experimented with hollowing the object. In these shell structures, outlets are simply holes and do not require tubes. This approach, however, requires per-object touch calibration while also requiring higher pressures for operation, as the air flows through the entire geometry of the object. In contrast, the spherical flow-distribution chamber has the same shape across objects and ensures a consistent airflow. To allow touch interactivity on small objects with AirTouch, the flow-distribution chamber should be small to fit objects of various geometries. Therefore, we fabricated three Stanford bunnies, each with identical outlet configurations but with varying flow-distribution chamber diameters of 15, 20 and 30 mm in diameter. We recorded the barometric pressure changes when covering the outlets and found that the volume of the chamber does not significantly impact the relative pressure variations between outlets. We did find, however, that smaller cavities yield a higher noise profile on the resulting signals; therefore, as

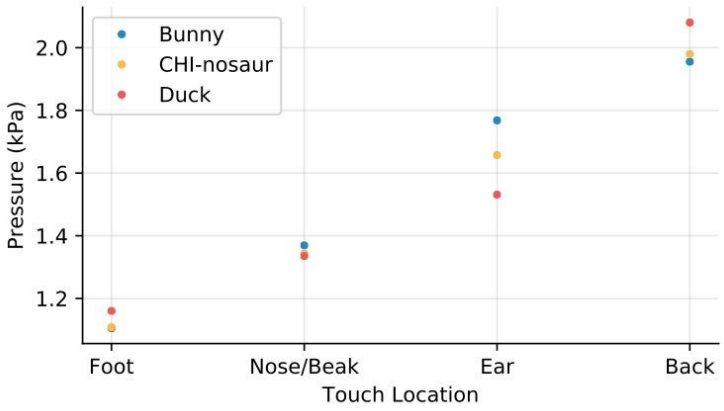


Figure 3.4: Pressure results by location on interactive animals. Note that the pressure difference between animal models is significantly smaller than the difference between touches on outlets of different sizes.

a compromise between size and performance, we used flow-distribution chambers of 30 mm diameter in most of our further experiments.

3.4.3 *Tubes*

The touch locations on the object's surface are connected to the flow-distribution chamber with cylindrical tubes. To allow tubes to be embedded in small objects, we want the diameter to be as small as possible. After experimenting with different diameters, we picked 5 mm tubes as the best tradeoff between size and printability. While 3 mm tubes often worked, we found that with this smaller size it was difficult to guarantee that all of the resin would drain from the tubes before post-print curing.

Based on fluid dynamics literature [14], we determined that the length of the tube as well as the area of the outlet influences the barometric pressure inside the object. First, we tested if varying the tube length produces enough difference in barometric pressure to identify the tube from which the airflow is blocked. To do so, we fabricated three objects of varying geometries: a duck, a bunny and a CHI-nosaur. Each has a 30 mm-diameter chamber and four outlets, placed at the foot (.35 mm^2), nose/beak (.65 mm^2), ear (.95 mm^2), and back (1.25 mm^2). Although these objects share the same cavity size and outlet configuration, their interior tube lengths vary between 3–100 mm. Figure 3.5 shows no significant difference in pressure responses when covering each of the outlets for all three objects. In the next section, we report on testing the impact of varying the area of the outlet.

Although we did not find significant difference between the pressure increases from interacting with desktop-sized objects of different geometries, we wished to identify the potential size limits of our technique. Because our available printers are limited in size, we used 6 mm polyethelyne (PE) tubing (hardness: Shore D 52) to simulate printed channels.

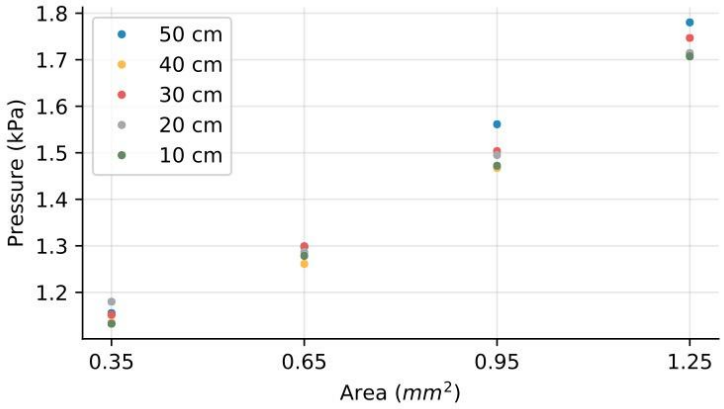


Figure 3.5: Pressure results by tube length. Note that the pressure difference between tube lengths is significantly smaller than the difference between touches on outlets of different sizes.

We fabricated a standalone flow-distribution chamber of 30 mm diameter with four 6 mm outlets. We also fabricated four 6 mm tube caps with outlets of the same sizes as the animals listed above. We connected one 50 cm length of tube to each outlet, terminating each with one cap, and recorded the pressure for each. We repeated this procedure four more times, shortening the tube length by 10 cm each time. Figure 3.5 illustrates our results: we see very little impact of tube length on pressure. As an additional informal experiment, we connected one outlet to a 40 m-long tube. At this length, the volume of air in the tube is significant (2 L) and is subject to both compression and losses due to friction [14]. Due to these effects we see a noticeable 2–3-second delay for the pressure to reach its full amplitude. While not suitable for interactions requiring immediate responsiveness, this example shows the versatility and scalability of our technique. Although these experiments suggests that AirTouch is capable of augmenting larger objects, more experimentation is still needed.

3.4.4 *Outlets*

In contrast to varying the length of the tubes, we did observe a significant difference in the barometric pressure response when blocking the airflow from tubes having different outlet diameters. To minimally disturb the object's original geometry and ensure outlets are always covered entirely when touched, we wanted to use outlets with diameters as small as possible. During our initial exploration phase, we found that our printer was unreliable in printing outlets smaller than 0.6 mm in diameter. Additionally, we observed that the difference between barometric pressure responses of small outlets are not significant in the presence of large outlets. Therefore, we empirically set the maximum outlet diameter to 1.50 mm.

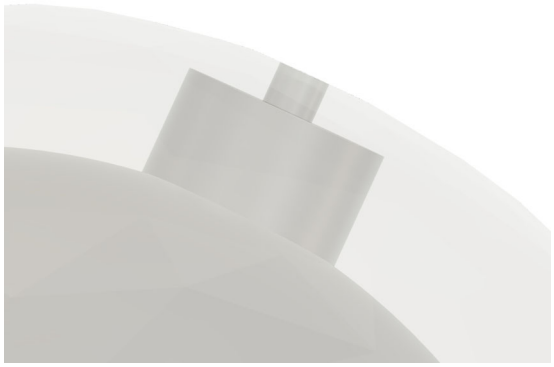


Figure 3.6: Close up view of the connection between the tube and the outlet.

With an operational range of 0.6–1.50 mm for outlet diameters, we optimized the step size between outlet diameters to maximize the number of outlets while ensuring significant differences between barometric pressure responses from all outlets. Equation 3.5 indicates that the pressure/area relationship is nearly linear for our range of diameters. We printed three test objects, each with outlets in the operational diameter range but with area steps of 0.02, 0.06, and 0.1 mm^2 between subsequent outlets. We then connected each to the testing setup and recorded touches on each outlet. We found that with the smaller area increments, the pressure change between neighboring outlet sizes was insufficient to offer a clear separation in the presence of sensor noise (approximately 0.01 kPa). Therefore, all of our subsequent objects are printed with at least a 0.1 mm^2 separation between hole areas.

As discussed earlier, the tubes connecting the outlets to the chamber are 5 mm in diameter to prevent clogging. We therefore reduce the last 1 mm of the tube to form the outlet, as shown in Figure 3.6.

3.5 SOFTWARE

In this section, we discuss the algorithms for recognizing and identifying touches as well as the software implementation that facilitates designing AirTouch's internal tube structure.

3.5.1 *Designing AirTouch Objects*

AirTouch-enabled objects require an object's internal structure to be augmented with an inlet for pressurized air, a flow-distribution chamber, uniquely sized outlets at desired touch locations, and a connection for the sensor. To facilitate designing AirTouch objects, we created an Autodesk Meshmixer script which automatically modifies the model's internal structure and adds appropriately-sized outlets to objects at user-selected locations. Our script embeds a flow-distribution chamber

inside the model, and uses the tube routing algorithm described in [103] to attach a tube spanning from the cavity to locations selected by the designer. If more precision is necessary, tubes can be added manually to objects using standard CAD software.

3.5.2 *Touch Recognition and Identification*

In order to identify when an outlet is covered, we first segment the signal coming from the sensor into 100-sample windows. We then calculate the mean and standard deviation for each window, and once the standard deviation surpasses an empirically set threshold, we assume there has been a touch or release event. To identify whether this change has been a touch or a release, we compare the mean of the current window with the mean of the previous one: if the current mean is higher than the previous, an outlet has been covered; if not, an outlet has been released.

Once we have determined that a touch event has happened, we identify which outlet has been covered. We take the mean of the 1000 samples previous to the touch event as the pressure baseline, and divide it by the mean of the 1000 samples following the touch. This ratio compensates for drift in the signal caused by minor atmospheric fluctuations and imprecisions in the regulator valve.

3.6 PERFORMANCE TESTING

To show the viability of AirTouch, we evaluated our recognition pipeline on a set of AirTouch-enabled objects of varying geometries and outlet configurations: an interactive bar chart, a Stanford bunny, a color hue selector and a dual-touch sensing sphere. For each outlet configuration, we train a machine learning model (SVM with rbf kernel) using a single instance consisting of mean and standard deviation for the registered pressures for a given touch. We proceed to cycle through the outlets of each object, recording the classification result for each touch. We repeat this process four times per object.

We obtained average accuracies of 95.5% (for the bar chart), 100% (Stanford bunny), 97.75% (color hue selector), and 91.6% (Grasp sensing cube). Figure 3.7 shows a detailed view of the performance for each object.

3.7 EXAMPLE DESIGNS AND APPLICATIONS

Because its ease of fabrication, and large number of interactive locations, AirTouch lends itself to the rapid prototyping of interactive devices. Below we present a number of example usages of AirTouch which illustrate its potential. All applications are developed in C# and WPF and receive data (over a socket connection) from Python code running the touch detection described above.

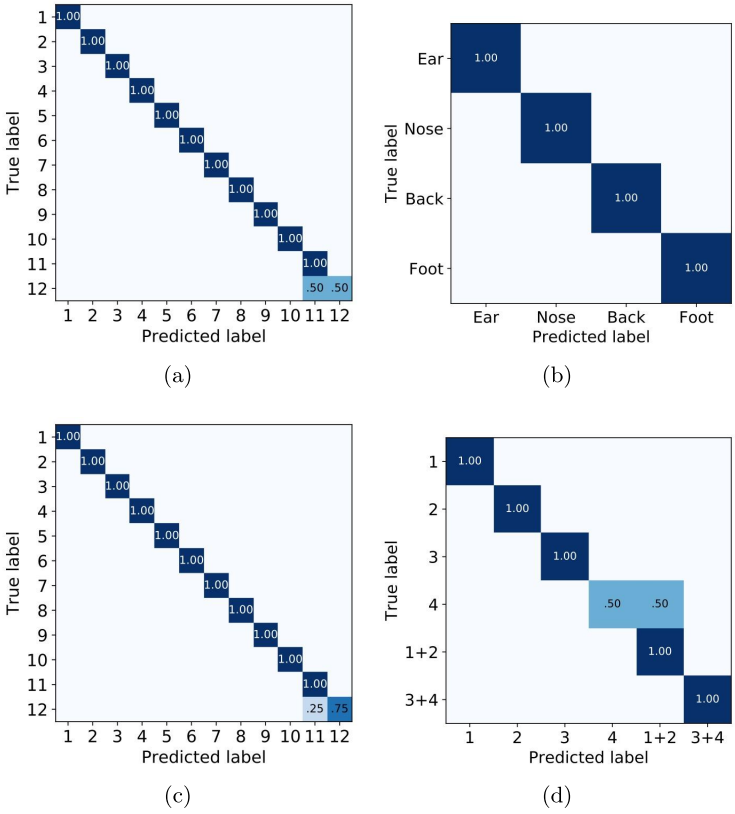


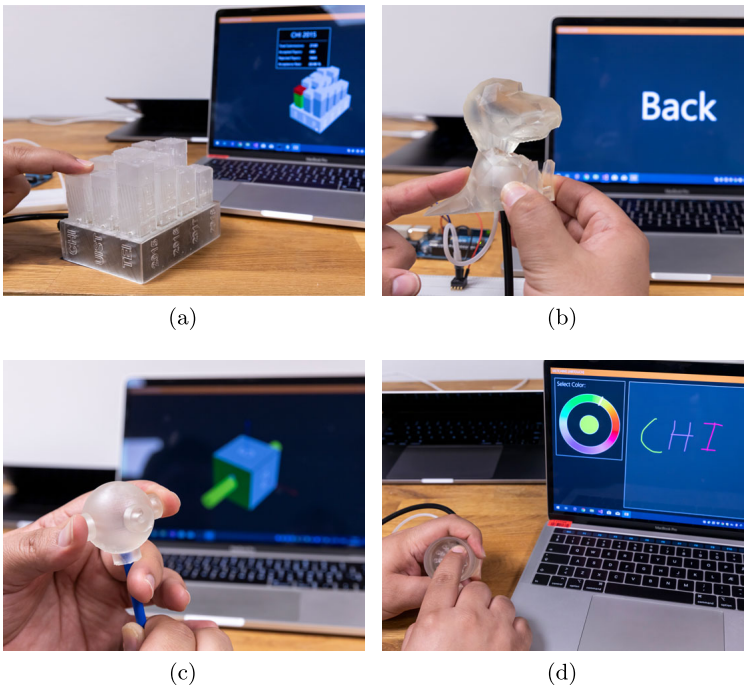
Figure 3.7: Confusion matrices of classification accuracies from our tests: interactive bar plot (a); augmented Stanford bunny (b); color hue selector (c); grasp-sensing sphere (d). Each cell indicates the number of classified touch interactions to a predicted class (row) for each actual class (column).

Interactive 3D Bar Chart

Recent work has highlighted the benefits of data physicalization [51]. We designed a three-dimensional bar plot displaying the relation between the submitted and accepted papers in the past four years to CHI, UIST, and TEI (Figure 3.8a). To obtain more information about the proceedings, the user touches the top of a bar, and the companion application shows the acceptance rate, number of accepted papers for the year and conference in question.

Interactive Animals

AirTouch can identify interactions on objects of varying geometries, but with the same outlet configurations, with a single, pre-trained machine learning model. We fabricated a set of interactive animals of different outer geometries, but sharing the same outlet configuration (i.e., an ear of different animals has the same outlet diameter). We augmented a Stanford bunny, a duck, and a CHI-nosaur with interactive locations on



(a) (b)

(c) (d)

Figure 3.8: Example AirTouch Applications. With AirTouch, interactive objects are fabricated as a single structure without any post-print assembly or calibration. We showcase objects of different geometries augmented with AirTouch: an interactive bar chart (a); interactive animals (b); grasp-sensing sphere (c); and a color hue selector (d).

the nose, ear, back and leg. When a location is touched, the corresponding label is displayed (Figure 3.8b).

Grasp Sensing

To showcase AirTouch's dual-touch sensing capabilities, we developed a touch-sensing sphere (Figure 3.8c). We augmented a sphere with four touch locations throughout its surface. When an outlet is covered, the companion application highlights which face is touched. When two outlets are covered simultaneously, the system highlights both faces.

Color Hue Selector

AirTouch can enable up to 12 interactive locations on 3D-printed objects. We designed a circular color hue selector for a drawing application (Figure 3.8d). The user selects a color using the selector, and sketch in the drawing window.

3.8 DISCUSSIONS AND LIMITATIONS

While AirTouch is successful in fabricating touch-sensitive objects and tangible input components without the need of any post-print assembly it has limitations. The most obvious one is the need for an air compressor to power the fabricated objects. Although we employed an air compressor to power our fabricated objects, other air sources might be used, granted they guarantee a constant stream of air. A miniature air pump, similar to the one used by Vázquez and collaborators in [131], can power AirTouch-enabled objects given their low pressure requirements.

Another limitation of our technique is that it's only able to augment objects fabricated using high-resolution 3D-printing technologies. We explored fabricating AirTouch-enabled objects using Fused Deposition Modeling (FDM) printers, but encountered a number of issues. Because of its layer-by-layer fabrication procedure, some objects fabricated with our Lulzbot Taz 6 and QiDi Technology X-One presented significant leaks in the internal structure of the object, hindering the technique's performance. We plan to explore the effects of employing smoothing techniques (e.g., acetone smoothing), and varying the shell thickness of the model when printing to reduce leaks. FDM printers also lack the precision of STL printers, meaning that when printing outlets. Future work can explore the different approaches to ensure the correct fabrication of outlet sizes on lower resolution equipment.

Although we experienced a noticeable latency in our 40m-long tube experiments between when covering the outlet and the signal reaching its full amplitude, these effects were not present in AirTouch-enabled objects—the pressure increase upon covering an outlet was instant. Future work can explore the use of our technique to enable richer gestures such as swiping and sliding on fabricated objects.

Employing the same principle used to detect individual touches, AirTouch can identify up to two simultaneous touches on different locations throughout the fabricated object. Because the increase in pressure is proportional to the outlet area, covering multiple outlets can be identified as a new touch location—as long as the sum of the areas of the covered outlets results in a unique change in pressure. In order to guarantee a $0.1\,\mathrm{mm}^2$ separation between the outlets and their respective combinations, we used outlets sizes of $0.4\,\mathrm{mm}^2$, $0.5\,\mathrm{mm}^2$, $0.6\,\mathrm{mm}^2$, and $0.8\,\mathrm{mm}^2$ in our test objects.

Finally, we experienced small variations in the measured barometric pressures inside our fabricated objects when compared to previous days. This is due to the everyday changes in environmental barometric pressure. To evaluate the effects of these everyday changes in our system's performance we performed a preliminary test over a period of four consecutive days where the ambient pressure varied from 0.1 to 0.7 kPa. Using the bunny model, we recorded touches each day and found that the pressure variation shifted the baseline of the measurements by an amount smaller than the separation between different touches.

3.9 CONCLUSION

In this paper we introduced AirTouch: a technique for fabricating touch-sensitive objects without the need of any post-print activities such as assembly or calibration. We presented the theory behind AirTouch, our explorations of parameters for both interaction and successful fabrication, and guidelines for designing AirTouch-enabled objects. We illustrated AirTouch's flexibility with several applications, and showed that AirTouch is able to identify interactions with accuracies of at least 91% with 12 interactive locations.

4

BLOWHOLE

While AirTouch is a successful technique for constructing tangible devices that can respond to user's touch, it had some serious limitations. The most prominent of these limitations was the need of a constant, pressurized source of air, like an air compressor. Additionally, having to connect the device using two points of entry (i.e., one to the air source, another to the sensing apparatus) meant that the designer had to perform a few post-print activities, albeit quick and easy ones.

Inspired by these limitations, and the ubiquity of microphones in our daily lives, I developed Blowhole. This next chapter, which is almost identical to the paper I published by the same name [121], and introduces a technique to fabricate tangible devices that can identify user's interactions by acoustically tag locations on the object. In this chapter, I introduce Blowhole's theory of operation, and characterization and evaluation procedures I carried out to assess its performance.

Blowhole improves on AirTouch in two main ways. First, as it employs acoustic sensing rather than pneumatic sensing, users can provide the air for it to operate by gently blowing on the object's openings. Second, thanks to the ubiquity of microphones, Blowhole-enabled objects can employ already existing microphones to identify user's interactions. Additionally, Blowhole uses a mathematical model to identify blowing locations based on the frequency emitted with gently blown. This means that an user can use a Blowhole-enabled device immediately after removing from the printer.

This chapter is based on the collaborative effort described below.

Title
Blowhole: Blowing-Activated Tags for Interactive 3D-Printed Models

Authors
Carlos E. Tejada, Osamu Fujimoto, Zhiyuan Li, Daniel Ashbrook

DOI
https://doi.org/10.20380/GI2018.18

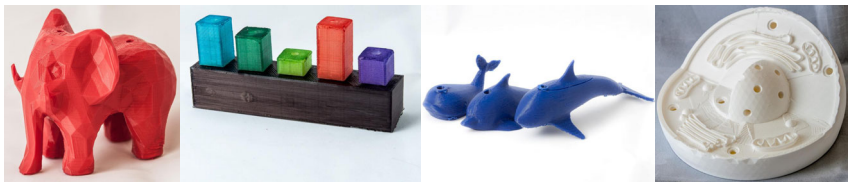
Venue
Proceedings of the 44th Graphics Interface Conference

What was the role of the PhD student in designing the study?
The PhD student was the first author of the paper, and responsible of the described studies.

How did the PhD student participate in data collection and/or development of theory?
The PhD student was responisble for study implementation, execution, data collection, and theory development.

Which part of the manuscript did the PhD student write or contribute to?
The PhD student contributed to all parts of the manuscript.

Did the PhD student read and comment on the final manuscript?
Yes.



(a) (b) (c) (d)

Figure 4.1: Example Blowhole applications. The elephant (4.1a) has six holes (one on the head, one on the back, one on each foot) which, when blown into, activate text-to-speech with different elephant facts. Each bar in the bar chart (4.1b) triggers a readout of the quantity, and can be moved to represent different data. The sperm whale, dolphin, and orca (4.1c) each produce a different tone to trigger videos of those animals. The cell model's blowholes interact with a quiz program to test knowledge of its components (4.1d).

Abstract

Interactive 3D models have the potential to enhance accessibility and education, but can be complex and time-consuming to produce. We present *Blowhole*, a technique for embedding blowing-activated tags into 3D-printed models to add interactivity. Requiring no special printing techniques, components, or assembly and working on consumer-level 3D printers, Blowhole adds acoustically resonant cavities to the interior of a model with unobtrusive openings at the surface of the object. A gentle blow into a hole produces a unique sound that identifies the hole, allowing a computer to provide associated content. We describe the theory behind Blowhole, characterize the performance of different cavity parameters, and describe our implementation, including easy-to-use software to automatically embed blowholes into preexisting models. We illustrate Blowhole's potential with multiple working examples.

4.1 INTRODUCTION

Adding interactivity to 3D-printed objects in an end-user-friendly way is a difficult undertaking. While doing so can enable a wealth of educational and accessibility applications (e.g., Figure 3.1), adding interactivity can require extra components [76, 99], special printing techniques [15, 87, 107], or large features that can disrupt the object's surface [40, 83, 102, 113] or make it larger [65]. While such solutions can offer rich interaction possibilities such as buttons, touch sensitivity, and grasp sensing, many useful applications can be enabled by simply "tagging" 3D models, allowing a system to know what part of a model a user is interested in and respond with information.

In this paper, we present *Blowhole*, a system for unobtrusively tagging 3D models or parts of models. Inspired by previous work in acoustic sensing, our system creates embedded, resonant cavities that a user gently blows into (Figure 3.1). Different-sized cavities create sounds

at different pitches that can be recognized by a computer. The main focus of Blowhole is to enable 3D-printing novices to create interactive objects which can be printed and immediately used with no during-print intervention, post-processing, or training necessary. The contributions of our work are as follows:

1. a technique for embedding resonant cavities—printable on commodity 3D printers—into existing 3D models, which produce distinct pitches when blown into;
2. characterization of the range of frequencies that can be generated and recognized by our system;
3. a design environment which enables non-expert users to indicate areas which they wish annotated, and automatically generates and embeds appropriate cavities;
4. and a user-independent system which recognizes the sounds of the user blowing into the cavities and plays back the associated annotations.

4.2 BLOWHOLE

Blowhole is based on the property of acoustic resonance; a familiar example is the sound created when blowing across the mouth of a bottle. Blowhole embeds cavities into 3D models, with tubular openings to the surface. Varying the volumes of the cavities and lengths of the tubes produces varying frequencies in response to gentle blowing into the holes, with the object held 5–10 cm away from the mouth. Our system recognizes the characteristic sound of each hole, linking the blow sound to an action associated with the hole's location on the model. Our design tool allows a user to select the placement of holes on arbitrary 3D models and associate actions with each hole; the software then optimizes blowhole size and placement, providing a printer-ready file.

The cavities used in Blowhole must satisfy several criteria: they must support sufficient variation in parameters to produce a range of frequencies when blown into; they must be sufficiently small to embed into models small enough to hold and manipulate; they should present a consistent hole appearance to the user; and they should be printable at any orientation and without support material on a consumer-grade printer.

This last criterion poses the strongest limitation due to the limited ability of FDM[1] 3D printers to print with *overhangs*—angles greater than 45° from gravity—and *bridging*—printing material with nothing underneath it as a support. We experimented with a number of shapes. Simple tubes (as used, for example, in Whoosh [94]) printed well at any orientation but quickly became too large to embed in smaller objects while supporting a range of frequencies. We tested several variations on

1 Fused Deposition Modeling—the most-common consumer printer technology

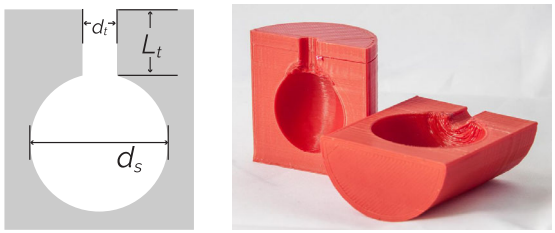


Figure 4.2: Left: an ideal spherical Helmholtz resonator, with tube diameter d_t, tube length L_t, and sphere diameter d_s. Right: cross-sections of two Blowhole test objects, showing the resonator structures.

shorter tubes connecting to larger cavities, which preserve a standard opening size but allow the production of a greater range of frequencies. We experimented with cavities shaped like spheres, cylinders, cylinders with cones on tops, cubes, and cubes with pyramids. The shape resulting in the best combination of clear sound and multi-orientation printability was a sphere with a tube connecting to the surface of the model. In the next section, we detail the theory behind the resonant properties of this structure.

4.2.1 *Cavity Resonator Theory*

Blowhole operates on the principle of acoustic resonance, where particular frequencies are amplified or attenuated due to the physical properties of a cavity. Blowhole uses spherical cavities inside a 3D-printed model with straight pipes opening onto the surface; the resonant frequency of a cavity depends on the area and length of the opening and the volume of the cavity, and is classically modeled using the Helmholtz resonance equation [43]:

$$f = \frac{cd_t}{\pi}\sqrt{\frac{3}{8(L_t + .75d_t)d_s^3}} \tag{4.1}$$

with c the speed of sound, d_s the diameter of the spherical cavity and d_t and L_t the diameter and length, respectively, of the tube connecting the cavity to the surface of the object. Figure 4.2 illustrates these parameters of Blowhole cavities.

4.2.2 *Blowhole Characterization*

In order for Blowhole to be of the most practical use, we want to understand how many different cavities we can fit inside a given object. As can be seen from Equation 4.1, we can vary three parameters—d_t, L_t or d_s—to change the resonant frequency of a Blowhole cavity. As the tube is the only user-facing element of Blowhole, its appearance should be consistent, with the size of the opening large enough to easily blow into, but not so large as to interfere with the features of the

Figure 4.3: A subset of our test cylinders with varying cavity volumes and tube lengths.

printed model. After some initial experimentation, we set d_t to 5 mm, leaving L_t and d_s as the available parameters to manipulate. Multiple combinations of these can produce the same predicted frequency; for example, $L_t = 2.5$ mm and $d_s = 35.3$ mm produce a prediction of 1000 Hz, as do $L_t = 5$ mm and $d_s = 28$ mm.

To understand the practical limits on the frequencies we could detect and differentiate between, we produced a large number of test objects (Figure 4.3) using consumer FDM printers (Qidi Technology X-One, LulzBot Taz 4, and LulzBot Taz Mini). Wanting to understand the practical limits on the frequencies we could detect and differentiate between, we produced 48 objects with cavities and tubes of different sizes. Holding d_t at 5 mm, we manipulated d_s from 8–40 mm in steps of 4 mm and tested L_t at 2.5, 3.5, 5, 7.5, 8.5 and 10 mm. These configurations gives us a frequency space ranging from 500 Hz, ($d_t = 5$ mm, $L_t = 10$ mm and $d_s = 40$ mm) to 5900 Hz ($d_t = 5$ mm, $L_t = 2.5$ mm and $d_s = 8$ mm).

We asked ten people to blow into each cylinder between one and four times, recording the data via a laptop computer's built-in microphone at a 44,100 Hz sampling rate. We extracted the fundamental frequencies of each blow using Welch's method [132]. Comparing the fundamental frequencies to the values predicted by the Helmholtz equation (Equation 4.1), we find deviation, sometimes significant. However, the deviation is not constant, but presents as noise, with two main patterns: frequencies under 1000 Hz are much noisier than those higher; and longer tubes exhibit more noise than shorter. Figure 4.4 illustrates the spectrogram from one user for a series of blows into different cavity sizes.

This behavior may be explained by several factors. First, the Helmholtz equation is a theoretical model known to be inexact for varying properties of cavity geometry [4, 110], assumes a reflective, smooth surface, and as L_t and d_s approach 2–5% of the resonant wavelength, the model begins to break down [110]. As shown in Figure 4.2, our 3D-printed models are not smooth. As the cavity size increases, the top of the sphere approaches

horizontal, which can cause drooping or stringing. These features of 3D prints may affect the resonance. To validate print features as a potential cause, we printed eight duplicate test objects on a resin-based printer (the FormLabs Form 2), resulting in hard, smooth objects. Blows into these objects resulted in frequencies much closer to the predicted values. However, our goal is to maintain broad accessibility of our technique, so while noting the potential for resin printers to produce better results, we continue to describe our results using FDM printers.

To validate the printability of our cavities, we tested multiple cavity sizes, from 5 mm to 60 mm in diameter. While all objects printed correctly, the smallest cylinders resulted in frequencies highly variable over the course of a single blow, and the 60 mm cavity failed to produce any strong harmonic at all.

We also tested the consistency of sound at different angular positions of the tube opening, from 0° (straight down) to 180° (straight up) in 22.5° increments with d_s as 16 mm and L_t as 5 mm. Although different orientations revealed different (minor) printing artifacts such as slight stringing and tube opening shape inconsistency, the results were consistent, with a mean deviation of under 240 Hz from the Helmholtz-predicted value.

We tested Blowhole with multiple printers: a LulzBot Taz 4, a LulzBot Taz Mini, two Qidi X-One v2 printers, and a Form Labs Form 2 resin-based SLA printer. All printed successfully; inspecting the spectrograms, we found little variance amongst the FDM prints, and that the SLA prints produce dominant frequencies on average 100 Hz closer to the Helmholtz-predicted frequency than the FDM prints and with less variation over the signal.

4.3 SYSTEM IMPLEMENTATION

As a system, Blowhole consists of three parts: the design software to modify existing 3D models to add blowholes; the physical printed-out models with resonant cavities and holes embedded; and the software that recognizes the sound of the user blowing into a cavity and performs an action. All pertinent code and designs can be found online [2].

4.3.1 *Design Software*

Our design software is built on top of Autodesk Meshmixer (Figure 4.5, left) using its Python API for scripting remote command execution. To add Blowhole tags to a model, a user simply imports an existing model and then clicks on the model to specify tag locations and desired actions. Currently supported actions include opening URLs, launching files such as images and movies, and reading text via text-to-speech. After the user indicates all of their desired blowhole positions, the software determines the best set of cavity sizes to embed in the model. The naïve approach would be to simply place the largest possible available cavity

2 https://github.com/fetlab/blowhole-gi18

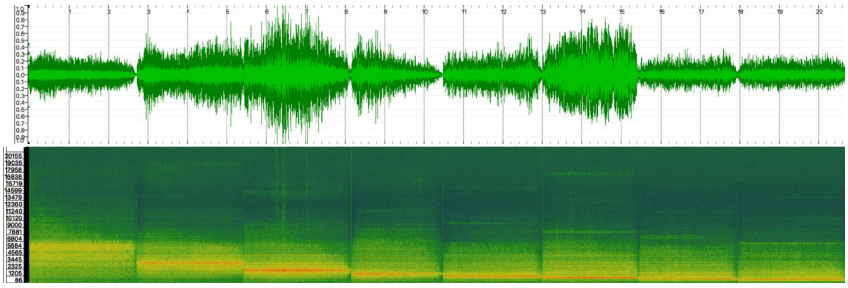


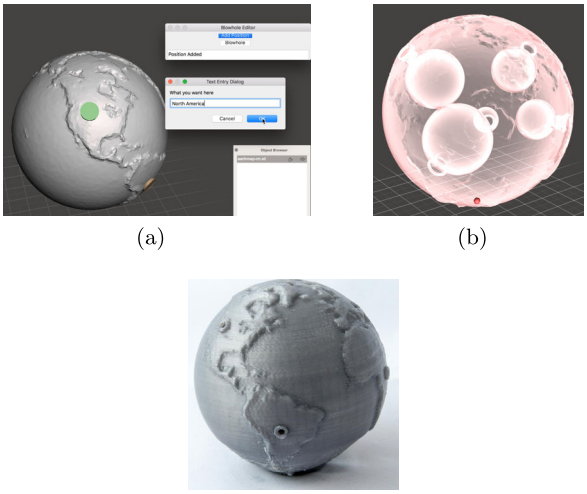
Figure 4.4: Waveform (top) and spectrogram (bottom) of blows into holes with a tube length L_t of 2.5 mm, with the cavity diameter d_s varying in steps of 2 mm from 4 mm on the left to 18 mm on the right.

in a location when the user selects it. However, this approach quickly fails; for example, if the user wished to place a blowhole in each eye of the elephant in Figure 4.1a, the first click would fill the elephant's head with a cavity and the second click would fail to find an acceptable cavity size. The opposite approach—choosing the smallest available size first—suffers from similar issues.

Instead, our search algorithm allows the user to add all requested positions first, then attempts to optimize cavity placement by finding a set of L_t and d_s that will fit all requested blowhole locations without cavities colliding (we can optionally fix L_t to a single value). The algorithm is based on depth-first search with backtracking. We represent the solution space—mapping a set of available cavities to desired blowhole positions—as a tree, with the root representing the original model, internal nodes as intermediate steps towards solutions, and leaves as final solutions. Each step of the algorithm takes as input a node, a list of candidate locations, and a list of unused cavity sizes. It tests each of the available cavities in the next candidate location until it finds one that fits within the model at that location. It then removes the candidate location and the cavity size from their respective lists and passes the newly modified model as a node to run the algorithm again. If, for a given node, no cavity can be found that fits within the candidate location, the tree is pruned at that point and the algorithm returns to the parent node to try the next child. When no candidate locations remain, we can output the current node's model as the solution; if candidate locations do remain but there are no more cavity sizes, we inform the user that we cannot find a solution. The search process takes under a minute.

The final result is a set of location/cavity pairs, which we then use to construct the model for printing (Figure 4.5, right). Once the cavities are placed, the software writes out a configuration file linking the cavity parameters L_t and d_s to the specified action. The final model may be exported to a STL file for 3D printing on a commodity printer.

In informal experiments with Blowhole-enabled objects, we found that locating the holes by feel alone could be challenging; on complex models like the elephant (Figure 4.1a) the hole gets "lost" in the model's geometry. Because one use scenario for Blowhole objects is as an aid for people with visual impairments, we added a "ring" feature. This simply



(a) (b)

(c)

Figure 4.5: (a,b): Detail view of our Blowhole design software, based on Meshmixer. (a) shows the software with the user inserting blowholes: clicking a point on the model results in a placeholder (green dot) and a dialog box where the user can specify the action to be taken upon blowing. (b) shows the interior of the model, illustrating the different-sized cavities the software inserted. (c) shows the final 3D-printed object with blowholes embedded.

adds a short (3 print layers, or about 0.64 mm high) ring extending 1 mm around the hole (Figure 4.7). In informal tests, we found that the ring is easily distinguishable by touch alone, allowing the hole to be easily located without vision. In order to avoid changing the sound produced, when we add the ring we shorten the inner end of the tube by the height of the ring, thereby maintaining the same L_t.

4.3.2 *Blowhole Objects*

Our software places blowholes into existing models, therefore models that are 3D-printable will remain so with the addition of cavities and openings. Because the cavities are spherical, and most hobbyist-level 3D printers can print up to 45° of overhang, the models can be produced on most printers with no modification; importantly, no support material is necessary inside the cavities or tubes.

Once a Blowhole-enabled object has been printed, some minor cleanup may be required: with larger spherical cavities, the top of the sphere becomes nearly horizontal, and the printer may produce some "3D printer spaghetti" (a small amount is visible in Figure 4.2) that can slightly muffle the sound. A simple solution is to simply insert a drill bit of the appropriate size and twist it by hand to quickly remove the strands.

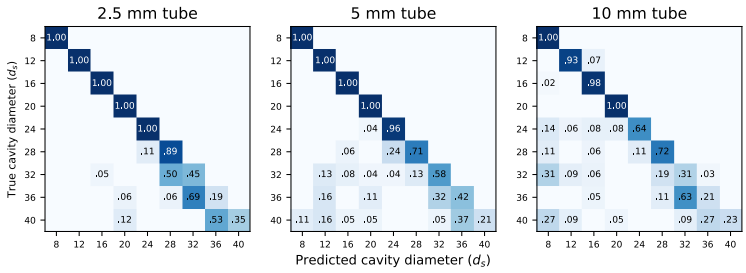


Figure 4.6: Confusion matrices across all test participants for the three tested tube lengths with sphere diameters from 8–40 mm.

4.3.3 *Blow Sound Recognition*

The last component of our system recognizes the sounds produced by the user blowing into the blowholes, producing the resonant frequency characteristic to cavity/tube combinations (Figure 4.1), allowing us to link the sound to the particular location the user is interacting with. Our software is implemented in Python running on a laptop, but is simple enough to run on phones and smartwatches as well.

To identify the resonant frequency, we window the 44,100 Hz incoming audio signal in 0.1s non-overlapping segments. We compute the RMS value of each and look for .5s worth of contiguous windows that exceed an empirically determined threshold. We apply Welch's method to extract the power spectrum of the signal [132], and use the strongest frequency as the resonance. We then take the set of cavity/tube (d_s/L_T) combinations available and match the resonant frequency to the Helmholtz-predicted frequencies to determine which hole the user is interacting with. Once a blow is classified, the system executes the action referenced in the configuration file produced by the design software.

Our main implementation is on a laptop computer, using its built-in microphone. We also tested with a LG-R Android smartwatch which transmits audio data to the same recognition pipeline. Our software runs in Python and uses the scikit-learn library for recognition.

4.3.4 *Performance Testing*

To validate our recognition procedure, we collected a total of 830 blow segments from ten participants as described in Section 4.2.2, with L_t of 2.5, 5, and 10 mm, and d_s varying from 8–40 mm in 4 mm increments. We divided the data according to the tube length and evaluated our recognition procedure both overall and on a per-user basis. Figure 4.6 presents a cross-validated evaluation of our classification system, where we can see that the reliability decreases as the diameter of the sphere increases.

We found that as we include larger spheres and larger tubes, the recognition accuracy decreases. The best performance/versatility tradeoff occurs at L_t of 2.5 mm and six spheres from 8–28 mm, which yields an

(a) (b) (c) (d)

Figure 4.7: Example Blowhole-augmented objects: (a) a model of Mt. Rushmore with each president tagged; (b) the bar graph from Figure 4.1b in a different arrangement; (c) a box that controls a music player by blowing; and (d) a 3D-printed tactile picture book for blind or low-vision children with a blowhole (upper right) which triggers text-to-speech of the Braille text.

overall 98% accuracy. Adding a 32 mm sphere decreases accuracy to 90%, and further spheres continue to decrease accuracy.

4.4 EXAMPLES

To illustrate the potential of Blowhole, we present several possible applications. Each was built with our software and works with our recognition algorithm.

Cell Model

We adapted an existing model of an animal cell[3] to add Blowhole tags to the different parts of the cell (Figure 4.1d). When the tags are activated, the listening computer application launches the Wikipedia page for the associated cell component.

Globe

Similar to the example in Tickers and Talker [113], we tagged the continents on a 3d-printed globe[4] (Figure 4.5c). When a user blows into the associated hole, our software speaks the name of the labeled continent using text-to-speech.

Interactive Animals

We printed three different cetaceans: a dolphin[5], a whale[6], and an orca[7], and adapted the position of the cavity to the location of the animal's blowhole (Figure 4.1c). When the user blows, the application plays a video about that animal.

3 http://www.thingiverse.com/thing:689381
4 http://www.thingiverse.com/thing:17336
5 http://www.thingiverse.com/thing:1121803
6 http://www.thingiverse.com/thing:232247
7 http://www.thingiverse.com/thing:665571

Music Controller

A "music box" with raised controls (Figure 4.7c) allows a user to control the flow of music by blowing. Each "button" has a different blowhole underneath it. Our segmentation algorithm described earlier is robust to background sound and in initial testing, its performance was not affected by the sound of the music playing.

Augmented Tactile Book

Previous research [56] has investigated 3D-printed tactile picture books for blind children. Some examples of these books have Braille text[8]. We created a set of custom rectangular resonators, thinner than our standard Blowhole spherical resonators, to add to each page of a 3D-printed book (Figure 4.7d). When the user blows into the hole, the computer reads aloud the text written in Braille on the page.

Reconfigurable Bar Chart

Figure 4.1b shows a Blowhole-enabled physical visualization [118]. Taking advantage of the Helmholtz property that varying L_t varies the frequency, the base of the bar chart contains cavities with identical d_s values. Each bar, being a different height, has a different L_t; when a bar is plugged into the chart, the tone produced is due to the size of the bar. This characteristic enables reconfiguring the bar chart, illustrating data in different orders while maintaining the labels on the individual bars. Placing the resonating cavities in the base rather than the bars allows the bars to maintain smaller cross-sections but still produce audible tones.

4.5 DISCUSSION AND FUTURE WORK

Our goal with Blowhole was to create a system to enable non-experts to design and 3D print interactive objects, with the particular aim of simplicity, avoiding interventions during printing, post-print processing, and complex training processes. Blowhole is usable for simple cases; with up to six cavities, the system achieves a high user-independent performance of 98%. As illustrated by our examples in Figures 3.1, and 4.7, six cavities are sufficient for many applications, and is a larger number of tags per object than have been demonstrated by other passive acoustic-based systems[40, 42, 61, 65, 94, 113].

While Blowhole is successful, there is room for future improvement. We are interested in further characterizing the behavior of the cavities with different print settings; for example, we have encountered some tentative evidence that the type and amount of infill the printer uses to fill the solid parts of the model may have an effect on the sound. Refinements to the cavity shapes may also have an effect, for example

8 https://tactilepicturebooks.org/

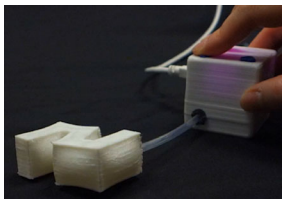
by shaping the top of the spheres to avoid greater than 45° overhang in order to prevent the "stringing" effect. Additionally, we are interested in exploring the user experience while using this tool. We believe that tools like Blowhole can be particularly useful for educational settings, specially for children, due to its playfulness, as well as to enable new ways low vision users can interact with their technology. We will carry out user studies in the future to assess the experience of these two populations while using Blowhole.

4.6 CONCLUSION

We presented Blowhole, a system for adding acoustic "tags" to 3D-printed models via embedded cavities which resonate at characteristic frequencies when a user blows into them. Our system enables high performance for up to nine different blowholes, provides simple point-and-click design software, and Blowhole-enabled models are ready to use immediately post-printing with no assembly or external components required. We detailed the theory behind Blowhole's operation, presented our characterization of its performance, and demonstrated high user-dependent and -independent recognition rates. We demonstrated Blowhole's potential through multiple examples, including educational models, 3D-printed book pages, and a reconfigurable physical visualization.

5

MORPHEESPLUGS



AirTouch and Blowhole are both successful in enabling the construction of tangible devices that can sense user's interactions without requiring assembly of parts, electronics, or calibration of machine learning models. These techniques are aimed to constructing devices that can *sense* interactions, meaning that they rely on external devices to provide output. This limits somewhat their utility. Additionally, as discussed in Chapter 2, the construction of tangible devices that provide output to interactions remains far-off from the Print-and-Play Fabrication ideal.

Inspired by these limitations, this chapter introduces MorpheesPlug: a toolkit for the construction of tangible devices that can change their physical form, providing visual output to the user. The MorpheesPlug toolkit is made up of three main components: a design environment, a set of air-powered, shape-changing widgets, and a control module to pneumatically actuate the widgets. This chapter, additionally, includes an evaluation of the usability of the MorpheesPlug toolkit, a set of inspiring applications constructed with our widgets, and thoughtful discussions and directions for future work.

While successful, the development of MorpheesPlug was not one without its challenges. The most prominent of these was the construction of the widgets using accessible fabrication equipment. Significant iterations on designs, and print settings, were needed to construct our designs using consumer-grade printers, while remaining airtight.

This chapter is based on the collaborative effort described below.

Title
MorpheesPlug: A Toolkit for Prototyping Shape-Changing Interfaces

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

DOI
https://doi.org/10.1145/3411764.3445786

Venue
Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems

What was the role of the PhD student in designing the study?
The PhD student aided the first author in the ideation of the study.

How did the PhD student participate in data collection and/or development of theory?
The PhD student was involved in the development of the software, testing of the printed objects, and visualization of their specific results.

Which part of the manuscript did the PhD student write or contribute to?
The PhD student contributed to all parts of the manuscript.

Did the PhD student read and comment on the final manuscript?
Yes.

5.1 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 3D-printable pneumatic SCIs; and (3) a hardware platform to control the widgets. To evaluate MorpheesPlug we carried out ten open-ended 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.

5.2 INTRODUCTION

Shape-Changing Interfaces (SCIs) are emerging as a new generation of devices that can change their shapes to support dynamic affordances [28], leverage human dexterity [89], and support the personalization of physical interfaces [57]. The current design space of SCIs covers a wide range of features [53], including variable length [28], volume [57], curvature [135], and porosity [19]. 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 [3]. One major barrier to the creation of SCIs is the lack of standardized toolkits for exploration and development [3]. 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 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 [9, 33] and tools to ease the communication between the digital and physical worlds [41].

Shape-Changing Interfaces 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 [117]. 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 [53] (e.g.,

length) and actuation methods [117] (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 [9, 33], but the bulk of the SCI literature focuses on novel techniques (e.g., [84]) 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 [53]. 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:

1. 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 publicly share design software and control module for the widgets [1].
4. We demonstrate the utility of our toolkit via five proof-of-concept applications and a qualitative user study.

5.3 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.

1 https://github.com/shape-changing-interfaces/MorpheesPlug

5.3.1 *Toolkits for Shape-Changing Interfaces*

Alexander et al. [3] identified twelve grand challenges in SCIs research. Although many types of SCIs have been explored in the literature [117], most are custom-made, one-off projects developed to illustrate an interaction technique, actuator, or application; hence, Alexander et al. [3] 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. [63]: 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. [63] 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 [38], 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 [91], a system of passive and active (motorized) building blocks that could record and re-play movements. LineFORM [72] and ChainFORM [73] 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 alloys or nitinol wires to actuate shape changes. shape-memory alloy-based actuation has the advantage of small size and flexibility, but at the expense of actuation speed. One early example, Bosu [85], offered a set of frames and fabric shapes on which the shape-memory alloy wires could be fixed. While these components formed a small library of transformable shapes, Bosu required users to assemble each component manually. NURBSforms [119] 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+ [53], a result of the low-amplitude length change possible with shape-memory alloys.

Some systems use pneumatic actuation to transform shapes. One of the earliest projects in this space was PneUI [135], 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 [98] and AeroMorph [84], both of which require custom-built equipment to create, and Siloseam [69], 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+ [53] 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.

5.3.2 *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 [33]. 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 [9], more powerful widgets [130], or novel form-factors [44]. 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 [104], for example, provided tools to help users fabricate customized

touch sensors that could wrap around objects of varying sizes; Pineal [62] added "remote widgets" to smartphones and watches via automated 3D modeling; and PaperPulse [93] 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.

5.3.3 *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 [57] illustrated multiple input and output behaviors, Harrison and Hudson's inflatable buttons provided dynamic haptics [39], and PneUI [135] 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 [69] 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 [131], and Lee et al. developed a system of Lego-compatible pneumatic blocks for experimenting with soft robotics [64]. 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 [136].

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.

5.4 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 [63]. 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 [81], 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 [135], or special machinery [84, 98]. 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 [32] and exploring design spaces that has not examined before [81]. To show that MorpheesPlug has such properties, we replicate one of the SCIs that had a huge impact in the field [28] 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 [3, 71]. SCI field is still at the early stage, and we believe that there are too few design guidelines to be generalized (e.g., [36, 54, 127]) 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.

5.5 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 5.1 shows the widgets we designed, and Figure 5.2 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.

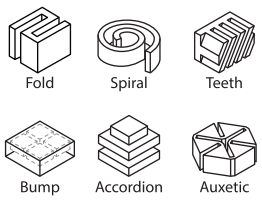


Figure 5.1: The six widgets that MorpheesPlug provide. The widgets can express different shape-change features such as length, curvature, etc.

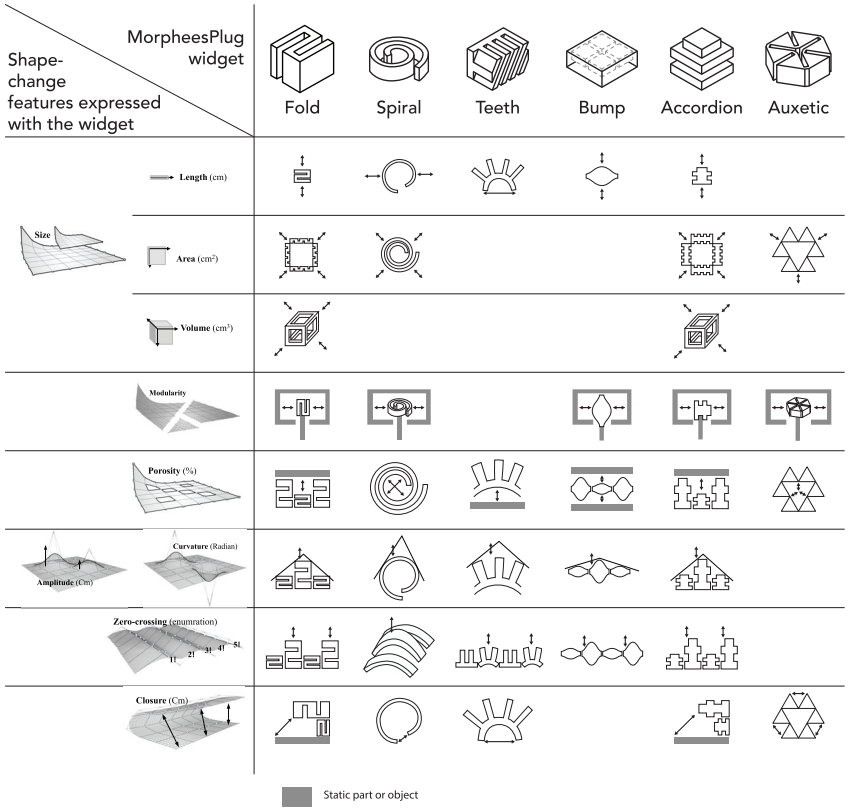


Figure 5.2: Top: Widgets provided by MorpheesPlug. Left: shape-change features from literature [53, 97]. Middle: Illustrations of how the widgets can express the shape-change features.

5.5.1 *Fold widget*

The Fold widget is a widget that is primarily designed to implement length change (Figure 5.2). 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 [16] 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 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.

5.5.2 *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 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.

5.5.3 *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.

5.5.4 *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.

5.5.5 *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 [22]. Thanks to the grooved surfaces on the four sides, it can have more length change than Bump widget.

5.5.6 *Auxetic widget*

We designed our auxetic widget to display porosity feature. I got inspired from the literature [34]. 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.

5.6 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 [53]; (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 under the MIT license.

5.6.1 *Design Software*

Our design environment is built on top of Autodesk Fusion 360 (Figure 5.3) 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.

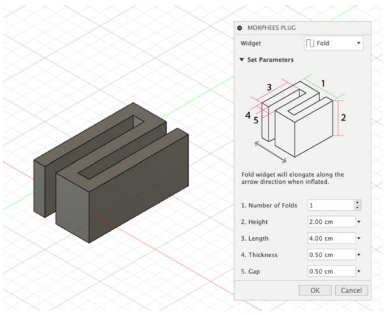


Figure 5.3: Detail view of our design environment, based on Autodesk Fusion 360. The user is presented with a window where they can specify the sizes for the different parameters that compose our widgets.

5.6.2 *Fabrication*

All MorpheesPlug widget designs are printed as a single structure using consumer-grade FDM [2] 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. When the overhang surface too large, we allowed support. An in detail view of the parameters can be found in Table 6.1.

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.

5.6.3 *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

2 Fused Deposition Modeling

Parameter	Value
Printing Speed (mm/min)	1200
Extrusion Multiplier	1.3
Top Solid Layers	10
Bottom Solid Layers	7
(For overhangs <4 mm) Interior fill	0%
(For overhangs >4 mm) Interior fill	10%
Combine Infill Every	2 layers
Combine Overlap	25%
Outline/Perimeter Shells	2

Table 5.1: List of modified printing parameters with their respective values used to fabricate our widgets.

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 5.3). 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 5.4, repeating each measure ten times.

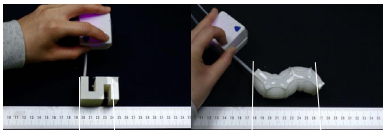


Figure 5.4: Our characterization setup. To measure the length change of the Fold widget, we placed a printed widget next to a ruler and measured the length of both deflated and inflated states.

Figure 5.5 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 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.

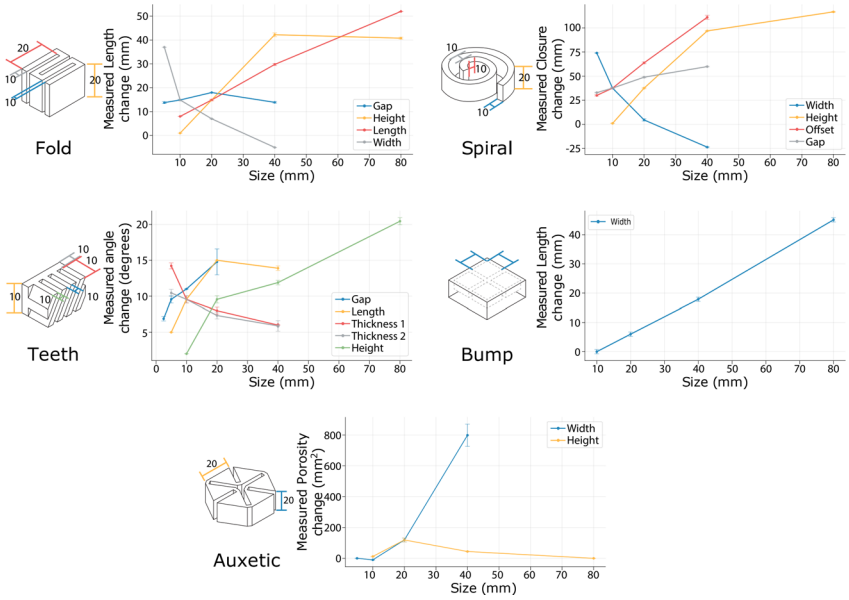


Figure 5.5: Results of the characterization. These plots show magnitude of the modified features versus the change of shape the widgets expressed. The numbers on the widgets show the baseline size of the features. For example, Fold widget had a baseline parameters of gap 10 mm, length 20 mm, height 20 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).

5.6.4 *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 5.6). 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.

5.7 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.

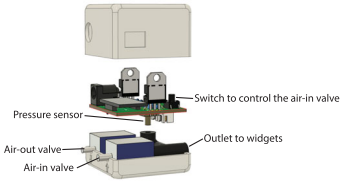


Figure 5.6: An exploded view of the control module. The module has two valves to let air in and out of the widgets.

5.7.1 *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 5.7 left). To create the umbrella pusher, we first created a spiral widget with 2cm 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 5.7 middle, right).

Figure 5.7: 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.

5.7.2 *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 5.8 left). When the phone case is inflated, it bend over the phone screen and block rain, strong sun-light on the phone (Figure 5.8 middle, right). It can also prevent someone from looking at private information similarly to a shape-changing phone [97].

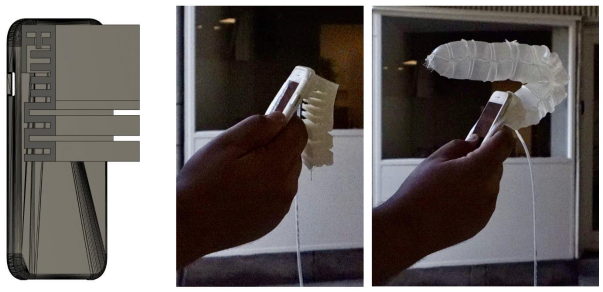


Figure 5.8: Left: The 3D model of anti-rain phone case. Middle: A user holding a 3D printed anti-rain phone case. Right: When it rains, the phone case can inflate and block rain drops over the phone.

5.7.3 *Posture-correcting cushion*

The posture-correcting cushion (Figure 5.9) 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.

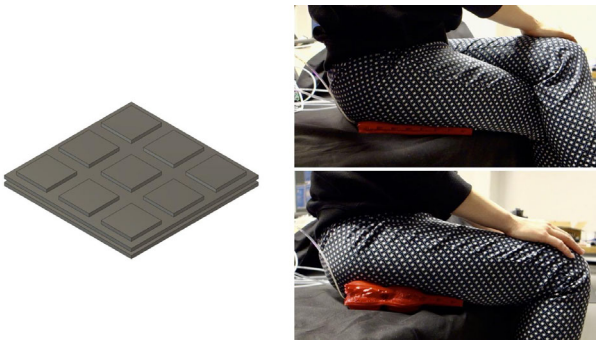


Figure 5.9: Left: The 3D model of posture-correcting cushion. Middle: A user sits on the cushion learning forward. The cushion recognizes higher pressure at the front. Right: The front cushion inflates and correct the user's posture.

5.7.4 *Physical bar chart*

The physical bar chart (Figure 5.10) was specifically designed to replicate pin-based SCIs (e.g., [23, 28, 38, 74]). 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 [28] has a 9.525 mm^2 footprint). Currently a widget has a footprint of 625 mm^2 (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.

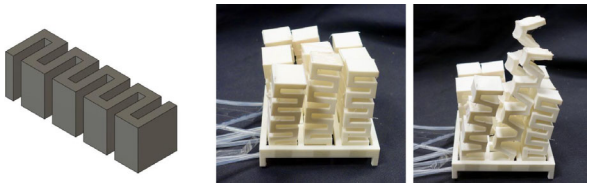


Figure 5.10: Left: A 3D model of Fold widget. Middle: We 3D printed nine copies of the 3D model and put them in a grid. Right: Some of the widgets are actuated.

5.7.5 *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 5.11 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 5.11: Left: The 3D model of the window blind. Middle: Deflated window blind. Right: Inflated window blind.

5.8 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.

5.8.1 *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 [46]. We compensated each participant with a $20 worth local product.

5.8.2 *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 [28, 135], 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 [114]. 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.

5.8.3 *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.

5.8.3.1 *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.

5.8.3.2 *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.

5.8.3.3 *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 plug-in 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.

5.8.3.4 *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 plug-in 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 5.12). They created Bump widgets and added rigid parts around them as a handle to hold a pen by squeezing one bump widget.

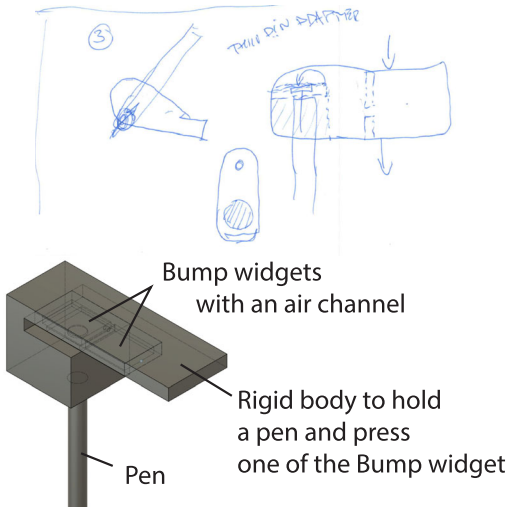


Figure 5.12: Idea sketch and its 3D model from P2. It is a pen-pressor to help people who lack fine motor skills. The two Bump widgets inside of the rigid body has an air channel between them. Users can press the rigid body to compress one of the widgets, and the air in the widget would travel to the other widget to press the top of the pen.

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.

5.8.3.5 *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 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).

5.9 DISCUSSION AND FUTURE WORK

MorpheesPlug's widgets are able to express seven of the eleven shape-changing features detailed in the Morphees+ taxonomy [53]. 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 [27], 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 shape-changing widgets using consumer-level elastic filament (Ninjaflex, shore 85A) on off-the-shelf hardware, the limited elasticity of this 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 [2] or manual [67].

5.10 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.

6

AIRLOGIC

Despite the success of AirTouch, Blowhole, and MorpheesPlug in facilitating the construction of tangible devices that can sense, and provide output to user's interactions, these techniques still require a computing device to process and identify the provided input, and to drive the output. This limitation inspired me to think about the construction of tangible devices where *everything* is encapsulated in the device: input sensing, logic processing, and output display. This upcoming chapter is the result of this exploration.

This chapter, based on unpublished work [122], introduces a toolkit for fabricating stand-alone tangible devices where all the input, logic processing, and output display are carried out in the device. In this chapter I present the set of air-powered widgets that make up the AirLogic toolkit, and the underlying technologies that enable them. Additionally, and similarly to the previously discussed techniques, I developed a design environment to enable users to embed AirLogic widgets into existing three-dimensional models. Last, this chapter closes with a set of illustrative applications, and discussions on AirLogic's performance and possible directions for future work.

The main challenge undertaken with AirLogic was the design the widgets that comprise the toolkit. Arriving at the final designs for the logic widget entailed significant iterations, because of the limited documentation on the operation of these designs.

This chapter is based on the collaborative effort described below.

Title
AirLogic: A Toolkit for 3D-printing Stand-Alone, Interactive Objects

Authors
Carlos E. Tejada, Hyunyung Kim, Mengyu Zhong, Raf Ramakers, Daniel Ashbrook

DOI
In Manuscript.

Venue
–

What was the role of the PhD student in designing the study?
The PhD student was the first author of the paper, and responsible of the described studies.

How did the PhD student participate in data collection and/or development of theory?
The PhD student was responisble for study implementation, execution, data collection, and theory development.

Which part of the manuscript did the PhD student write or contribute to?
The PhD student contributed to all parts of the manuscript.

Did the PhD student read and comment on the final manuscript?
Yes.

Abstract

The promise of on-demand fabrication of custom, interactive devices is closer to reality thanks to recent developments in 3D-printing of interactive devices. While recent work has presented novel ways to 3D-print artifacts such as speakers, electromagnetic actuators, and hydraulic robots; these efforts are non-trivial to instantiate, requiring assembly of circuits or mechanical parts. The present work introduces AirLogic: a toolkit for the creation of stand-alone, interactive objects using pneumatic widgets. Objects constructed using AirLogic, require no electronic circuits, and little to no assembly of physical components. AirLogic is comprised of a set of 12 pneumatic widgets, and a design environment, which designers can use to embed input, logic processing, and output capabilities to existing 3D models. We present an evaluation of the performance of our widgets, and a four applications that illustrate AirLogic's potential.

6.1 INTRODUCTION

In recent years, digital fabrication research has broadened its focus from fabricating 3D shapes to creating functional and interactive objects, including speakers [49], electromagnetic actuators [86], and hydraulic robots [66]. While such objects offer useful functionality, the fabrication process is often laborious, requiring that end users modify object geometry [62], assemble circuits [70], or manually insert non-printable materials [42]. We envision a future where functional devices can be printed and instantly used, without the need for intervention during printing, post-print assembly, or training of machine learning models.

As a step towards this vision, this paper presents AirLogic, a novel technique to fabricate interactive 3D-printed devices that encapsulate all input, logic, and output as an integral part of the printed structure, and which are immediately usable once printed. AirLogic accomplishes this by updating classic work in *fluerics*[1] [18], a nearly forgotten area of research that uses jets of air to perform computation without electricity or moving parts. While flueric technology was popular in the 70s it largely became obsolete with the advent of smaller, cheaper, and faster transistors. In this paper, we show how advances in additive manufacturing enable current generations of off-the-shelf fused-deposition modeling (FDM) printers to produce flueric input, output, and logic structures. In contrast to approaches requiring embedding non-printable material into 3D prints, AirLogic's flueric structures are 3D printed as part of the object itself. As such, AirLogic allows designers to prototype objects that are instantly interactive once 3D printed.

This work contributes to the existing track of research on embedding functionality in objects during the fabrication process in order to facilitate and speed-up prototyping interactive devices [48, 86, 128]. AirLogic

1 More commonly called *fluidics*; we use *fluerics* to avoid confusion with microfluidics.

is therefore a step in fulfilling the vision of fully 3D-printed interactive devices. This work contributes:

1. A set of 12 pneumatic widgets, consisting of input, output, and logic gates that can be fabricated using consumer-grade FDM 3D-printers.
2. A software plugin for Autodesk Fusion 360 that makes the pneumatic widgets available to users when designing AirLogic objects.
3. A technical evaluation to characterize the workings and performance of the pneumatic widgets.
4. A set of example applications that illustrate AirLogic's use in various scenarios.

6.2 AIRLOGIC OPERATING PRINCIPLE

AirLogic operates using pneumatics; however, unlike previous approaches which required complex fabrication techniques [42, 115, 131] or were limited to sensing input points [121, 123], AirLogic uses a single-step fabrication process and senses a variety of input events, performs simple computations based on those events, and creates output based on the computations. The key principle is that—inspired in part by fluerics—we use 3D-printed geometry to enable a continuous flow of air to act as a *power source*, allowing AirLogic to perform functions analogous to those preformed by electrical circuits. Here we briefly explain how each of AirLogic's three main parts (input, logic, output) work in the context of the sample object illustrated in Figure 6.6, A; later sections describe these components in greater detail.

After designing and printing the bunny, the user can immediately connect it to a pressurized air source using the single input channel embedded in the object. This air input is analogous to VIN or V+ in an electronic circuit. The air flows through channels and splitters fabricated in the body of the object (analogous to wires). The designer has specified two touch points on the bunny's surface. These are designed such that air vents to the atmosphere (analogous to ground) in the absence of touch. When blocked, however, the channels route the air to a flueric `OR` gate (described below). With either touch sensor covered, the air continues to flow to the oscillating actuator (very roughly analogous to a motor) embedded in the bunny's tail, which then wiggles up and down with the force of the air striking the paddle on its way to the atmosphere.

While the functioning of the input and output widgets is fairly intuitive, the operation of flueric logic gates is less so. These operate on the principle of *momentum transfer* between jets of air. The idea is that the course of an air jet can be changed by the force of another jet striking it from the side. In the case of the `OR` gate (schematically illustrated in Figure 6.2, B), a single jet proceeds directly at an angle through the central "interaction area" to the output. When both jets are present,

they collide with each other, canceling each other's angle, and form a single coherent jet that exits the output. Of particular note are the vents to either side of the interaction area in the gate. While the same gate could be simply formed as a "Y"-shaped pipe, any loading on the output (e.g. being blocked) would cause back-pressure throughout the system and impair the functionality of other components. The vents allow any such negative flow to be safely discharged to atmosphere.

6.3 RELATED WORK

Our work touches upon multiple areas of research including the fabrication of interactive objects, fluidics, and analog computing. I have previously presented an overview of the fabrication of tangible devices literature (Chapter 2), thus this section describes the current state of the fluidics, analog computing, and toolkit research.

6.3.1 *Physical User Interface Toolkits*

AirLogic contributes a technique to fabricate stand-alone interactive objects using pneumatic principles to the field of physical user interface toolkits.

Early efforts in physical user interface toolkits adopted a *prefabricated* approach, where the different components of the toolkit are fabricated by a third-party, and end users assemble, but don't usually modify, them. One of the first efforts introducing physical user interface toolkits in the HCI literature is Phidgets [33]. Its authors applied the concepts of Graphical User Interfaces (GUIs) widgets, to construct physical interaction controls using reusable components. Further iterations on this concept introduced connections between the components [10], novel form-factors [44], or more powerful components [130]. While prefabricating the different components of the toolkit reduces the design and assembly work for end users, they are limited to the component manufacturer's designs.

To allow more customization, later endeavors resort to custom fabrication of their widgets and components. These efforts provide assistance to designers to construct custom widgets. Efforts like Midas [104], Pineal [62], and PaperPulse [93] enable users to construct widgets in order to fabricate interactive objects using custom touch sensors to wrap around existing objects, using "remote widgets" on smartphones and watches, or fabricating predefined widgets using conductive inkjet printing. The main advantage of this approach is the increased flexibility it allows designers to include different types of widgets in different sizes, as needed. However, this increased flexibility comes at the cost of greater time and effort during the design process.

AirLogic draws inspiration from both types of physical user interface toolkits. We provide a set of predefined input, logic, and output widgets, which are customizable during the design stage, embedded in existing 3D models, and fabricated using commodity 3D printers. AirLogic supports

designers by providing a series of pre-validated widgets that can be re-used if desired, but that allow for enough customization to be used in a variety of applications.

6.3.2 *Non-electrical computing systems*

There is a long history in computing of non-electrical computation. The earliest computing devices, developed before the advent of electrical circuits, were mechanical: the earliest known computer, the Antikythera Mechanism (ca. 250–100 BCE) [20], was based on a complex system of gears, as was Babbage's proposed Analytical Engine (1837) [13]. Liquids were also used for pre-electronics computation of complex algebraic [21] and differential [68] equations.

Despite the modern dominance of electronic computers, researchers have continued to explore alternative computing substrates, aiming to overcome limitations imposed by a reliance on electrical circuitry. Thorsen et al. developed complex microfluidic processors [126] with application in biology and chemistry. However, fabricating such devices requires complex industrial processes, and—due to their "micro" nature—they operate at pressures and flow rates too low for actuating interactive devices [126]. Aiming at robots composed entirely of soft components, Preston et al. created flexible pneumatic logic circuits based on kinking embedded soft tubes [90]. Although able to demonstrate AND, OR, and NOT operations, fabricating the gates necessitates a complex manual molding process, and incorporation into interactive objects requires assembly of both circuits and objects.

Recently, Ion et al. demonstrated fully functional 3D-printed interactive digital devices comprised of metamaterial-based logic cells [48]. Although theoretically capable of extending logical operations through any number of gates due to per-gate energy storage via a buckling mechanism, these devices must be manually reset after each use to recover the lost energy.

6.3.3 *Fluerics*

Although many non-electrical computing systems have potential, none described in the preceding section are viable for achieving our goal of fabricating interactive devices requiring little to no user intervention in the production process. To do so, we require a computing substrate with the following properties:

- it should be able to be completely produced on a single printer with no user intervention;
- it should be capable of sensing, logic, and output;
- it should be embeddable in objects during printing;
- and it should be accessible to non-expert users.

The only available technology that meets all of these requirements is *fluerics* or *fluidic logic*, a nearly forgotten field started in the 1960s and active through the early 1980s.

The basic principle underlying fluerics is simple: a constant stream of fluid moving in one direction can be deflected by the momentum of a second, less powerful, stream applied perpendicular to the first, by an amount proportional to the strength of the deflecting jet [18, p. 64]. By creating specific geometrical arrangements of channels that shape and direct the fluid streams, a multitude of operational elements can be created [30], as shown in Figure 6.2.

Although fluerics was an active research area for nearly three decades, with widespread commercial application [25], the majority of work in the space occurred before the popularization of interactive computing in the 80s (e.g., [26]), limiting flueric interaction elements to simple buttons [25, p. 240] and one-bit displays [25, p.698]. Flueric technology was eventually eclipsed by the development of high-speed integrated electronics; today, the remaining research in fluerics largely concentrates on its potential for aerospace [24] or industrial [60] applications, leaving the potential for fluerics-based interactive devices largely unexplored.

In order to make them as small and efficient as possible, fluidic logic devices were historically produced via chemical etching or machining, achieving channel sizes as small as 0.5 mm [120]. Such processes are inaccessible to end users. AirLogic starts with basic flueric concepts and structures and updates and extends them to enable production on consumer-grade 3D printers. Using 3D printing allows us to achieve a high level of integration, incorporating flueric inputs, outputs, and logic gates directly into the structure of an object.

6.4 AIRLOGIC WIDGET TOOLKIT

To facilitate making AirLogic interactive objects, we present a widget toolkit consisting of pneumatic structures for sensing input, for providing output, and for performing basic logic operations. All widgets have compatible characteristics and thus allow for interconnecting them to realize the desired interactive behavior. Below we describe the operational details of our 13 widget designs.

6.4.1 *Input widgets*

The internal design of our input widgets are based on an inverted T-joint as shown in Figure 6.1, A. In its most basic design, pressurized air is injected on the left and will only continue its trajectory on the right side of the object when the air vent at the top is blocked.

1. *Touch.* As shown in Figure 6.1A, touch widgets use the most basic T-joint design, touching (or otherwise obstructing) the vent at the top allows the air to continue its trajectory inside the object.

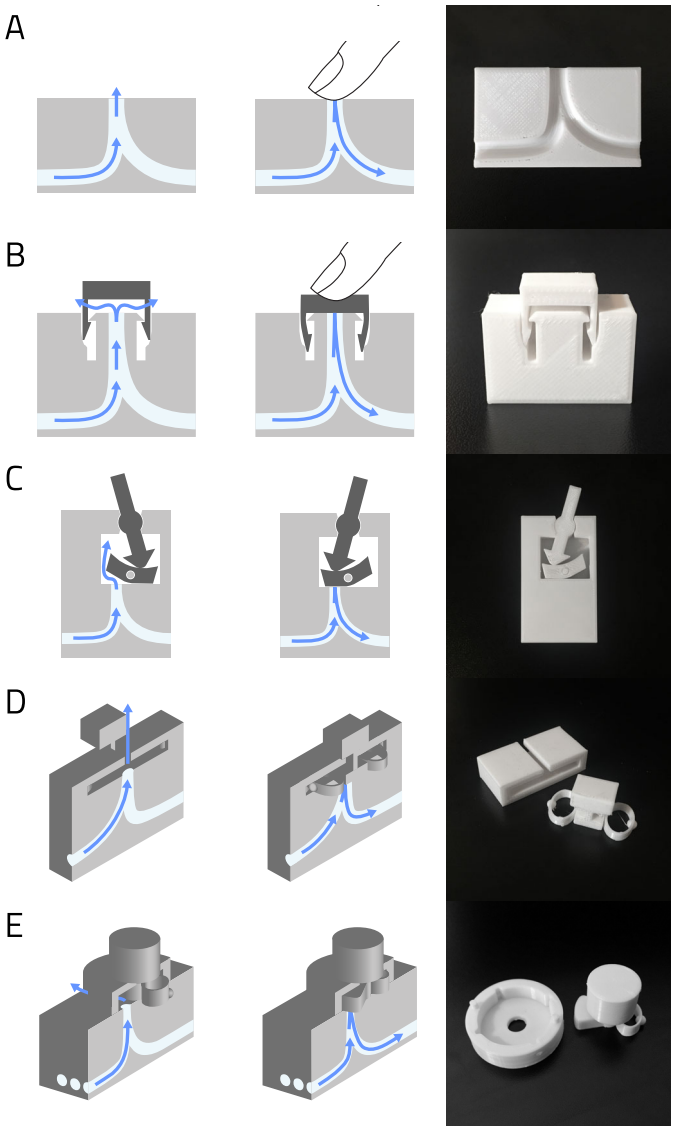


Figure 6.1: Our five input widgets. (A) Touch, (B) Button, (C), Switch, (D) Slider, (E) Dial.

2. *Push Button.* Embedding a cap and slots inside the T-joint structure realizes a push button (Figure 6.1, B). The cantilever spring design ensures the button cap always returns to its original position when released.

3. *Switch.* Figure 6.1, C shows the design of a switch. This widget design integrates a lever and a wedge on top of our basic T-joint structure. When moving the lever, the wedge covers the sensing structure, allowing the air flow to continue inside the object.

4. *Slider.* Our slider widget consists in a series of our basic sensing structure connected in parallel to the air source in a straight line. On top of them, is a rail where a handle covers the sensing structures as it slides. We added stops at each sensing location to aid the user in finding them.

5. *Dial.* Similarly to the slider, the dial widget is comprised of a set of sensing structures, arranged in a circle. Atop these structures is a channel that guides a handle in circular motions. This handle has a structure in the bottom that covers our sensing structure once on top. The dial widget also contains stops at each of the sensing locations to allow the user to better find them.

6.4.2 *Logic Widgets*

Our logic widgets were inspired by classic works in fluid logic regarding jet deflection devices. Our widgets use the momentum of interacting jets of air to change the direction of the airflow in the system. These structures lend themselves to the implementation of logic functions (e.g., `AND, OR, XOR, NOT`) for their capability to modify the output of a system based on what inputs are present.

While the majority of our input, and output widgets require moving parts in order to operate, our logic widgets are capable of computing digital logic operations by interacting jets of air (see Section 6.3.3). Our use of flueric structures to represent digital logic operations was motivated by their capability to carry out logic operations without any mechanical parts, or electronics. This capability highlights two main benefits. First, *printability.* Because there is no mechanical or electronic parts to assemble, we are able to fabricate the main parts of an AirLogic object as a single structure, requiring only the minimal assembly of external moving parts. Second, *reliability.* The lack of moving parts means that the inner workings of the object will not degrade with use, with the added bonus that AirLogic objects are particularly robust against movements and vibrations.

Below we describe the operation of our four logic widgets.

1. `AND`. We designed our `AND` logic gate widget (Figure 6.2-A,1) inspired by classic works in flueric jet deflection devices. This design presents our input controls to the left, and a single output on the middle right, with vents at the top and bottom. When only

a single input is present (Figure 6.2-A,2), the flow is directed to the respective vent channel. If both inputs are present, however, their corresponding jets collide, redirecting the flow to the output channel (Figure 6.2-A,3).

2. `OR`. Our `OR` logic gate widget (Figure 6.2-B,1) is also based on classic works in fluerics. This design is made up of two inputs channel in the left, an output channel to the right, and two vents in the top and bottom. This design operates as an "inclusive or", meaning that if any of the inputs are active (Figure 6.2-B,2), the flow is directed to the output channel. Additionally, when both inputs are active, the flow from each combines and the resulting jet is directed to the output channel.

3. `XOR`. In order to implement our exclusive or (`XOR`) logic gate widget, we make use of the same design previously discussed for our `AND` logic gate, varying the location of the output and vents. In this configuration, the output channels are located in the top, and bottom right, while the input remain unchanged from the `AND` design. When a single input is present, the air jet is directed to the respective output vents (Figure 6.2-C,1). If both inputs are present, however, their respective jets collide, redirecting the flow to the center vent (Figure 6.2-C,2).

4. `NOT`. Last, our `NOT` logic gate widget, we likewise utilize the same design previously discussed for our `AND` and `XOR` logic widgets, with some changes on the location of output and vents. When employed as a `NOT` logic gate widget, this design presents an input and a power channel on the left, and an output in the lower right with two vents above it (Figure 6.2-D,1). In this configuration the power channel provides a constant flow of air, which is directed at the output channel (Figure 6.2-D,2). If there's flow incoming from the input channel, this jet collides with the power jet, causing the flow to be redirected to a vent (Figure 6.2-D,3).

6.4.3 *Output Widgets*

We were interested in displaying output in as many modalities as possible. We developed different air powered widgets that can present acoustic, visual and vibrotactile output as the result of the logic operations carried out by our logic widgets.

1. *Pin.* Inspired by research in shape-changing interfaces [28], we designed our pin widget to provide visual output as a result of our logic operations (Figure 6.3-A). Our pin widget is comprised of a cylindrical piston inside a camber. This piston can be actuated using compressed air, reaching a height of up to 4 cm.

2. *Whistle.* Our whistle widget (Figure 6.3-B) is used to provide acoustic output as a result of a logic operation. This widget is

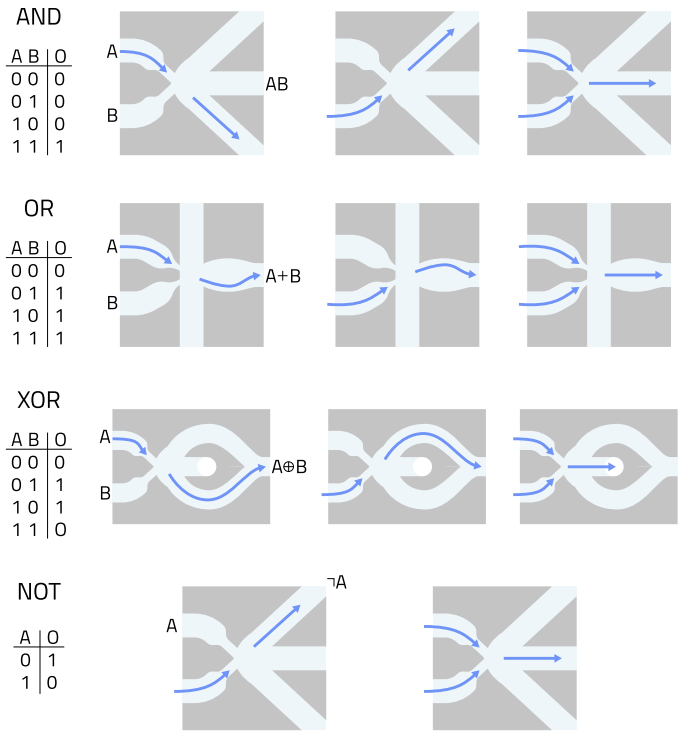


Figure 6.2: AirLogic logic widgets. (A) `AND`, (B) `OR`, (C) `XOR`, (A) `NOT`.

made up of three main components: an intake, a fipple, and a chamber. Air resulting from our logic operations enters the whistle widget through the intake, and exhausts through the fipple, making a tone while doing so. By varying the size of the internal chamber in our whistle, we can change the tone generated [43].

3. *Oscillating actuator.* Another manner to provide visual output is to agitate sections of an AirLogic object. To do so, we constructed a wiggler widget (Figure 6.3-C), comprised of a lever that is pushed by incoming jets of air. When moving, the lever shortly falls out of phase with the air jet, and returns to its original position, causing it to be pushed once more.

4. *Vibration motor.* In order to provide vibrotactile feedback on an AirLogic object, we designed a vibration motor widget (Figure 6.3-D). This widget operates similarly to electronic vibration motors commonly found on smartphones, where a mass is spun up to create different vibration patterns. In our design, air incoming from our logic widgets spins a fan-like structure, causing vibrations in the AirLogic object.

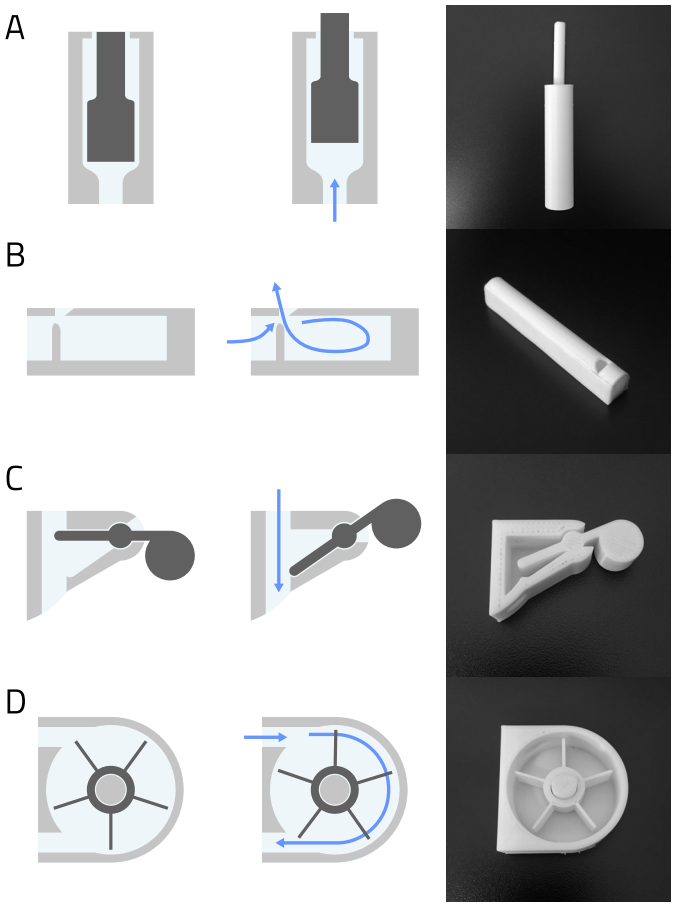


Figure 6.3: Output widgets. (A) Pin, (B) whistle, (C) oscillating actuator, (4) Vibration Motor

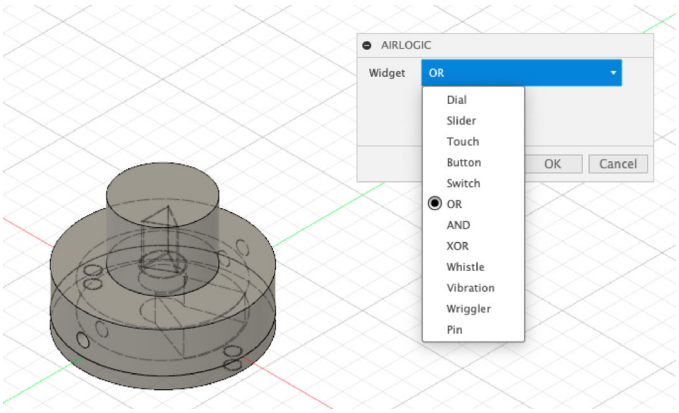


Figure 6.4: The widget library (right) and a Dial widget dropped by using the interface. In the interface, users can choose widgets and insert them into their design.

6.5 DESIGNING AIRLOGIC OBJECTS

AirLogic offers two strategies for designing interactive objects: a prototyping workflow that allows for rapid testing, and a design pipeline for embedding AirLogic widgets inside existing 3D models.

When prototyping the desired interaction, the designer can construct a transit using our encapsulated widgets. These widgets can be connected using off-the-shelf 6 mm wide plastic tubing, and powered by a constant air source. Once the design is completed, it can be translated to an existing 3D model.

Further, to aid designers augment their 3D models with AirLogic functionalities, we developed a widget library, available as a plugin for Autodesk Fusion 360, shown in Figure 6.4. Our library contains instantiated versions of all of our widgets, enabling the designer to embed input, logic processing, and output capabilities to existing 3D models in a single step. The design process starts by importing an existing geometry into the CAD software. Once loaded, the designer selects the desired input, logic and output widgets to use. Last, using an algorithm similar to PipeDream [103], we create channels connecting the widgets.

While prototyping with encapsulated widgets is a heavily manual process for the designer and involves making many connections, fabricating and assembling an AirLogic-powered 3D model with internal widgets requires little intervention. The majority of the object is printed as a single structure, involving only the assembly of moving parts (e.g., button, slider), and aesthetic covers.

6.6 IMPLEMENTATION

Our logic widget designs are comprised of three main components: two input channels, an interaction area, and output channels and vents. We systematically explored the effect of each of these components by

individually varying them while recording the barometric response of our designs. We found that using input, and output channels 2 mm, and 5 mm in width gave us the best trade-off between performance, and pressure output to further drive our widgets. In order to avoid unnecessary changes in pressure within our system, we use 5 mm channels to chain our widgets together.

In order to make our approach accessible to a broader audience, we chose to implement AirLogic using consumer-grade 3D-printing hardware. We constructed our test and prototype objects using both FDM and SLA technologies: an Ultimaker S3, a Creality Ender 3 Pro, and a Formlabs Form 2. During our tests, we discovered that the printing layers can cause unwanted turbulence when air flows through our air channels, negatively affecting the performance of our designs. We mitigate this issue by fabricating our prototypes using a minimum layer height of 0.03 mm.

In addition to focusing our efforts to enable AirLogic with consumer-grade 3D-printing equipment, we also designed AirLogic to be implemented using single-material printers. While this allow us to reach a broader audience, it came at the cost of very minimal assembly of moving parts. Multi-material printers can construct AirLogic objects as a single structure by embedding dissolvable, or breakable support materials, however AirLogic objects constructed with single material printers must have their widgets with moving parts (e.g., buttons, sliders, vibration motors) printed separately and manually assembled.

We constructed an experimental setup to test our AirLogic prototype designs. This setup was comprised of JunAir 2000-40PD air compressor, and Festo MS4-LR-1/4-D5-AS valve.

6.7 VALIDATION

To validate AirLogic and assess its practical feasibility, we empirically evaluated our widgets' capabilities to fabricate stand-alone interactive objects, and developed four illustrative applications.

6.7.1 *Technical Evaluation*

In order to understand AirLogic's capabilities to fabricate stand-alone interactive devices, we empirically evaluated our widgets. We wish to understand the barometric pressure requirements of our designs, and how they can connect to each other. We recorded the barometric pressure responses of our designs by measuring the relation between the barometric pressure provided to our designs with the barometric pressure they discharge. We repeated this measure for 5 distinct input pressures, for all of our widgets, while noting if the widgets were working as expected.

This experiment enables us to understand two main qualities of our designs. First, it allows us to understand the operating pressure ranges for our designs by identifying the pressures where our widgets perform

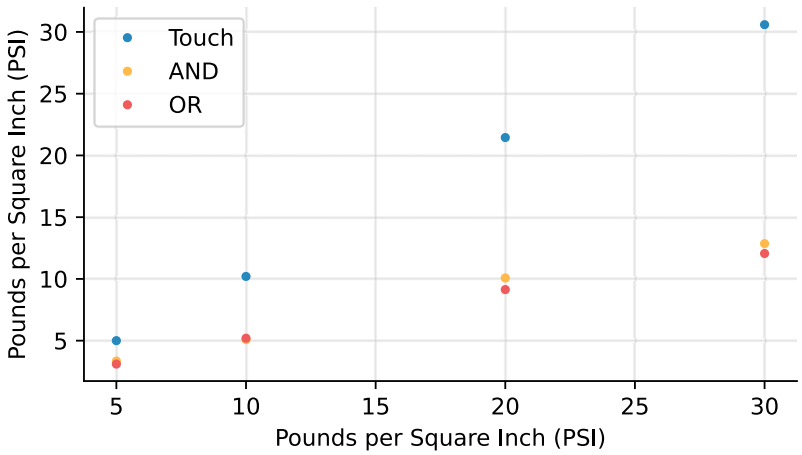


Figure 6.5: Pressure losses

bests. Second, it informs the pressure needs of our designs by establishing the losses in barometric pressures to be expected in our designs.

To carry out our evaluations we constructed an evaluating setup consisting of a JunAir 2000-40PD air compressor, a Festo MS4-LR-1/4-D5-AS valve, an analog Panasonic PS-A (ADP5) barometric sensor. We printed encapsulated versions of our widgets, and connected their output channels to our barometric pressure sensor using off-the-shelf rubber tubes.

6.7.1.1 *Results*

Our tests uncovered that the operating pressures of our widgets vary with their type. Our input widgets perform best when powered with a constant air source with pressures from 50 kilopascals (kPa), to 200 kPa. In contrast, our logic widgets presented an operating range of 50 to 300 kPa, while our output widgets operate best when actuated with pressured from 150 to 250 kPa.

In addition, we found that, in average, our logic widgets lose half of the input pressure when processing their respective operations, while our input widgets designs present a neglible pressure loss. We did not evaluate the pressure losses of our output widgets, as they are intended to be the last element in our designs. A detailed view of our results can be found in Figure 6.5.

6.7.2 *Example Applications*

In this section we present a series of applications constructed using AirLogic. These applications illustrate AirLogic's capabilities and potential for fabricating stand-alone, custom interactive objects without the use of any electronics. These applications were not constructed, but their behavior was simulated using fluid dynamics simulation software [7].

6.7.2.1 *Interactive bunny*

Using our touch, `OR`, and wiggler widgets, we constructed an interactive bunny that wiggles its tail when pet in the correct locations.

Figure 6.6: Interactive bunny

6.7.2.2 *Pitch switch*

We used our slider, button, `AND`, and whistle widgets to fabricate a pitch selector. The user selects the frequency she wants to play using a slider, and when the "play" button is pressed, the desired tone is played using the respective whistle.

6.7.2.3 *Interactive puzzle*

By modifying our button widget, we constructed an interactive puzzle using the letters U, I, S, and T. When arranged in the correct order to spell UIST, a pin with a flag attached is raised.

6.8 DISCUSSION & LIMITATIONS

6.8.1 *Chaining logic widgets*

While our applications illustrate the capacity of using multiple logic widgets in an AirLogic object (see Figure 6.7.2.2 and Section 6.7.2.3), this functionality is limited in our current implementation. During exploratory tests we found that our logic widget designs could be connected

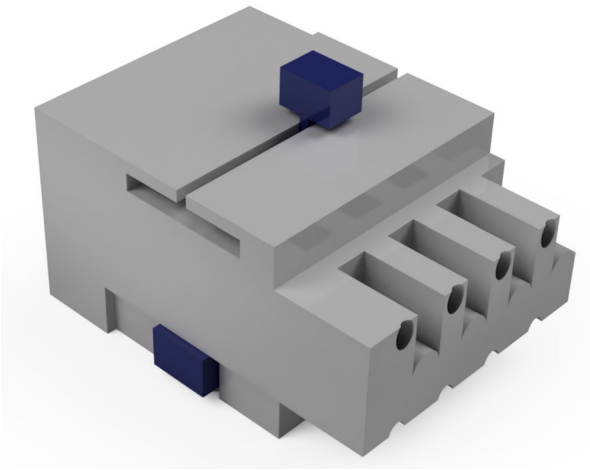


Figure 6.7: Pitch slider

three different ways: in parallel, balanced chained `AND`, and chained `OR`. We are not able to support unbalanced chained `AND`, and combinations.

The paramount reason that we can't do these things is that our `AND` design is reliant on identical pressures on both input channels in order to function correctly. If one channel has more pressure, and therefore more momentum, than the other, it pushes the jet towards the top-right channel, rather than the middle right. This means that, if an `AND` widget were to be connected after an `OR`, which can be activated using one or two inputs, we would have to dynamically regulate the pressure of the second input for our `AND` widget depending on the number of inputs used in the previous `OR` widget.

In future iterations of this work we aim to tackle this issue by standardizing the output from our logic widgets. Doing so will guarantee that the results from our logic operations will have the same pressure profile than our input widgets, no matter how many inputs are used to actuate them. To achieve this, we will explore two avenues. First we will research the use of active fluidic designs to construct our logic widgets. Our current designs are based on passive fluidic logic devices, where the the operation is performed using flow coming from the inputs. Using active designs, which contain a continuous air jet inputs interact with, will mean that the output barometric pressure for our widgets is always the same: that of the power air jet. Second, we will explore the use of fluidic amplifiers [18]. These structures increase the velocity of an incomming jet of air by switching an existing, constant jet from one outlet to another.

6.8.2 *Exploring different fabrication methods.*

During our explorations we constructed our widgets using both FDM and SLA printers. Interestingly, despite their higher resolution of 25 microns, widgets fabricated using SLA printers performed worse than those

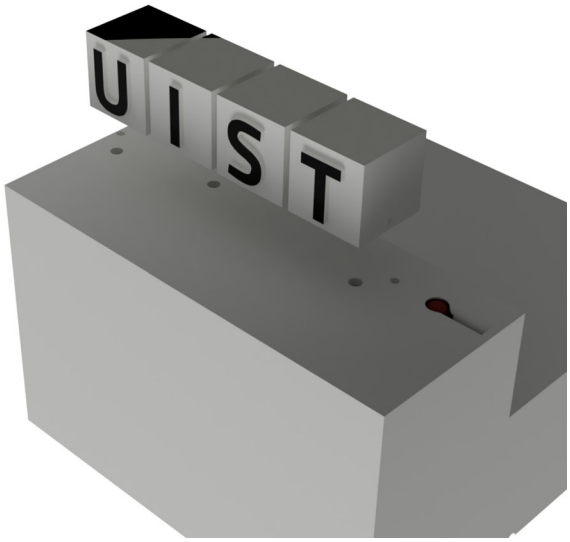


Figure 6.8: Interactive Puzzle

constructed with FDM ones. We suspect this is due to the fabrication pipeline of SLA printers where the resulting object needs to be washed before it is cured. If some resin deposits remain in our air channels after the wash, they will turn into partial blockages once cured. These blockages, given the small size of our channels, and high pressure sensitivity of our designs, can adversely affect our performance. To validate this theory, we fabricated a single `AND`, and washed it using fresh isopropyl alcohol, obtaining promising results, however, more experimentation is still needed.

Additionally, we wish to explore subtractive fabrication methods for constructing our widgets. We believe that because of its high precision and clean cuts, laser cutting equipment presents a promising alternative to fabricate AirLogic objects. While AirLogic objects constructed using laser cutting equipment would require significantly more manual assembly than our current implementation and would likely be limited to 2D, we envision a further iteration of our design environment to provide stencils and assembly guides for AirLogic designs fabricated on such machinery.

6.8.3 *Comparison with electronics*

Despite the potential interactions that AirLogic enables designers to embed into 3D-printed objects, it falls short of what can be enabled using off-the-shelf electronic toolkits such as Arduino [2]. While electronic toolkits allow for greater flexibility and variety of interactions, it comes at the cost of complex assemblies. In order to create an interactive device using an off-the-shelf electronic toolkit, the designer must posses engineering expertise in order to chose the correct components, assemble

2 https://www.arduino.cc/

them into a working circuit, and translate said circuit into a digitally fabricated enclosure. In contrast, AirLogic objects require no electronics to operate, and minimal assembly of physical parts.

In order to increase the expressivity of our technique, we will explore novel structures to represent more intricate operations such as timers, proximity, temperature, and light sensors. However, even with the inclusion of these new capabilities, we envision AirLogic objects not as a technique to replace traditional electronic components, but to complement them. Objects constructed using future iterations of AirLogic, can serve as single purpose computers, where they fulfill very specific actions, without the need of any electronics. For example, a designer can fabricate an umbrella reminder that is triggered when going for the door, and it is raining.

6.8.4 *Other air sources*

All interactive devices require a power source to operate. In the case of electronic devices is electricity, and for AirLogic devices is air. The main issue with AirLogic objects is that, while electricity can be contained inside batteries, they require a constant air source to operate. We utilized an air compressor (JunAir 2000-40PD) to power our prototypes and widgets, however other air sources can be used to power our widgets. Informal experiments showed that users can power our logic structures using their lungs, and designers can use our characterization results to calculate the specific pressure requirements of their designs, and choose their air source accordingly.

6.9 CONCLUSION

In this paper, we presented AirLogic, a physical toolkit to fabricate interactive 3D printed objects. We presented twelve pneumatic widgets that can used used as input, logic gates, and output. Those widgets allowed us to fabricate small interactive intelligent object without any electronics or coding. To enable a wide range of users such as designers, teachers, parents, and researchers, we shared the raw 3D models of the twelve widget. Further, we created a software user interface to copy, modify, and extend them via an open source platform. We believe our work brings fabrication research one step closer to make "objects that can think."

Part III

FINAL WORD

7

DISCUSSIONS & IMPLICATIONS

This thesis has presented a set of techniques that enable the construction of interactive, tangible devices without requiring any domain experience (e.g., assembly of printed parts and circuits, or calibration of machine learning models). To conclude, I discuss how each of the proposed techniques moves the state-of-the-art closer to the Print-and-Play Fabrication ideal, and highlight some paths for future work.

7.1 DISCUSSION OF PROJECTS

The main takeaway from this thesis is this: in order to realize the fabrication revolution, and enable designers to construct on-demand tangible devices, the fabrication of this devices should be as streamlined as possible. Ideally, digital fabrication equipment should abstract most, if not all, of these complications and enable designers to fabricate tangible, interactive devices that are immediately usable after fabrication. This ideal scenario is what I call *Print-and-Play Fabrication*: a fabrication paradigm where tangible devices are printed, not assembled.

In this thesis, I have presented four Print-and-Play techniques: AirLogic, Blowhole, AirLogic, MorpheesPlug; each aimed to address specific barriers in sensing, processing, and providing output to user's interactions. While this thesis has been aimed for the understanding of the general scientific community, the design and schematics of each of these projects have been open-sourced upon the publication of their respective manuscripts, enabling not only researchers but non-expert designers to experiment with the proposed techniques.

I envision that, mirroring the process of creating graphical user interfaces, a Print-and-Play Fabrication toolkit will abstract the interactive mechanisms from their respective techniques. For example, designers can specify "a touch location here", "a linear actuator there", and the system would create the desired tangible device, with the specified interaction modes. Continuing, once the tangible device has been designed successfully, the toolkit must be able to aid with the required software deployment. For tangible devices that can sense user's interactions, I envision this to be approached in two manners. For non-expert users, the Print-and-Play Fabrication toolkit can allow some form of programming by example, where designers specify locations in the model and link them to specific actions; similarly to how Blowhole's design environment operates. More expert users, on the other hand, can obtain pre-trained machine learning models to embed into their applications.

In addition to contributing to the digital fabrication community, the work I present in this thesis builds on the larger trend in computer science, specifically Human-Computer Interaction, where computing devices are imbued with domain expertise to support users. Although

previous explorations of this concept in a digital fabrication setting have been aimed at preserving user creativity and agency [137], the work discussed above aims to abstract the complexity of creating tangible devices into the fabrication process using 3D printers. By removing the complexities of constructing tangible, interactive devices and instead abstracting them into fabrication pipelines, we can enable non-expert designers and hobbyists to create on-demand tangible devices without requiring extensive domain expertise, making the fabrication revolution ideal closer to reality.

A potential development that might change the performance of the projects discussed above is the advent of new fabrication machines, or the increased accessibility of higher-end, current ones. While some of the barriers that previous endeavors face may be overcome by improvements in digital fabrication equipment (e.g., complex support material removal [61], or manual fabrication procedures [42]), others efforts present intrinsic barriers in the way they enable the construction of tangible devices. For example, in Sauron [99], Savage et al. use video cameras placed in the interior of the object to track user's interactions. Although fabrication methods will improve over time, to create Sauron-enabled objects, designers will be required to embed cameras, mirrors, and reflectors inside their fabricated objects, with all the complications this process entails. Another effort that presents similar barriers is `./trilaterate` [109], by Schmitz et al. While their fabricated objects are printed as a single structure using multi-material printers, the chosen way to identify user's interactions, trilateration, will require post-print calibration by user, and by object. This stands in contrast to the techniques included in this thesis, where the advent of new fabrication technologies will only improve their performance. For example, higher resolution printing can enable AirTouch to embed more touch-sensitive locations throughout the model, or breakthroughs in flexible materials can allow MorpheesPlug to construct a wider range of shapes without the use of internal support.

Last, in this thesis I have presented a series of projects that enable Print-and-Play Fabrication using air-powered objects, via custom internal structures. As mentioned previously, the use of air as a driving mechanism for the proposed techniques was motivated by two main factors. First, the ample study on fluid behavior, allows for the design and construction of custom interior structures to leverage various physical phenomena for constructing tangible devices that can sense, process, and provide output to user's interactions. Second, thanks to the broad understanding of fluid behavior, we can employ these concepts to construct pre-trained machine learning models, or use their respective mathematical equations, to identify user's interactions. While these principles of fluid behavior are applicable to most fluids (e.g., water, oil, other gases), the use of air permits for a "cleaner" operation. Compressed air sources are widely available, and used air can be safely discharged into the atmosphere. This does not mean, however, that only air-powered approaches can be Print-and-Play. As mentioned previously, any technique that can allow the construction of tangible, interactive devices without the need of complex post-print activities, or prohibitive fabrication pipelines is a

Print-and-Play Fabrication technique. For example, recent work from Schmitz and colleagues present a very interesting way to tackle assembly: using magnets and conductive filament [108]. The objects fabricated using this technique are constructed in two parts: an "Oh Snap! board" which houses a microcontroler with the logic, and the printed object itself.

7.2 DIRECTIONS FOR FUTURE WORK

Throughout this thesis, I have laid out possible directions future researchers can improve each project. Moving from per-project improvements, this section aims to discuss how Print-and-Play Fabrication can be improved by future explorations.

7.2.1 *Support for multi-material printing*

All through the projects that make up this thesis, I have made a conscious decision to construct tangible, interactive devices using single-material fabrication pipelines in order to increase the potential audience of each technique. Future efforts researching new Print-and-Play techniques for constructing tangible devices can explore the use of multi-material fabrication pipelines. Significant work has already been carried out by Schmitz [105] using a combination of conductive and nonconductive materials to construct tangible devices. While these efforts are not truly Print-and-Play, as they require post-print activities including carefully pouring liquids into the object, machine learning calibrations, or attaching multiple points of contact, they highlight the promising possibilities of using multi-material fabrication pipelines to construct interactive objects.

In addition to using conductive and nonconductive material to construct tangible devices, future work can explore the use of metamaterials to create new composite materials, tailored for specific functions. The use of multi-material fabrication pipelines has the potential for enriching the vocabulary of interaction modalities, while still remaining Print-and-Play. For example, the use of auxetic metamaterials can enable a variety of haptic feedback when interacting with the printed device.

7.2.2 *Support for other interaction modalities*

The work discussed above focuses on demonstrating the possibilities of constructing tangible devices with little designer intervention post-print, rather than exploring a variety of interaction modalities. An interesting direction for future work is to investigate Print-and-Play Fabrication techniques to construct tangible devices that expand on the interaction modalities presented in this thesis. I speculate future research can approach this in three main ways: techniques that provide richer touch input, techniques that provide rich output, and techniques for constructing tangibles that can sense their environments.

In order to enable richer touch interaction on tangible devices, future research on Print-and-Play Fabrication techniques should explore two avenues: deformation sensing, and multi-touch. All the sensing techniques presented in this thesis are tailored for interaction on hard tangible devices. An interesting direction for future work to explore is the implementation of Print-and-Play techniques that enable reliable deformation sensing in soft, and flexible objects. Regardless of the technique chosen to implement this interaction modality (e.g., pneumatic, acoustic, or electric sensing), I believe the main challenge in implementing this interaction modality in a Print-and-Play fashion will be the removal of per-object calibration of machine learning models. Because of each object's varying geometry, interacting with different object should yield different results. Inspired by AirTouch's success in reusing machine learning models for tangible devices of varying geometries, future work can explore the use of custom, stable interior structures for deformation (e.g., press, squeeze) sensing.

Continuing, I have presented two techniques for the construction of efforts that are able to respond to touch interactions: AirTouch and AirLogic. While successful, these efforts can only reliably sense single touch locations. I have laid out guidelines on how to pneumatically sense more than one location in Chapter 3, but this approach limits the number of interaction locations if a multi-touch setup is desired. Future explorations can investigate other Print-and-Play-friendly techniques for constructing tangible devices capable of sensing touch interaction in various locations.

Another direction future work on Print-and-Play Fabrication should explore is providing richer output to user's interactions. With AirLogic I explored ways to physicalize the output of logical operations. These outputs can also be expanded by future work. For example, AirLogic presents a vibrotactile motor to provide haptic feedback to its users; future work can explore other haptic interaction modalities as output, like temperature. Similarly, AirLogic presents acoustic feedback to its users using whistles. Future work can explore Print-and-Play ways to construct speakers, similar to those presented in [49], and provide richer acoustic output. Continuing, while with AirLogic and MorpheesPlug I explored different ways to provide visual output to users, future work can enrich this output modality by investigating in novel ways to embed displays into fabricated objects. Similarly how [129] employs techniques closely related to AirLogic to develop a seven-segment display without requiring electronics or moving parts, future work can investigate novel, Print-and-Play-friendly techniques to incorporate rich visual output in tangible devices.

Last, an interesting avenue to explore is the construction of tangible devices that not only respond to user's interactions, but to changes in the environment they sit in. The inclusion of "smart" materials can aid in the creation of such devices. The use of materials which properties change depending on their environment can enable the easy construction of "tangible sensors". These sensors can respond to environmental temperature, humidity, or sound, and act accordingly.

7.2.3 *Support for other fabrication equipment*

In this thesis I have presented four Print-and-Play ways to construct tangible, interactive devices using 3D printers. This does not mean, however, that Print-and-Play devices can only be fabricated using 3D printers. The main benefit of using 3D printers instead of other fabrication devices is their capability for constructing three-dimensional shapes without the need user intervention; whereas using other fabrication devices require manual assembly post-fabrication. A possible research avenue to construct tangible devices using laser cutters and CNC routers is to not only use these devices for the fabrication of the object, but for the assembly tasks as well.

Despite early explorations in this concept have yielded promising results [52, 80], they remain far from the Print-and-Play Fabrication ideal: [52] is not able to construct tangible devices, but more objects that can be mechanically actuated, and [80] requires intricate calibration of the laser cutter mid-print to solder conductive tracings, and combine previously cut parts. Future endeavors expanding this concept can explore the use of other materials, e.g., conductive filaments, to simplify the connection between electronics.

8

CONCLUSIONS

In closing, in this thesis I have introduced the concept of Print-and-Play Fabrication: a fabrication paradigm where tangible, interactive devices are fabricated instead of assembled. To explore different ways we can enable Print-and-Play Fabrication, I have introduced four techniques: AirTouch, Blowhole, AirLogic, MorpheesPlug; each of these tailored to tackle a specific facet of tangible devices. With AirTouch and Blowhole, I explore novel ways to provide input to tangible devices. Continuing, AirLogic investigates interesting ways to represent encapsulate logic in tangible devices, without requiring custom electronics. Last, MorpheesPlug researches novel ways of constructing devices that can change in shape, and provide physical output to input and computations.

The work of this thesis illustrates that a promising approach for enabling Print-and-Play Fabrication is to embed custom interior structures in three-dimensional models to leverage fluid, in this case air, behavior to enable sensing, logic processing, and output display. I have demonstrated that using this concept we can construct tangible, interactive devices that can sense, process, and display output using consumer-grade 3D printers that are immediately usable after fabrication.

The suite of tools I present in this thesis, the techniques used to enable them, and the space they explore, will hopefully empower designers to construct tangible, interactive devices using 3D printers, and inspire researchers to continue exploring different ways to enable Print-and-Play Fabrication.

BIBLIOGRAPHY

[1] *3D Printing Market Size, Share: Industry Report, 2021-2028*. 2020. URL: https://www.grandviewresearch.com/industry-analysis/3d-printing-industry-analysis.

[2] Parastoo Abtahi and Sean Follmer. "Visuo-Haptic Illusions for Improving the Perceived Performance of Shape Displays." In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI '18: CHI Conference on Human Factors in Computing Systems. Montreal QC Canada: ACM, Apr. 19, 2018, pp. 1–13. ISBN: 978-1-4503-5620-6. DOI: 10.1145/3173574.3173724. URL: https://dl.acm.org/doi/10.1145/3173574.3173724 (visited on 05/25/2021).

[3] Jason Alexander, Anne Roudaut, Jürgen Steimle, Kasper Hornbæk, Miguel Bruns Alonso, Sean Follmer, and Timothy Merritt. "Grand Challenges in Shape-Changing Interface Research." In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. New York, New York, USA: ACM Press, 2018, pp. 1–14. ISBN: 978-1-4503-5620-6. DOI: 10.1145/3173574.3173873. URL: http://dl.acm.org/citation.cfm?doid=3173574.3173873.

[4] M Alster. "Improved calculation of resonant frequencies of Helmholtz resonators." In: *Journal of Sound and Vibration* 24.1 (Sept. 1972), pp. 63–85.

[5] Chris Anderson. *Makers: The New Industrial Revolution*. Random House, 2012.

[6] *Arduino : an open-source electronics prototyping platform*. URL: https://www.arduino.cc/.

[7] *Autodesk CFD: Computational Fluid Dynamics Simulation Software*. URL: https://www.autodesk.com/products/cfd/overview.

[8] Rafael Ballagas, Sarthak Ghosh, and James Landay. "The Design Space of 3D Printable Interactivity." In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2.2 (July 2018), pp. 1–21. DOI: 10.1145/3214264. URL: http://dl.acm.org/citation.cfm?doid=3236498.3214264.

[9] Ayah Bdeir. "Electronics as Material: littleBits." In: *Proceedings of the Third International Conference on Tangible and Embedded Interaction (TEI'09)*. New York, New York, USA: ACM, Feb. 2009, pp. 397–400. ISBN: 978-1-60558-493-5. DOI: 10.1145/1517664.1517743.

[10] Ayah Bdeir. "Electronics as Material: littleBits." In: *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction*. TEI '09. New York, NY, USA: Association for Computing Machinery, Feb. 16, 2009, pp. 397–400. ISBN: 978-1-60558-493-5. DOI: 10.1145/1517664.1517743. URL: https://doi.org/10.1145/1517664.1517743.

[11] Daniel Bernoulli. *Hydrodynamica: sive de viribus et motibus fluidorum commentarii*. Johannis Reinholdi Dulseckeri, 1738.

[12] Eric Brockmeyer, Ivan Poupyrev, and Scott Hudson. "PAPILLON." In: *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*. New York, New York, USA: ACM Press, 2013, pp. 457–462. ISBN: 978-1-4503-2268-3. DOI: 10.1145/2501988.2502027. URL: http://dl.acm.org/citation.cfm?doid=2501988.2502027.

[13] A. G. Bromley. "Charles Babbage's Analytical Engine, 1838." In: *IEEE Annals of the History of Computing* 20.4 (Oct. 1998), pp. 29–45. ISSN: 1934-1547. DOI: 10.1109/85.728228.

[14] Glenn O Brown. "The History of the Darcy-Weisbach Equation for Pipe Flow Resistance." In: *Environmental and Water Resources History Sessions at ASCE Civil Engineering Conference and Exposition 2002*. Washington, D.C., United States: American Society of Civil Engineers, Nov. 2002.

[15] Moritz Bächer, Benjamin Hepp, Fabrizio Pece, Paul G Kry, Bernd Bickel, Bernhard Thomaszewski, and Otmar Hilliges. "DefSense." In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. New York, New York, USA: ACM Press, 2016, pp. 3806–3816. ISBN: 978-1-4503-3362-7. DOI: 10.1145/2858036.2858354. URL: http://dl.acm.org/citation.cfm?doid=2858036.2858354.

[16] Federico Carpi, Claudio Salaris, and Danilo De Rossi. "Folded dielectric elastomer actuators." In: *Smart Materials and Structures* 16.2 (2007), S300–S305. DOI: 10.1088/0964-1726/16/2/s15. URL: https://doi.org/10.1088%2F0964-1726%2F16%2F2%2Fs15.

[17] Géry Casiez, Nicolas Roussel, and Daniel Vogel. "1 € Filter." In: *Proceedings of the 2012 CHI Conference on Human Factors in Computing Systems*. New York, New York, USA: ACM, May 2012, pp. 2527–2530. ISBN: 978-1-4503-1015-4. DOI: 10.1145/2207676.2208639. URL: http://dl.acm.org/citation.cfm?doid=2207676.2208639.

[18] Charles Belsterling. *Fluidic Systems Design*. 1971. URL: http://archive.org/details/BELSTERLINGFluidicSystemsDesign1971.

[19] Marcelo Coelho and Pattie Maes. "Shutters: A Permeable Surface for Environmental Control and Communication." In: *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction - TEI '09*. The 3rd International Conference.

Cambridge, United Kingdom: ACM Press, 2009, p. 13. ISBN: 978-1-60558-493-5. DOI: 10.1145/1517664.1517671. URL: http://portal.acm.org/citation.cfm?doid=1517664.1517671 (visited on 05/25/2021).

[20] Kyriakos Efstathiou and Marianna Efstathiou. "Celestial Gearbox." In: *Mechanical Engineering* 140.09 (Sept. 1, 2018), pp. 31–35. ISSN: 0025-6501. DOI: 10.1115/1.2018-SEP1. URL: https://doi.org/10.1115/1.2018-SEP1.

[21] DR Arnold Emch. "Two Hydraulic Methods to Extract the Nth Root of Any Number." In: *The American Mathematical Monthly* 8.1 (Jan. 1, 1901), pp. 10–12. ISSN: 0002-9890. DOI: 10.1080/00029890.1901.12000520. URL: https://doi.org/10.1080/00029890.1901.12000520.

[22] Aluna Everitt and Jason Alexander. "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*. 2017, pp. 1283–1294.

[23] Aluna Everitt, Faisal Taher, and Jason Alexander. "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*. CHI'16: CHI Conference on Human Factors in Computing Systems. San Jose California USA: ACM, May 7, 2016, pp. 2778–2782. ISBN: 978-1-4503-3362-7. DOI: 10.1145/2858036.2858316. URL: https://dl.acm.org/doi/10.1145/2858036.2858316 (visited on 05/25/2021).

[24] Michele Ferlauto and Roberto Marsilio. "Numerical Investigation of the Dynamic Characteristics of a Dual-Throat-Nozzle for Fluidic Thrust-Vectoring." In: *AIAA Journal* 55.1 (2017), pp. 86–98. ISSN: 0001-1452. DOI: 10.2514/1.J055044. URL: https://doi.org/10.2514/1.J055044.

[25] *Fluidic Components and Equipment 1968–9*. Elsevier, 1968. ISBN: 978-0-08-013446-8. DOI: 10.1016/C2013-0-02239-6. URL: https://linkinghub.elsevier.com/retrieve/pii/C20130022396.

[26] J. D. Foley, V. L. Wallace, and P. Chan. "The Human Factors of Computer Graphics Interaction Techniques." In: *IEEE Computer Graphics and Applications* 4.11 (Nov. 1984), pp. 13–48. ISSN: 1558-1756. DOI: 10.1109/MCG.1984.6429355.

[27] Sean Follmer, Daniel Leithinger, Alex Olwal, Nadia Cheng, and Hiroshi Ishii. "Jamming User Interfaces." In: *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*. New York, New York, USA: ACM, Oct. 2012, pp. 519–528. ISBN: 978-1-4503-1580-7. DOI: 10.1145/2380116.2380181. URL: http://dl.acm.org/citation.cfm?doid=2380116.2380181.

[28] Sean Follmer, Daniel Leithinger, Alex Olwal, Akimitsu Hogge, and Hiroshi Ishii. "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*. UIST '13. New York, NY, USA: Association for Computing Machinery, Oct. 8, 2013, pp. 417–426. ISBN: 978-1-4503-2268-3. DOI: 10.1145/2501988.2502032. URL: https://doi.org/10.1145/2501988.2502032.

[29] Neil A Gershenfeld. *Fab: The Coming Revolution on Your Desktop–from Personal Computers to Personal Fabrication*. Basic Books (AZ), 2005.

[30] H. H. Glaettli. "Digital Fluid Logic Elements." In: *Advances in Computers*. Ed. by Franz L. Alt and Morris Rubinoff. Vol. 4. Elsevier, Jan. 1, 1964, pp. 169–243. DOI: 10.1016/S0065-2458(08)60221-1. URL: http://www.sciencedirect.com/science/article/pii/S0065245808602211.

[31] Kristian Gohlke, Eva Hornecker, and Wolfgang Sattler. "Pneumatibles." In: *Proceedings of the 10th International Conference on Tangible, Embedded, and Embodied Interaction*. New York, New York, USA: ACM Press, 2016, pp. 308–315. ISBN: 978-1-4503-3582-9. DOI: 10.1145/2839462.2839489. URL: http://dl.acm.org/citation.cfm?doid=2839462.2839489.

[32] Saul Greenberg. "Toolkits and interface creativity." In: *Multimedia Tools and Applications* 32.2 (2007), pp. 139–159. ISSN: 1573-7721. DOI: 10.1007/s11042-006-0062-y. URL: https://doi.org/10.1007/s11042-006-0062-y.

[33] Saul Greenberg and Chester Fitchett. "Phidgets." In: *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology*. New York, New York, USA: ACM, Nov. 2001, pp. 209–218. ISBN: 1-58113-438-X. DOI: 10.1145/502348.502388. URL: http://portal.acm.org/citation.cfm?doid=502348.502388.

[34] Joseph N Grima and Kenneth E Evans. "Auxetic behavior from rotating triangles." In: *Journal of materials science* 41.10 (2006), pp. 3193–3196.

[35] Daniel Groeger, Elena Chong Loo, and Jürgen Steimle. "HotFlex: Post-Print Customization of 3D Prints Using Embedded State Change." In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. CHI'16: CHI Conference on Human Factors in Computing Systems. San Jose California USA: ACM, May 7, 2016, pp. 420–432. ISBN: 978-1-4503-3362-7. DOI: 10.1145/2858036.2858191. URL: http://dl.acm.org/doi/10.1145/2858036.2858191.

[36] Erik Grönvall, Sofie Kinch, Marianne Graves Petersen, and Majken K. Rasmussen. "Causing Commotion with a Shape-Changing Bench: Experiencing Shape-Changing Interfaces in Use." In: *Pro-*

ceedings of the SIGCHI Conference on Human Factors in Computing Systems. CHI '14: CHI Conference on Human Factors in Computing Systems. Toronto Ontario Canada: ACM, Apr. 26, 2014, pp. 2559–2568. ISBN: 978-1-4503-2473-1. DOI: 10.1145/2556288.2557360. URL: https://dl.acm.org/doi/10.1145/2556288.2557360 (visited on 05/25/2021).

[37] Timo Götzelmann and Christopher Althaus. "TouchSurfaceModels." In: *The 9th ACM International Conference*. New York, New York, USA: ACM Press, 2016, pp. 1–8. ISBN: 978-1-4503-4337-4. DOI: 10.1145/2910674.2910690. URL: http://dl.acm.org/citation.cfm?doid=2910674.2910690.

[38] John Hardy, Christian Weichel, Faisal Taher, John Vidler, and Jason Alexander. "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*. New York, New York, USA: ACM Press, 2015, pp. 19–28. ISBN: 978-1-4503-3145-6. DOI: 10.1145/2702123.2702599.

[39] Chris Harrison and Scott E Hudson. "Providing Dynamically Changeable Physical Buttons on a Visual Display." In: *Proceedings of the 2009 CHI Conference on Human Factors in Computing Systems*. New York, New York, USA: ACM, Apr. 2009, pp. 299–308. ISBN: 978-1-60558-246-7. DOI: 10.1145/1518701.1518749. URL: http://dl.acm.org/citation.cfm?doid=1518701.1518749.

[40] Chris Harrison, Robert Xiao, and Scott E Hudson. "Acoustic Barcodes: Passive, Durable and Inexpensive Notched Identification Tags." In: *UIST '12: Proceedings of the 25th annual ACM symposium on User interface software and technology*. New York, New York, USA: ACM Press, 2012, pp. 563–567.

[41] Björn Hartmann, Leith Abdulla, Manas Mittal, and Scott R. Klemmer. "Authoring Sensor-Based Interactions by Demonstration with Direct Manipulation and Pattern Recognition." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '07. San Jose, California, USA: Association for Computing Machinery, Apr. 29, 2007, pp. 145–154. ISBN: 978-1-59593-593-9. DOI: 10.1145/1240624.1240646. URL: https://doi.org/10.1145/1240624.1240646.

[42] Liang He, Gierad Laput, Eric Brockmeyer, and Jon E Froehlich. "SqueezaPulse." In: *Proceedings of the 11th International Conference on Tangible, Embedded, and Embodied Interaction*. New York, New York, USA: ACM Press, 2017, pp. 341–350. ISBN: 978-1-4503-4676-4. DOI: 10.1145/3024969.3024976. URL: http://dl.acm.org/citation.cfm?doid=3024969.3024976.

[43] Herman L F Helmholtz. *On the Sensations of Tone as a Physiological Basis for the Theory of Music*. 2nd ed. London, United Kingdom: Longmans, Green, and Co., 1885.

[44] Steve Hodges, Nicolas Villar, Nicholas Chen, Tushar Chugh, Jie Qi, Diana Nowacka, and Yoshihiro Kawahara. "Circuit Stickers: Peel-and-Stick Construction of Interactive Electronic Prototypes." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '14. New York, NY, USA: Association for Computing Machinery, Apr. 26, 2014, pp. 1743–1746. ISBN: 978-1-4503-2473-1. DOI: 10.1145/2556288.2557150. URL: https://doi.org/10.1145/2556288.2557150.

[45] Jonathan Hook, Peter Wright, Thomas Nappey, Steve Hodges, and Patrick Olivier. "Making 3D printed objects interactive using wireless accelerometers." In: *CHI '14 Extended Abstracts on Human Factors in Computing Systems*. ACM, Apr. 2014, pp. 1435–1440.

[46] Larry L Howell. "Compliant mechanisms." In: *21st century kinematics*. Springer, 2013, pp. 189–216.

[47] Charles Hudin, Sabrina Panëels, and Steven Strachan. "INTACT." In: *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. New York, New York, USA: ACM Press, 2016, pp. 2719–2725. ISBN: 978-1-4503-4082-3. DOI: 10.1145/2851581.2892351. URL: http://dl.acm.org/citation.cfm?doid=2851581.2892351.

[48] Alexandra Ion, Ludwig Wall, Robert Kovacs, and Patrick Baudisch. "Digital Mechanical Metamaterials." In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. Denver, CO, USA: ACM Press, 2017, pp. 977–988. ISBN: 978-1-4503-4655-9. DOI: 10.1145/3025453.3025624. URL: http://dl.acm.org/citation.cfm?doid=3025453.3025624.

[49] Yoshio Ishiguro and Ivan Poupyrev. "3D Printed Interactive Speakers." In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. 2014, pp. 1733–1742.

[50] Vikram Iyer, Justin Chan, Ian Culhane, Jennifer Mankoff, and Shyamnath Gollakota. "Wireless Analytics for 3D Printed Objects." In: *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*. New York, New York, USA: ACM Press, 2018, pp. 141–152. ISBN: 978-1-4503-5948-1. DOI: 10.1145/3242587.3242639. URL: http://dl.acm.org/citation.cfm?doid=3242587.3242639.

[51] Yvonne Jansen, Pierre Dragicevic, Petra Isenberg, Jason Alexander, Abhijit Karnik, Johan Kildal, Sriram Subramanian, and Kasper Hornbæk. "Opportunities and Challenges for Data Physicalization." In: *CHI '15: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. New York, New York, USA: ACM, Apr. 2015, pp. 3227–3236.

[52] Shohei Katakura, Yuto Kuroki, and Keita Watanabe. "A 3D Printer Head as a Robotic Manipulator." In: *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and*

Technology. New Orleans, LA, USA: ACM Press, 2019, pp. 535–548. ISBN: 978-1-4503-6816-2. DOI: 10.1145/3332165.3347885. URL: http://dl.acm.org/citation.cfm?doid=3332165.3347885.

[53] Hyunyoung Kim, Celine Coutrix, and Anne Roudaut. "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*. CHI '18. Montreal QC, Canada: Association for Computing Machinery, 2018. ISBN: 9781450356206. DOI: 10.1145/3173574.3174193. URL: https://doi.org/10.1145/3173574.3174193.

[54] Hyunyoung Kim, Céline Coutrix, and Anne Roudaut. "KnobSlider: Design of a Shape-Changing Parameter Control UI and Study of User Preferences on Its Speed and Tangibility." In: *Frontiers in Robotics and AI* 6 (2019), p. 79. ISSN: 2296-9144. DOI: 10.3389/frobt.2019.00079. URL: https://www.frontiersin.org/article/10.3389/frobt.2019.00079.

[55] Hyunyoung Kim, Aluna Everitt, Carlos Tejada, Mengyu Zhong, and Daniel Ashbrook. "MorpheesPlug: A Toolkit for Prototyping Shape-Changing Interfaces." In: *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. CHI '21: CHI Conference on Human Factors in Computing Systems. Yokohama Japan: ACM, May 6, 2021, pp. 1–13. ISBN: 978-1-4503-8096-6. DOI: 10.1145/3411764.3445786. URL: https://dl.acm.org/doi/10.1145/3411764.3445786 (visited on 05/24/2021).

[56] Jeeeun Kim and Tom Yeh. "Toward 3D-Printed Movable Tactile Pictures for Children with Visual Impairments." In: *CHI '15: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. New York, New York, USA: ACM, Apr. 2015, pp. 2815–2824.

[57] Seoktae Kim, Hyunjung Kim, Boram Lee, Tek-Jin Nam, and Woohun Lee. "Inflatable Mouse: Volume-Adjustable Mouse with Air-Pressure-Sensitive Input and Haptic Feedback." In: *Proceeding of the Twenty-Sixth Annual CHI Conference on Human Factors in Computing Systems - CHI '08*. Proceeding of the Twenty-Sixth Annual CHI Conference. Florence, Italy: ACM Press, 2008, p. 211. ISBN: 978-1-60558-011-1. DOI: 10.1145/1357054.1357090. URL: http://portal.acm.org/citation.cfm?doid=1357054.1357090.

[58] Scott R. Klemmer, Björn Hartmann, and Leila Takayama. "How Bodies Matter: Five Themes for Interaction Design." In: *Proceedings of the 6th ACM Conference on Designing Interactive Systems - DIS '06*. The 6th ACM Conference. University Park, PA, USA: ACM Press, 2006, p. 140. ISBN: 978-1-59593-367-6. DOI: 10.1145/1142405.1142429. URL: http://portal.acm.org/citation.cfm?doid=1142405.1142429.

[59] Yong Lin Kong, Ian A Tamargo, Hyoungsoo Kim, Blake N Johnson, Maneesh K Gupta, Tae-Wook Koh, Huai-An Chin, Daniel A Steingart, Barry P Rand, and Michael C McAlpine. "3D Printed Quantum Dot Light-Emitting Diodes." In: *Nano letters* 14.12 (2014), pp. 7017–7023.

[60] Arnaud Lacarelle and Christian O. Paschereit. "Increasing the Passive Scalar Mixing Quality of Jets in Crossflow With Fluidics Actuators." In: *Journal of Engineering for Gas Turbines and Power* 134.021503 (Dec. 20, 2011). ISSN: 0742-4795. DOI: 10.1115/1.4004373. URL: https://doi.org/10.1115/1.4004373.

[61] Gierad Laput, Eric Brockmeyer, Scott E Hudson, and Chris Harrison. "Acoustruments." In: *Proceedings of the 2015 CHI Conference on Human Factors in Computing Systems*. 2015. DOI: 10.1145/2702123.2702416. URL: https://dl.acm.org/citation.cfm?id=2702414.

[62] David Ledo, Fraser Anderson, Ryan Schmidt, Lora Oehlberg, Saul Greenberg, and Tovi Grossman. "Pineal." In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. Denver, CO, USA: ACM Press, 2017, pp. 2583–2593. ISBN: 978-1-4503-4655-9. DOI: 10.1145/3025453.3025652. URL: http://dl.acm.org/citation.cfm?doid=3025453.3025652.

[63] David Ledo, Steven Houben, Jo Vermeulen, Nicolai Marquardt, Lora Oehlberg, and Saul Greenberg. "Evaluation Strategies for HCI Toolkit Research." In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. New York, New York, USA: ACM Press, 2018, pp. 1–17. ISBN: 978-1-4503-5620-6. DOI: 10.1145/3173574.3173610. URL: http://dl.acm.org/citation.cfm?doid=3173574.3173610.

[64] DoYoung Lee, Youryang Lee, Yonghwan Shin, and Ian Oakley. "Designing Socially Acceptable Hand-to-Face Input." In: *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*. New York, New York, USA: ACM Press, 2018, pp. 711–723. ISBN: 978-1-4503-5948-1. DOI: 10.1145/3242587.3242642. URL: http://dl.acm.org/citation.cfm?doid=3242587.3242642.

[65] Dingzeyu Li, David I W Levin, Wojciech Matusik, and Changxi Zheng. "Acoustic Voxels." In: *ACM Transactions on Graphics* 35.4 (July 2016), pp. 1–12. DOI: 10.1145/2897824.2925960. URL: http://dl.acm.org/citation.cfm?doid=2897824.2925960.

[66] Robert MacCurdy, Robert Katzschmann, Youbin Kim, and Daniela Rus. "Printable Hydraulics: A Method for Fabricating Robots by 3D Co-Printing Solids and Liquids." In: 2016 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2016, pp. 3878–3885.

[67] John C. McClelland, Robert J. Teather, and Audrey Girouard. "Haptobend: Shape-Changing Passive Haptic Feedback in Virtual Reality." In: *Proceedings of the 5th Symposium on Spatial User Interaction*. SUI '17: Symposium on Spatial User Interaction. Brighton United Kingdom: ACM, Oct. 16, 2017, pp. 82–90. ISBN: 978-1-4503-5486-8. DOI: 10.1145/3131277.3132179. URL: https://dl.acm.org/doi/10.1145/3131277.3132179 (visited on 05/25/2021).

[68] A. D. Moore. "The Hydrocal." In: *Industrial & Engineering Chemistry* 28.6 (June 1, 1936), pp. 704–708. ISSN: 0019-7866. DOI: 10.1021/ie50318a022. URL: https://doi.org/10.1021/ie50318a022.

[69] Hedieh Moradi and César Torres. "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. New York, NY, USA: Association for Computing Machinery, 2020, pp. 1995–2006. ISBN: 978-1-4503-6974-9. DOI: 10.1145/3357236.3395473. URL: https://doi.org/10.1145/3357236.3395473.

[70] Roderick Murray-Smith, John Williamson, Stephen Hughes, and Torben Quaade. "Stane: Synthesized Surfaces for Tactile Input." In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. 2008, pp. 1299–1302.

[71] Brad A Myers, Scott E Hudson, and Randy Pausch. "Past, Present, and Future of User Interface Software Tools." In: *ACM Transactions on Computer-Human Interaction (TOCHI* 7.1 (Mar. 2000), pp. 3–28. DOI: 10.1145/344949.344959.

[72] Ken Nakagaki, Sean Follmer, and Hiroshi Ishii. "LineFORM: Actuated Curve Interfaces for Display, Interaction, and Constraint." In: *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. UIST '15: The 28th Annual ACM Symposium on User Interface Software and Technology. Charlotte NC USA: ACM, Nov. 5, 2015, pp. 333–339. ISBN: 978-1-4503-3779-3. DOI: 10.1145/2807442.2807452. URL: https://dl.acm.org/doi/10.1145/2807442.2807452 (visited on 05/25/2021).

[73] Ken Nakagaki, Luke Vink, Jared Counts, Daniel Windham, Daniel Leithinger, Sean Follmer, and Hiroshi Ishii. "Materiable: Rendering Dynamic Material Properties in Response to Direct Physical Touch with Shape Changing Interfaces." In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. CHI'16: CHI Conference on Human Factors in Computing Systems. San Jose California USA: ACM, May 7, 2016, pp. 2764–2772. ISBN: 978-1-4503-3362-7. DOI: 10.1145/2858036.2858104. URL: http://dl.acm.org/doi/10.1145/2858036.2858104.

[74] Ken Nakagaki, Luke Vink, Jared Counts, Daniel Windham, Daniel Leithinger, Sean Follmer, and Hiroshi Ishii. "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*. New York, New York, USA: ACM, May 2016, pp. 2764–2772. ISBN: 978-1-4503-3362-7. DOI: 10.1145/2858036.2858104.

[75] Kosuke Nakajima, Yuichi Itoh, Yusuke Hayashi, Kazuaki Ikeda, Kazuyuki Fujita, and Takao Onoye. "Emoballoon." In: *Advances in Computer Entertainment*. Ed. by Dennis Reidsma, Haruhiro Katayose, and Anton Nijholt. Cham: Springer International Publishing, 2013, pp. 182–197. ISBN: 978-3-319-03161-3.

[76] Thomas Nappey, Steve Hodges, Peter Wright, Jonathan Hook, Steve Hodges, and Patrick Olivier. "Making 3D Printed Objects Interactive Using Wireless Accelerometers." In: *Proceedings of the 2014 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. New York, New York, USA: ACM, Apr. 2014, pp. 1435–1440. ISBN: 978-1-4503-2474-8. DOI: 10.1145/2559206.2581137. URL: http://dl.acm.org/citation.cfm?doid=2559206.2581137.

[77] Kristin Neidlinger, Khiet P Truong, Caty Telfair, Loe Feijs, Edwin Dertien, and Vanessa Evers. "AWElectric: That Gave Me Goosebumps, Did You Feel It Too?" In: Proceedings of the Eleventh International Conference on Tangible, Embedded, and Embodied Interaction. 2017, pp. 315–324.

[78] Ryuma Niiyama, Xu Sun, Cynthia Sung, Byoungkwon An, Daniela Rus, and Sangbae Kim. "Pouch Motors." In: *Soft Robotics* 2.2 (June 2015), pp. 59–70. DOI: 10.1089/soro.2014.0023. URL: https://www.liebertpub.com/doi/10.1089/soro.2014.0023.

[79] Ryuma Niiyama, Xu Sun, Lining Yao, Hiroshi Ishii, Daniela Rus, and Sangbae Kim. "Sticky Actuator." In: *Proceedings of the 8th International Conference on Tangible, Embedded, and Embodied Interaction*. New York, New York, USA: ACM Press, 2015, pp. 77–84. ISBN: 978-1-4503-3305-4. DOI: 10.1145/2677199.2680600. URL: http://dl.acm.org/citation.cfm?doid=2677199.2680600.

[80] Martin Nisser, Christina Chen Liao, Yuchen Chai, Aradhana Adhikari, Steve Hodges, and Stefanie Mueller. "LaserFactory: A Laser Cutter-Based Electromechanical Assembly and Fabrication Platform to Make Functional Devices & Robots." In: *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. CHI '21: CHI Conference on Human Factors in Computing Systems. Yokohama Japan: ACM, May 6, 2021, pp. 1–15. ISBN: 978-1-4503-8096-6. DOI: 10.1145/3411764.3445692. URL:

https://dl.acm.org/doi/10.1145/3411764.3445692 (visited on 05/22/2021).

[81] Dan R Olsen. "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, Oct. 2007, pp. 251–258. ISBN: 9781595936792. DOI: 10.1145/1294211.1294256.

[82] Makoto Ono, Buntarou Shizuki, and Jiro Tanaka. "Touch & Activate." In: *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*. New York, New York, USA: ACM Press, 2013, pp. 31–40. ISBN: 978-1-4503-2268-3. DOI: 10.1145/2501988.2501989. URL: http://dl.acm.org/citation.cfm?doid=2501988.2501989.

[83] Jifei Ou, Gershon Dublon, Chin-Yi Cheng, Felix Heibeck, Karl Willis, and Hiroshi Ishii. "Cilllia: 3D Printed Micro-Pillar Structures for Surface Texture, Actuation and Sensing." In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. CHI'16: CHI Conference on Human Factors in Computing Systems. San Jose California USA: ACM, May 7, 2016, pp. 5753–5764. ISBN: 978-1-4503-3362-7. DOI: 10.1145/2858036.2858257. URL: http://dl.acm.org/doi/10.1145/2858036.2858257.

[84] Jifei Ou, Mélina Skouras, Nikolaos Vlavianos, Felix Heibeck, Chin-Yi Cheng, Jannik Peters, and Hiroshi Ishii. "aeroMorph - Heat-Sealing Inflatable Shape-Change Materials for Interaction Design." In: *Proceedings of the 29th Annual ACM Symposium on User Interface Software and Technology*. New York, New York, USA: ACM Press, 2016, pp. 121–132. ISBN: 978-1-4503-4189-9. DOI: 10.1145/2984511.2984520. URL: http://dl.acm.org/citation.cfm?doid=2984511.2984520.

[85] Amanda Parkes and Hiroshi Ishii. "Bosu: A Physical Programmable Design Tool for Transformability with Soft Mechanics." In: *Proceedings of the 8th ACM Conference on Designing Interactive Systems - DIS '10*. The 8th ACM Conference. Aarhus, Denmark: ACM Press, 2010, p. 189. ISBN: 978-1-4503-0103-9. DOI: 10.1145/1858171.1858205. URL: http://portal.acm.org/citation.cfm?doid=1858171.1858205 (visited on 05/25/2021).

[86] Huaishu Peng, François Guimbretière, James McCann, and Scott Hudson. "A 3D Printer for Interactive Electromagnetic Devices." In: *Proceedings of the 29th Annual Symposium on User Interface Software and Technology - UIST '16*. The 29th Annual Symposium. Tokyo, Japan: ACM Press, 2016, pp. 553–562. ISBN: 978-1-4503-4189-9. DOI: 10.1145/2984511.2984523. URL: http://dl.acm.org/citation.cfm?doid=2984511.2984523.

[87] Huaishu Peng, Jennifer Mankoff, Scott E Hudson, and James McCann. "A Layered Fabric 3D Printer for Soft Interactive Objects." In: *Proceedings of the 2015 CHI Conference on Human Factors in Computing Systems*. New York, New York, USA: ACM

Press, 2015, pp. 1789–1798. ISBN: 978-1-4503-3145-6. DOI: 10.1145/2702123.2702327. URL: http://dl.acm.org/citation.cfm?doid=2702123.2702327.

[88] Henning Pohl, Tor-Salve Dalsgaard, Vesa Krasniqi, and Kasper Hornbæk. "Body LayARs: A Toolkit for Body-Based Augmented Reality." In: *26th ACM Symposium on Virtual Reality Software and Technology*. VRST '20. New York, NY, USA: Association for Computing Machinery, 2020. ISBN: 978-1-4503-7619-8. DOI: 10.1145/3385956.3418946. URL: https://doi.org/10.1145/3385956.3418946.

[89] D. Prattichizzo, M. Malvezzi, I. Hussain, and G. Salvietti. "The Sixth-Finger: A modular extra-finger to enhance human hand capabilities." In: *The 23rd IEEE International Symposium on Robot and Human Interactive Communication*. 2014, pp. 993–998.

[90] Daniel J. Preston, Philipp Rothemund, Haihui Joy Jiang, Markus P. Nemitz, Jeff Rawson, Zhigang Suo, and George M. Whitesides. "Digital Logic for Soft Devices." In: *Proceedings of the National Academy of Sciences* 116.16 (Apr. 16, 2019), pp. 7750–7759. ISSN: 0027-8424, 1091-6490. DOI: 10.1073/pnas.1820672116. pmid: 30923120. URL: https://www.pnas.org/content/116/16/7750.

[91] Hayes Solos Raffle, Amanda J Parkes, and Hiroshi Ishii. "Topobo: A Constructive Assembly System with Kinetic Memory." In: *CHI '04: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. New York, New York, USA: ACM, Apr. 2004, pp. 647–654. ISBN: 1-58113-702-8. DOI: 10.1145/985692.985774.

[92] Raf Ramakers, Fraser Anderson, Tovi Grossman, and George Fitzmaurice. "RetroFab: A Design Tool for Retrofitting Physical Interfaces Using Actuators, Sensors and 3D Printing." In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. CHI'16: CHI Conference on Human Factors in Computing Systems. San Jose California USA: ACM, May 7, 2016, pp. 409–419. ISBN: 978-1-4503-3362-7. DOI: 10.1145/2858036.2858485. URL: http://dl.acm.org/doi/10.1145/2858036.2858485.

[93] Raf Ramakers, Kashyap Todi, and Kris Luyten. "PaperPulse: An Integrated Approach for Embedding Electronics in Paper Designs." In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. CHI '15. New York, NY, USA: Association for Computing Machinery, Apr. 18, 2015, pp. 2457–2466. ISBN: 978-1-4503-3145-6. DOI: 10.1145/2702123.2702487. URL: https://doi.org/10.1145/2702123.2702487.

[94] Gabriel Reyes, Dingtian Zhang, Sarthak Ghosh, Pratik Shah, Jason Wu, Aman Parnami, Bailey Bercik, Thad Starner, Gregory D Abowd, and W Keith Edwards. "Whoosh: Non-Voice Acoustics for Low-Cost, Hands-Free, and Rapid Input on Smartwatches." In: Proceedings of the 2016 ACM International Symposium on Wearable Computers. 2016, pp. 120–127.

[95] Michel Rieutord. *Fluid Dynamics*. Graduate Texts in Physics. Cham: Springer International Publishing, 2015.

[96] Jan Rod, David Collins, Daniel Wessolek, Thavishi Ilandara, Ye Ai, Hyowon Lee, and Suranga Nanayakkara. "UTAP - Unique Topographies for Acoustic Propagation." In: *Proceedings of the 11th International Conference on Tangible, Embedded, and Embodied Interaction*. Yokahama, Japan: ACM Press, 2017, pp. 141–152. ISBN: 978-1-4503-4676-4. DOI: 10.1145/3024969.3024987. URL: http://dl.acm.org/citation.cfm?doid=3024969.3024987.

[97] Anne Roudaut, Abhijit Karnik, Markus Löchtefeld, and Sriram Subramanian. "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*. New York, New York, USA: ACM Press, 2013, pp. 593–602. ISBN: 978-1-4503-1899-0. DOI: 10.1145/2470654.2470738.

[98] Harpreet Sareen, Udayan Umapathi, Patrick Shin, Yasuaki Kakehi, Jifei Ou, Hiroshi Ishii, and Pattie Maes. "Printflatables." In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. Denver, CO, USA: ACM Press, 2017, pp. 3669–3680. ISBN: 978-1-4503-4655-9. DOI: 10.1145/3025453.3025898. URL: http://dl.acm.org/citation.cfm?doid=3025453.3025898.

[99] Valkyrie Savage, Colin Chang, and Björn Hartmann. "Sauron." In: *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*. New York, New York, USA: ACM Press, 2013, pp. 447–456. ISBN: 978-1-4503-2268-3. DOI: 10.1145/2501988.2501992. URL: http://dl.acm.org/citation.cfm?doid=2501988.2501992.

[100] Valkyrie Savage, Sean Follmer, Jingyi Li, and Björn Hartmann. "Makers' Marks: Physical Markup for Designing and Fabricating Functional Objects." In: *UIST '15: Proceedings of the 28th annual ACM symposium on User interface software and technology*. New York, New York, USA: ACM Press, Jan. 2015, pp. 103–108.

[101] Valkyrie Savage, Sean Follmer, Jingyi Li, and Björn Hartmann. "Makers' Marks: Physical Markup for Designing and Fabricating Functional Objects." In: *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. UIST '15: The 28th Annual ACM Symposium on User Interface Software and Technology. Charlotte NC USA: ACM, Nov. 5, 2015, pp. 103–108. ISBN: 978-1-4503-3779-3. DOI: 10.1145/2807442.2807508. URL: https://dl.acm.org/doi/10.1145/2807442.2807508.

[102] Valkyrie Savage, Andrew Head, Björn Hartmann, Dan B Goldman, Gautham Mysore, and Wilmot Li. "Lamello." In: *Proceedings of the 2015 CHI Conference on Human Factors in Computing Systems*. New York, New York, USA: ACM Press, 2015, pp. 1277–1280. ISBN: 978-1-4503-3145-6. DOI: 10.1145/2702123.2702207.

URL: http://dl.acm.org/citation.cfm?doid=2702123.2702207.

[103] Valkyrie Savage, Ryan Schmidt, Tovi Grossman, George Fitzmaurice, and Björn Hartmann. "A Series of Tubes." In: *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*. New York, New York, USA: ACM Press, 2014, pp. 3–12. ISBN: 978-1-4503-3069-5. DOI: 10.1145/2642918.2647374. URL: http://dl.acm.org/citation.cfm?doid=2642918.2647374.

[104] Valkyrie Savage, Xiaohan Zhang, and Björn Hartmann. "Midas." In: *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*. Aug. 2012, pp. 1–9.

[105] Martin Schmitz. "3D-Printed Interaction." Feb. 2019. URL: https://tuprints.ulb.tu-darmstadt.de/8431/1/Schmitz_Thesis_3D-Printed-Interaction.pdf.

[106] Martin Schmitz, Martin Herbers, Niloofar Dezfuli, Sebastian Günther, and Max Mühlhäuser. "Off-Line Sensing." In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. New York, New York, USA: ACM Press, 2018, pp. 1–8. ISBN: 978-1-4503-5620-6. DOI: 10.1145/3173574.3173756. URL: http://dl.acm.org/citation.cfm?doid=3173574.3173756.

[107] Martin Schmitz, Mohammadreza Khalilbeigi, Matthias Balwierz, Roman Lissermann, Max Mühlhäuser, and Jürgen Steimle. "Capricate." In: *Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology*. New York, New York, USA: ACM Press, 2015, pp. 253–258. ISBN: 978-1-4503-3779-3. DOI: 10.1145/2807442.2807503. URL: http://dl.acm.org/citation.cfm?doid=2807442.2807503.

[108] Martin Schmitz, Jan Riemann, Florian Müller, Steffen Kreis, and Max Mühlhäuser. "Oh, Snap! A Fabrication Pipeline to Magnetically Connect Conventional and 3D-Printed Electronics." In: *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. CHI '21: CHI Conference on Human Factors in Computing Systems. Yokohama Japan: ACM, May 6, 2021, pp. 1–11. ISBN: 978-1-4503-8096-6. DOI: 10.1145/3411764.3445641. URL: https://dl.acm.org/doi/10.1145/3411764.3445641 (visited on 05/20/2021).

[109] Martin Schmitz, Martin Stitz, Florian Müller, Markus Funk, and Max Mühlhäuser. "./Trilaterate." In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. Glasgow, Scotland, United Kingdom: ACM, May 2019, pp. 454–13. ISBN: 978-1-4503-5970-2. DOI: 10.1145/3290605.3300684. URL: http://dl.acm.org/ft_gateway.cfm?id=3300684&type=html.

[110] A Selamet, N S Dicky, and J M Novak. "Theoretical, Computational and Experimental Investigation of Helmholtz Resonators With Fixed Volume: Lumped Versus Distributed Analysis." In: *Journal of Sound and Vibration* 187.2 (Oct. 1995), pp. 358–367.

[111] Rita Shewbridge, Amy Hurst, and Shaun K. Kane. "Everyday Making: Identifying Future Uses for 3D Printing in the Home." In: *Proceedings of the 2014 Conference on Designing Interactive Systems*. DIS '14. New York, NY, USA: Association for Computing Machinery, June 21, 2014, pp. 815–824. ISBN: 978-1-4503-2902-6. DOI: 10.1145/2598510.2598544. URL: https://doi.org/10.1145/2598510.2598544.

[112] Lei Shi, Ross McLachlan, Yuhang Zhao, and Shiri Azenkot. "Magic Touch: Interacting with 3D Printed Graphics." In: *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility*. ASSETS '16: The 18th International ACM SIGACCESS Conference on Computers and Accessibility. Reno Nevada USA: ACM, Oct. 23, 2016, pp. 329–330. ISBN: 978-1-4503-4124-0. DOI: 10.1145/2982142.2982153. URL: https://dl.acm.org/doi/10.1145/2982142.2982153.

[113] Lei Shi, Idan Zelzer, Catherine Feng, and Shiri Azenkot. "Tickers and Talker." In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. New York, New York, USA: ACM Press, 2016, pp. 4896–4907. ISBN: 978-1-4503-3362-7. DOI: 10.1145/2858036.2858507. URL: http://dl.acm.org/citation.cfm?doid=2858036.2858507.

[114] Pao Siangliulue, Kenneth C. Arnold, Krzysztof Z. Gajos, and Steven P. Dow. "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*. CSCW '15: Computer Supported Cooperative Work and Social Computing. Vancouver BC Canada: ACM, Feb. 28, 2015, pp. 937–945. ISBN: 978-1-4503-2922-4. DOI: 10.1145/2675133.2675239. URL: https://dl.acm.org/doi/10.1145/2675133.2675239 (visited on 05/25/2021).

[115] Ronit Slyper and Jessica Hodgins. "Prototyping Robot Appearance, Movement, and Interactions Using Flexible 3D Printing and Air Pressure Sensors." In: *The 21st IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, Sept. 2012, pp. 6–11. ISBN: 978-1-4673-4604-7. DOI: 10.1109/ROMAN.2012.6343723. URL: http://ieeexplore.ieee.org/document/6343723/.

[116] Yuanping Song, Robert M. Panas, Samira Chizari, Lucas A. Shaw, Julie A. Jackson, Jonathan B. Hopkins, and Andrew J. Pascall. "Additively Manufacturable Micro-Mechanical Logic Gates." In: *Nature Communications* 10.1 (Dec. 2019), p. 882. ISSN: 2041-1723. DOI: 10.1038/s41467-019-08678-0. URL: http://www.nature.com/articles/s41467-019-08678-0.

[117] Miriam Sturdee and Jason Alexander. "Analysis and Classification of Shape-Changing Interfaces for Design and Application-

Based Research." English. In: *ACM Computing Surveys (CSUR)* 51.1 (Apr. 2018), pp. 2–32. DOI: 10.1145/3143559.

[118] Saiganesh Swaminathan, Conglei Shi, Yvonne Jansen, Pierre Dragicevic, Lora A. Oehlberg, and Jean-Daniel Fekete. "Supporting the Design and Fabrication of Physical Visualizations." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '14: CHI Conference on Human Factors in Computing Systems. Toronto Ontario Canada: ACM, Apr. 26, 2014, pp. 3845–3854. ISBN: 978-1-4503-2473-1. DOI: 10.1145/2556288.2557310. URL: https://dl.acm.org/doi/10.1145/2556288.2557310 (visited on 05/25/2021).

[119] Yasaman Tahouni, Isabel P. S. Qamar, and Stefanie Mueller. "NURBSforms: A Modular Shape-Changing Interface for Prototyping Curved Surfaces." In: *Proceedings of the Fourteenth International Conference on Tangible, Embedded, and Embodied Interaction*. TEI '20: Fourteenth International Conference on Tangible, Embedded, and Embodied Interaction. Sydney NSW Australia: ACM, Feb. 9, 2020, pp. 403–409. ISBN: 978-1-4503-6107-1. DOI: 10.1145/3374920.3374927. URL: https://dl.acm.org/doi/10.1145/3374920.3374927 (visited on 05/25/2021).

[120] J. W. Tanney. "Fluidics." In: *Progress in Aerospace Sciences* 10 (Jan. 1, 1970), pp. 401–509. ISSN: 0376-0421. DOI: 10.1016/0376-0421(70)90008-4. URL: http://www.sciencedirect.com/science/article/pii/0376042170900084.

[121] Carlos E Tejada, Osamu Fujimoto, Zhiyuan Li, and Daniel Ashbrook. "Blowhole." In: *Proceedings on the 44th Anual Conference on Graphics Interface*. 2018. URL: http://danielashbrook.com/pubs/2018%20Tejada-Blowhole.pdf.

[122] Carlos E. Tejada, Hyunyoung Kim, Mengyu Zhong, and Daniel Ashbrook. "AirLogic: A Toolkit for 3D-printing Stand-Alone, Interactive Objects." In: *In Manuscript*. 2021.

[123] Carlos E. Tejada, Raf Ramakers, Sebastian Boring, and Daniel Ashbrook. "AirTouch: 3D-Printed Touch-Sensitive Objects Using Pneumatic Sensing." In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. CHI '20. New York, NY, USA: Association for Computing Machinery, 2020, pp. 1–10. ISBN: 978-1-4503-6708-0. DOI: 10.1145/3313831.3376136. URL: https://doi.org/10.1145/3313831.3376136.

[124] *The 2021 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 1982–Present.

[125] *The ACM Symposium on User Interface Software and Technology*. New York, NY, USA: Association for Computing Machinery, 1988–Present.

[126] Todd Thorsen, Sebastian J. Maerkl, and Stephen R. Quake. "Microfluidic Large-Scale Integration." In: *Science* 298.5593 (Oct. 18, 2002), pp. 580–584. ISSN: 0036-8075, 1095-9203. DOI: 10.1126/science.1076996. pmid: 12351675. URL: https://science.sciencemag.org/content/298/5593/580.

[127] John Tiab and Kasper Hornbæk. "Understanding Affordance, System State, and Feedback in Shape-Changing Buttons." In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. CHI'16: CHI Conference on Human Factors in Computing Systems. San Jose California USA: ACM, May 7, 2016, pp. 2752–2763. ISBN: 978-1-4503-3362-7. DOI: 10.1145/2858036.2858350. URL: http://dl.acm.org/doi/10.1145/2858036.2858350.

[128] Tom Valkeneers, Danny Leen, Daniel Ashbrook, and Raf Ramakers. "StackMold: Rapid Prototyping of Functional Multi-Material Objects with Selective Levels of Surface Details." In: *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*. New Orleans, LA, USA: ACM Press, 2019, pp. 687–699. ISBN: 978-1-4503-6816-2. DOI: 10.1145/3332165.3347915. URL: http://dl.acm.org/citation.cfm?doid=3332165.3347915.

[129] J Van Der Heyden. "Fluidic Displays." In: *Recent Advances in Display Media*. 1968, pp. 71–80.

[130] Nicolas Villar, James Scott, Steve Hodges, Kerry Hammil, and Colin Miller. ".NET Gadgeteer: A Platform for Custom Devices." In: *Pervasive Computing*. Ed. by Judy Kay, Paul Lukowicz, Hideyuki Tokuda, Patrick Olivier, and Antonio Krüger. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2012, pp. 216–233. ISBN: 978-3-642-31205-2. DOI: 10.1007/978-3-642-31205-2_14.

[131] Marynel Vázquez, Eric Brockmeyer, Ruta Desai, Chris Harrison, and Scott E Hudson. "3D Printing Pneumatic Device Controls with Variable Activation Force Capabilities." In: *Proceedings of the 2015 CHI Conference on Human Factors in Computing Systems*. New York, New York, USA: ACM Press, 2015, pp. 1295–1304. ISBN: 978-1-4503-3145-6. DOI: 10.1145/2702123.2702569. URL: http://dl.acm.org/citation.cfm?doid=2702123.2702569.

[132] P Welch. "The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms." In: *IEEE Transactions on Audio and Electroacoustics* 15.2 (June 1967), pp. 70–73.

[133] Karl Willis, Eric Brockmeyer, Scott Hudson, and Ivan Poupyrev. "Printed Optics." In: *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*. Cambridge, Massachusetts, United States: ACM, Oct. 2012, pp. 589–598.

ISBN: 978-1-4503-1580-7. DOI: 10.1145/2380116.2380190. URL: http://dl.acm.org/citation.cfm?doid=2380116.2380190.

[134] Junichi Yamaoka, Ryuma Niiyama, and Yasuaki Kakehi. "Blow-Fab." In: *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. New York, New York, USA: ACM Press, 2017, pp. 461–469. ISBN: 978-1-4503-4981-9. DOI: 10.1145/3126594.3126624. URL: http://dl.acm.org/citation.cfm?doid=3126594.3126624.

[135] Lining Yao, Ryuma Niiyama, Jifei Ou, Sean Follmer, Clark Della Silva, and Hiroshi Ishii. "PneUI." In: *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*. New York, New York, USA: ACM Press, 2013, pp. 13–22. ISBN: 978-1-4503-2268-3. DOI: 10.1145/2501988.2502037. URL: http://dl.acm.org/citation.cfm?doid=2501988.2502037.

[136] Hong Kai Yap, Hui Yong Ng, and Chen-Hua Yeow. "High-Force Soft Printable Pneumatics for Soft Robotic Applications." In: *Soft Robotics* 3.3 (Sept. 2016), pp. 144–158. DOI: 10.1089/soro.2016.0030. URL: https://www.liebertpub.com/doi/10.1089/soro.2016.0030.

[137] Amit Zoran and Joseph A. Paradiso. "FreeD: A Freehand Digital Sculpting Tool." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '13: CHI Conference on Human Factors in Computing Systems. Paris France: ACM, Apr. 27, 2013, pp. 2613–2616. ISBN: 978-1-4503-1899-0. DOI: 10.1145/2470654.2481361. URL: https://dl.acm.org/doi/10.1145/2470654.2481361 (visited on 05/20/2021).

DECLARATION

Put your declaration here.

Copenhagen, June 2020

Carlos E. Tejada