Dynamic Physical Affordances for Shape-Changing and Deformable User Interfaces

by

Sean Weston Follmer

B.S., Stanford University (2009)
S.M., Massachusetts Institute of Technology (2011)

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Media Arts and Sciences

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2015

Signature redacted

Author ....
Program in Media Arts and Science
January 9, 2015

Certified by............................ Signature redacted

Prof. Hiroshi Ishii

Jerome B. Wiesner Professor of Media Arts and Sciences

Program in Media Arts and Sciences

Thesis Supervisor

Signature redacted

Accepted by ....

Prof. Pattie Maes
Alexander W. Dreyfoos (1954) Professor of Media Arts and Sciences
Associate Academic Head
Program in Media Arts and Sciences

Dynamic Physical Affordances for Shape-Changing and Deformable User Interfaces

by

Sean Weston Follmer

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning
on January 9, 2015, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Media Arts and Sciences

Abstract

The world is filled with tools and devices designed to fit specific needs and goals, and their physical form plays an important role in helping users understand their use. These physical affordances provide products and interfaces with many advantages: they contribute to good ergonomics, allow users to attend to other tasks visually, and take advantage of embodied and distributed cognition by allowing users to offload mental computation spatially. However, devices today include more and more functionality, with increasingly fewer physical affordances, losing many of the advantages in expressivity and dexterity that our hands can provide.

My research examines how we can apply shape-changing and deformable interfaces to address the lack of physical affordances in today's interactive products and enable richer physical interaction with general purpose computing interfaces. In this thesis, I introduce tangible interfaces that use their form to adapt to the functions and ways users want to interact with them. I explore two solutions: 1) creating Dynamic Physical Affordances through shape change and 2) user Improvised Physical Affordances through direct deformation and through appropriation of existing objects. Dynamic Physical Affordances can provide buttons and sliders on demand as an application changes, or even allow users to directly manipulate 3D models or data sets through physical handles which appear out of the data. Improvised Physical Affordances can allow users to squeeze, stretch, and deform input devices to fit their needs, creating the perfect game controller, or shaping a mobile phone around their wrist to form a bracelet.

Novel technical solutions are needed to enable these new interaction techniques; this thesis describes techniques both for actuation and robust sensing for shape-changing and deformable interfaces. Finally, systems that utilize Dynamic Physical Affordances and Improvised Physical Affordances are evaluated to understand patterns of use and performance. My belief is that shape-changing UI will become increasingly available in the future, and this work begins to create a vocabulary and design space for more general-purpose interaction for shape-changing UI.

Thesis Supervisor: Prof. Hiroshi Ishii
Title: Jerome B. Wiesner Professor of Media Arts and Sciences
Program in Media Arts and Sciences

Dynamic Physical Affordances for Shape-Changing and Deformable User Interfaces

by

Sean Follmer

Signature redacted

Thesis Reader ....

Björn Hartmann

Assistant Professor of Electrical Engineering and Computer Science

University of California, Berkeley

Dynamic Physical Affordances for Shape-Changing and Deformable User Interfaces

by

Sean Follmer

Signature redacted

Thesis Reader ....

Joseph Paradiso

Associate Professor of Media Arts and Sciences

Massachusetts Institute of Technology

Acknowledgments

I would like to thank Prof. Hiroshi Ishii, my advisor and mentor for my time at the MIT Media Lab. His support and inspiration has made all of this work possible. Hiroshi would always challenge me to think bigger, often asking, "What is your vision?" But most of all, his excitement was contagious and his energy and passion undying, which will always give me something for which to strive. In addition, I would like to thank my thesis committee, Prof. Joseph Paradiso and Prof. Björn Hartmann. Their advice and guidance through my research process has been invaluable. Björn has long been guiding and inspiring me, ever since I was an undergrad at Stanford, and has served as a model and a benchmark for what good HCI research should look like. I would also like to thank Hayes Raffle and Scott Klemmer, two people who helped me find my way to the Media Lab and both showed me how much fun HCI could be.

I have had the pleasure of collaborating with some of the most interesting and inspiring researchers at the Media Lab. Specifically, Daniel Leithinger, whose friendship and collaboration I am most thankful for. The rest of the members of the Tangible Media Group have been a pleasure to bounce ideas off of and work shoulder to shoulder with in the late nights leading up to sponsor weeks or CHI deadlines. Alex Olwal, though not always in TMG, has been a close collaborator and a source for helping to organize and order our crazy ideas. Beyond TMG, many others at the lab have made me consider new ideas and crazy schemes over drinks at the Muddy Charles, tea at Voltage, or lunch at Sloan or Momogoose (formerly Gooseberry), like Dave, David, and Davíd. I especially enjoyed borrowing things from High Low tech as an excuse to hang out in Sam, Jenifer, and Dave's office. Thanks to Seth, Marcelo, and Jamie for meeting with me even before I started and for helping me find my way when I did. J.B. for guiding me when I first got to the lab. James Patten and Jen Lusker for their awesome help with the TRANSFORM and Cooper Hewitt version of inFORM. Brian Mayton helped me understand electric field sensing. Mark Feldmeier always helped to solve any electrical engineering problem that ever came his way. Even though he might not believe it, I think that Matt Hirsch was the best student at the Lab, and I thank him for helping me and for being around.

This work could not have been possible without the tremendous help and collaboration of MIT undergraduate researchers. Akimitsu Hogge was always there when we needed him. A large cast of amazing mechanical engineers helped with the design and implementation of inFORM: Guangtao Zhang, Alyx Daly, Cheetiri Smith, Keenan Sunderwirth, Pat Capulong and David Doan.

So many wonderful people at MIT have greased wheels and done too many favors for me. Specifically, Mary Tran Niskala, who has long been my office mate and a all around great person to be around. Natalia, Sarah, and Lisa for helping out before that. It has been a pleasure over the years to get to know Kevin Davis one of the nicest and most helpful people I have ever met. I thank Tom Lutz and John DiFrancesco for their help in the shop. Also, the amazing press and communications staff at the Media Lab: Alexandra and Stacy. MAS staff Aaron, Bill, and Keira always brightened my day. And, last but not least, the great Linda Peterson.

Finally, I would like to thank my family for their support and love. My dad for taking me to SIGGRAPH '98 in Orlando, not only because of Disney World, but also because of the amazing inventions and technologies on display. My mom for always encouraging me to study, and for instilling a life long love of education and reading in me. And Kendall, for putting up with me even when I was stressed about finishing this document, and for fixing all of my spelling mistakes!

This research was supported in part by the National Science Foundation Graduate Research Fellowship under Grant No. 1122374. In addition I was supported as a Research Assistant at the MIT Media Lab through the Media Lab Consortium.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Designers of interactive products have long sought to match form with function. The physical affordances of a design make it easier for users to interact with products like cameras, cars, or music production tools when people need to attend to many tasks simultaneously, and when precision and expressivity is required. Physical affordances provide products and interfaces with many advantages: they provide strong clues for use, often contribute to good ergonomics, allow users to attend to other tasks visually, and take advantage of embodied and distributed cognition by allowing users to offload computation spatially.

However, today we live in an Internet of Things where devices are becoming more connected and include more functions as computation becomes cheaper and faster. Designers often lack the tools to support all of these new features gracefully; they cannot always include the proper physical affordances for each interaction. Instead they often rely on complex on-screen menus, embracing the ease of graphical display while ignoring physical affordances. Today we even use interactive screens to turn on and off lights through an application on our mobile device. We can see this divorce between interactivity, control, and the physical products that we use in our dally lives. This trend is led by the fact that the digital world provides many advantages - for example programability, extensibility, maintaining state, and sharing information on the internet. You can control your thermostat from work to make sure your home is the right temperature when you arrive. But we are beginning to lose a way to

(a) A desk in 1984. (b) A desk in 2014.

Figure 1-1: These photos illustrate the convergence of digital devices. Images courtesy of Doug Thomsen and bestreviews.com.

interact with devices and appliances that is not through a graphical user interface.

The general purpose nature of the personal computer has revolutionized the way we interact with machines. Prior to the PC it was common to have a different device and thus a different interface for every task. But the power of computation and the pioneering work of Doug Englbart, Xerox PARC, and many other researchers has led direct manipulation interfaces, which utilize a keyboard and mouse or touch screen, to dominate the way we control and interact with most machines.

We see Graphical User Interfaces (GUIs) in all places, from controlling complex machinery such as CNC machines to automatic bank tellers and thermostats. The screen and the GUI are ubiquitous. In many ways we have arrived at Weiser's concept of Ubiquitous Computing - we have Pads, Tabs, and Boards - but the reality is that though computation is everywhere, for the most part all of our interaction is confined to flat surfaces and screens. These devices that are always with us and everywhere are doing more with increasingly fewer physical UI controls.

Nowhere is this more apparent than in the case of the mobile phone, a computer interface that is most intimate and with us at almost all times. The mobile phone is - of course - a telephone, but it is also a messaging device, camera, camcorder, game player, television/video player, web browser, and about a thousand different other things. There are over 1.2 million applications available for download on the Apple

Appstore [129]. Yet for all of these different tasks and all of its different uses, it is limited to the same static affordances - a flat rectangular screen that can sense the x/y position of up to ten fingers, and three physical buttons, and one physical switch, all encased in a rigid rectangular prism. It instead leverages the interactivity and flexibility of the screen. However, the best affordances and UIs for typing an email are not the same for playing a game or 3D modeling.

My belief is that the physical form of products and devices must reflect this interactivity better. Not only should there be interactive devices that have richer affordances and more degrees of freedom of input, but we also need devices that can physically adapt to different types of interaction and different tasks. We need to have devices that allow users to shape the affordances they need, rather than a one size fits all approach. We should not be using the same physical interface to play a game that we do to send an email.

Form must become as dynamic, as malleable, as the pixels of a screen - able to change at over 60 frames per second. The recent field of shape-changing and deformable interfaces has begun to make this a possibility. HCI researchers have been expanding beyond of the limitations of GUIs for decades now. Tangible and Graspable interfaces can provide richer physical affordances [72]. But most tangible interfaces are limited by the static nature of the physical world - they cannot keep up with the speed of digital computation, whereas processors can complete millions of floating point operations per second. Shape-changing and deformable interfaces begin to change this equation, not only allowing for more complex interaction but also for physical form to be updated by digital computation, through motors, smart materials, or pneumatics. However, in order for these new types of interfaces to begin to address the lack of physical affordances in interactive devices, new interaction techniques and technologies must be developed.

My research looks at how we can apply shape-changing and deformable interfaces to address the lack of physical affordances in today's interactive products. Previously, much of the work on shape-changing interfaces and shape displays has focused on representing content physically. This thesis instead focuses more on the role of

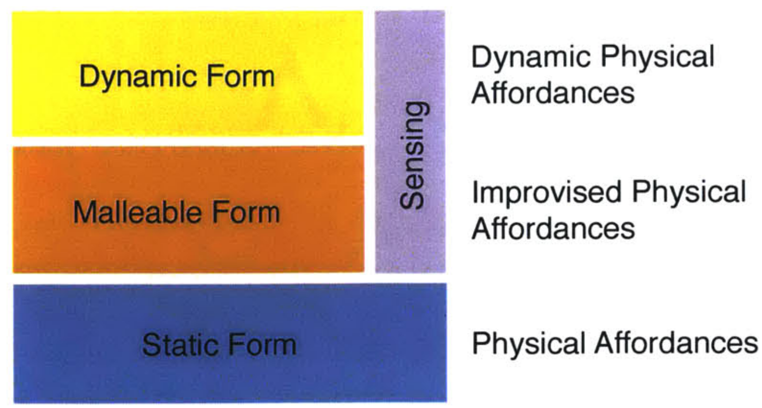


Figure 1-2: Physical Affordances have only been investigated for static objects; this thesis investigates physical affordances for deformable and shape-changing interfaces.

shape change in providing necessary affordances for interaction. In this thesis, I describe tangible interfaces that adapt their form to the functions and ways users want to interact with them. My belief is that shape-changing UI will become increasingly available in the future, and this thesis begins to create a vocabulary and design space for more general-purpose interaction for shape-changing UI. Previously, physical affordances have been investigated only for devices that have a static form. This thesis goes further and investigates physical affordances for deformable and shape-changing interfaces, see Figure 1-2. I explore two solutions: 1) creating Dynamic Physical Affordances through shape change, and 2) user Improvised Physical Affordances through direct deformation and by appropriating existing objects. However, these deformable and shape-changing structures must also be coupled with rich sensing to enable them to be interactive devices. This thesis therefore pushes the concept of physical affordance further, and seeks to find where these new "Dynamic Physical Affordances" can help guide interaction with computing devices.

This thesis introduces Dynamic Physical Affordances, in which a Shape-Changing UI is used to render physical affordances on demand to adapt to a variety of different applications [42]. Dynamic Physical Affordances can be *Directly Rendered*, allowing the user to touch them directly, or *Object Mediated*, where the user interacts with the shape-changing interface through a mediating tangible object on the shape-changing interface.

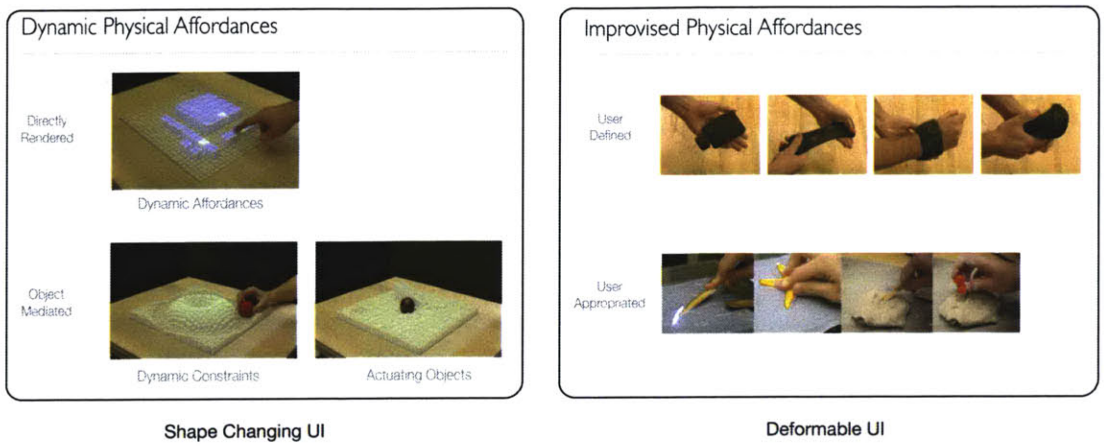


Figure 1-3: Framework exploring affordances for Shape-Changing and Deformable Interfaces.

Beyond having the computational system render physical affordances on demand, it is important to have systems that users can adapt to their needs. When users are faced with problems in the real world, they often appropriate the tools or objects they have on hand to solve the problem, creating an improvised solution. It is often much easier to improvise and adapt in the physical world - a little glue or tape goes a long way. How can users improvise new ways of interacting with computational information? For example if a user plays a game on her phone and she wants to have different buttons to control it, or an entirely different form to grip the game device, how can she do that easily?

In this thesis I propose two approaches to Improvised Physical Affordances: 1) User-Defined Physical Affordances 2) Appropriated Physical Affordances. Both of these types of Improvised Physical Affordances utilize malleable and deformable interfaces to allow for more flexibility in input.

The first type of Improvised Physical Affordances are Appropriated Physical Affordances. How can we allow the user to select existing physical objects that have both different handling affordances (how the user grips the tool) and effector affordances (how the tool can modify the operand)? A user can take existing physical objects and use them to deform a malleable input device, which can capture and track the geometry of arbitrary objects pressed into their surfaces, provide passive haptic feedback, and allow for co-located graphical feedback.

User-Defined Physical Affordances allow users to deform interfaces to create the necessary physical affordances for interaction. This assumes a deformable input device that can sense both its shape, and a user's touch on the surface of the device. For example, a user could deform the device to create a number of buttons to control parameters in a game. Or the user could deform the entire shape of the device to have different ergonomic support for a different task. In order to accomplish this task of easily shifting between different physical forms, I suggest using particle jamming to change the stiffness of a malleable interactive device.

Through these novel interaction techniques, and technology to support them, I have demonstrated a new way of considering affordances of interactive devices that allows for richer interaction and more expressive input.

1.1 Thesis Contributions

This thesis makes contributions to the field of Human Computer Interaction in three areas:

1. Techniques for providing Dynamic Physical Affordances through shape change.
   (a) An exploration of the design space of Dynamic Physical Affordances and Constraints.
   (b) Explorations in the use of motion and animation for physical affordances.
   (c) State-of-the-art system for fast, real-time 2.5D shape actuation, co-located projected graphics, object tracking, and direct manipulation.
   (d) Three applications that demonstrate the potential of these interaction techniques for HCI.
   (e) An evaluation of the performance of dynamic physical affordances.
   (f) An evaluation of the perceptual qualities of motion in shape change for physical affordances.

2. An Investigation of User Appropriated Physical Affordances.

    (a) An exploration of the design space of User Appropriated Physical Affordances.

    (b) A novel hardware system, deFORM, to support User Appropriated Physical Affordances through a real-time 2.5d deformable surface interface that uses infrared (IR) structured light 3D scanning and projected visual feedback.

    (c) Techniques for tracking arbitrary and tagged tangible tools (phicons), touch and hand gestures.

    (d) A number of application prototypes that make use of User Appropriated Physical Affordances.

    (e) A study exploring the use of User Appropriated Physical Affordances to support 3D modeling for children.

3. Techniques for supporting User-Defined Physical Affordances through direct deformation.

    (a) An exploration of the design space of User-Defined Physical Affordances.

    (b) Applying particle jamming for use as a variable stiffness material to enable User-Defined Physical Affordances.

    (c) A review of the state of the art in jamming from an HCI perspective.

    (d) A novel hydraulic-based jamming technology, for rapid activation, silent actuation, and embedded optical sensing.

    (e) Two techniques for high-resolution, integrated and embedded sensing for jamming interfaces: optical sensing, using index-matched fluids and particles; and electrical sensing, using capacitive and electric field sensing.

    (f) A small, low-power jamming system for mobile and embedded organic user interfaces.

    (g) Motivating prototypes to highlight how jamming can be applied to HCI, and particularly User-Defined Affordances.

1.2 Dissertation Outline

The theoretical background that supports this thesis work is overviewed in Chapter 2. It covers the sensory and biomechanical aspects of the grasping and touching with the human hand, embodied and distributed cognition, and affordances.

Chapter 3 looks at related work and describes how other researchers have applied the theoretical work in Chapter 2 to Human Computer Interfaces. This chapter surveys research topics in HCI that leverage physical interaction and manipulation, such as Virtual and Augmented Reality, Ubiquitous Computing, Haptic Interfaces, Tangible and Graspable Interfaces. It also provides a more in-depth review of Shape-changing and Deformable Interfaces, the area of research most closely related to this thesis. In addition, analysis of related work is provided and the work in this thesis (Dynamic Physical Affordances, User Appropriated Affordances, and User Defined Affordances) is positioned in the context of the related work.

In Chapter 4 Dynamic Physical Affordances are introduced and described in detail. This chapter proposes utilizing shape displays in three different ways to mediate interaction: to *facilitate* by providing dynamic physical affordances through shape change, to *restrict* by guiding users with dynamic physical constraints, and to *appropriate* existing objects as dynamic physical affordances by actuating them through shape change. This chapter outlines potential interaction techniques and introduces *Dynamic Physical Affordances and Constraints* with our inFORM system, built on top of a state-of-the-art shape display, which provides for variable stiffness rendering and real-time user input through direct touch and tangible interaction. A set of motivating examples demonstrates how dynamic affordances, constraints, and object actuation can create novel interaction possibilities. Finally, the results of a lab based study evaluating the performance of dynamic physical affordances over mid-air 3D pointing in a mesh manipulation task is presented.

The second part of the thesis concerns Improvised Physical Affordances - User Appropriated Affordances and User Defined Affordances. User Appropriated Affordances are introduced in Chapter 5. This chapter explores how we can create interfaces that allow users to use existing tools and objects as expressive input devices, leveraging their physical affordances. To support this, we introduce a novel input device, deForm, that enables use of 2.5D touch gestures, tangible tools, and arbitrary objects through real-time structured light scanning of a malleable surface of interaction. deForm captures high-resolution surface deformations and 2D grey-scale textures of a gel surface through a three-phase structured light 3D scanner. This technique can be combined with IR projection to allow for invisible capture, providing the opportunity for co-located visual feedback on the deformable surface. We describe methods for tracking fingers, whole hand gestures, and arbitrary tangible tools. We outline a method for physically encoding fiducial marker information in the height map of tangible tools. In addition, we describe a novel method for distinguishing between human touch and tangible tools, through capacitive sensing on top of the input surface. We motivate our device through a number of sample applications. An in lab study of use patterns with a system that enables User Appropriated Affordances for a 3D sculpting application, called KidCAD, is discussed.

User Defined Affordances are detailed in Chapter 6. How can we allow users to shape their own devices and affordances? Malleable and organic user interfaces have the potential to enable radically new forms of interactions and expressiveness through flexible, free-form and computationally controlled shapes and displays. This work specifically focuses on particle jamming as a simple, effective method for flexible, shape-changing user interfaces in which programmatic control of material stiffness enables user driven shape change and haptic feedback. We introduce a compact, low-power pneumatic jamming system suitable for mobile devices, and a new hydraulic-based technique with fast, silent actuation and optical shape sensing. We enable jamming structures to sense input and function as interaction devices through two contributed methods for high-resolution shape sensing using: 1) index-matched particles and fluids, and 2) capacitive and electric field sensing. We explore the design space of malleable and organic user interfaces enabled by jamming through four motivational prototypes that highlight jamming's potential in HCI, including applications for for portable shape-changing mobile devices and tablets.

Chapter 7 provides a discussion of the thesis work in context and suggests avenues for future work.

A conclusion follows, which reviews the contributions of this thesis and discusses future work.

1.3 Statement on Multiple Authorship and Prior Publication

The work presented in this thesis was a collaborative effort with many individuals at the MIT Media Lab and beyond. Most closely I collaborated with Daniel Leithinger and Alex Olwal on the inFORM system and Jamming User Interfaces which contribute to Chapters 4 and 6. In addition, Guangtao Zhang, Alyx Daly, Cheetiri Smith, Keenan Sunderwirth, and Pat Capulong contributed to the mechanical design and implementation of inFORM. We worked with Ryan Wistort to design the motor control printed circuit boards for inFORM. Akimitsu Hogge contributed to the software development for inFORM. Lee Gross contributed to the software and study design for the emotional content study in Chapter 4. Travis Rich and Kevin Hu helped with collecting data on Amazon Mechanical Turk and analyzing the data for that study. Kimo Johnson helped with the development of gels for the deFORM system. Nadia Cheng provided advice for and helped to implement the Jamming User Interfaces. As such, following the introduction I will use 'we' in this thesis to describe our process.

The work in this thesis is based on papers previously published in ACM conference proceedings. Those papers are: deFORM [40], KidCAD [39], Jamming User Interfaces [41], and inFORM [42]. I am the primary author on these publications, except for a paper on the Sublimate system, in which a user study that I conducted was published [100]. That user study appears in Chapter 4.

Chapter 2

Background

2.1 Introduction

Humans have great ability and capacity for complex interaction with the environment around them. What are the mechanisms for our understanding of the world and our ability to manipulate it? This section focuses on the theoretical, anatomical, and neurological basis for physical and embodied interaction in design. It explores the complexities of the hand and our ability as humans to reason through objects, space and gesture. New frontiers in cognitive science have expounded on the importance of the body and the material world in the cognitive process, leading to Embodied and Distributed Cognition. Recent literature has also investigated how the mind, hands, and tools can work in concert to accomplish skilled tasks. Finally, what role does a designer have in shaping these interactions with objects and tools?

2.2 Hands

Hands probe, explore, handle, deform, and manipulate. What gives our hands such dexterity, such ability to adapt to different tasks and different tools? This section reviews the basic anatomy and somatic nervous system of the hands, which gives rise to such complex motion and senses. Hands are not simply the end-effectors of the mind, but also play a large part in our cognitive process. Finally, how do we use our hands to manipulate objects, and what are the basic modes of prehension? Literature in neuroscience and robotics has expanded our understanding of hands and these have direct implications for designing tangible and deformable interfaces.

Much of the dexterity and capacity of the hand comes from the arm [184]. The hand has 27 degrees of freedom, yet the muscles and tendons that control its motion extend further up into the forearm. Further, the hand is attached at the end of the arm and guided by the arm to reach objects or gesture. The whole body works to manipulate objects.

If we wish to understand how the hand moves and how it senses, we must focus on the hand's nervous system. The somatic nervous system extends into the arms and hands, and is composed of afferent nerves, which provide sensory feedback, and efferent nerves, which control muscle contraction and motor function [78]. The cutaneous (skin) and kinesthetic (muscle, tendon, joint) nerves provide sensory information. Proprioceptive perception is the understanding of stimuli that are produced by the body itself, such as knowing where one's arm is in space. Exteroceptive perception relates to understanding stimuli that are outside of the body. These perceptual systems often work through the same nerve cells, though much of proprioceptive perception happens directly in the tendons to provide kinesthetic feedback [184].

There are a variety of different cutaneous afferent nerves in skin cells in the hand that work together to provide responses to different stimuli, such as motion, texture, form, skin stretch, contact, and vibration [78]. These sensory nervous cells provide the basis for the limits of our haptic tactile perception. These cells are distributed in different quantities in different parts of the hand. Slowly adapting afferent cells in the tip of fingers are able to detect features as small as 0.94mm while moving over the target [170].

The four main types of sensory cells in the hand and their distributions are shown in Figure 2-1, [78]. These four types are:

- RA, (Meissner Corpuscle) Quickly adapting, sensitive to motion across skin. Has lower spatial resolution that slowly adapting afferents, but used in detection of surface textures. (Black in figure)

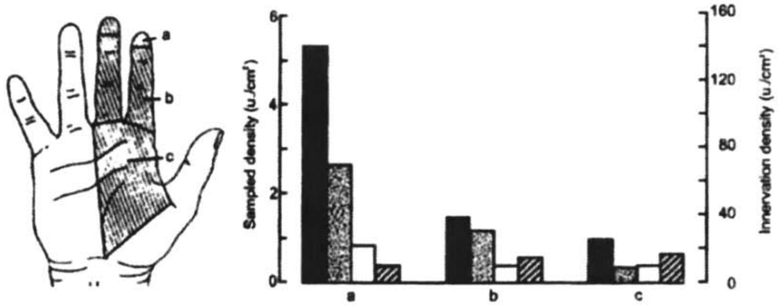


Figure 2-1: Distribution of sensory cells in different parts of the hand. The bar charts on the right refer to the distribution of parts a, b, and c of the hand on the left. The sensory cells are RA(Black), SA-I(Grey), PC (White), and SA-II (Slashed) (Source [78]).

- SA-I (Merkel Cell-neurite complex) Slowly adapting afferents, especially sensitive to corners, edges, and curvature. Essential for tactile perception of form and texture of objects held. (Grey in figure)
- PC (Pacinian Corpuscle) Senses contact, vibration, lateral movement. (White in figure)
- SA-II (Ruffini Corpuscle)Sensitive to stretch of the skin, sometimes with orientation preferences. Sends back regular impulses in response to sustained mechanical stimulation of the skin. Could be used for motor control, as is not perceptible. (Slashes in figure)

But perceiving texture and other features such as form, or weight, are better done actively than passively [46]. Klatzky suggests that there are a variety of *Exploratory Procedures* through which people determine different haptic properties [86]. Different exploratory procedures or hand movements are used to maximize the ability to perceive different features, see Figure 2-2: for example, lateral motion to perceive texture or applying pressure to determine hardness. In addition, different forces are applied against the target object to sense different properties: 0.2N for surface friction, 0.5N for tactile exploration, 1.0N for roughness, and 1.5N for temperature [79].

Hands were not made only for touching, but also for prehension, or grasping, and manipulating. Our unique hand geometry, with the addition of the opposable thumb,

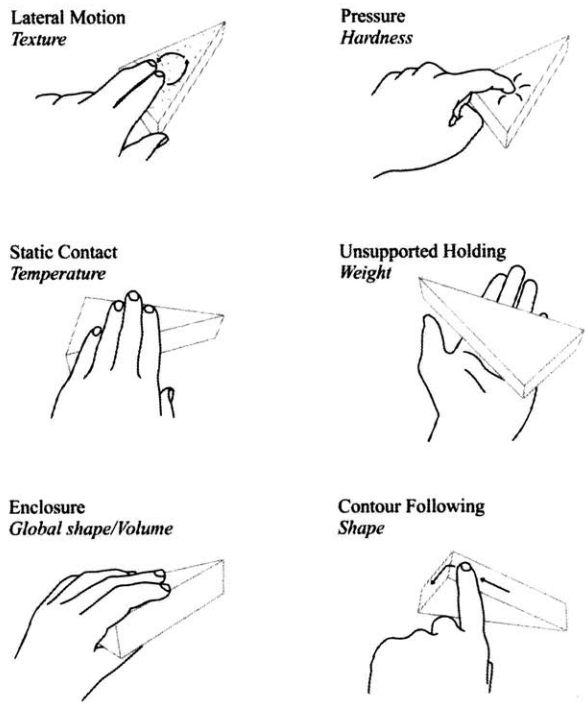


Figure 2-2: Exploratory procedures for haptic perception (Source: [118] adapted from [86]).

plays a large role in defining our prehensile abilities [184]. However, there are many factors that are accounted for in the type of grip used to manipulate an object: the geometry of the object, the surface properties of the object, and the type of task or forces we wish to apply to the object. Much of this is done unconsciously with the help of sensory cells in the hand, which provide feedback on a variety of features and forces. For example, sweat glands in the hand actually help improve grip performance, providing a "boundary lubricator" which increases adhesion [105].

There are many types of grips that can provide different amounts of force, support, or control. Naipier introduced the concept of the Power and Precision requirements for prehension. The power requirements "relate to the ability to apply forces and resist arbitrary forces that may be applied to an object" [105]. The precision requirements "involve small adjustments of posture in order to control the direction in which the force is being applied." These requirements map to different grips; Napier defined two main grip categories the *power grip*, in which the fingers press the object against the

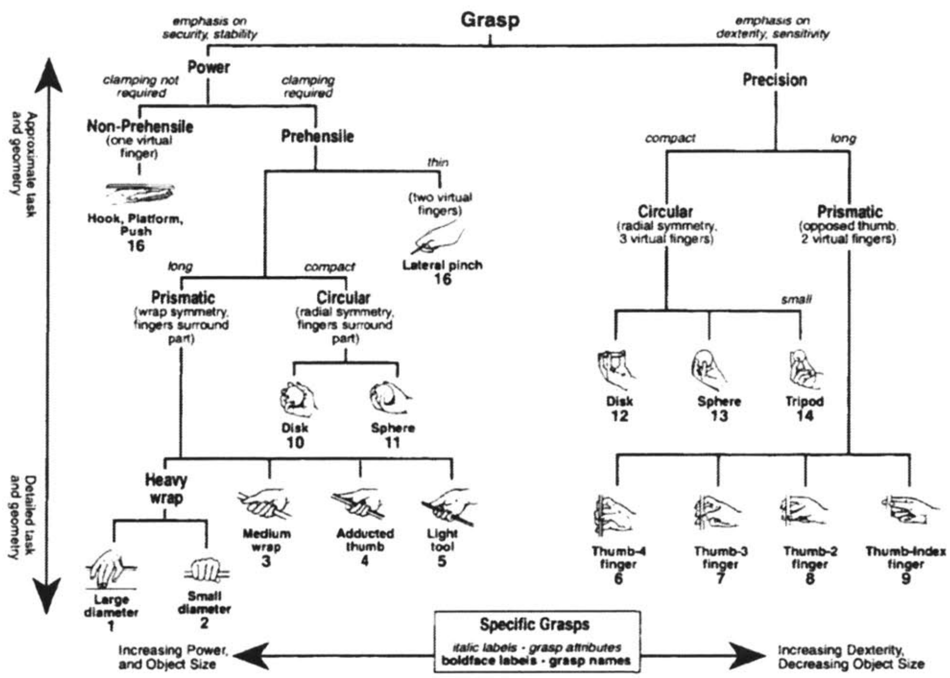


Figure 2-3: Cutkosky and Howe extended Naipier's Power and Precision grip taxonomy (Source: [31]).

palm, and the *precision grip*, in which fingers support the object against the thumb allowing for more rotational forces to be applied. Cutkosky and Howe extended this taxonomy to include a wider range of grips [31], see Figure 2-3. There is a continuum between power and precision: the tripod or three-jaw chuck grip provides much dexterity in manipulating objects, the lateral pinch grip provides less control but more power, and finally the heavy wrap provides much power but little control.

By understanding the geometry, sensory system, and prehensile ability of the hand we are able to build better tangible, graspable, and deformable interfaces. The constraints of the somatic nervous system provide boundaries for haptically representing information. Designers should remember that active haptic exploration is more precise than passive exploration. Finally, the power or precision requirements of a task or interface dictate different tool geometries to enable different grips.

2.3 Embodied Cognition

How do we understand and reason about the world? How do we know how to manipulate it? These questions lead us to the fields of philosophy, psychology, and cognitive science. The idea that the mind is separate from the body has long influenced our understanding of cognition. This separation was much influenced by Rene Descartes's 'Dualism.' The foundations of Cognitive Science and Artificial Intelligence are based on what Winnograd and Flores call a "rationalistic tradition," one that tries to take a logical and objective view of of cognition, and asserts that people have an internal, stable, representation of the world inside their minds [187]. The sensory system updates that model of the world, some goal is created, and the motor system then is used to manipulate the world to achieve this goal. However, recent research suggests that this is not the case; instead cognition is directly tied to our physical bodies and to the physical world and cannot be separated from the two. Embodied Cognition suggests that "minds are not passive representational engines, whose primary function is to create internal models of the external world" and that we use our entire bodies in the process of thinking [65].

Embodied Cognition was influenced by the work of several philosophers, such as Husserl, Heidegger, and Merlau-Ponty, who focused on the importance of experience and introduced the study of Phenomenology [34]. Heidegger's work focused on what he called Dasein, or "Being-in-the-world," and suggested that the notion of "being is inseparable from the world in which it occurs." Winnograd and Flores specifically highlight the importance of Heidegger's exploration of 'breakdowns,' moments when we move from being in the world, from acting, to noticing or reflecting on the 'equipment' we use to act. This concept relates especially to the use of tools, or 'equipment.' Tools can be *ready-to-hand*, when they disappear and become an extension of self, or *present-at-hand*, when the details of the object itself come to the forefront. For example, when using a hammer to strike a nail, we use it without thinking, and it is *ready-to-hand*. However, if the hammer breaks, or something goes wrong, the hammer itself is now the center of attention and *present-at-hand*.

Merleau-Ponty saw the importance of the body in perception, suggesting "a theory of the body is already a theory of perception." However, Merleau-Ponty expanded that "the body can no longer be regarded as an entity to be examined in its own right but has to be placed in the context of a world." We are not observers alone, but rather, through bodily experience, we shape our understanding of the world, and our physical embodiment (the size, shape, appendages and the senses we have) play a deep role in this.

Embodied cognition has sought to understand the importance of the body and situated cognition and to tie phenomenology to cognitive science. This is to say that there is as much importance in our interactions in the world as in our brains alone [24]. Cognition takes place and is situated in our world, thus it is tied to our physical embodiment and to our environment [185]. In contrast to the rationalistic tradition of planning, and other approaches to artificial intelligence, our cognitive process is time pressured because it takes place in the real world, where there may be predators or other dangers. Thus cognition is comprised of many small systems that work quickly, and build up complexity through layering systems on top of each other.

The importance of embodiment is manifested also in our perception. Enactive Perception suggests that our perception of the world is an active process, not something that we passively do. Alva Noe has looked at the importance of the body in perception, noting studies that have shown that our perceptual system cannot develop without the active use of our body [115]. He describes a study that looks at two kittens' development of depth perception. One kitten is able to move freely about the space, the second is mounted on top of the first kitten and unable to move on its own. Thus both cats receive all of the same visual stimuli, however only the cat that can move under its own will develops depth perception properly. Noe explains "only through self-movement can one test and so learn on the relevant patterns."

We do not form a complete image of the world in our minds, but instead leverage our body, our senses, and the world to construct the experience of a full representation. Noe explains that a moving animal can sense much more information about the world than about a single image from their eye, thus it would be wrong to suggest that all of visual perception takes place in the eye and the brain alone. Instead we use the world as a reference, and not only the things that are in view of the retina. Noe explains, "phonomenological reflection on the character of perceptual presence suggest that the features are present as available, rather than as represented." As a result of this we think we have a much richer representation of the world around us, because we are able to move around in the real world to fill in the gaps. For the most part, the world remains constant and this allows us to use the world as offloaded memory. Phenomena like change blindness, only experienced when these systems breakdown, highlight this fact.

When we are not acting, our cognitive processes are still tied to the same areas of our brain that process motor actions. "Observing actions made by others activates the cortical circuits responsible for the planning and execution of those same actions" [165]. These mirror neurons even fire when participants were read action related sentences. There seems to be little difference between thinking about acting and acting.

In addition our body itself helps us think about abstract concepts. Goldwin Meadow demonstrated that gesturing helps us lower our cognitive load while preforming complex abstract tasks [48]. In his experiment, participants completed a number of written math problems, after which they had to memorize a list of words or letters. Next, participants had to explain the math problem they had done either with the use of gesture or without gesture (hands still on the table). After that participants had to recall the list of letters from memory. Participants were able to recall significantly more information when allowed to use gesture in describing the math problems, suggesting a higher cognitive load while not gesturing.

2.4 Distributed Cognition and Tools

A broader view of cognition looks beyond a single body alone. Distributed Cognition "explores how cognitive activity is distributed across internal human minds, external cognitive artifacts, and groups of people, and across space and time" [194]. This relates to much research that explores how we leverage tools and adapt tools to our bodies. How do we begin to think through objects? Secondly, how do we think with other people? Hollan et al. suggests that we must consider both our notions of the scale at which we think of cognition, ie its boundries beyond one mind alone, and the "mechanisms" that we consider part of that process in order to have a full view of cognition [65].

Ed Hutchins has explored these issues in depth in the context of pilotage and navigation [68]. He explores how navigational tools such as the chart, as well as other people, become part of the cognitive process: "Both objects and people become buffers of information, and distributed memory." Hutuchins explains that we encode information and knowledge into physical objects, and he describes western navigational tools as "based on the principle of building computational constraints of the task into the physical structure of the artifact." An astrolabe can be set to show the positions of different heavenly bodies at different times of day, or different dates, or different latitudes (i.e. the astrolabe functions as a representation of information). But, an astrolabe can also serve as a computer to help to solve navigational tasks.

We leverage the world around us to think better and faster. Kirsh introduced the concept of pragmatic and epistemic action. Pragmatic action helps us get physically closer to a goal. Epistemic actions help us solve complex problems by offloading mental compuation to the physical world [84]. Kirsh observed this type of epistemic action in expert Tetris players, where players rotate objects not only to move them, but also to help aide in mental rotation - which is a complex cognitive task, see Figure 2-4. Experts use more epistemic action than novices. Kirsh states epistemic action can:

- reduce "the memory involved in mental computation (space complexity)"
- reduce "the number of steps in mental computation (time complexity)"
- reduce "the probability of error of mental computation (unreliability)" [84]

Another aspect to distributed cognition is that we use conceptual external representations as tools for thinking; different representations have different benefits and

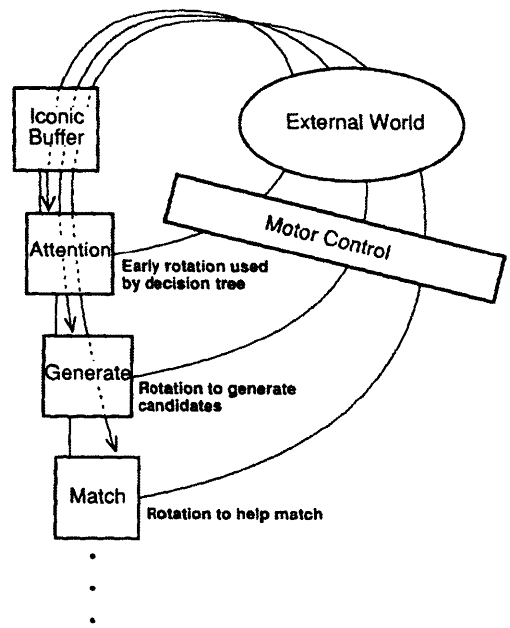


Figure 2-4: Epistemic action: here we see different uses of epistemic action and the external world for different parts of a cognitive task, such as attention, generating possible rotations, and pattern matching (Source: [84]).

weakness. Zhang has explored the use of External Representations in a variety of different contexts and looked at isomorphic representations of the same task, such as tic-tac-toe or the towers of Hanoi [193, 192]. These external representations are in contrast to our internal representations we have in our mind, i.e. our mental models. Zhang showed that as more information about the task is externally represented, performance increases. Zhang explains that External Representations have three roles:

- "External Representations provide information that can be directly perceived and used without being interpreted and formulated explicitly."
- "External Representations can anchor cognitive behavior. That is, the physical structures in external representations constrain the range of possible cognitive actions in the sense that some actions are allowed and others prohibited."
- "External Representations change the nature of tasks."

Tools change the way we see the world and can become extensions of the body. The possibilities for our actions, or Gibson's affordances, change when we have different tools; we can do more things [83]. For example, tools prime us to see different things; in the kitchen "at each moment what a chef sees is partly primed by the tools in their hand."

Furthermore, Kirsh explains, "When we use a tool to reach for a distant object it is as if we are extending our motor capability and we treat our hand as if it is elongated to the tip of the tool." And our notion of self extends to encompass those tools. Iriki et al. showed that in Japanese macaques trained to use tools such as a rake to extend their reach, the neurons that fire when looking at one's body also fire when the monkeys look at the tools [69].

There is some indication that our ability to integrate objects into our notion of self has much to do with time and the lack of delay between our movement and our perception of an external object's movement. A study of self-produced tactile stimulation measured the tickle response of participants [11]. Participants were unable to tickle themselves when using a tool, but when a delay was added to that same tool's movement, the tickle response reached close to the response of an external tactile stimulation. This has a great bearing on the refresh rate of our interactive systems if we intend to use them as a kind of prosthesis.

We do not use single tools alone, but rather we use many objects and artifacts at once in an environment. We group, cluster, arrange, and prepare our tools and our spaces to speed cognition [82]. Experts plan very little as they are performing a task because of such factors as muscle memory and experience, but also because they have arranged their spaces and tools to aide them. They organize and remove objects to limit distractions, and group and cluster objects to simplify perception and ease of access.

Embodied and Distributed cognition have direct implications for how we design interfaces. From considering how we perceive the world to creating tools that can leverage epistemic action to finding the right external representation, there is much that we as designers must consider. The next chapter reviews work that has sought to integrate lessons from these fields to HCI practice. However, there is also theoretical work that developed along with these traditions and out of perceptual psychology and philosophy - that of affordances.

2.5 Affordances

Gibson introduced the theory of affordances as "what [an object or environment] offers the animal, what it provides or furnishes, either for good or ill." This can be viewed as the set of action potentials for an object [47]. Gibson's theory of affordances tries to explain how we perceive the world, suggesting that we understand it not through colors or patterns that we see in objects, but rather by how we can make use of the world around us.

Norman first applied affordance to design and HCI, focusing on "perceived affordances," what he now calls 'signifiers' that the designer creates to provide interaction clues, or suggestions, to the user [116]. Norman explains, "Affordances determine what actions are possible. Signifiers communicate where the action should take place" [117].

- "Affordances are the possible interactions between people and the environment. Some affordances are perceivable, others are not."
- "Perceived affordances often act as signifiers, but they can be ambiguous."
- "Signifiers signal things, in particular what actions are possible and how they should be done. Signifiers must be perceivable, else they fail to function"

Affordances and signifiers can be the same thing, for example the holes of a scissor are "both affordances- they allow the fingers to be inserted - and signifiers - they indicate where the fingers are to go" [117]. Figure 2-5 positions the features that allow users to understand how to use an affordance. Here the qualities of an object that are perceivable contribute to the user's understanding, but in addition higher level features like symbols and iconography, made up by lower level features, also contribute. Finally the user understands the constraints of the affordance, and thus can act upon it.

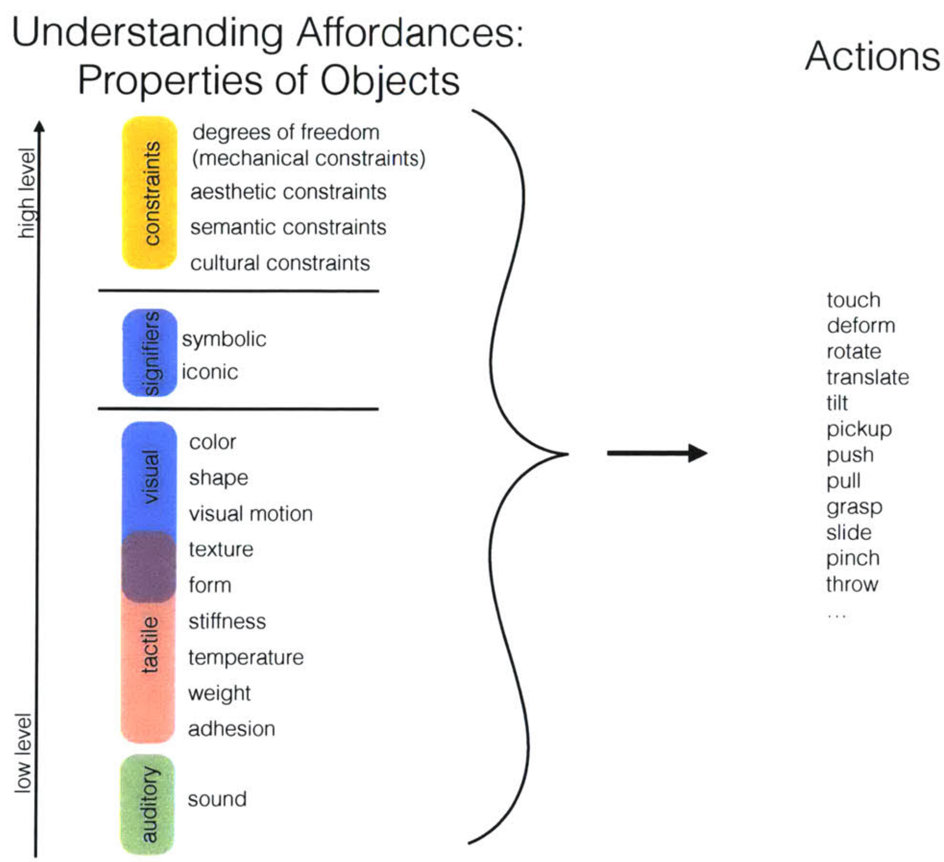


Figure 2-5: Positioning the perceptual qualities of objects, and Normans signifiers and constraints in the context of affordances.

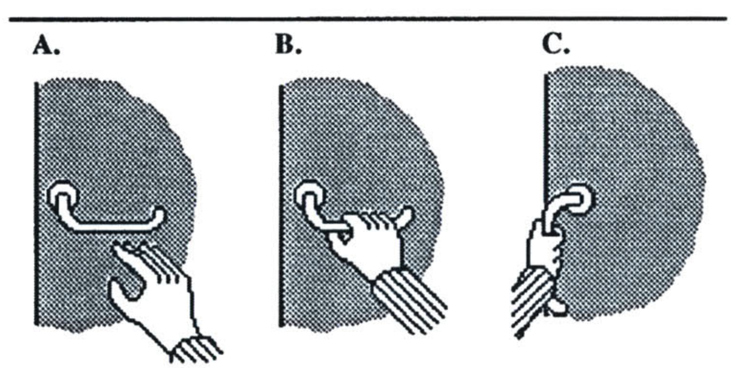


Figure 2-6: Gaver's Sequential Affordances: A) First the affordance of the handle implies grasping, B) Then rotating, C) And finally pulling (Source: [45]).

Gaver [45] defines his technology affordances as "properties of the world that are compatible with and relevant for people's interactions." He emphasizes the importance of perceptible affordances, since mismatched or hidden affordances interfere with the interface's legibility, which may confuse the user and result in improper operation. Gaver highlights that these affordances can be perceived visually, tactilely, or aurally. He also expands on sets of affordances: nested affordances, which are grouped spatially, and sequential affordances, which are grouped temporally, see Figure 2-6. Nested affordances help us explore interaction possibilities: "For instance, a handle alone only appears to afford pulling. A door alone may suggest an affordance for manipulation due to its partial separation from the wall, but not what sort of manipulation will be effective. Only by seeing the affordance of pulling the handle as nested within an affordance of pulling the door can opening the door be a perceptible affordance." Sequential affordances are often used in GUI based interaction, in which graphical perceived affordances can be rendered quickly and then disappear. Gaver explains that "Affordances are not passively perceived, but explored." In contrast to Gibson who focused on visual perception of affordances, Gaver casts a wider net, suggesting we can perceive affordances not only through vision, but also through our other senses such as touch.

Hartson elaborates on Norman and Gaver's work describing four types of affordances as cognitive affordance, "design features that help users in knowing something", physical affordance, "design features that help users in doing a physical action in the interface", sensory affordance, "design features that help users sense something" and functional affordance, "design features that help users accomplish work (i.e., the usefulness of a system function)" [58]. Kaptelinin et al. further splits both cognitive and physical affordances in two parts, describing the handling affordance, the affordances with which a user interacts, and the effecter affordance, the affordances with which a tool manipulates an object [81]. They suggest that these must be tightly coupled.

Feedback and feedforward also play a large role in shaping our understanding of affordances. Norman describes feedforward as "the information helps answer the question of execution (doing)," and feedback as "the information that aids in under-

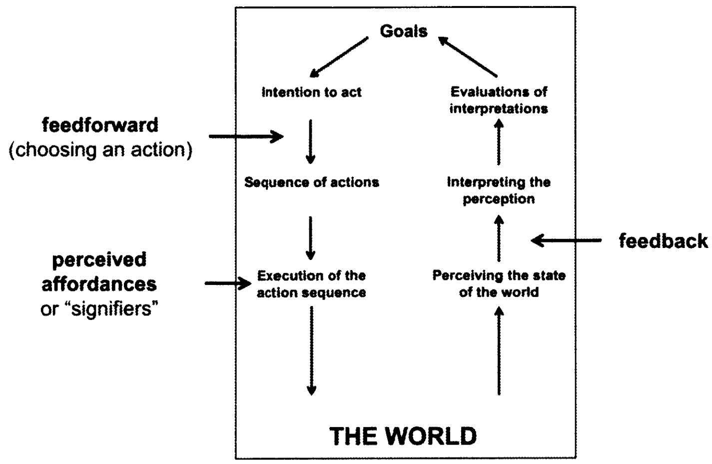


Figure 2-7: Affordance, signifier, feedfoward, and feedback positioned on Norman's Stages of Action (Source: [173]).

standing what has happened" [117]. Vermeulen positions feedforward in the context of affordance, and reviews feedback and affordances [173], see Figure 2-7.

The work in this thesis expands the notion of affordance to consider the role of shape change. Figure 2-8 shows the focus of this thesis. This thesis investigates how shape change modifies the notion of affordance. Specifically it investigates how changes in texture, form, and stiffness can contribute to affordances. The means of accomplishing this shape change fundamentally change how users perceive and use affordances - the shape change can be driven by Computational Control (in the case of self-actuated Shape -changing interfaces), by users replacing tools with different tools, or by the user creating the shape change (in the case of a malleable interface). These three means of shape change lead to 3 new types of affordances: Dynamic Physical Affordances, Appropriated Physical Affordances, and User Defined Physical Affordances.

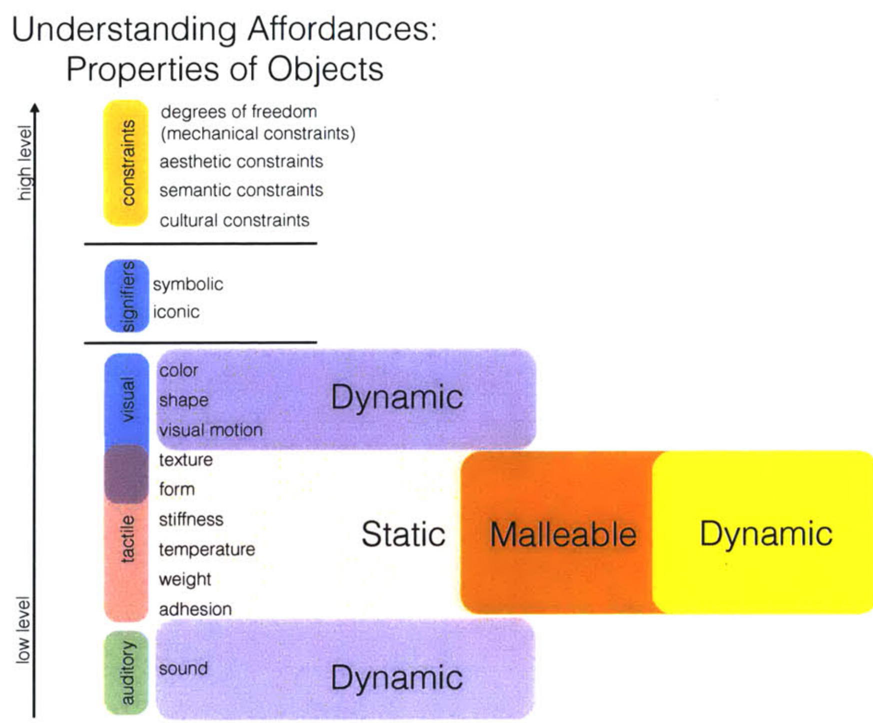


Figure 2-8: Highlighting the focus of this thesis, which examines how malleable and dynamic shape-changing interfaces modify our understanding and use of physical affordances.

Chapter 3

Related Work

A wide set of researchers in the field of Human Computer Interaction have sought to apply the intuition and knowledge of embodied cognition, theory of affordances, and biomechanics of hands (discussed in the previous chapter) to computer interfaces. Virtual Reality leverages much of the notions of egocentrism and the importance of the body in interaction, but often only focuses on visual information to recreate a new reality. Haptic rendering provides a tactile view into a virtual world. Augmented Reality seeks to overlay digital information into our lives, and particularly spatial augmented reality makes use of many of the spatial cues and interaction with digital information grounded in the world around us. Tangible computing attempts to provide rich physical affordances and also leverages spatial offloading of computation, and distributed cognition. Finally, a new field has begun to emerge - shape-changing user interfaces - which can combine the physical affordances of Tangible interfaces with computational control.

3.1 Virtual, Augmented, and Mixed Reality

The goal of Virtual Reality (VR) is to replace the current sensory perception of the world around us with a virtual world. Much work in VR has focused on the visual aspects of perception, by utilizing Head Mounted Displays [139] or view dependent rendering using large displays [1], see Figure 3-1, or computer assisted virtual environ-

(a) Two users viewing different views of the same information.

(b) Two users pointing at a 3D model. Each user has a unique view.

Figure 3-1: Manipulating information with the Two User Responsive Work Bench.

ments (CAVEs) [29]. These systems can allow the user to move around a space, and often track the user's head movements to give users the perception of an immersive virtual environment. VR can also be combined with a variety of spatial input devices or gesture and pose tracking to allow the user to feel as though she is naturally interacting with the virtual world. Because VR creates a spatial 3D environment similar to our natural experience, users can leverage many of the qualities of embodiment, spatial and kinesthetic understanding, that were explored in the last chapter. However, VR has a number of limitations: you cannot see the real world, or the other users' faces, and hardware to provide more than a visual virtual world can be quite complicated and expensive.

In contrast to VR, Augmented Reality (AR) overlays digital information over the real world, visually blurring the boundary between the two. Spatial Augmented Reality situates virtual information in our 3D world, such that the virtual 3D information is aligned with the user's view of the real world, and that as she moves through that space these remain aligned. This allows for more physical interaction with the world around us.

Both Virtual and Augmented Reality owe a great deal to Ivan Sutherland, who in the posited that "Real and Synthetic objects will coexist" in the "Ultimate Display" [162]. In 1968 he created the first head-mounted 3D display and tracking system [163].

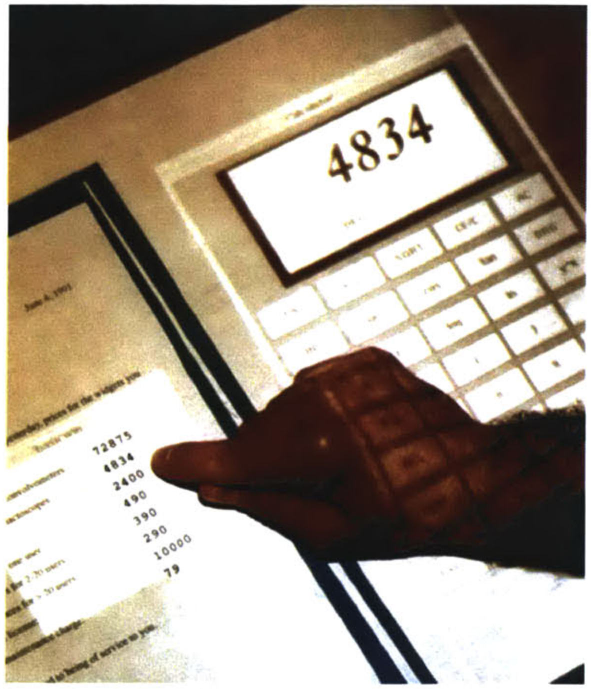


Figure 3-2: The Digital Desk by Pierre Wellner allowed users to interact with projected information overlaid on real paper.

The system used two small CRT displays and mirrors to display 3D information directly in front of the user's eyes as they move around, tracked by a mechanical arm attached to the user's helmet. The displays would change content based on the location of the user in physical space, rendering the appropriate 3D scene from that vantage point in the digital model. The system also allowed for semi-transparent displays with the CRTs reflected off of glass to the user's eyes, allowing the user to see both the real world and the digital 3D display at the same time.

More recently either video-see-through or transparent head mounted displays have been used to overlay information. Myron Krueger developed many early interactions for video AR with his Videoplace system [90]. On the other hand, projected AR has the advantage that many users can see the augmented virtual information without the necessity for wearing any hardware. Pierre Wellner ushered in a new era for Augmented Reality and Ubiquitous Computing with his Digital Desk concept video [182]. Wellner used digital projection, as opposed to head mounted see through displays, to augment the physical world. This system would allow users to interact

Figure 3-3: The Phantom haptic device.

with digital information and the physical world at the same time. For example, users could copy numbers from a physical receipt into a projected digital spread sheet.

Using a Shader Lamp [134] approach to augmented reality, designers can digitally paint on physical 3D objects using 6 degrees of freedom tracked tools and a projection setup [98]. A number of tools can be used, such as digital stencils and spray cans for bimanual manipulation.

Virtual and Augmented Reality open many new possibilities for interaction, but often without the consideration of tactile experience or haptic feedback.

3.2 Haptics

While Virtual Reality often focuses on rendering visual content, it can also be useful to create a tactile experience of a virtual information, whether that is a virtual world, a remote location, or other abstract data. Haptic Rendering is concerned with using a proxy device that is typically used to render tactile illusions of physical touch, collisions and constraints in the interface [145]. Haptic rendering is often combined with Virtual Reality - to create an even more immersive environment.

Electromagnetism can, for example, be used to control a finger or device with an attached magnet [180, 178]. Such haptic interfaces can effectively provide guidance and feedback, but lack perceivable affordances in their static state; the system can only suggest operation during interaction or movement.

In contrast to Haptic Rendering, which often leverages tactile perception alone, this thesis will focus on creating actual shape change or leveraging passive haptics. Thus the interfaces described in this thesis can be both understood visually and tactilely allowing for richer affordances and the ability for these interfaces to be used outside of the lab.

3.3 Tangibles and Graspable Interfaces

As opposed to Haptic Interfaces, Tangible interfaces have explored using passive physical props to control dynamic computation. John Frazer, an architect and inventor, explored some of the earliest tangible interfaces by designing interactive architectural models [43]. Some of his work managed to couple the tangible input with digital output by embedding LEDs into the building blocks, allowing the physical design to inform the user.

Ken Hinckley's pioneering work on passive real-world interface props for neurosurgical visualization demonstrated the advantage of using physical objects control complex computational tasks, such as selecting cross sections in CT data scans [64]. These physical props harnessed the power of bimanual interaction, allowing one hand to hold and manipulate a reference object and the other to point. Fitzmaurice et al. [38] demonstrated the benefits of graspable UIs with Bricks, in which physical affordances are mapped to the control and representation of virtual objects and abstract actions, see Figure 3-4. Here much of the interaction is situated on a smart surface, so the physical interaction with the 'bricks' is co-located with the visual output from the screen, allowing for a more tight coupling between input and output.

A more formal definition of Tangible Interfaces, interfaces with which the user can manipulate physical objects to change a digital model, can be found in Ishii's Tangible Bits [72]. Fishkin provides a good overview of many tangible interfaces and organizes

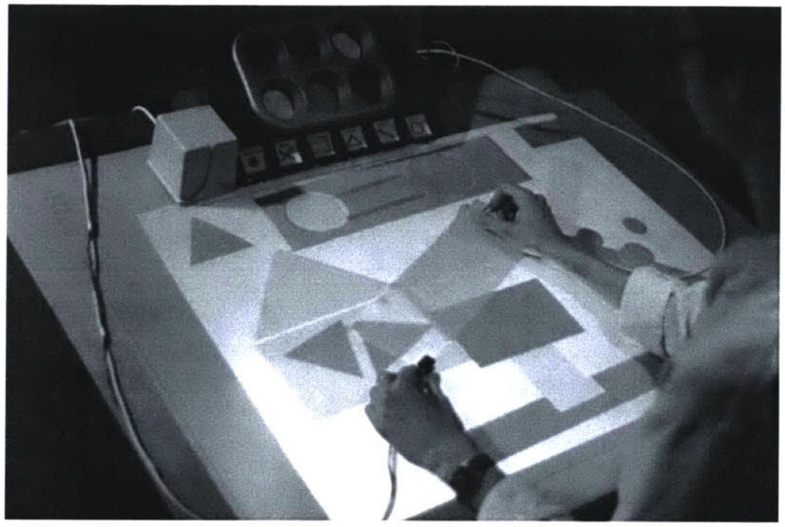


Figure 3-4: The Grasp Draw interface on the Active Desk.

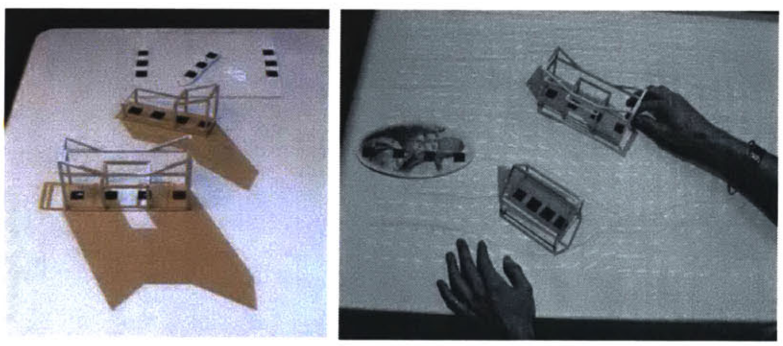


Figure 3-5: URP is a Tangible User Interface for Urban Planners.

them across level of embodiment and other axes [37]. The importance of Fishkin's work is that it clearly explains both tangible interfaces that are fully embodied, in which input and output are fully coincident or collocated, as well as more "distant" tangible interfaces, in which physical objects change a digital representation on a traditional screen, similar to Hinckley's work [64].

The URP project, built on top of the I/O Bulb platform, has an even stronger coupling between tangible blocks, or Phicons, and digital feedback by co-locating projected feedback around the tangible Phicons [169]. URP is an urban planning workbench on which physical models of buildings represent the digital models, allowing the user to easily move buildings around a proposed site. Physical models cast digital projected shadows, and a time dial allows users to see how the shadows change

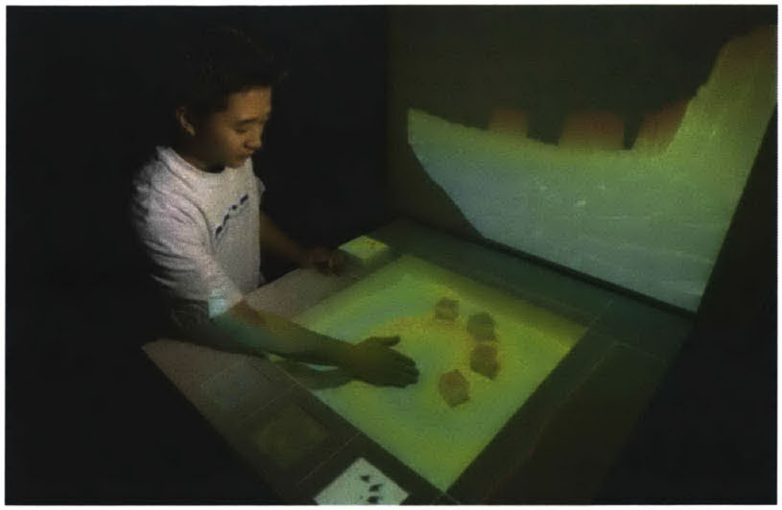


Figure 3-6: Sandscape is a Tangible User Interface for Landscape Designers.

over the course of the day. In addition, wind simulations are projected around the buildings. This digital feedback allows designers to make informed decisions about the placement of buildings, and allows them to easily and quickly try alternatives simply by moving the buildings.

Other tangible systems harness co-located projected feedback to inform the user about design alternatives, or to provide instruction, such as suggested locations for building blocks. CADCast uses projection on wooden blocks and a micro switch to show users step by step building instructions for LEGO or Other Block models [130]. Tangible building block systems can also provide feedback on a variety of different parameters other than placement alone. For example, Senspectra allows users to build structures that can be deformed, and the level of deformation on individual vertices is displayed through color LEDs [95].

In contrast to the discrete world of tangible tabletop, token, or block interfaces, there can also be more continuous input dimensions. Illuminating clay and Sand Scape allowed land scape designers to manipulate a physical clay or sand models of landscapes with their hands [130]. Analog changes in these models were scanned in at 1 Hz using a 3D laser range finder. Projected digital feedback on top of the clay or sand could show the designer simulated water runoff or erosion patterns over time based on the current physical model. In this system the designers are limited to only mirroring the physical sculpture to the digital world, and thus are limited by the constraints of the physical world, for example no undo function, or loading and saving.

Other tangible interfaces have had a closer connection with visualizing and exploring digital information, by enhancing them with physical affordances and constraints. DataTiles [137] are tangible toolglasses [10], that can be arranged on a screen and act as transparent lenses or props that guide the user's pen interaction with the underlying virtual content. The different tile types embed grooved widgets that physically steer the user's interaction through motion constraints. The Token+constraint framework [168] explores this design space further, highlighting the importance of mechanical constraints, and allows users to build database queries by arranging tangibles where rules and meaning are inferred from spatial relationships. SLAP widgets [181] demonstrate how passive controls can provide control and input affordances through optical tracking and sensing.

3.3.1 Actuated Tangible User Interfaces

A natural extension to user manipulation of TUIs is the ability to computationally control them through actuation. The Actuated Workbench [123] uses electromagnetism for 2D movement of tracked tangibles on an interactive surface, while Madgets [179] extends the concept to enable height actuation, to control mechanical mechanisms, and to power circuits through induction in passive tangible assemblies. PICO [127] introduces mechanical constraints as a direct way for the user to specify behavior and rules for actuated tangibles. While the system can communicate system constraints through graphics and object actuation, it is limited in its ability to manifest mechanical constraints; only the user can create those. Other techniques for moving objects on a 2D surface include vibration [138] and robotics [88]

AR-Jig uses a row of 12 actuated linear sliders, which the user holds in one hand [4]. The tool's position is tracked in 6 degrees of freedom in 3D space, and the user wears a head-mounted display to see the digital 3D model. As the AR-Jig tool is moved around the digital model, the sliders change to physically represent a slice of

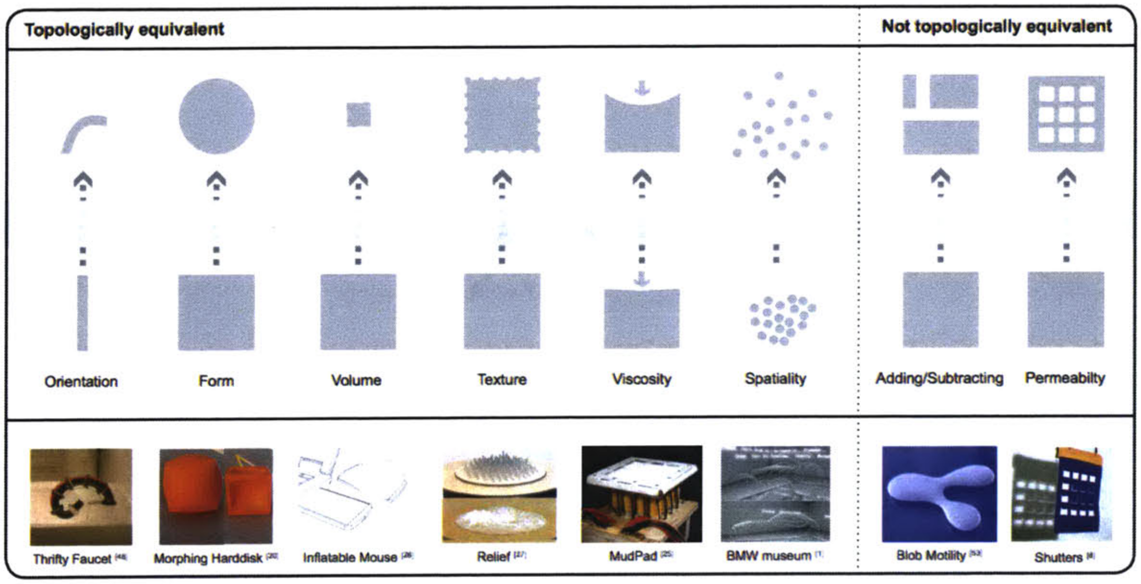


Figure 3-7: Rasmussen et al. introduced this parameter space of shape-changing UI.

the digital model. This slice can be deformed by the user by pushing and pulling on the physical sliders, to create a desired curve.

3.4 Shape-Changing Interfaces and Dynamic Physical Affordances

While affordances can be mechanically designed to be manipulated by the user through tangible artifacts, there is even more interesting potential in the ability for a system to computationally control also physical form in its adaptation of affordances to context and user. Coelho and Zigelbaum [27] and Rasmussen et al. [135] review the design spaces for shape-changing interfaces, where actuation actively modifies the shape of an interface or object. Rasmussen et al. highlight a number of different parameters of shape change, see Figure 3-7. Shape Resolution, based on NURBS surfaces, is another frame work describing the design space of shape change, see Figure 3-8 [142]. Ishii describes his vision of Radical Atoms, and suggests Dynamic Physical Affordances as an interesting area of further research [71].

Less work has explored using shape change for Dynamic Physical Affordances.

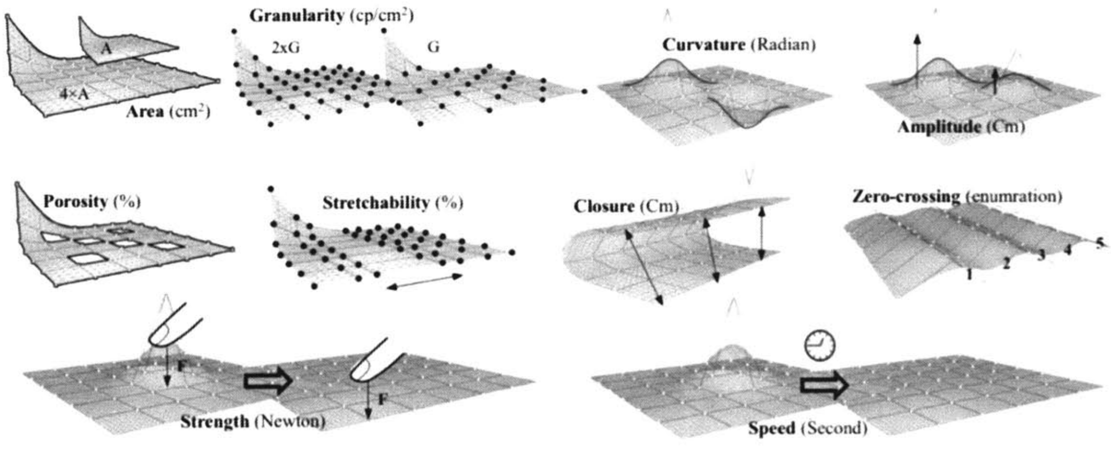
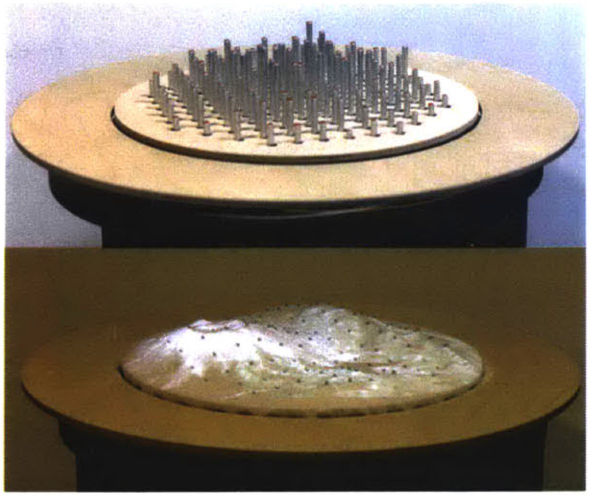


Figure 3-8: Roudaut et al. introduced the shape resolution framework.

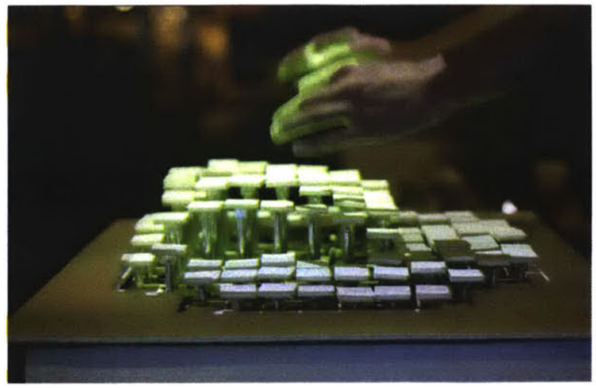
Most current shape-changing interfaces that address on-demand affordances provide a specific transformation, which limits their use for general purpose UIs and 3D interaction. The haptic chameleon by Michelitsch et al. [110] introduces the concept of shape-changing control devices and reports on experiments with early prototypes. Hemmert et al. [59] manipulate tapering and weight shift in conceptual mobile devices. Bubblewrap [8] embeds electromagnetic actuators in textiles to control form, while MudPad [75] uses magnetorheological fluid to manipulate viscosity. Jamming User Interfaces can change the stiffness of an input device to change its affordances through particle jamming [41]. Harrison and Hudson [53] employ pneumatics, whereas Tactus Technologies [23] use microfluidics, to inflate predefined physical buttons, for on-demand tactile affordances on touch screens. Madgets also provide affordances dynamically, by moving them around a surface or mechanically raising or locking elements [179].

3.4.1 Shape-Changing Mobile Devices

Much work on shape-changing mobile devices has focused on tactile notification. Hemmer's work on shape-changing mobile devices, as previously mentioned, explored tapering and weight shifting as a means to display different information, such as navigational directions [59].



(a) The Relief Shape Display



(b) Gestural interaction with the Relief Shape Display.

Figure 3-9: The Relief Shape Display allowed for gestural and direct manipulation of information.

3.4.2 Shape Displays

Shape displays enable more general topologies and greater degrees-of-freedom than other Shape-Changing Interfaces. Shape displays render physical shapes through the use of an array of actuators [73]. This physical rendering is often limited to 2.5D shapes due to the common use of linear actuators, although other topologies have been proposed such as Georgia Tech's digital clay project which described both a bed of pins topology, similar to Iwata's, and a formable crust topology [141]. For the bed of pins Shape Display, each pixel is a physical pin attached to a linear actuator that can move up and down to render 2.5D shapes.

A variety of different technological approaches have been applied for Shape Display actuation: DC motors with Lead screws [73], Rotational Servos [73], pnuematic actuators, and Shape Memory Alloys [132].

Shape displays tend to primarily focus on content representation through graphics and shape; the generated shapes can respond to the user's touch [73], gestures [102], or other objects' presence [97]. Poupyrev et al. do, however, mention the potential for on-demand UI elements in the description of Lumen, a 13×13 array of actuated illuminated rods [132].

This thesis work will explore in more depth the notion of Dynamic Physical Affordances with a general-purpose 2.5D shape display, which allows us to support dynamic adaptation of the form based on user interaction, application context, and scenario. The combination of dynamic surface topologies, their actuated control of tangible objects, user sensing, and object tracking, provides a rich set of capabilities for dynamically controlled perceptible affordances that can optimize user guidance and interaction.

3.5 Malleable and Deformable Interfaces

Organic or Deformable Interfaces allow for more expressive input through more degrees of freedom than traditional Keyboard and Mouse interfaces, as well as many tangible interfaces. In contrast to shape-changing UI, Malleable and Deformable Interfaces are not often self-actuated and instead rely on user's deformation to change an interface's shape. This thesis proposes to apply Malleable and Deformable interface technology to allow for Improvised Physical Affordances. Particularly, this thesis will explore how variable stiffness through Particle Jamming can be applied to deformable interfaces to enable a wide variety of physical affordances.

Particle based material such as beads or sand have been used previously in malleable interfaces, such as in Sandscape [70]. Fluid based interfaces have been also explored [66, 63]. Passive springs can be used to provide haptic feedback for malleable input through the use of mechanical springs [122], gels [177, 40], or foam [157].

One other approach is to use passive deformable props along with active sensing of 3D hand position to approximate deformations on a 3D object; the tracked hand can press into the foam prop to sculpt onscreen graphics [152]. The passive deformable prop can also be tracked in 3D space and used to squash, stretch, or twist 3D models. The passive deformable prop gives haptic feedback and resistance to the user, mimicking the sensation of deformation. In this case, unlike Sand Scape, the foam prop returns to its normal state when the force of the hand is removed and is never truly deformed. A similar approach can be used even if the prop is deformed. For example, this system tracks a foam-cutting hot-wire tool, which cuts a known piece of foam. The changes in the foam block are interpreted based on the 3D movements of the foam cutting tool through the known location of the foam, and reflected back to the digital model [106]

All of these interfaces have a static modulus of elasticity, which provides passive feedback to the user. Organic UIs have explored bending and stretching as a means for input with flexible devices [150, 92, 61]. Jamming user interfaces can contribute to this area of research by controlling the degree to which users can interact with these interfaces and the feeling of interacting with them, i.e. bending a very flexible device vs a very rigid one.

3.5.1 Stiffness Changing Interfaces

Other work in HCI has explored variable stiffness materials for haptic feedback. One common technique has been MR fluid, described previously. Interfaces have used MR fluid for localized haptic feedback on a touch surface [166, 8, 75]. Variable stiffness can also be conveyed through mechanical actuation and take a variety of forms from hand held squeezing [51] to a mouse like interface [107].

Particle Jamming has the ability to vary the stiffness over much larger areas than these other techniques [14]. For fast localized haptic feedback MR fluid should be used. But jamming enables a wide variety of other applications and scales. Wearable force display by Mitsuda et al. is a haptic feedback device for virtual reality applications [111]. It consists of body-worn soft tubes filled with styrofoam particles, which can be jammed to constrain user motion.

HoverMesh by Mazzone et al. applies a similar jamming technology with polystyrene beads to a tangible user interface [108]. It consists of a soft mesh, which can morph into different shapes through computer controlled pneumatic cells. Jammable chambers in the skin of the interface solidify the shape when it's deformation goal is reached. ClaytricSurface by Sato et al. combines a malleable tabletop jamming surface with a ceiling-mounted depth sensing camera and projector as a sculpting interface [147]. The malleable surface contains a pneumatic jamming apparatus, which allows for variable stiffness control. A depth sensing camera is utilized to sense touch input rather than capture the actual 3D structure of the surface and interact with 3D digital content.

3.6 User-defined affordances

A number of researchers have explored interfaces to allow users to quickly adapt and define their own physical affordances to create custom interfaces. Voodoo I/O lets users quickly arrange physical widgets such as buttons, sliders or switches with physical affordances around a 2D deformable surface and easily map these widgets to digital programs [176]. Sauron allows users to easily 3D model new physical interactive interfaces with custom physical affordances that can be 3D printed [148]. Other research has investigated how users may cut or shape their own interfaces using printable electronics [119, 49]. This thesis seeks to enable users to define physical affordances faster than previous approaches by leveraging shape-changing, deformable, and variable stiffness interfaces.

Chapter 4

Dynamic Physical Affordances and Constraints

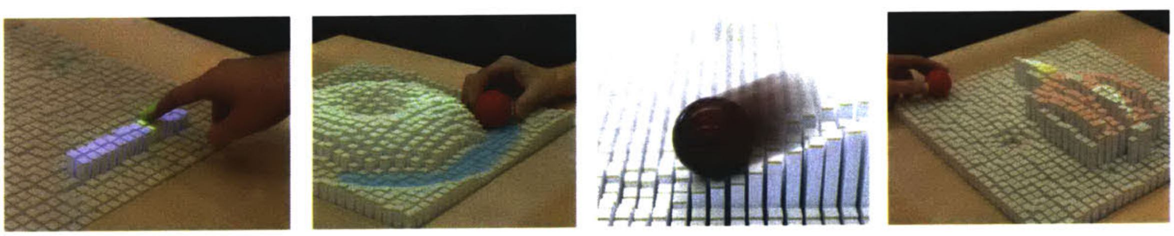


Figure 4-1: inFORM enables new interaction techniques for shape-changing UIs. Left to right: On-demand UI elements through Dynamic Physical Affordances; Guiding interaction with Dynamic Physical Constraints; Object actuation; Physical rendering of content and UI.

Our goal is to overcome the limitations of Graphical User Interfaces that lack rich physical affordances, such touch screen based interaction. As described in the previous chapter, much work on Tangible User Interfaces has sought to provide physical affordances for digital information. While GUIs have the ability to change signifiers rapidly to adapt them to different content and contexts, TUIs primarily exploit the affordances inherent in physical form, as well as their physiological and cognitive advantages [87]. For example, the Token and Constraint framework introduced by Ullmer uses mechanical constraints to provide physical affordances for interacting with tangible controllers, such as tokens [168]. However, TUIs, such as those outlined by Ullmer, are often limited by the static nature of most man-made physical artifacts, and thus cannot easily change their form.

We seek to bring the dynamism of signifiers in GUIs to physical interaction by utilizing shape-changing UIs. This chapter explores Dynamic Physical Affordances which can transform shape, size, location, and orientation, in addition to being able to appear and disappear. They provide appropriate affordances on demand by changing their physical properties based on program states and the context of the user or other objects in the interaction area to *facilitate* interaction. Buttons can, for example, grow in size to ease target acquisition, or move out of the way of an object. These are what we call directly rendered Dynamic Physical Affordances, meaning that a user interacts directly with them through touch and deformation, see Figure 4-2. We also introduce Dynamic Physical Constraints, which help mediate interaction between the interface and tangible tokens or tools. We call these *object mediated* Dynamic Affordances, as a user interacts with them through a tool or object. Dynamic Physical Constraints not only provide affordances to the user, but also serve to mechanically *restrict* object motion. Constraints limit the degrees of freedom through which users interact with the system, allowing for more precise input in each dimension. These build on and expand Ullmer's Token and Constraint work [168].

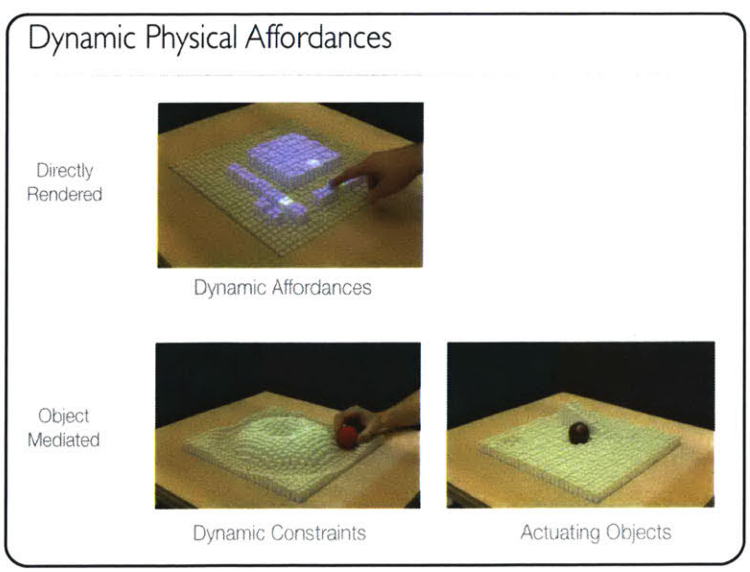


Figure 4-2: The two types of Dynamic Physical Affordances: Directly Rendered and Object Mediated.

In addition to creating affordances and constraints for physical objects and tools, we also show how shape change can be utilized to *manipulate* passive objects to appropriate them as Dynamic Physical Affordances. Mechanical forces can push objects, causing them to roll, slide, or tumble in one direction or another. Other constraints can be programmed to rotate or raise passive objects. Passive tangible tokens can be moved to maintain state, and devices, such as phones, can be raised to draw attention to them. Our techniques allow for a wide variety of physical objects to be actuated. This opens up interaction possibilities that point towards tabletop systems that can more easily interact with the world around us.

To explore these techniques and interactions, we introduce the inFORM system, a state-of-the-art 2.5D shape display that enables Dynamic Physical Affordances, Constraints and actuation of passive objects, see Figure 4-3. Shape displays allow for more general-purpose shape change than many other actuated or shape-changing interfaces, and thus are ideal research platforms. The inFORM system supports fast 2.5D actuation, malleable input, and variable stiffness haptic feedback. While shape displays still remain limited in scale and cost, this work is an exploration of the interaction capabilities and is meant to inspire further research in this area. Our belief is that shape-changing interfaces will become increasingly available in the future, and this work tries to push towards creating a vocabulary and design space for more general-purpose interaction for shape displays, including rendering of both content and UI elements.

In this chapter, we discuss the design space of Dynamic Physical Affordances and Dynamic Physical Constraints, and provide methods for using these concepts to also actuate objects. We describe three implemented demonstration applications that highlight different aspects of our concepts, followed by a technical overview of our system. We also present the results from two user studies concerning Dynamic Physical Affordances: the first concerning using motion in Dynamic Physical Affordances, and the second regarding their performance in 3D manipulation tasks. Finally we discuss the implications, limitations, and future potential of the inFORM system.

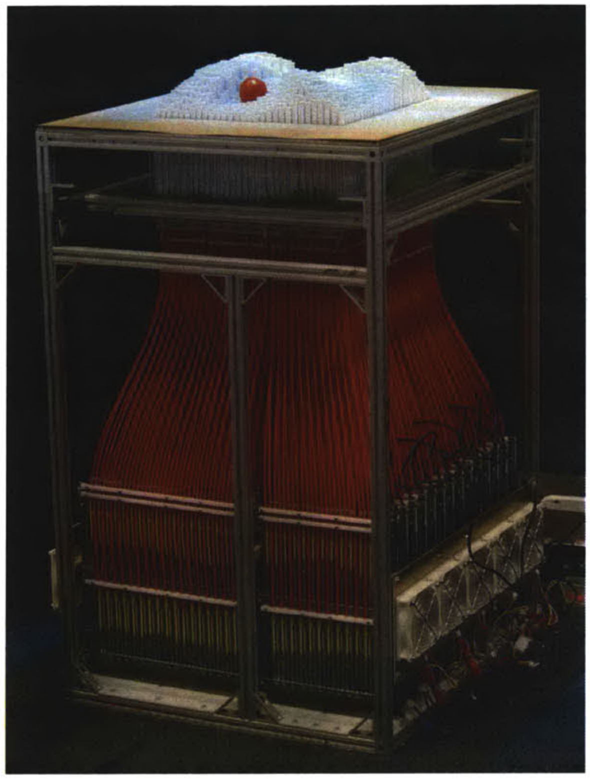
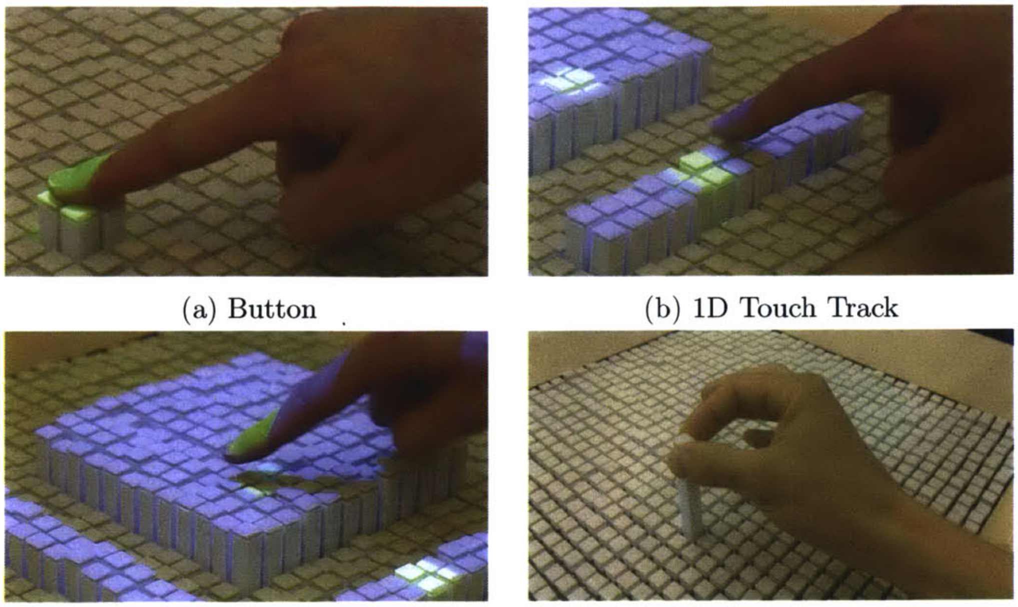


Figure 4-3: The inFORM shape display used to render Dynamic Physical Affordances.

4.1 Contributions

- An exploration of the design space of Dynamic Physical Affordances and Constraints.
- Explorations in the use of motion and animation for physical affordances.
- Actuation of physical objects through shape displays.
- State-of-the-art system for fast, real-time 2.5D shape actuation, co-located projected graphics, object tracking, and direct manipulation.
- Three applications that demonstrate the potential of these interaction techniques for HCI.
- An evaluation of the performance of Dynamic Physical Affordances.
- An evaluation of the perceptual qualities of motion in shape change for physical affordances.



(a) Button (b) 1D Touch Track (c) 2D Touch Surface (d) Handle

Figure 4-4: Dynamic Physical Affordances transform the UI to facilitate interactions.

4.2 Dynamic Physical Affordances and Constraints

Past research on shape displays has primarily focused on rendering content through shape output, with less emphasis on investigating dynamically changing UI elements. We propose an analysis of dynamically generated physical features with specific affordances that guide the user on how the system can be used and provide passive haptic feedback, enabling interaction at a lower cognitive cost [87]. We believe that shape displays need to provide three types of functionality for creating dynamic UIs: to *facilitate* through Dynamic Physical Affordances, to *restrict* through Dynamic Physical Constraints, and to *manipulate* passive objects through shape change.

4.2.1 Directly Rendered Dynamic Physical Affordances

Dynamic Physical Affordances function both as perceived affordances and "real" affordances, as they are rendered physically and provide mechanical support for interaction. We can combine graphical perceived affordances with Dynamic Physical Affordances, or switch between these states.

In the inFORM system, these affordances are physical elements that the user can touch. Depending on how they are rendered by the system, they either directly react to touch, or react to displacement from the user pushing or pulling them. Figure 4-4 depicts a set of different Dynamic Physical Affordances rendered on our system. Examples of UI controls with Dynamic Physical Affordances that our system supports are:

Binary Switches: Buttons

Buttons are formed by raising pins from the surrounding surface. Users activate a button by touching it or by pushing it into the surface, which is registered as a binary input.

1D input: Touch tracks

Touch tracks consist of a line or curve of adjacent raised pins, which the user can touch at different locations, or slide over. These touch points are registered in one input dimension.

2D input: Touch surfaces

Touch surfaces are created using multiple pins, which are aligned to form surfaces. These surfaces, which can be non-planar, map each touch point to two dimensions.

Handles

Handles provide interaction in the Z dimension. These raised pins can be grabbed, and then pulled up or pushed down along one dimension.

Interactions with Dynamic Physical Affordances

Affordances can change shape to reflect a changing program state. For example, when a user presses a play button (triangle shape) it can transform into a stop button (square shape).

Shape-changing affordances can also enable smooth transitions between input dimensions. For example, pressing a button could cause it to transform into a 2D touch panel, as illustrated in Figure 4-5.

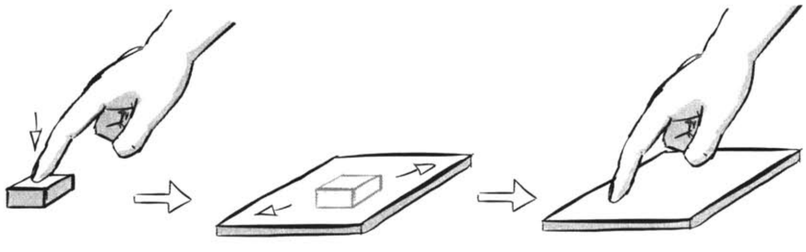
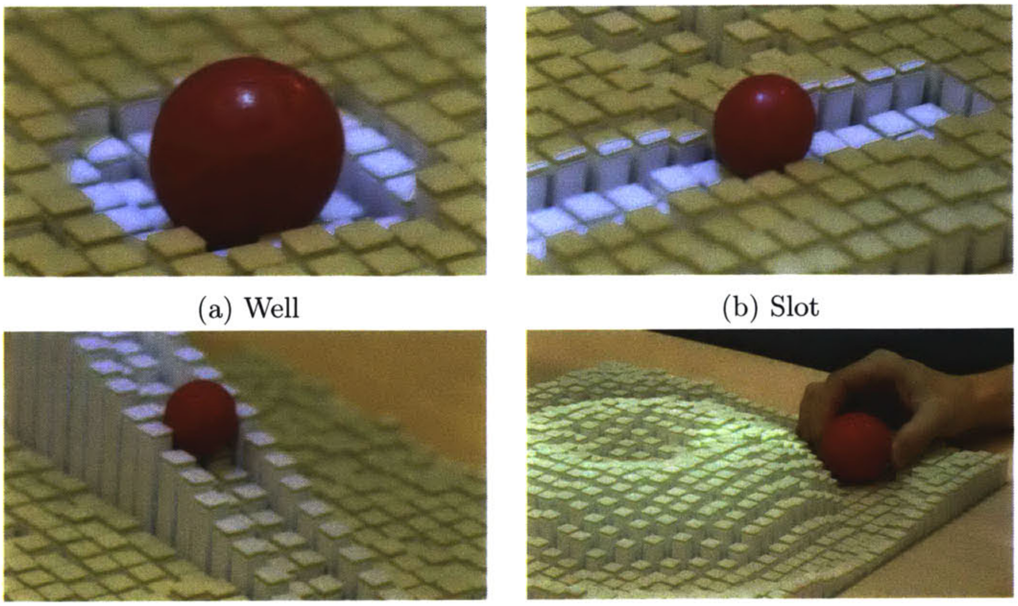


Figure 4-5: *Dynamic Physical Affordances* can physically transform between UI elements, for example, by transforming a button into a touch surface.

It can be advantageous to let the user's proximity inform shape change. Affordances can, for example, increase in size as a user's hand approaches them, making them easier to acquire according to Fitt's law [109].

Affordances can also move out of the way of physical objects, or rearrange to provide more space for interaction. Besides avoiding physical objects, affordances can compliment them to increase their functionality. As a physical object is moved, the affordances can appear or follow it. For example, as a device is placed on the table, relevant physical UI controls can appear; a phone could be complemented with a large answer button next to it, or a tablet could have buttons to control games appear around it.

The inFORM system is particularly well-suited to guide complex interactions and adapt the affordances when the user may perform multiple actions using the same controls. Gaver refers to *sequential affordances* [45] in situations where the affordances change based on the interaction, to enable new possibilities or restrict actions. A UI control can, for example, be rendered with different stiffness to provide further affordances. Two flat surfaces might appear the same, and both afford touching, but once the user touches their surface, a stiff surface affords touch interaction, where a more compliant surface affords pressing. More interestingly, with systems like inFORM, such qualities can be dynamically changed and updated based on interaction



(a) Well (b) Slot

(c) Ramp (d) Surface

Figure 4-6: *Dynamic Physical Constraints* guide the user by limiting possible interactions.

and context.

4.2.2 Object Mediated Interaction with Dynamic Physical Constraints

While Dynamic Physical Affordances facilitate user interactions, Dynamic Physical Constraints limit the possibilities, making some interactions difficult or impossible to perform. These Dynamic Physical Constraints make the system more legible, but also guide the user in performing certain interactions through physical interaction with the constraints. They can also help mediate interaction through tangible tokens or tools.

When an object is placed on a shape display, it physically interacts with the shapes generated by the display. In the context of our work, we refer to the physical objects as tokens and the shapes interacting with them as constraints. Figure 4-6 depicts different types of constraints. Constraints like wells, slots, and ramps limit the movement of the token through their shape, thus guiding user interaction, similar to [168].

As our system can sense how tokens interact with constraints, it can dynamically modify their parameters (shape, size, location, orientation) to adapt to user input or to reflect changing program states. Examples of techniques to guide interactions using shape change:

Holding Tokens and Sensing Presence: Wells

Wells act as containers to hold objects. Placing a token inside a well or removing it, is sensed as a binary action. The shape of the well and the shape of the token determine if the token can be rotated in the well, which adds another degree of freedom.

Restricting Movement to 1D: Slots

Slots are grooves which constrain the direction in which a user can move a token. In addition to the same actions that are supported by the wells, tracked tokens can also be moved in the slots to, for example, control a 1D parameter.

Affecting Movement: Ramps and Curved Surfaces

The surface geometry can be changed to make it easier or harder for the user to move tokens in a certain direction. Ramps can, for example, be used to facilitate or restrict movement of a token due to gravity.

Interaction with Dynamic Physical Constraints

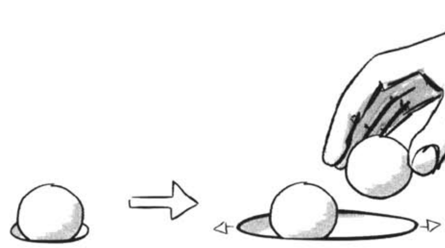


Figure 4-7: A well transforms in size to accommodate additional tokens.

Wells can transform in size and shape to adapt to the size, shape and number of tokens. They can also deepen to move tokens outside the users reach (see Figure 4-8).

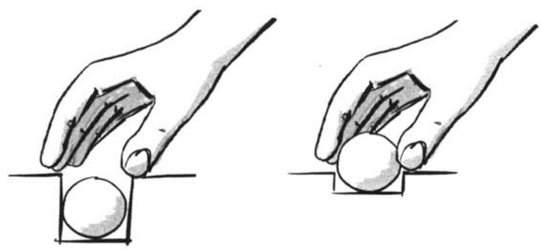


Figure 4-8: The depth of a well changes if the user is able to grasp a token contained in it.

Slots change in size, shape and location to reflect a changing program state. A user can, for example, place a token inside a well, which is equivalent to selecting a top level menu item. The well then transforms into a slot, similar to an expanding menu. As the user moves the item inside the well, its shape can transform and branch out to present selectable options.

Slots can also transform their shape to promote the movement of tokens in a certain direction or to hinder it. An example is shown in Figure 4-9 (right), where a ramp-shaped slot allows users to roll a token with ease in the direction sloping downwards, while requiring deliberate effort to move it in the upwards direction. In addition to ramps, slots can contain further constraints to provide haptic feedback as the user moves the token through the slot. Such feedback can be provided through slots with detents at the bottom (Figure 4-9, left), ramps with drops, or vibration of the entire slot.

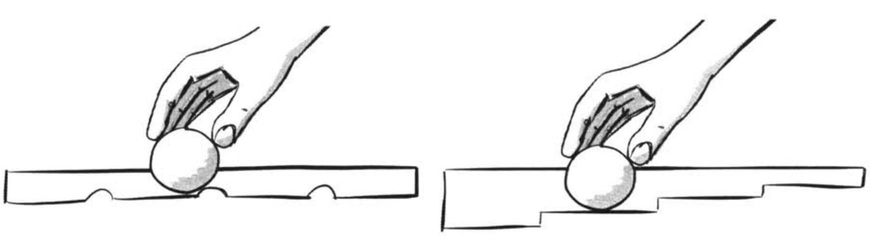


Figure 4-9: Slots with indentations and ramps can be used to guide the user's interaction or to provide haptic feedback.

4.2.3 Manipulate: Actuating Objects with Shape Displays

Shape displays can appropriate passive objects by independently actuating and manipulating them to create *object mediated* Dynamic Physical Affordances. This way, passive objects can be augmented with dynamic capabilities, expanding their possible use as tangible tokens or tools that represent program state or other functionality.

The shape display can apply mechanical force to an object and cause it to move in a variety of ways. This greatly expands opportunities for interaction, and inter-material interaction, as well as solving a problem inherent in passive tangible systems: keeping tokens' physical state synchronized with the digital state. Additionally, it allows the shape display to output greater degrees of freedom (e.g., lateral movement), and enables greater degrees of freedom afforded to the user for input. Our techniques for actuating passive objects do not require an active or special material (such as magnets), but instead manipulate geometrical shapes, with the limitation that certain geometries (such as a ball) are easier to move than others. Other factors to consider include the mass of the object, the force of the motors, and the friction between the shape display surface and the passive object.

Manipulating Objects on the Surface through Actuation

Any object can be lifted vertically by the system as long as the actuation force is sufficient (1.08 N/actuator for inFORM). Objects placed on the table can also be tilted to lean in one direction, with computational control of tilt angle and orientation. This can be used, for example, to orient an object's surface towards a user.

In addition to lifting and tilting, objects can be translated on the X-Y surface through three techniques. Firstly, objects can be lifted and caused to slide or roll down an inclined plane rendered by the surface, essentially using gravity to cause it to move (see Figure 4-13). Secondly, given the right shape, the vertical actuator movement can push an object sideways or induce rolling (see Figure 4-10). This works by applying a force on the object offset from its center of mass, inducing a moment on the object, either causing it to roll or slide out of the way. This actuation method works best for objects with angled or rounded features, like spheres, cones and cylinders. To ensure overlap with at least four pins at a time, our current system actuates spherical objects of at least 25.4 mm diameter. Thirdly, tilt can be used for controlled tumbling of objects about the X- or Y-axis, by alternating tilting and catching, to move the object on the surface. Tumbling works well for cubes or other angular geometry. These different techniques have varying levels of legibility to the user—the inclined plane, for example, makes it clear where the ball will move.

Objects can also be rotated about the surface's Z-axis through similar techniques as used for X-Y translation. Currently only certain objects, with conical or rectangular shapes can be rotated. However, a simple surface feature to allow locking an object at an anchor point could allow for rotation through the aforementioned methods of pushing-induced sliding.

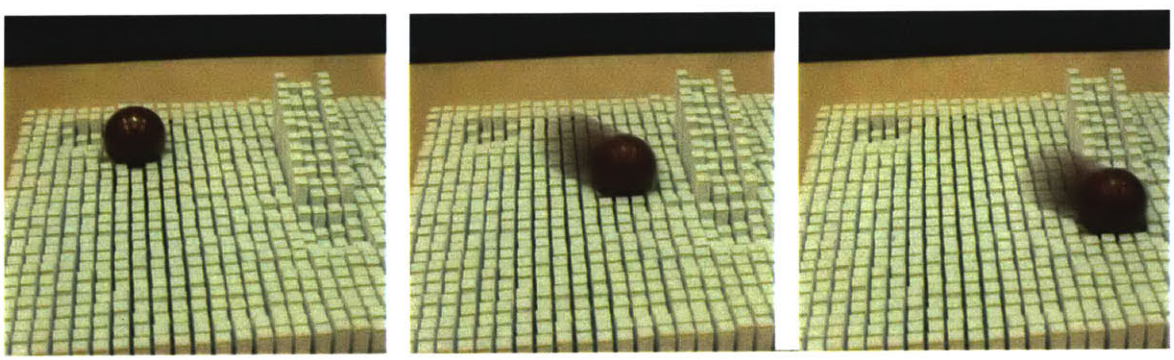


Figure 4-10: inFORM can lift, translate, tilt, or rotate objects on the surface by changing the surface geometry. Here, a ball is moved on the surface.

In-air movement through Ballistics

Objects can be projected ballistically from the table into air. The use of multiple pins allows the launch angle to be computationally controlled. With sufficient tracking, the object could also be smoothly caught to dampen impact and avoid subsequent bounces. Currently, we are able to launch a 7 g ball with 20 mm diameter, approximately 80 mm above the maximum pin height of the surface.

Vibrations for Haptic Feedback or Attention

Vibrations by the actuators underneath an object can cause the object to shake. This can provide haptic feedback or draw attention to that object.

User Interaction with Actuated Objects

Similar to PICO [127], users can physically stop the interface from actuating objects by lifting them or holding them in one position. This can be used to control interface behavior of moving parts. In addition, users can place static physical barriers on the table to prevent tokens from moving. These objects can be of arbitrary shape or make use of the space in between the actuators. An example is a wall barrier made from a cardboard sheet, or a cup that is placed over an object to prevent it from being moved. The constraints defined by the shape display surface could also be user defined, for instance, by deforming the shapes directly with bare hands.

4.2.4 Parameter space

The physical properties of a UI element strongly influence its perceived and real affordances. In his definition of affordances, Gibson [46] lists a number of such properties: "When the constant properties of constant objects are perceived (the shape, size, color, texture, composition, motion, animation, and position relative to other objects), the observer can go on to detect their affordances." This definition includes dynamic parameters like motion and animation, which static TUIs do not possess as real affordances. Shape-changing interfaces, on the other hand, have the ability to add such parameters, and can be categorized as changes in orientation, form, volume, texture, viscosity, spatiality, adding/subtracting, and permeability [135].

We find it attractive to utilize 2.5D shapes display to render physical affordances, as their hardware capabilities enable simultaneous control over multiple parameters. In the following list, we identify parameters that both contribute to affordances and can be dynamically controlled by inFORM.

Shape: The shape of UI elements can provide multiple affordances, real affordances (how the shape can be touched), and cultural affordances (what the shape represents). Their quality and expressiveness is tightly coupled to the possible resolution and degrees of freedom [102]. This creates interesting interface design challenges that must be considered, in particular, for 2.5D shape display hardware.

Size: The size of a UI element is constrained by the user's physiology and available space. It has to be sufficiently large for the user to manipulate, while also small enough to fit in the interface. While static physical UIs have to compromise between these two factors, dynamically resizing UI elements can enable better ergonomics and use of space, given that they can provide sufficiently smooth and continuous transitions at their spatial and temporal resolution.

Position and Orientation: The spatial relationship between objects is an important parameter for TUIs. While these can be dynamically modified, users cannot easily grasp, lift, and rearrange objects on 2.5D shape displays. Therefore, we propose to complement them with passive physical tokens that enable these interactions, while they can also be constrained and actuated by the display surface shape.

Color: Color and visual texture can be applied to provide additional graphical perceived affordances, using embedded display [132], projection [102], or augmented reality [100].

Haptic feedback: The material and haptic feedback of an element communicates to the user if it affords actions like deformation. Systems with mechanical actuation can provide haptic feedback by dynamically changing the resistance of a pin when pressed by the user.

Visibility: By rapidly changing the size of a dynamically rendered element, it can appear and disappear. An advantage compared to static systems is that physical objects can be rendered in succession, rather than having to permanently share valuable space. The rate of change is hardware dependent.

Motion: Motion describes the change of the above parameters over time. While motion to switch between predefined static affordances has been explored previously [53], carefully choreographed motion adds a compelling dimension to the interaction. If the UI element transforms continuously as the user interacts with it, the motion itself turns into an expressive affordance. The quality of motion in the context of HCI has previously been described by the path, volume, direction, and velocity of an object [172].

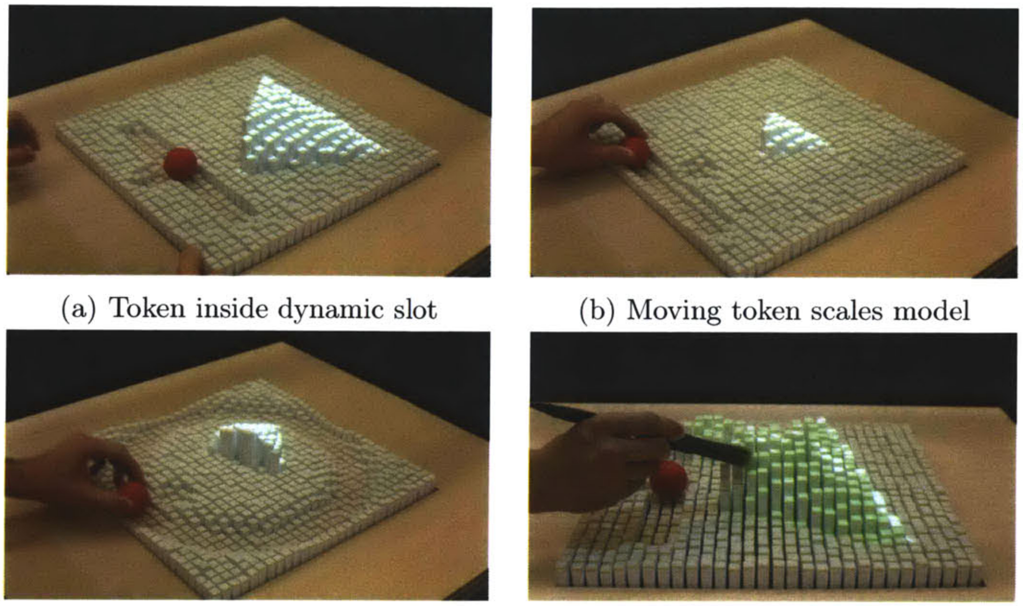
4.3 Demonstration Applications

4.3.1 3D Model Manipulation (handles, constraints, context)

The 3D Model Manipulation application demonstrates how the inFORM system's dynamic capabilities can be used to render physical representations of 3D models that the user can flip through, and then use tokens and tools to transform, edit or paint (see Figure 4-11).

Users move a token through a slot to browse the different 3D models and can select it by placing the token in an adjacent well (Figure 4-11a). When the token is placed in the well, it transforms into a slot. Moving the token to any end of the slot will select the function of the token; rotation, translation or scale. The slot transforms again to represent the degrees of freedom of the current mode. In rotation mode, the slot is a circle circumscribing the model (Figure 4-11c), scale mode uses a linear slot (Figure 4-11b), while translation mode has no constraints. As the token is moved in these constraints, the object dynamically transforms. To exit the current mode, the user places the token in the *select well.*

The user can also use a 3D brush tool to add geometry, erase geometry, or paint on the 3D model, as shown in Figure 4-11d. To change modes for the brush, the user presses a foot pedal to activate a context menu which is rendered physically as a 2D Dynamic Physical Affordance. The menu appears offset from the current location of the brush in 3D space, allowing for quick selection. The bristles of the brush move smoothly over the shape display surface, while being optically tracked by the system.



(a) Token inside dynamic slot

(b) Moving token scales model

(c) Rotation in circular slot

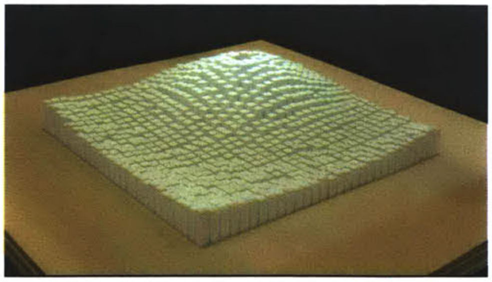
(d) Painting geometry with tool

Figure 4-11: The 3D Model Manipulation application uses tokens and tools to browse, transform, and edit physical representations of 3D models.

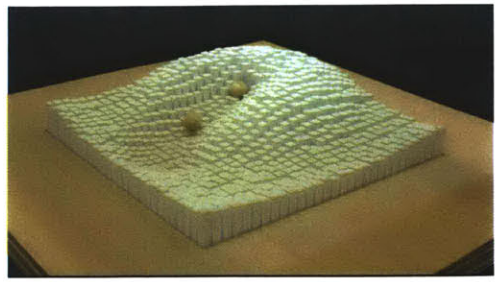
4.3.2 Marble Answering Machine

The Marble Answering Machine [28], is a tangible interface to receive, store and play back voice messages that are represented as physical marbles. Its iconic form, sketched by Durrell Bishop, is a continuous surface with a raised hill and a hole in it. Our homage to the original design, implemented with the inFORM system, uses a dynamically changeable form, and demonstrates how Dynamic Physical Constraints transform to reflect changing program states (Figure 4-12).

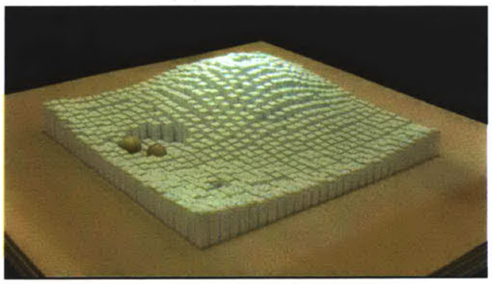
New messages are represented by marbles, which are ejected from the hole (Figure 4-12b) to roll into a newly formed well that stores messages (Figure 4-12c). To listen to a message, users pick up a marble from the *new message well* and place it inside the *play well* (Figure 4-12d). The *play well* transforms into a slot, and as the message is played back, the marble is moved inside the slot, representing its relative playback position. At any given time, users can pick up the marble to stop the message, or scrub it in the playback slot to replay parts of the message. Once the message is played, the machine moves the marble to the *old message well*. Dropping the marble back into the hole on top of the machine erases it.



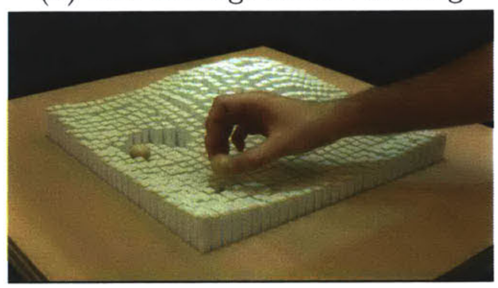
(a) Idle state



(b) New message marble arriving



(c) New messages in well



(d) Moving marble to *play well*

Figure 4-12: inFORM's version of the Marble Answering Machine uses dynamic shape and constraints to reflect program state.

This system demonstrates the ability to render both aesthetic form and UI elements. In addition, it highlights the ability for the user to directly intervene while an object is being actuated. In the spirit of PICO [127], the user can also introduce mechanical constraints to control computational behavior. By placing a rigid sheet in the path of the playback track, the user can limit message playback to the beginning of the audio file.

4.3.3 Actuating Interactive Devices on the Surface

Beyond passive tokens, we can also actuate interactive devices, such as phones or tablets. Such devices can be similarly moved, tilted, rotated, and vibrated. A smartphone could, for example, be tilted towards the user upon a phone call or alert (see Figure 4-13), or to preserve privacy in a collaborative session when private information should not be seen by others.

The dynamic capabilities of inFORM could be used to render physical UI elements on-demand, for example, by having physical buttons and touch elements emerge when a tablet is placed on the surface. These UI elements provide physical affordances for

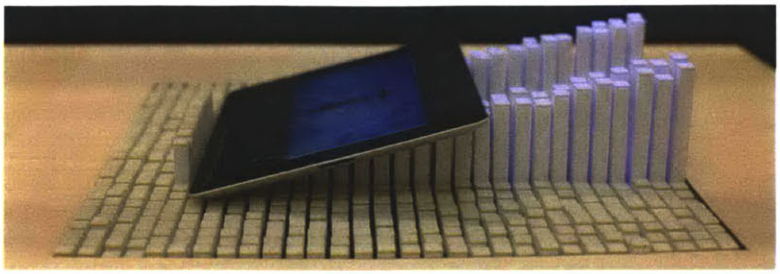


Figure 4-13: inFORM can actuate devices, for example, by sliding and tilting a tablet towards the user.

applications on the tablet. Vibration and haptic feedback could be used to augment gameplay, for instance.

4.4 inFORM Shape Display Implementation

4.4.1 Prototyping

In order to find the right resolution and scale, the inFORM system was first prototyped through a stop motion animation. We first built a number of different resolution passive prototypes to investigate spacing, and then found that 9.525mm spacing was adequate to display a wide variety of objects. A full scale passive prototype constructed out of plywood was created to be used in the video prototyping process, see Figure 4-14. In the video prototype, a user creates a 3D model of a boat using the shape display. Based on the results of this prototype, we decided to move forward with the production of inFORM.

4.4.2 Final Implementation

The system uses 30×30 motorized white polystyrene pins, in a 381×381 mm area. The pins have a 9.525 mm^2 footprint, with 3.175 mm inter-pin spacing, and can extend up to 100 mm from the surface. Push-Pull rods are used to link each pin with an actuator, to enable a dense pin arrangement independent of actuator size, giving the system a height of 1100 mm. The linkage, a nylon rod inside a plastic housing (Sullivan Gold-N-Rods), transmits bi-directional force from a motorized slide poten-

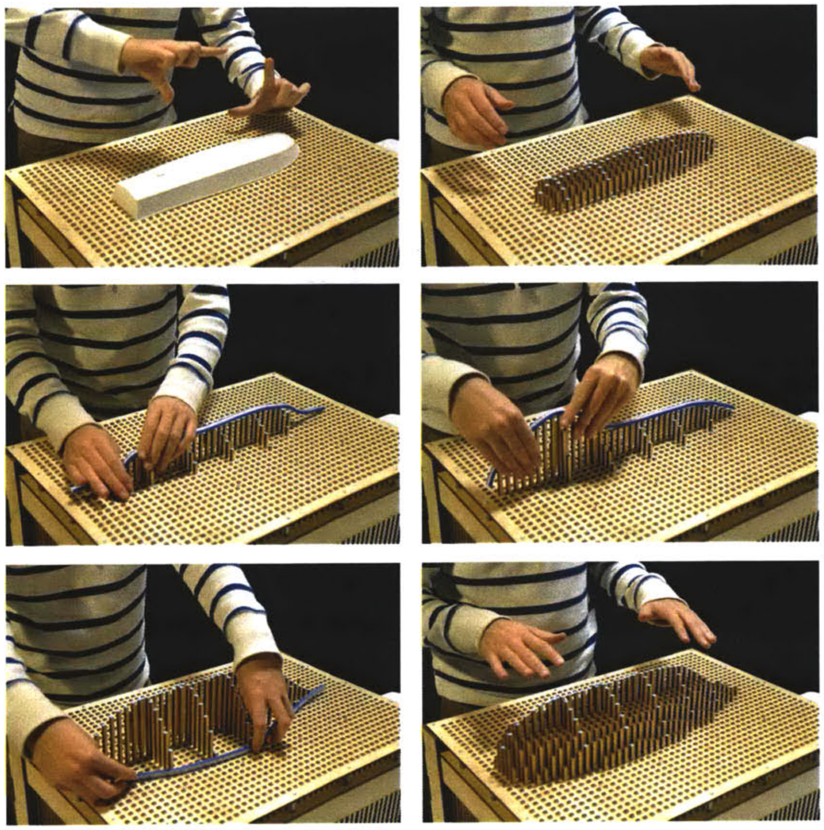


Figure 4-14: A stop motion prototype was created to envision what interacting with a shape display similar to inFORM could be like.

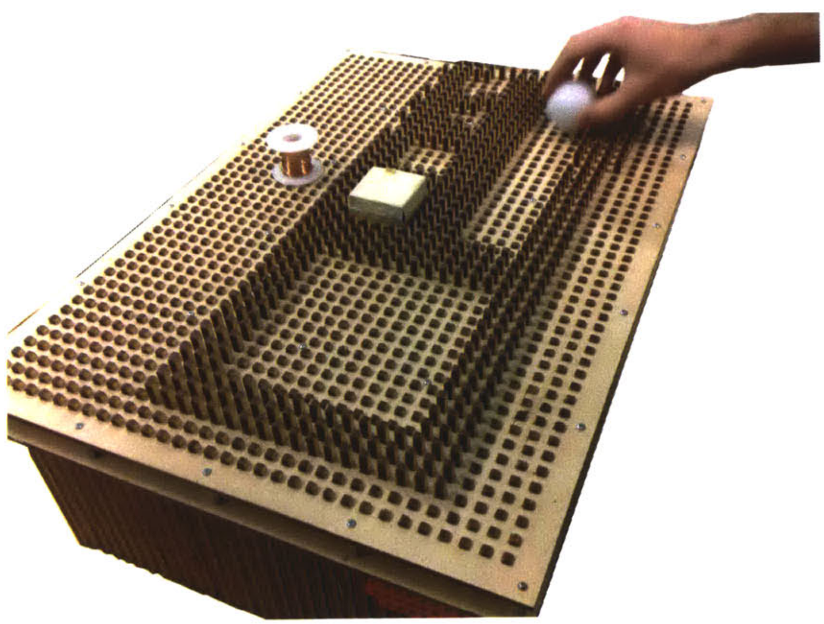


Figure 4-15: Prototyping different Dynamic Physical Affordances with the passive prototype.

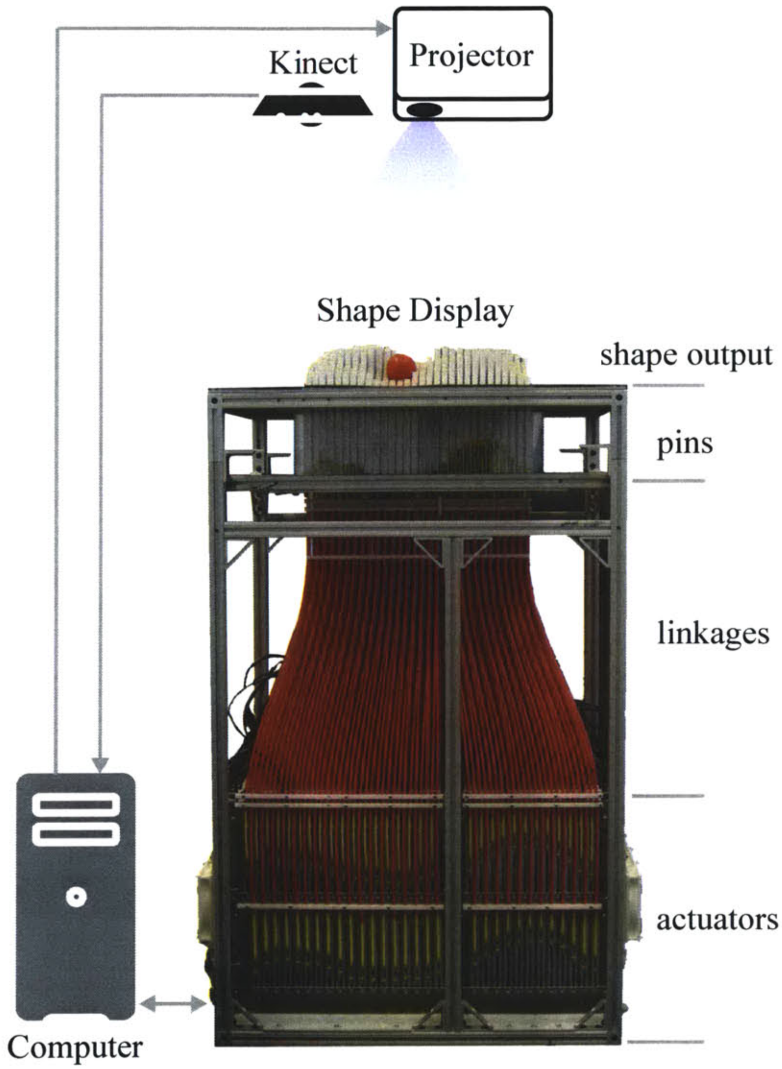


Figure 4-16: The inFORM system actuates and detects shape change with 900 mechanical actuators, while user interaction and objects are tracked with an overhead depth camera. A projector provides visual feedback.

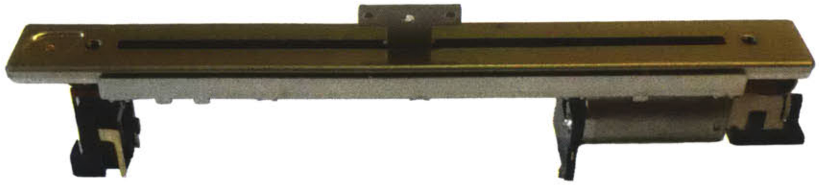
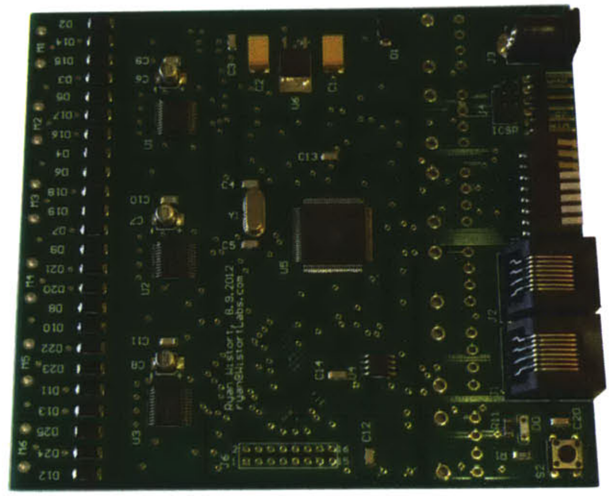


Figure 4-17: A single Alps RSA0N11M9A07 Motorized Slide Potentiometer used in inFORM.

(a) A single Motor Module with 6 actuators.



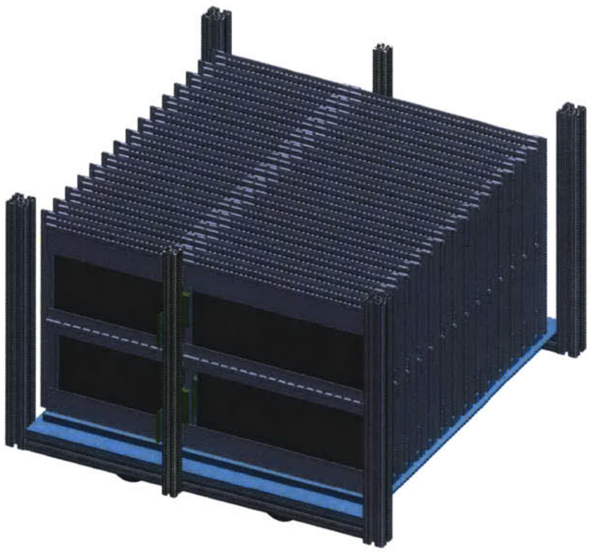
(b) The PCB Control Circuit for inFORM

Figure 4-18: The inFORM Motor Module and PCB design allows for modular assembly of different shape displays.

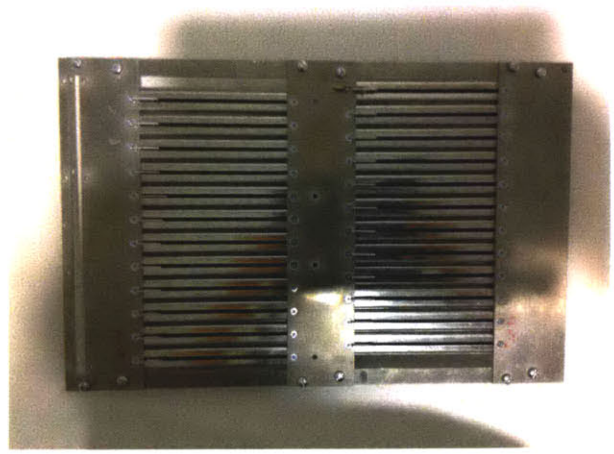
tiometer (ALPS RSA0N11M9A07, shown in Figure 4-17), through a bend. Six slide potentiometers are mounted onto a custom-designed PCB (see Figure 4-18), powered by an Atmel ATMega 2560, and TB6612FNGCT-ND motor drivers, see Appendix A for more details. The linear positions are read by the 10-bit A/D converters on the microcontroller, and allow for user input, in addition to servoing their position using PID control.

150 boards are arranged in 15 rows of vertical panels, each with 5×2 boards, see Figure 4-19. The boards communicate with a PC over five RS485 buses bridged to USB. The system has a 60 Hz refresh rate, determined by the 115200 bps RS485 bus speed, the 8 byte control message, and 30 boards on each RS485 bus.

For each pin, we can update both position and PID terms to provide haptic feedback and variable stiffness, for example, to create haptic detents or buttons that



(a) CAD design of the lower assembly of inFORM's actuation panels.



(b) A single actuator panel with 4 Motor Control Boards and 24 actuators.

Figure 4-19: inFORM's actuation panels contain 4 or 6 motor control boards for a total of 24 or 36 actuators, respectively.

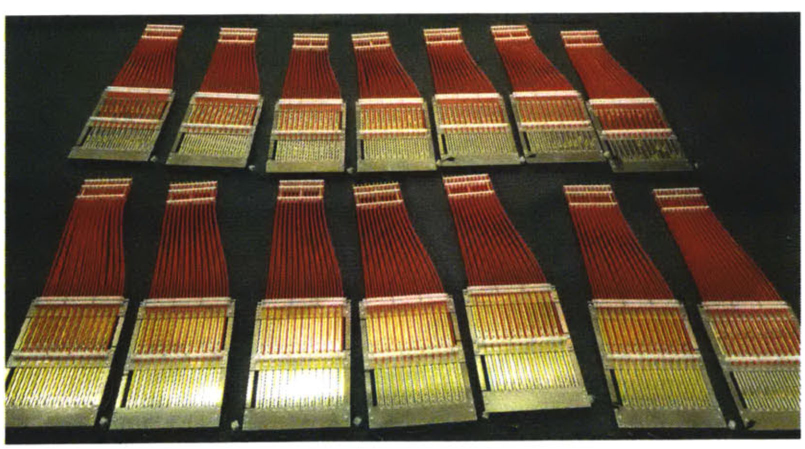


Figure 4-20: inFORM Actuation Panels with flexible linkages mounted.

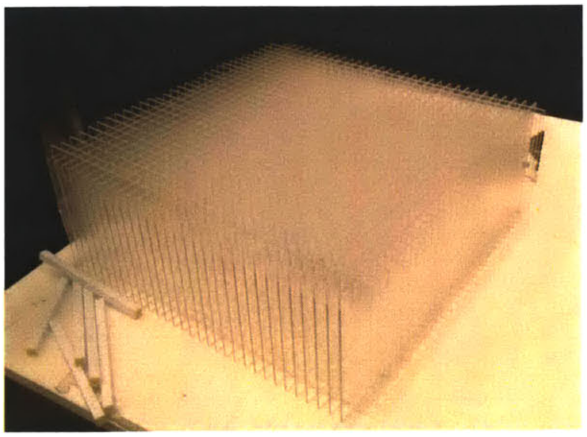


Figure 4-21: The top of the shape display contains a acrylic grid which the 900 plastic pins slide in.

are harder to press. This control also allows us to limit power consumption per pin to avoid burning out the motors.

Pin height and stiffness is represented in software as an 8-bit height map, which can be produced by OpenGL shaders, through different shape primitive classes, or by directly writing data. The height map is then sent to the microcontrollers. Similarly, all 900 pin heights can be received over the RS485 bus, and used to track user's surface deformations as input.

Each pin can exert a force of 1.08 Newtons (equivalent to 100 g weight), which was measured using a precision digital scale. The effective average upwards and downwards speeds (0.644 m/s and 0.968 m/s, respectively) were measured using a high speed camera.

In theory, the system's 900 pins could consume up to 2700 W due to a peak 3 W power consumption per actuator. In practice, however, we measure the effect to approximately 700 W when in motion. Due to friction, the pins maintain their static position unpowered. Therefore, the power consumption to maintain a shape is less than 300 W (mainly due to the 83 % efficiency of our 12 V power supplies). Heat dissipation remains a critical design criteria for the actuation assembly and we use two rows of five 120 mm fans to cool the actuators.

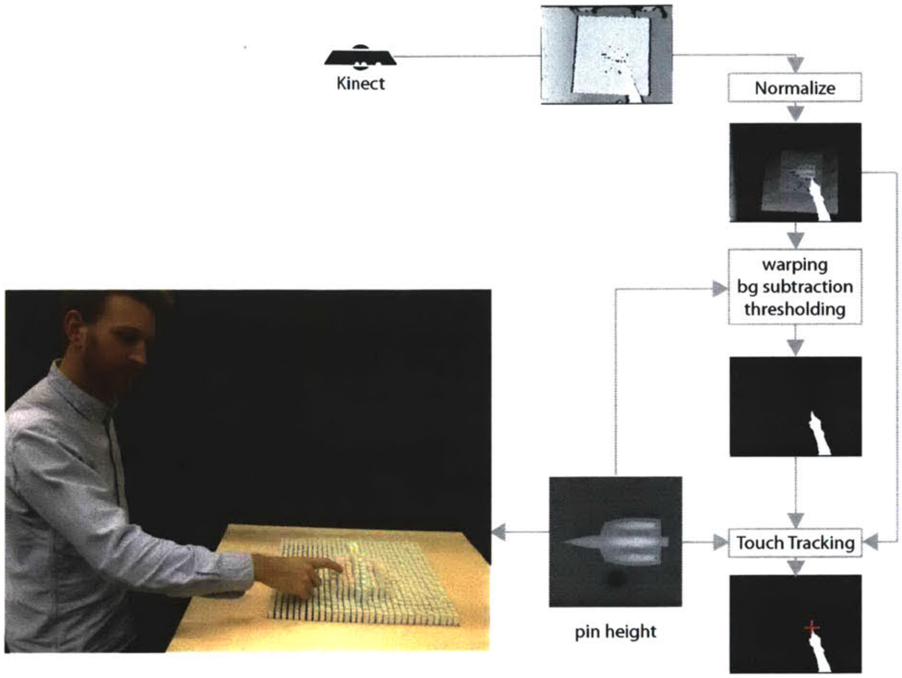


Figure 4-22: The inFORM Touch Tracking Pipeline.

User and Object Tracking

Our current system uses an overhead depth camera to track users' hands and surface objects, as shown in Figure 4-22. A Microsoft Kinect with a 640×480 pixel depth sensor is mounted 1200 mm above the surface and calibrated for extrinsic and intrinsic camera parameters. We combine a static background image of the table surface with the surface's real-time height map to form a dynamic background image that is used for subtraction and segmentation. The height map is scaled and a homography is applied to warp the image into camera space. We find all pixels above the computed dynamic background model, and threshold the image. For hand and fingertip tracking, we use OpenCV to compute contours of the threshold image, followed by convex hull and convexity defects. The 3D finger tip coordinates are transformed to surface space, and an arbitrary number of finger tip contacts can be tracked and detected to trigger touch events. For object tracking, we currently rely on color tracking in the HSV space. However, other approaches to object tracking would be straightforward to implement. Objects and finger positions are tracked at 2 mm resolution in the 2D surface plane, and at 10 mm in height. The touch tracking can be combined with input from the slide potentiometers to, for example, distinguish button presses from light touch.

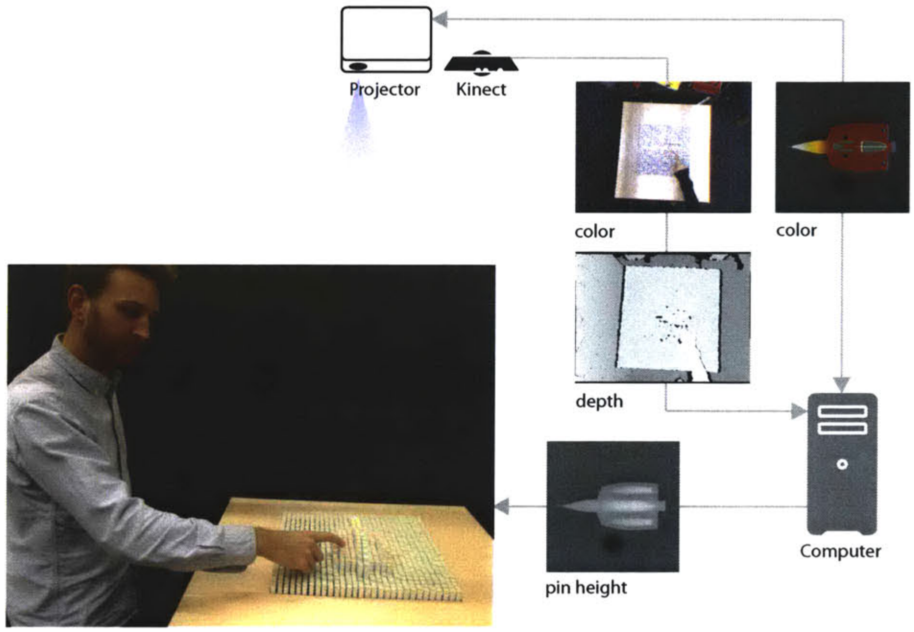


Figure 4-23: inFORM I/O System Diagram.

Graphics and Projection

Graphics are rendered using OpenGL and openFrameworks. The projector's world coordinates were determined after calibration with the Kinect color camera, while distortion coefficients and intrinsic parameters were recovered using ProCamCalib [5]. The projector's 1400×1050 pixel output is then warped to correct for the parameters and projected over an 87×66 cm area, see Figure 4-23.

Technical Limitations

As an alternative to the limited depth camera touch tracking, it may be interesting to embed touch sensors, e.g., capacitive or optical, directly in the pins. More sophisticated object tracking, for example, using multiple cameras, markers, or sensing in the electromagnetic or RF domain, could be used to address user occlusion, object identification, and color interference with projector.

The limited scalability due to the current architecture with one actuator per pin, has significant implications on cost and footprint to move beyond the current relatively low resolution. Also, the current implementation is limited to 1D input and output for each pin, which would be difficult to increase without significantly increasing complexity. While stiffness can be controlled, pin shape, spacing, and material choices limit the affordances and constraints that the system can generate and how they interact with external objects. We, however, believe that the system's spacing, resolution and the interactive speeds rendered, are sufficient to allow the prototyping of many interactions that would be challenging on other existing shape displays. The current cost and scale of our shape display hardware limits its primary use to research.

4.5 Exploring Motion in Dynamic Physical Affordances

Motion has been explored as an important feature of perceived affordances and signifiers in graphical user interfaces [76]. However, motion is less thoroughly investigated in physical affordance, due to the more complex nature of making physical objects move. The physical motion of an object provides affordances - for example something that is bouncing is harder to pickup than a static object, or a fast moving object, such as a fan blade, affords not touching. Motion can also provide context and preview potential functionality of physical affordances, as well as drawing attention.

Physical motion and people's perceptions of it has been explored in the domain of Social Robotics and Human Robot Interaction. Some research has investigated different qualities of robot motion to make a robot's movements be perceived as safer [91]. Others have studied how motion plays a role in animacy, likability, and perceived intelligence [6]. Another approach has been to use motion to guide human collaborators and help them understand the intent of robots [32, 112]. These types of research questions apply directly to that of shape-changing interfaces, and their role in Dynamic Physical Affordances.

Some researchers have begun to investigate users' perception of motion in shape-changing user interfaces. Pedersen et al. studied effects of varying different parameters of shape change [128]. It used the model of Roudaut et al. used to generate the different shape changes [142]. They narrowed down the 10 parameters from the model into 5: Area, Curvature, Amplitude, Zero-crossing and Speed. They rendered video of shape change and had users on Mechanical Turk answer questions about their perception of the interfaces. However, the space of motion they analyzed was very narrow.

We were interested in exploring more expressive types of shape-changing motion, to see how they could be applied to creating richer physical affordances that leverage the dynamic abilities of shape displays. We were particularly interested in exploring how affordances could be imbued with emotional content through motion. For example, what would an excited button act like, and how would people want to interact with it compared to a sad or depressed button? In order to investigate this space we wanted to understand how users would perceive Shape Change, and how different motion parameters would effect changes in perceived affect of affordances, and perceived usability of affordances. Our goal was to develop specific motion patterns that would be richer than simply changing speed or size by working with a puppeteering system, to allow expert users to create animations.

4.5.1 Generating Motion Affordances Through Puppeteering Shape Change

We wanted to investigate if a rich set of expressive notifications and affordances could be developed for a Shape Display. Previously, we had animated all of the motion using key frame animation, but in order to more quickly create shape change we developed a system that would allow for puppeteering of shape change. Because the

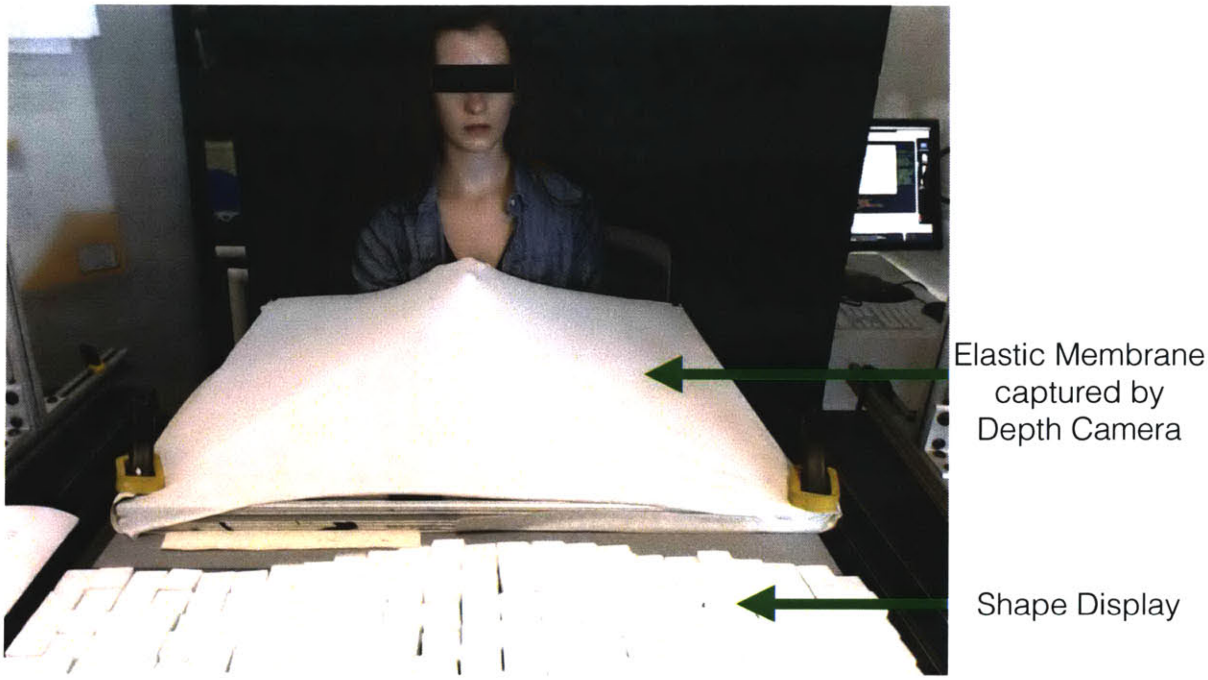


Figure 4-24: The puppeteering system used to create shape change animations.

Shape Displays we have developed can only output 2.5D shapes, it is a good match for shape capture using a 2.5D depth sensing camera. Given that Clynes was able to have participants convey emotion through expressive touch (pressure over time) using two fingers, we believed it would be possible for users to convey emotion through our system [26, 25].

A pilot study with two users was run to compare how well motion animations could be puppeteered using 3 different tool conditions: bare hands, a sheet of 11x17 paper, and an elastic mesh. A depth camera was mounted on a frame 30 inches above the area of interaction, and users would place their hands, paper, or the mesh in that area. For the elastic mesh condition, users deformed the surface of the mesh from below. Changes in the shape of the paper or mesh were rendered directly to the shape display. We found that the interpolation from the mesh, the passive haptic feedback from the spring forces, as well as the fact that it did not capture the users arms created a much easier puppeteering interface. Figure 4-24 shows the final puppeteering system in action.

Once we had found an ideal puppeteering interface, we wanted to capture a large set of motion animations from experts. Our goal was to have participants create animations for shape display which could be applied to affordances, but particularly had different emotional content. We choose 13 different emotions we wished to capture. The 13 emotions were roughly evenly spaced around Russell's circumplex model of affect [143]. They were: alarmed, angry, bored, calm, content, delighted, excited, frustrated, happy, relaxed, sad, tense, tired.

We wanted to find people skilled in conveying emotion through motion. We selected dancers as the participants as they use their bodies skillfully to create many emotional motifs. Nine amateur dancers from a local university participated (8 Female). They responded to an advertisement to campus dance group mailing lists and were compensated 25USD for their time.

Procedure

The participants performed individually. They signed a consent form, were informed of the procedures of the study, and how the system worked. First participants had 10 minutes to practice with the puppeteering system to get accustomed to how it worked and what its limits were. Next participants saw one of 13 emotions on a screen, and had 20 seconds to prepare, and then 10 seconds to record shape change that conveyed that emotion as best they could. Then the next emotion would come on the screen. The order of the 13 emotions was randomized for each participant. After all 13 emotions were performed a post test survey was completed.

Motion Animation Data Set

Nine participants generated 13 videos each for a data set of 117 unique animations that can be displayed on a shape display. One participant was excluded from the study as he or she did not complete all of the videos, leaving us with 108 unique animations.

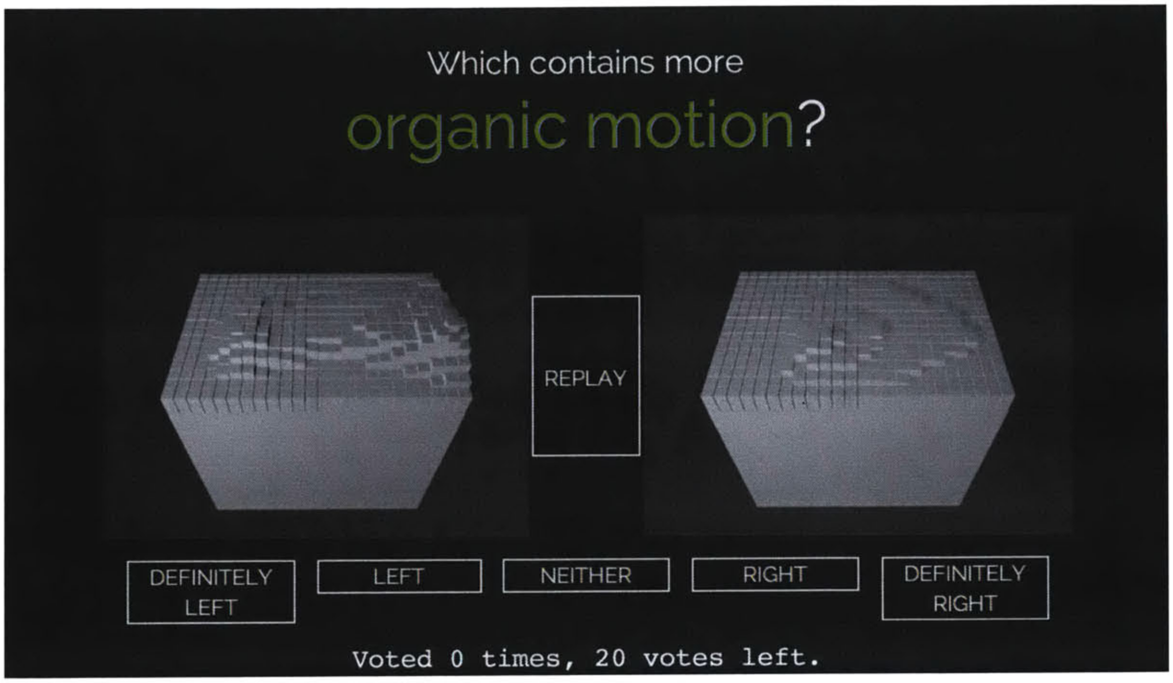


Figure 4-25: Online voting tool, based on QUANTIFY system.

4.5.2 Online Evaluation of Perceived Emotion and Perceived Motion Qualities of Physical Motion Affordances

Once we had recorded all of the participants' puppeteered emotion animations we wanted to know both how the intended emotions correlated with perceived emotion, but also how motion parameters would correlate with the perceived emotions. In order to more quickly have a large number of participants in our study we choose to render the shape change as video animations and use participants on Mechanical Turk to rate them, similar to [128]. We used a system developed by the MIT Viral Spaces and Macro Connections groups, called QUANTIFY, to host the videos and record votes. The QUANTIFY tool has been used to rate the perceived emotional qualities of animated gifs previously . The study interface displayed two videos sequentially, and then after both videos had finished playing allowed users to vote, see Figure 4-25.

We asked participants to watch two 10 second videos, and to vote on which video better conveyed a given emotion. 518 unique participants each voted on 20 pairs of videos, giving us over 10,000 votes. Participants were paid 0.70 USD for approximately 3.2 mins of work.

We used a similar apparatus to measure perceived motion qualities in the previously generated video data set. We used a subset of Young et al.'s motion parameters to label the videos (Young 2005). The parameters were Volume (size of shapes), direction (horizontal, vertical), Path (linear, circular, rhythmic), Speed (velocity, acceleration, energy), and Quality (Organic / Mechanical / Random). See Appendix B for more information on the questions asked. Again, participants on Amazon Mechanic Turk viewed two 10 second videos, and voted on which video had faster motion, etc. For the motion classification 241 unique participants cast 5338 votes. Participants were paid 0.70 USD for approximately 3.2 mins of work.

Results

The QUANTIFY tool uses TrueSkill, a bayesian rating system from Microsoft Research, to rank each video for each question [60]. TrueSkill rankings converge much faster than other ranking algorithms (often in 12-36 contests), and it has been previously used to rank the qualitative data from photos, such as which photo looks like a safer place [113].

In this fashion, each video was ranked for each perceived emotion, and each perceived motion parameter. Thus each video is given a score for each emotion and each perceived motion parameter, scaled to 0-50. Next the Pearson's r correlation between each perceived motion parameter with each perceived emotions was computed, and the P-values for each correlation were also computed. See the chart in Figure 4-27 to see the correlations between perceived emotional content and perceived motion qualities.

There are much stronger correlations in the arousal axis of the circumplex emotion space, than in the positive/negative valence axis, suggesting that it may be easier to convey intensity level but harder to convey positive or negative emotions through shape change. Faster Motion correlates with positive activation/ arousal in the circumplex affect space ($p<0.01$). Similar results were found when measuring affect in arm movements [131], which suggests that motion parameters used to influence affect in other domains such as 3D animation [57, 54] can map to shape output.

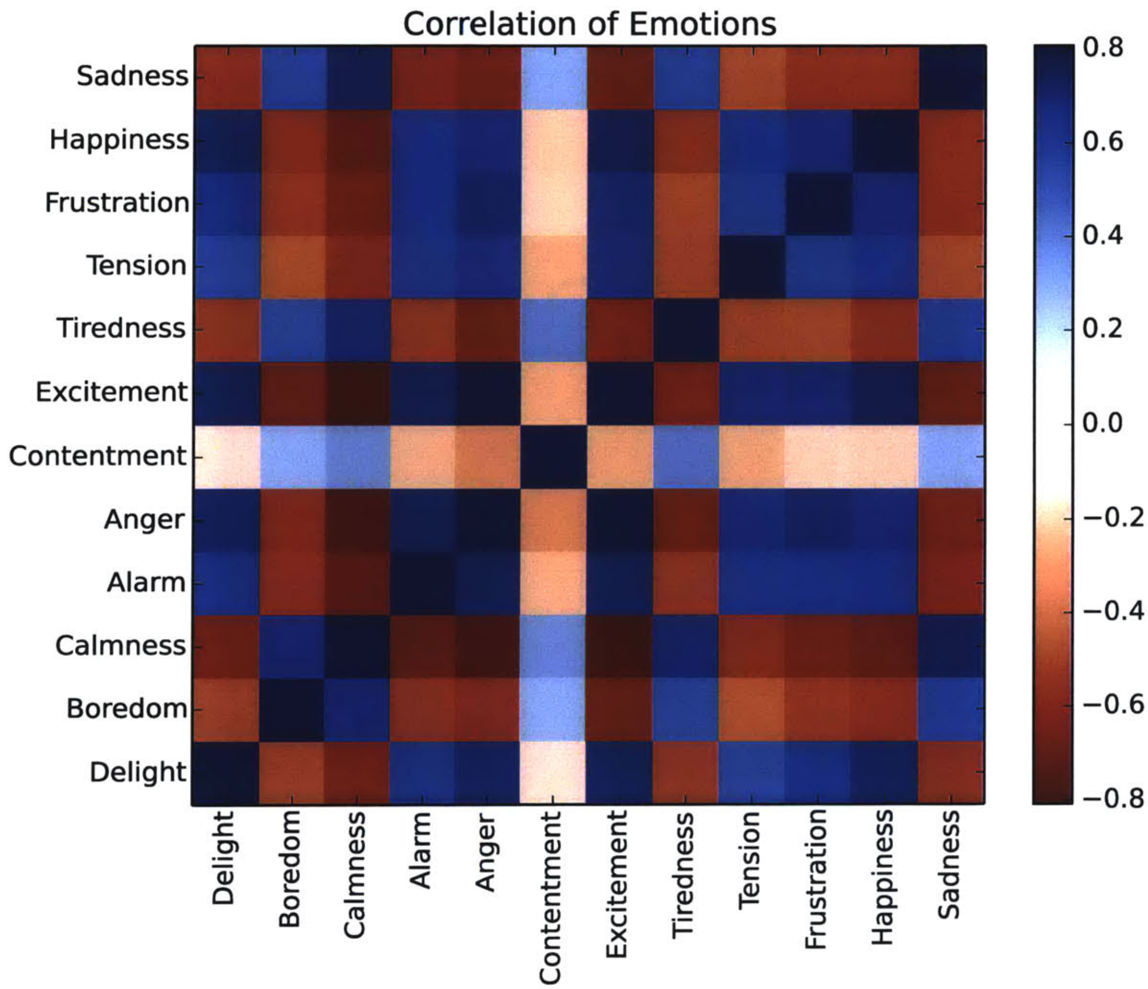


Figure 4-26: Correlations between perceived emotional content and other perceived emotional content.

row.names	relaxation	delight	boredom	calmness	alarm	anger	contentment	excitement	tiredness	tension	frustration	happiness	sadness
organic.motion	-0.2	0.3	-0.2	-0.2	0.3	0.3		0.3		0.3	0.4	0.4	
more.energetic	-0.6	0.6	-0.5	-0.7	0.6	0.7	-0.2	0.7	-0.6	0.6	0.7	0.7	-0.5
random.motion	-0.4	0.3	-0.3	-0.3	0.3	0.3		0.3	-0.2	0.3	0.4		-0.3
faster.movement	-0.8	0.5	-0.6	-0.7	0.7	0.7	-0.4	0.6	-0.6	0.5	0.6	0.6	-0.5
more.sudden.movements	-0.7	0.6	-0.6	-0.6	0.7	0.7	-0.4	0.7	-0.6	0.6	0.6	0.5	-0.6
changes.shape.more	-0.6	0.7	-0.6	-0.7	0.7	0.7	-0.2	0.8	-0.5	0.7	0.7	0.8	-0.6
rhythmic.motion	-0.3	0.5	-0.3	-0.5	0.4	0.4		0.5	-0.3	0.4	0.4	0.5	-0.3
mechanical.motion	-0.4	0.6	-0.5	-0.6	0.6	0.6		0.7	-0.4	0.6	0.6	0.7	-0.6
vertical.movement	-0.5	0.6	-0.4	-0.6	0.6	0.6		0.6	-0.5	0.5	0.5	0.6	-0.5
horizontal.movement	-0.2	0.3	-0.3	-0.4	0.4	0.4		0.4	-0.3	0.5	0.4	0.5	-0.3
more.circular													
bigger.shapes	-0.5	0.6	-0.5	-0.6	0.6	0.7		0.7	-0.5	0.7	0.6	0.7	-0.5

Figure 4-27: Correlations between perceived emotional content and perceived motion qualities. Green is positive correlation, red is negative correlation, white squares did not have a statistical significance ($p<0.05$)

However there were some motion parameters that correlated more strongly along the positive/negative valence axis. For example, rhythmic motion correlates more to happiness, than rhythmic motion to frustration ($p< 0.05$), even though both happiness and frustration correlate to faster motion. Beyond this, random motion correlates more strongly to frustration (Pearson's $r = 0.43035164$), than happiness (Pearson's $r = 0.17539522$), ($p<0.05$). This suggests that if one wants to evoke a positive emotion through shape change one should use more rhythmic motion, and for negative emotion more random motion.

4.6 Evaluating Performance Using Dynamic Physical Affordances

In order to evaluate the performance of Dynamic Physical Affordances, we conducted a user study in which we attempted to measure the advantages of shape output combined with spatial graphics. In the study, we tested the following hypotheses:

1. Physical input using Dynamic Physical Affordances is easier and faster than mid-air gestural input for spatial manipulation tasks when interacting with spatial 3D information. We believe that the haptic feedback provided by shape output is

advantageous compared to mid-air interaction with only virtual graphics.

2. Multi-point, two handed manipulation of a 3D surface is easier and faster than single point haptic interaction; users can use their full hand to interact with the system as opposed to just fingers or single points and this is advantageous.

As highlighted by [135] there are few user evaluations of shape displays, we hope to provide and contribute insight into that area as well.

4.6.1 Experiment

In order to investigate these hypotheses we wanted to choose a task domain that would be in the area of actual use that we imagine for the Dynamic Physical Affordances, and allow for bimanual interaction, so we chose a CAD related task: 2.5D mesh manipulation. We wanted to compare manipulation of the mesh using 1) a mid air pointing device or wand, 2) manipulating the physical shape display through physically manipulating its pins, with either single or two handed manipulation. The single-handed manipulation was constrained to interacting with a single pin at a time in order to simulate an interaction close to a phantom. As users had to grab each pin to move it, we hypothesized that it would be easier to push than pull the pins. We did not test a multi hand wand condition, as we used the non-dominant hand to toggle selection. Because the pin display is limited to one degree of freedom per pin, we constrained the mesh vertices only to y-displacement in all conditions. All interaction was direct.

We chose to run our study using a See-Through-AR display mounted above a shape display, called the Sublimate system (see Figure 4-28), as it provides for higher accuracy matching of graphics and shape output and leaves two hands free for input [100]. Sublimate uses an earlier, lower resolution Shape display, called Relief [101]. Participants wore active shutter stereo glasses. We used a Vicon motion capture system to track the user's head with a tag placed on the stereo glasses. We rendered view dependent graphics based on the head position. The virtual graphics were the same 3D scene in all conditions, and they were rendered at screen resolution of 1920X1080 on a 27" screen in portrait mode.

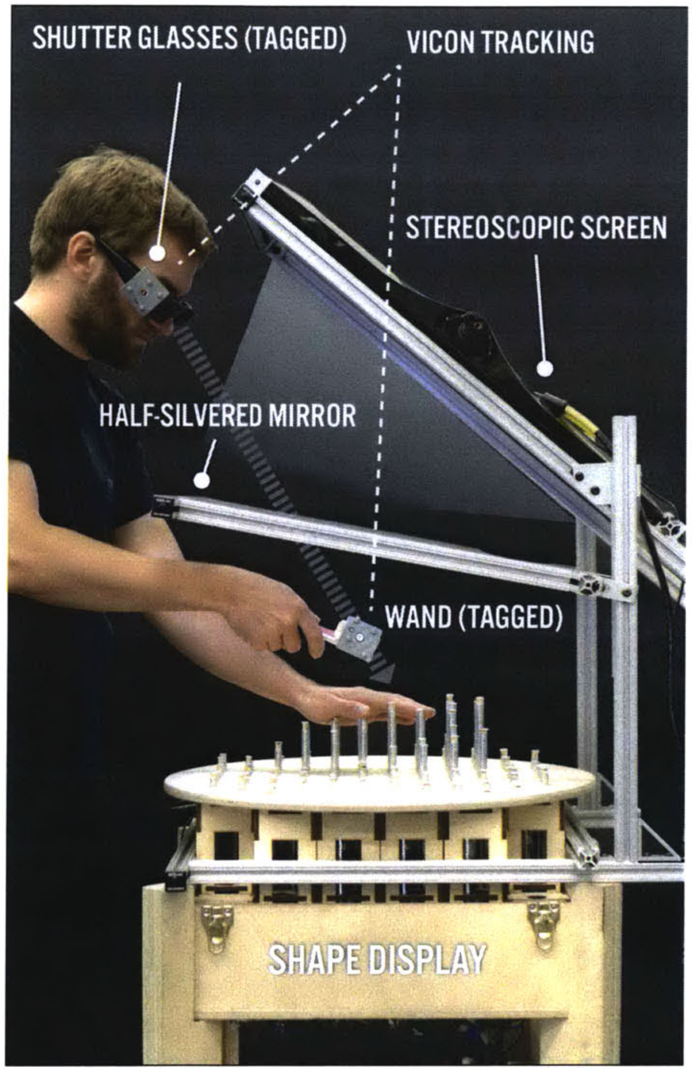


Figure 4-28: The Sublimate system combines spatial AR with a shape display and was used to study the performance of interacting with shape displays.

To insure accurate tracking of 3D input we opted to use a Vicon motion capture system and a pointing wand, as opposed to a depth camera. To avoid errors caused by shaking induced by button presses on the wand we used a secondary button triggered by the non-dominant hand for selection with the wand. For physical input and output we made use of the shape display's physical pins without using the mesh top. The pins were 10mm in diameter, and had a vertical travel of 100mm.

Participants

Ten participants (4 female, 6 male) between the ages of 23 and 40, were recruited through a department email list to participate in the study. One participant was left handed. All participants were regular computer users, 8 had used some type of 3D display before (including 3D movies), and 4 were at least monthly users of 3D input devices such as a Wii Mote or Sony Move.

3D surface manipulation task

In the 3D surface manipulation task, the user is asked to match a target surface with a collocated input surface. Both the input surface and the target surface are displayed as a wire-mesh rendering. Our hypothesis is that it is easier and faster to match the surfaces by modifying a physical shape compared to using a wand. Our second hypothesis is that it is easier and faster to modify the physical shape of the surface using two hands as opposed to one. We developed the following conditions:

1) Single point manipulation of virtual graphics (using a wand with vicon marker, and pressing button with non-dominant hand)

2a) Single point manipulation with all pins starting up,

2b) Single point manipulation with all pins starting down,

2c) multi hand and multipoint manipulation with all pins starting up.

The two meshes were always co-located, and the goal was to match the input mesh to the target mesh. The target mesh was always displayed virtually in green. In the conditions where the users manipulated the physical shape display manually, each of the vertices was rendered physically by the height of the pin, and virtual

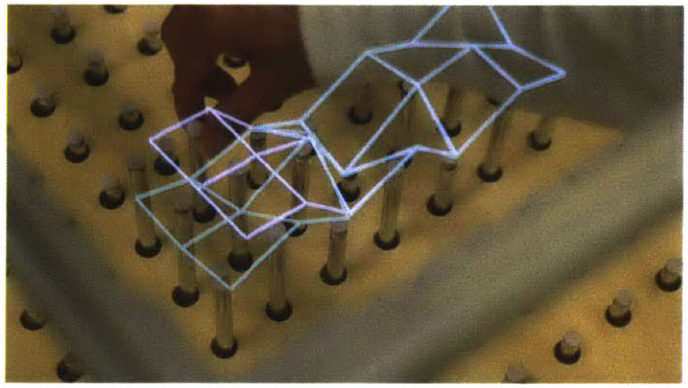


Figure 4-29: 3D Surface manipulation task, with single hand manipulation of shape display condition.

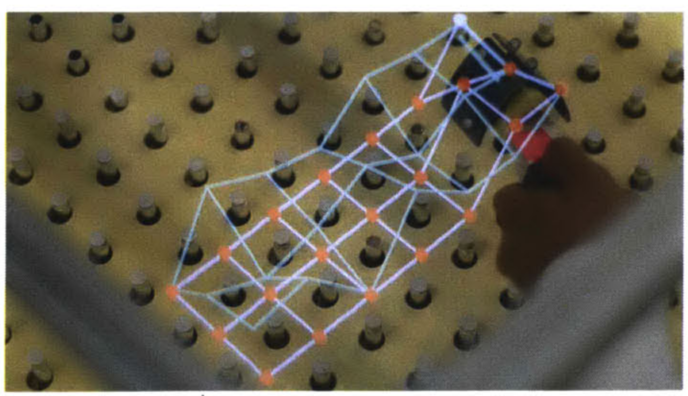


Figure 4-30: 3D Surface manipulation task, with wand condition.

graphics displayed edges connecting the pins, see Figure 4-29. When using the wand, both meshes were displayed virtually, see Figure 4-30. Each mesh had 7x3 vertices, spaced evenly in the x and z dimensions, 38.1mm apart. The meshes were randomly generated and vertices were normalized between the upper and lower bounds 100mm apart.

For the wand condition, users had to select and move vertices using the end of a virtual cursor that was overlaid on the physical wand. The non-dominant hand was used to press a button to select the vertices. The virtual vertices were rendered to be sphere 10mm in diameter, the same size as the pin diameter. The wand control would select any vertex that was closest.

In the single handed pin manipulation conditions participants were instructed to only manipulate one pin at a time, to be similar to the wand condition. In the bimanual condition users could manipulate as many pins at once as they wanted, using their fingers, palms or any surface of their two hands. We wanted to also compare the effects of the pins starting down vs starting up, which force the user to either pull or push on the pins primarily.

A total of 10 pairs of meshes are displayed per trial. As soon as the user matched all vertices with the two meshes, the current mesh is cleared and a new target mesh is displayed after a 3 second timeout, during which the screen flashes red, yellow, then green to alert the user that the next mesh will be displayed.

Procedure

We used a within-subjects repeated measure design. The order of the 4 different conditions was counterbalanced. Users were instructed to complete the tasks quickly and informed that it was a time trial task. After completing each condition, users would take a 30 second break and fill out a short form based on the NASA Task Load Index to gauge mental and physical demands of the completed task. The experiment lasted 60 minutes in total. Users were observed and video recorded for later analysis. Following the tasks users filled out a post test questionnaire and were also interviewed about the conditions and qualitative feedback on the system.

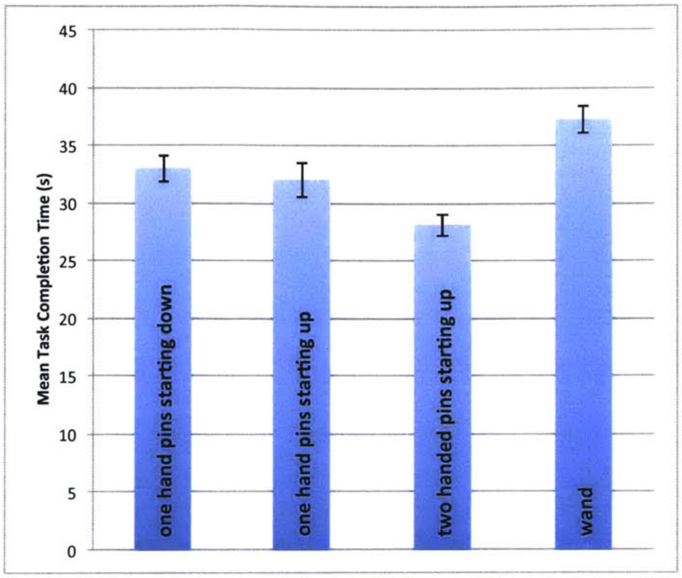


Figure 4-31: Task completion time between different input conditions. Error bars are +/- SEM.

4.6.2 Results

We present the results of the mesh matching task. The average task completion time of a single 3x7 mesh for all conditions was 32.55 seconds. With one-way repeated-measure ANOVA, we found a significant difference between the four input conditions ($F(3,27)=8.033$, $p < 0.01$, partial $eta^2 = 0.47$)). Figure 4-31 shows the mean task completion time for all conditions. The bimanual pin manipulation condition was the fastest 28.10s, next was single handed pin manipulation with pins starting up 31.97s, followed by single handed pin manipulation with pins starting down 32.94 and then the wand condition 37.20. When running post-hoc pair-wise comparisons (Bonferroni corrected) we found a significant difference between the bimanual pin manipulation condition and the wand input, and the bimanual condition and the single handed pin manipulation with pins starting up condition ($p < 0.05$). There was no significant difference in accuracy across conditions.

Wand Versus Pin Manipulation

Our hypothesis was that the physical pin manipulation would be faster than mid air interaction with the wand. The results show that while the task completion times for all of the pin manipulation conditions were lower than using the wand, only the multihand pin condition is statistically significantly better than the wand. The actuated shape display was designed for two handed pin manipulation, and the dominant method of input using the shape display; therefore we feel that this study validates the hypothesis that the shape display can perform better than a mid air 3D pointing device. The physical pins provide many benefits in this controlled scenario, such as constrained movement and haptic feedback.

There may be several reasons for the lack of significance in the one hand pin conditions. Firstly, the wand condition allowed the users to select the closest vertex with a single button press - thus there was no penalty for accuracy of target acquisition using the wand as long as the users were closer to the given target than any other targets. The single handed pin conditions had no such snapping effect. Secondly, one area that participants mentioned that made the physical pin conditions more challenging was that they would obstruct the interaction with other pins. The virtual condition did not have this problem at all. "The wand was nice, because it was not in the way," according to P3. Many, users mentioned this problem, and even those who did not prefer the wand thought the lack of obstruction while using it was a clear advantage: "The wand is better at not getting the pins in the way, but it tires you more and it doesn't feel too precise" P4. Users developed several strategies to minimize the obstruction of interaction from surrounding pins, which limited this problem; "I had to be careful about strategy and order," according to P5. Some users felt that the bimanual condition alleviated some of this problem. This concern of pin obstruction has been discussed before [102]. This may be one of the key limitations of manipulating and interacting with physical shape displays, which may be addressed by different interaction techniques. Thirdly, the force required to move pins is higher due to friction in the motors.

Another limitation of the shape display is that users were more surprised when the shape display cleared all of the pins, than in the all virtual display case. Almost all participants appeared visibly surprised at at least one point, when the pins changed dramatically. It is unclear if this had any effect on performance. This is a possible limitation of sublimation based interaction techniques, where the physical shape changes quickly.

In addition it is worth noting that other interaction techniques could be chosen for the wand condition and for pin manipulation as well. Snapping to grid for example would change task completion times for this study dramatically. Also, the mesh modification in this case was limited to a 2.5D mesh, constraining vertices x and z movement. Other interaction techniques would have to be developed to allow a 2.5D shape display to manipulate a 3D mesh, and the wand input clearly has more degrees of freedom which can easily be mapped to that interaction.

We also wanted to look at pin manipulation task completion times and how these were effected by pin starting location; was it significantly easier to push or pull the pins? We had assumed that pushing would be easier. The results show that it was faster, but not significantly. However, we limited interaction to a single pin at once in both of these conditions; it is possible that one could push multiple pins more easily than pulling up multiple pins with one hand. In addition in post test questionnaire users preferred pushing (mean 5 out of 7) to pulling (mean 3.5 out of 7) ($p < 0.05$). Users also reported different strategies for ordering interaction, between pushing and pulling; when pulling many users started at the back of the mesh, and when pushing many users began at the front.

Bimanual Interaction

Bimanual pin manipulation, with pins starting up, was significantly more effective than both the pin manipulation condition, with pins starting down, and the wand interaction ($p < 0.05$). Users also often commented in post test questionnaires that using two hands was much easier and felt more intuitive than the single hand or wand conditions. "Two handed interaction felt the most natural. I felt like I was molding the pins into shape" (P1). "It felt more organic" (P5). "Doing the experiment with two hands really showed off some of the potential of such a display" (P11). During the bimanual input condition users could also manipulate multiple pins with one hand, which also added to the ease of use at times.

Single hand pin interaction was difficult for some users, and often frustrating because they often wanted to use both; "I have another hand!" (P1). There seemed to be some cognitive load involved with making sure not to use two hands.

There were a number of different strategies with the bimanual condition. Some users switching between hands, using their left hand for pins on the left side, and their right hand for right side. Others used their dominant hand for more accurate control and their non dominant hand for roughing out the shape. "I like the two handed version the best because it allowed me to do an unrefined pass with my left hand and a refined pass with my right hand" (P2). However, some users felt that though they could be faster with the bimanual condition, it felt more taxing; "I found that I made more errors when using two hands, which I had to later go back and correct - though this method felt faster than using one hand" (P3). Other users primarily used a only single hand during the bimanual condition, "I didn't really use my second hand, I found it was too confusing to use" (P6).

4.7 Discussion

Shape displays allow for new ways to create physical interfaces, beyond functionality alone. Aesthetic form is an important part of many of the devices and objects that we interact with on a daily basis. Shape displays begin to allow interface designers to create radically different physical forms for different applications. The Marble Answering Machine example points towards this type of use, in which form is more than functional; it is also evocative and emotional. This introduces an opportunity for physical motion design. It also points towards uses of shape displays for prototyping new physical interfaces.

The inFORM system with Dynamic Physical Affordances has been used by roughly 50 people and we have collected qualitative feedback from these encounters. Users tried the 3D Model Manipulation application and an example program to move passive objects on the surface autonomously. Initial feedback was generally very positive, with users commenting on the advantages of having physical UI elements appear and transform on demand, as well as expressing general delight with the autonomous movement of passive objects on the table. However, we observed that in the 3D Model Manipulation application, users sometimes struggled with physically overlapping content and UI elements. While we believe the main reasons to be the limited resolution of the shape display hardware and the software not adapting well to content changes, solving the physical overlap of content and interface elements rendered both as shapes is a very interesting new challenge of such interfaces.

We also noticed on multiple occasions how rapid shape transitions were jarring to users, an observation we have made in earlier studies as well [100]. The question remains how to best communicate shape transitions to the user before they occur, to avoid surprise. We see this question as an important next step for research on shape displays. One potential solution is for smooth, slow transitions, that can be explored calmly, as in Lumen [132].

Smooth transitions may not suffice to adequately inform the user; possibilities for shape change may need to be more legible. As Gaver explains: "Affordances are not passively perceived, but explored" [45] and we must find a way for these new affordances to gracefully be explored, potentially by more tightly coupling their motion to the motion of the user. We think this is a rich area for future exploration. For example, it can be viewed as a feedforward problem [173]. Or, one potential direction to explore legibility for potential shape change could be to combine shape change with augmented reality, similar to [100]. More theoretically, considering the shape display as an autonomous agent, may suggest looking towards research in human—robot interaction, where robots may want to convey to the user how they will move through more subtle means.

Along these lines, the legibility of how a passive object will move on the table is of interest. We described two ways to move a ball on the shape display surface: pushing the ball or rolling it down a slope. Rolling down a slope is much more legible to the user; toy marble runs are very legible, because the marble only has one path and the user can easily follow its trajectory, which is powered only by gravity. A designer, using the inFORM system, can make a ball's trajectory more legible by creating a slot that it will move in. But, such features for legibility alone may take up space and not scale well. New interaction techniques can be explored to address this legibility.

We believe that the *facilitate*, *restrict* and *manipulate* techniques described here are merely one part of a larger space of Dynamic Physical Affordances, which will emerge as shape-changing UIs mature. The Dynamic Physical Affordances in this chapter focus on affordances rendered on a 2.5D surface. However, it is interesting to look at the larger space of possibilities for actuation and shape change: the user, the tool handle, tool, object, and physical surface must be considered. In this work, we have focused on dynamically changing the physical surface, but these other areas and their combinations provide many interesting possibilities for new interactions. For example, a tool can change shape as the interaction surfaces change shape as well. It is at these intersections between different materials and different interaction elements where shape change and actuation begin to open new opportunities for human—computer interaction. This suggests the importance of considering the whole ecology of interaction and interactive devices. We believe that proxemic interaction for shape-changing UIs is another important area to explore, as well as multi-user co-located and remote collaboration.

4.8 Conclusion

In this work we have explored the design space of Dynamic Physical Affordances and Constraints, and described methods for actuating physical objects on actuated shape displays. Many prior approaches to shape-changing user interfaces have relied on special-purpose or bistable shape change. Instead, we explored dynamic shape change's more general-purpose role, similar to the flexibility of a bitmap screen for GUIs. This opens possibilities for using shape change both for content and as UI elements. However, this approach requires complex hardware and does not allow users to define their own affordances. In the next two chapters we investigate how to off load the mechanism of shape change from computer controlled actuators to the user to create User Defined and User Appropriated Physical Affordances for malleable and deformable interfaces.

Chapter 5

Appropriated Physical Affordances

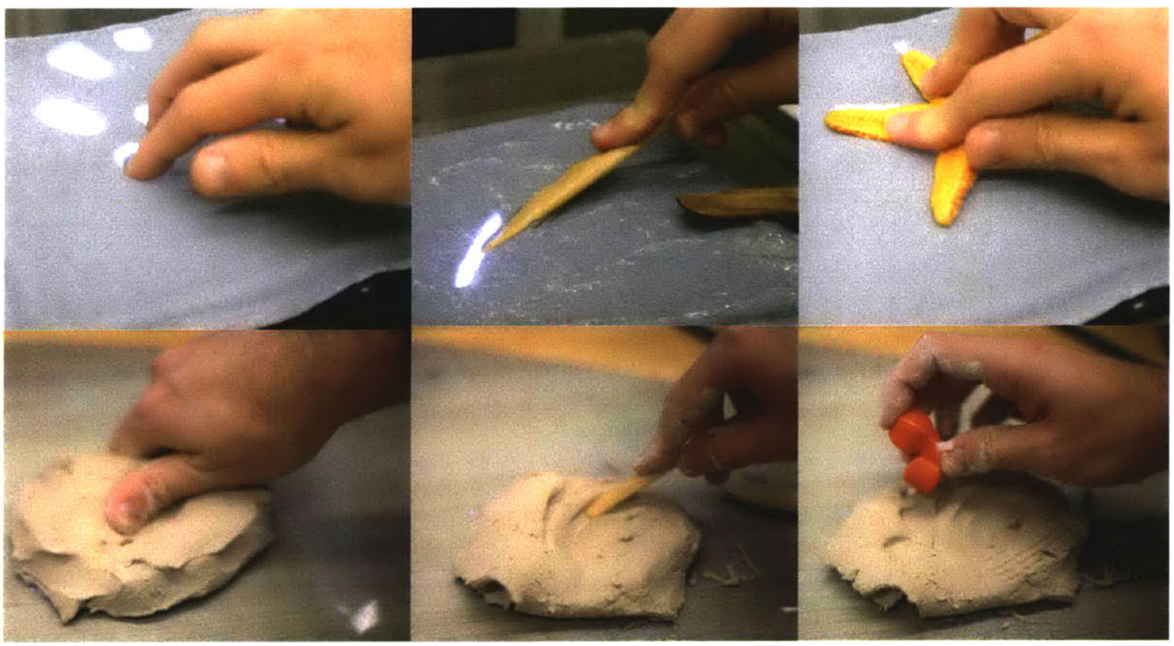


Figure 5-1: Hands, Tools and Objects used with clay and the deFORM System.

In contrast to the previous chapter that discussed projects in which the underlying physical properties of affordances are modified through computational control of shape change, this chapter and the following chapter, investigate how users can provide the mechanisms for shape change through user improvisation. This chapter explores how users can appropriate existing objects and tools and utilize their affordances for interaction. The following chapter looks at how users can improvise physical affordances through deformation of interactive devices.

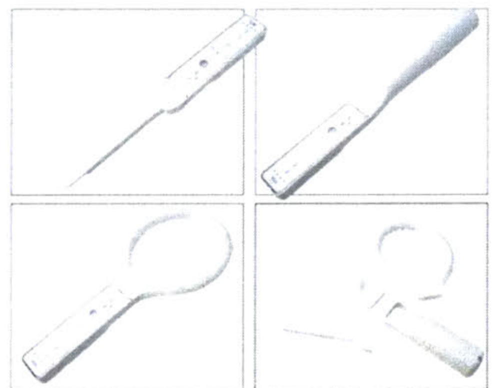
Expert users are often motivated to modify and adapt their input devices and interfaces to suit their needs. We see the need to have richer physical affordances that users can adapt to general purpose input devices. For example, there are a wide variety of add-ons available to extend the physical handling affordances of Nintendo Wii game controllers, see Figure 5-2. These add-ons are often completely passive and only change the handling affordances of the device; however they can easily change a users precision in game play. These range from steering wheels and guns, to tennis rackets, golf clubs, and even saxophones. Here these add-ons provide not only richer handling affordances, but they also change the center of mass and provide strong cultural constraints for their use. By appropriating these add-ons users have taken a general purpose input device and made it a more constrained, effective, single purpose device.

Other researchers have investigated allowing users to adapt existing objects as handling affordances for input devices. Zhang's Control Freaks [190] allows users to attach a sensor to objects like a chair or a frying pan to use them as game controllers. Building on Everyday Play [191] combined Control Freaks with Hartman's Exemplar tool [55] for even more flexibility in adapting existing objects as input devices, and leverages the handling affordances of those objects. Using Exemplar, end users can quickly author appropriate mappings between sensor data and meaningful user input. onObject allowed users to appropriate existing objects as smart interactive devices by attach RFID tags to arbitrary objects, combined with a hand mounted RFID sensor and accelerometer [22].

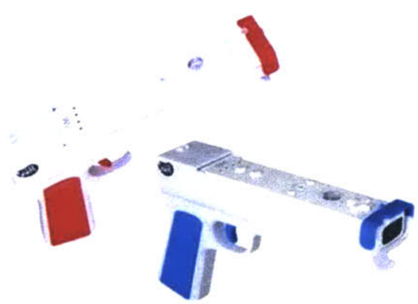
This prior work in appropriated physical affordances was limited to handling affordances, and investigated how existing objects can be utilized for their effector affordances as well. However, while many of these devices leverage the kinesthetic and proprioceptive nature of handling a physical object, and potentially change this greatly by modifying the grip or center of mass, they do not provide much in the way of haptic feedback.

This chapter considers how to leverage both the physical handling and effector affordances of existing objects and tools as input devices for 3D interaction. It focuses

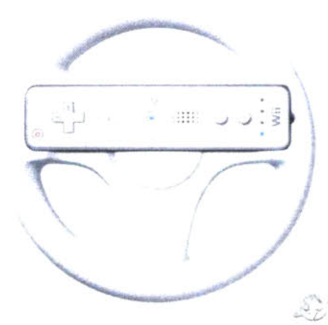
(a) Wii-Mote add-ons for music.



(b) Wii-Mote add-ons for sports.



(c) Wii-Mote add-ons for shooting games.



(d) A Wii-Mote add-on for driving games.

Figure 5-2: Examples of commercially available Wii-mote add-ons.

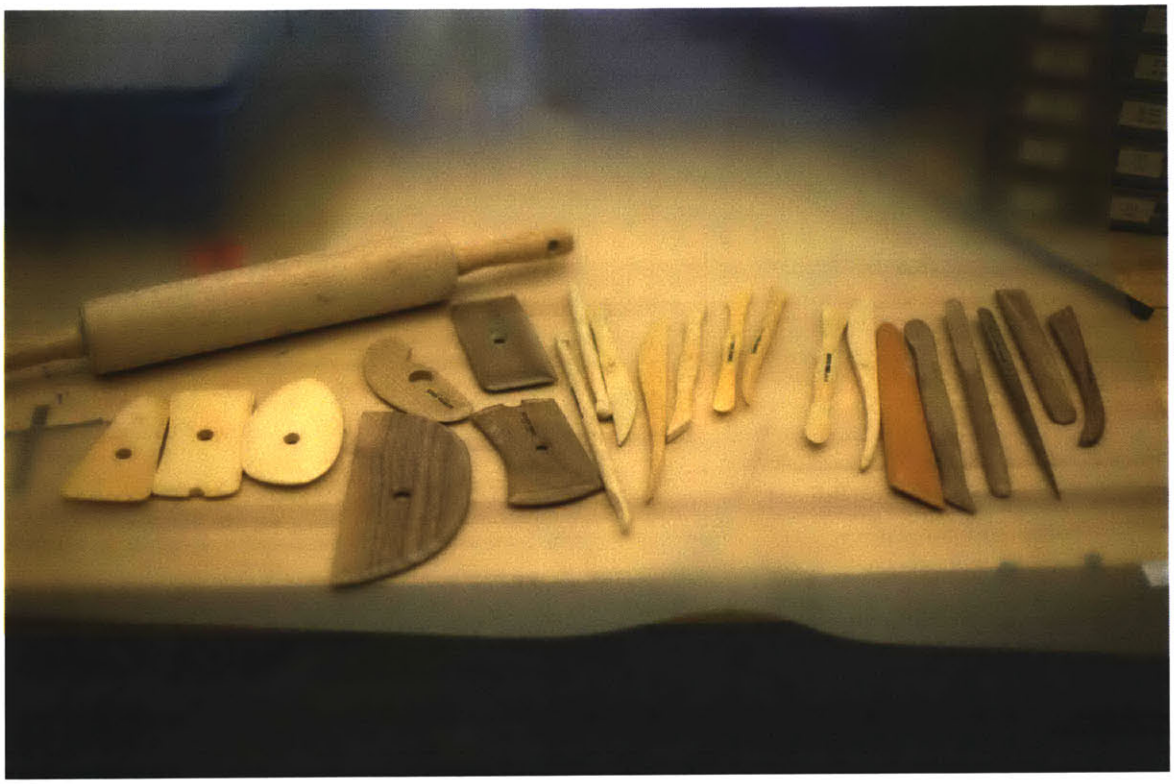


Figure 5-3: Traditional Sculpting Tools.

on a specific application domain - digital sculpting, in which the effector affordances can map directly to deformations in a 3D model.

When interacting with highly malleable and deformable physical surfaces and forms in the real world, such as clay, there are diverse possibilities for input. Sculptors use their entire hands to shape and deform, not just their fingertips, providing nuanced control. Sculptors also use a variety of tools with complex shapes to displace clay or to add texture, see Figure 5-3. These tools afford higher precision and more variety than hands alone. But in addition to sculpting tools, any arbitrary object can be used to deform clay.

When sculptors deform clay, they also feel the feedback of the clay pressing back. This enables sculptors to accurately gauge how much material they are removing or the manner in which they are shaping the medium. By combining these various inputs, sculptors transform blocks of clay into expressive and meaningful forms.

What if we could combine the expressivity of clay with the benefits of digital interaction to allow for input from hands, tools and arbitrary objects with co-located

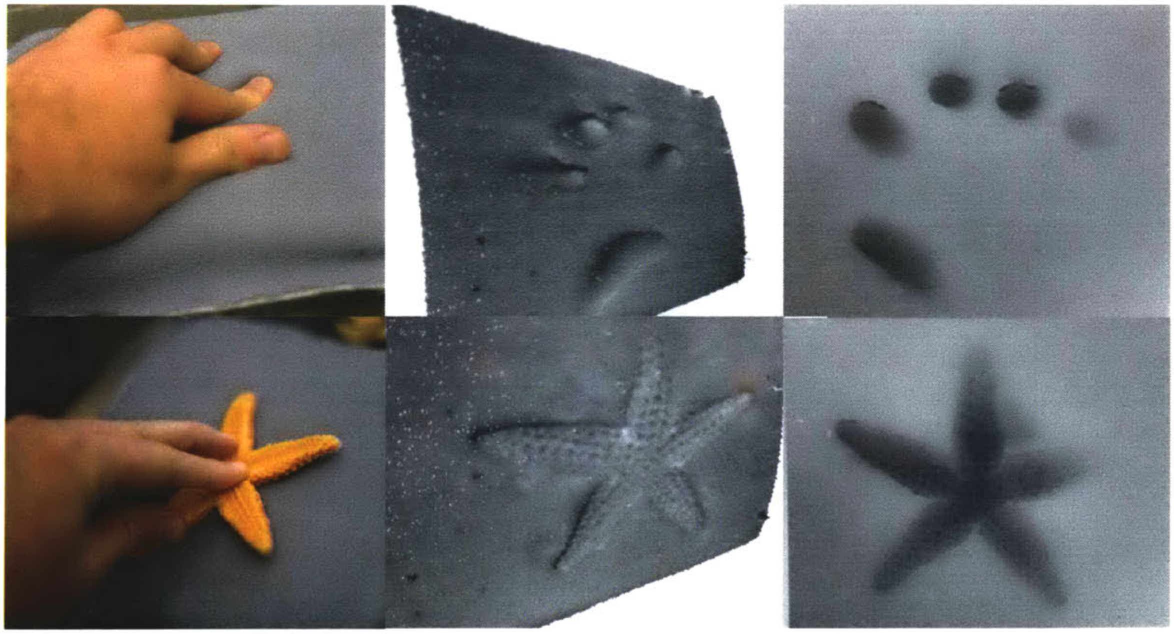


Figure 5-4: The deForm system allows users to appropriate the affordances of existing objects and tools.

visual feedback? Users could use their fingers and hands to pinch and de-press the form. They could use a physical sculpting tool to add fine detail, and find a physical object to imprint a complex texture into the form. Users could also feel the deformations while producing them, because of the immediate feedback from an elastic input surface.

Just as in the real world where sculptors leverage the physical affordances of tools, and can easily swap physical tools quickly, we want to enable users of digital interactive systems to be able to appropriate physical tools. However the question of how to support this interaction remains.

To capture complex interactions of hands, tools and arbitrary objects, we propose using high resolution real-time 3D scanning with a passive deformable surface. Dense real-time 2.5D/3D input has only recently become available and affordable, bringing the flexibility to use arbitrary objects as input. Some camera-based solutions often focus on mid-air interaction, and lack the physical feedback of real-world malleable surfaces. Other researchers have shown that passive haptic feedback can enhance precise, expressive input [96, 154, 175].

Our solution combines a passive deformable surface with real-time 2.5D capture to support a wide variety of input. Instead of directly tracking tools, objects, or hands, our system indirectly senses them through deformations of a highly pliable surface. This approach provides passive haptic feedback, and makes clear to the user where the surface of interaction begins and when objects are being scanned. Any object can be used as input, and its shape and 2D grayscale texture, or albedo, are captured as it deforms the surface of the device. A high-resolution 2.5D capture system allows for increased complexity, overcoming the limitations of low-resolution generic deformations in order to achieve the desired clay-like sculpting.

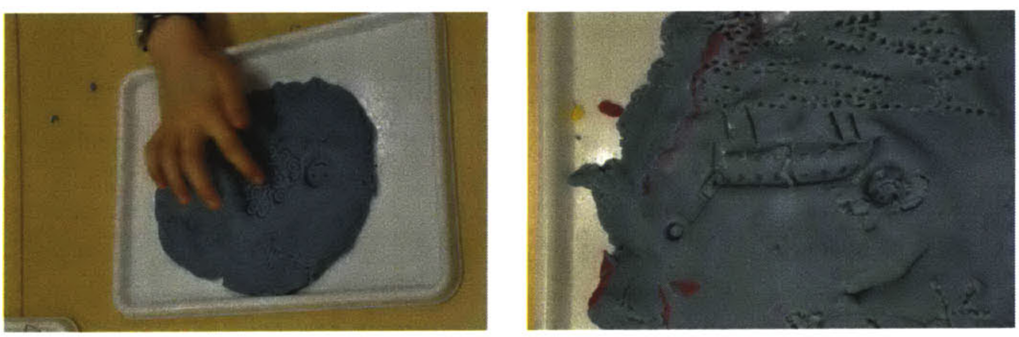
We introduce deForm, a real-time 2.5d surface interface that uses infrared (IR) structured light scanning and projected visual feedback. This system enables the use of Appropriated Physical Affordances. We also detail our solution for tracking arbitrary and tagged tangible tools (phicons), touch and hand gestures. A discussion of limitations follows. Finally, the results from a qualitative user study with 12 children are presented.

5.1 Background Research: Appropriating Affordances in Sculpting

We conducted an initial exploration to explore the use of many tools in sculpting. We choose the domain of children's sculpting clay to purposefully contrast to expert sculptors, hoping that there would be more of a playful environment for improvisation beyond traditional sculpting tools. As an analog for the interface we would end up designing, we used Play Dough. Play Dough is a very malleable sculpting material with which young children can easily play.

We selected children aged seven to ten years old as our target audience, and as such found a class of second graders ages seven and eight to participate. Two groups of six children each participated in the study, with a total of six girls and six boys. Children were split up into two tables, each given approximately one pound of Play

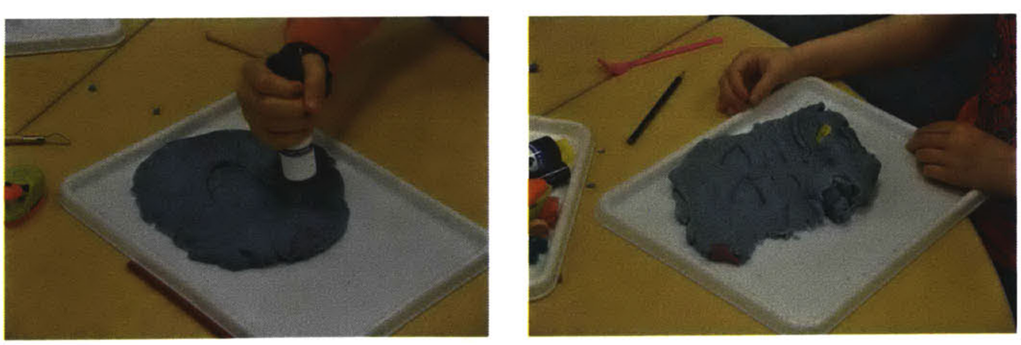
Figure 5-5: A child's design in Play Dough made by stamping objects and toys.



(a) A child using his hand to smooth out an error.

(b) Small details are filled in with a fine pencil on top of stamped designs.

Figure 5-6: More designs in Play Dough.



(a) The children used a variety of objects to sculpt with.

(b) Repeated shapes form textures created by stamping.

Figure 5-7: More designs in Play Dough.

Dough to work with. All Play Dough was colored blue, as we only wanted to explore shape and form in this study. During the session the children's task was to create animals by stamping objects into rolled out Play Dough 1 inch thick. The rolled out Play Dough was intended to be an analog for our remixing interface. A number of toys, blocks, knives, pencils and other objects were laid out for children to use with the clay.

We observed some interesting trends that seemed to be exhibited in a number of children's designs. The most prevalent was the use of stamping to create a patterned texture.

There was often a combination of many different objects in addition to drawing into the clay. Many of the children used over 5 different tools or toys to create their animal. Children seemed quite resourceful in using existing toys or objects to create new designs. More importantly, the complex nature of different objects provided both handling and effector affordances for children to explore. We saw a wide variety of different techniques employed with the same tool or object. Different grips encouraged different uses as well.

However, almost all designs utilized drawing. Children tended to use existing objects to layout the general shape, and then use drawing to fill in more details. This speaks to the need to support a wide variety of input in future design tools.

Hands tended to be used to clean up mistakes, and erase areas, but were not used as often to create geometry. Although a number of times children used their entire hands as geometry, but there was not as much sculpting with fingers as we had expected.

From this initial investigation it was clear to us that there was a strong need to incorporate the ability to improvise in a 3D modeling tool. We wanted to harness the flexibility of clay, allowing for an almost infinite set of tools to be used to modify it, with the power of digital computation such as undo, scale, rotate, etc.

5.2 Technical Related Work

In this section we summarize 3D input, and its limitations. We then describe how related work has sought to bring 3D input to 2D surface input.

5.2.1 3D Input

Advances in stereo vision and structured light scanning have made 2.5D real-time video capture a possibility. Most recently the Microsoft Kinect, made by Primesense, uses structured lighting to capture real-time, 30hz, 2.5D geometry at a 640x480 resolution, but is tuned for room scale interactions with a wide angle lens. Custom structured lighting systems have been shown to capture realistic geometry at very high frame rates, by projecting fixed or time sequenced patterns onto objects [195].

One disadvantage of using 3D capture of points or video for input is that it does not provide physical feedback. In addition, these systems provide no physical mechanism to highlight to the user which information is being captured; there is only on-screen feedback, in some cases. The work of haptic interfaces such as The Phantom have explored adding mechanical actuators to 3D input to provide tactile feedback [144]. But these systems only allow for single point interactions and contain many moving parts.

One successful approach has been to combine materials that can provide unactuated, passive haptic feedback with 3D sensing. Illuminating Clay used a laser scanner to scan the front of a clay surface at 1 Hz and projected feedback directly onto the clay [130]. However, the user's hands interfered with scanning, as a result of the camera's location above the surface. Passive foam blocks tracked with a Vicon system and tracked fingers and tools have been used to enable 3D sculpting [152]. However, this system required augmenting hands and tools with markers, and only provided a simulation of deformations, as opposed to capturing true deformations in the surface. We hope to expand on this work by adding real-time 2.5D scanning to a passive malleable surface to capture real deformations with any object.

5.2.2 Extending Surface Input to 2.5D

There has been a wealth of research on 2D surface interaction [5]. Recently many researchers have explored adding more dimensionality to surface input through both above the surface interactions and into the surface interactions.

Visual Touchpad used stereovision to enable above the surface 2.5D interaction [18]. More recent work has harnessed depth-sensing cameras to facilitate above the surface interaction [62, 74]. Although these systems allow for much larger areas of interaction, they lose some of the advantages of tabletop surface systems, such as passive haptic feedback and co-located input and output. More closely related to our work, into the surface 2.5D interaction allows users to press hands and objects against or into the surface to capture more dimensionality. Some of these systems measure pressure through force sensitive resistors [140], or mechanical deformations [122]. Other systems employ magnetic sensors and deformable magnetic material [66, 75].

Another approach is to allow the surface to be deformable and to measure its deformation with a camera. Our system takes this approach, and as such we closely reviewed other systems in this domain. One approach uses a deformable projection screen made of lycra fabric or latex rubber, which stretches when force is applied to it, either tracked by reflected pixel intensity [17] or by tracking dots on the surface.

A number of these 2.5D surfaces have used a deformable liquid bag or gel as their basis. These systems can more clearly resolve concave shapes. This occurs because the gel or liquid applies a stronger force back on the surface to fill in concavities.

One category of gel/liquid based 2.5D systems provide pressure-sensitive input through pixel intensity from a camera mounted below the surface. Pigment dispersed in a liquid contained in a bag reflects more light the deeper an object is pressed [155]. The liquid-based approach does not provide for high-resolution 3D scans, cannot allow 2D texture information to be captured, and has physical stability issues due to fluid movement [63].

Gel-based input systems provide a stable deformable surface with which to interact. Photo-elastic Touch utilizes polarizers to measure the photo-elastic effect of deformations into gel surfaces [146]. This provides a fairly low resolution spatial pressure map, limited to finger scale detail. Furthermore, spatial resolution decreases dramatically with increased input force. Smith et al. showed that a deformable gel on top of an FTIR multitouch system can provide pressure information [156].

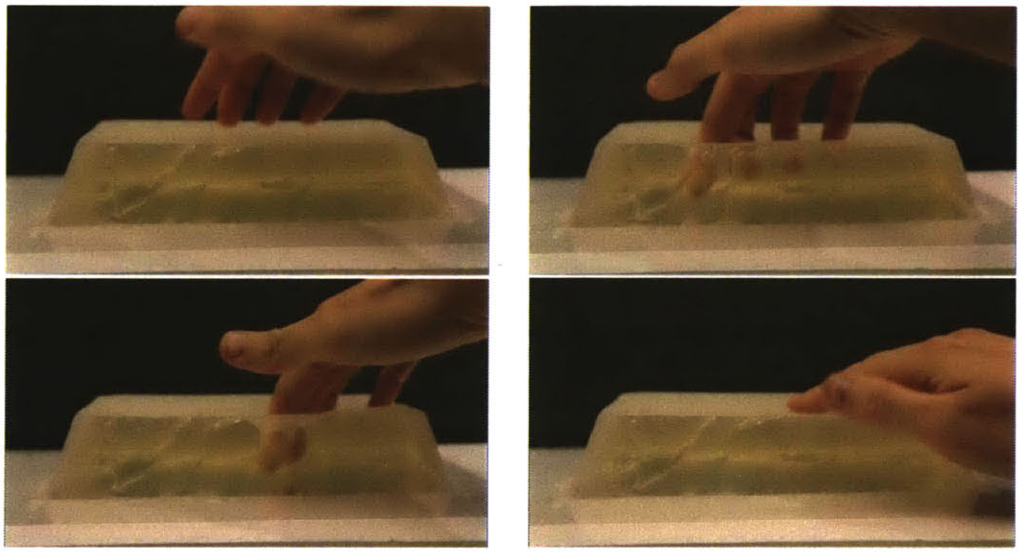


Figure 5-8: The Thermoplastic Elastomer used in deFORM deforms when force is applied but returns to its normal state quickly.

A more sophisticated marker-based system, Gelforce, uses two grids of visible markers vertically offset in the gel and a single camera to derive true 3D force vectors applied to the gel [177]. This system has many benefits, but its resolution is limited by the size of the dots. These optical dots also obscure the surface and preclude 2D texture reconstruction.

GelSight uses a gel with a painted surface and a photometric stereo system to capture 2.5D surface normals [77]. This system is limited to only accurately reconstructing shallow surfaces because photometric stereo does not capture precise depth disparities [114]. In addition, Gelsight is highly dependent on surface color, requiring a thick layer of paint. Furthermore, it cannot capture the independent 2D texture image of an object. Our system uses structured lighting to triangulate surface geometry and is less sensitive to depth discontinuities.

Our system provides many benefits beyond existing work in into the surface 2.5D input. It allows for high-resolution dense surface reconstruction, 2D texture capture in the IR spectrum, and simultaneous 2D visible light feed-back at interactive rates. This chapter also introduces depth-based fiducials.

5.3 System Description

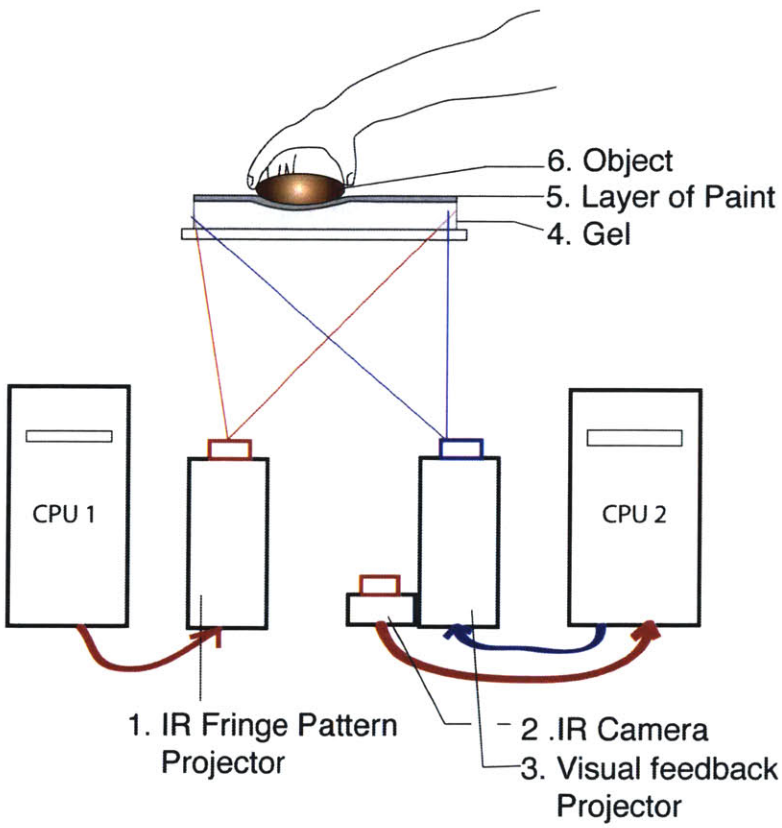


Figure 5-9: The deFORM System has two parts: 1) a deformable gel coated in paint, and 2) a camera projector system.

Our system for 2.5D input consists of two parts: a passive, deformable gel surface coated with a thin layer of paint and a camera projector system for real-time 3D scanning of the paint surface from below.

We use a 1 inch thick gel surface, which is cut into a square measuring 8 by 8 inches. The gel is deformable, but very elastic, and returns to its normal state after the object is removed. The gel is optically transparent, and the surface is painted with a gray paint to capture only the geometry of the surface of the gel as opposed to objects above the gel. The painted surface can also be used as a projection screen. The gel sits on a piece of clear glass through which the pattern is projected onto the gel, see Figure 5-9.

deForm uses a structured light system to capture deformations in the surface of the gel in 3D. Our system implements the Three-Phase structured light scanning techniques described by Zhang [195]. Three sinusoidal fringe patterns are projected on to the gel surface in sequence and captured by a high-speed point grey camera. The patterns are time sequenced, which means our system requires three projected and captured frames for one 2.5D reconstruction.

With this system we are able to achieve a high-resolution, 640 by 480, depth map at interactive rates of 20 Hz. Figure 5-10 shows a single reconstruction captured in three frames at 60fps. Three-phase structured light scanning can also reconstruct a greyscale texture image of the surface of the gel from the three phase images without requiring an additional camera or a reduction in frame rate [195]. The thin paint used lets through much of the surface color and texture, allowing us to simultaneously map the surface image of the object to its 3D scan.

Instead of projecting patterns in the visible light spectrum, the IR light spectrum is used to 'invisibly' capture geometry. This allows for simultaneous 2.5D input in IR and projection of visible light interfaces on the gel surface for interaction.

We initially attempted to use a Microsoft Kinect camera for our 3D input, but found that it was not appropriate because it was designed for room scale interactions. The 70 degree field of view, combined with an active sensing area starting 30in from the device, results in a minimum sensing area of roughly 42X31 inches. At its 640 by 480 resolution the maximum spatial resolution is roughly 15PPI, far lower than our system's 80PPI. The Kinect also has a very limited z-depth resolution, at close to 0.5cm accuracy.

5.3.1 Accuracy

Our system is currently able to capture surface geometry with features as small as 0.8mm with spacing between features as small as 1.6mm. We evaluated our system

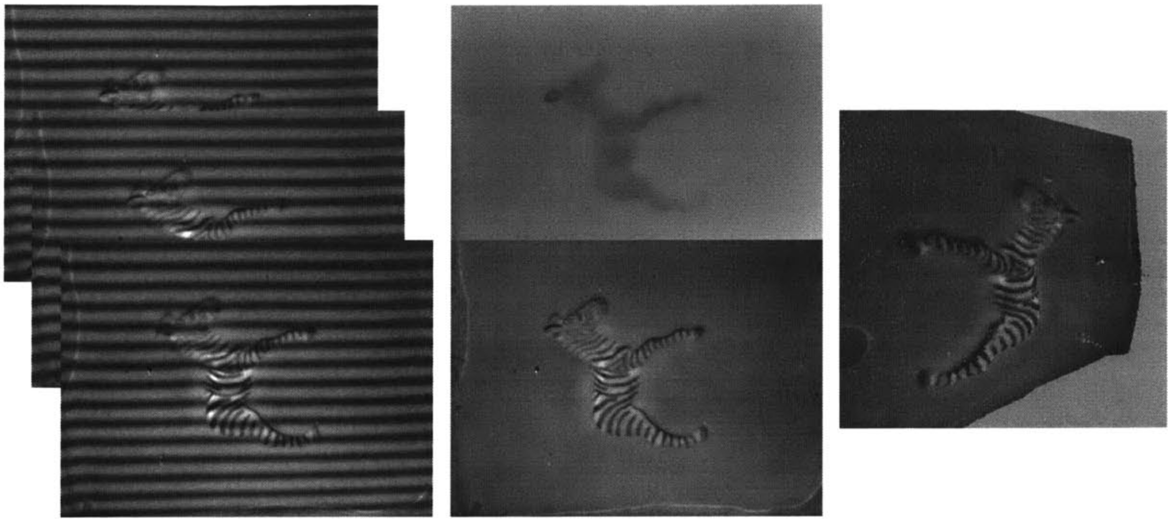


Figure 5-10: 2.5D structured light reconstruction. Left, 3 phase shifted sinusoidal fringe patterns projected in IR on gel surface. Middle Top, 2.5D depth map of Zebra toy. Middle Bottom, greyscale 2D texture reconstructed from fringe patterns. Right, 3D view with 2D texture applied.

using a number of lasercut depth targets, see Figure 5-11. We are able to capture the overall geometry of a Lego gear, a fairly complex 2.5D object. There is some reduced accuracy due to the gel surface, but this is minimal. Deep concavities are not accurately reconstructed.

5.3.2 Tracking

Using a background subtraction algorithm on the reconstructed depth map, our system is able to easily detect objects, fingers, and tangible tools pressed into the surface. After segmentation and labeling, we are able to track these objects and classify their average and maximum depth if necessary. We can also estimate the relative rotation and orientation of the object, providing 6 Degree of Freedom input. We estimate the pitch and roll by averaging the normal vectors over the object. The rotation or yaw can be estimated by finding the major axes, but this approach only works with non-rotationally symmetric objects.

The system can also estimate the force applied by the object, based on both its depth in the gel and the surface area of the object in contact with the gel. The gel

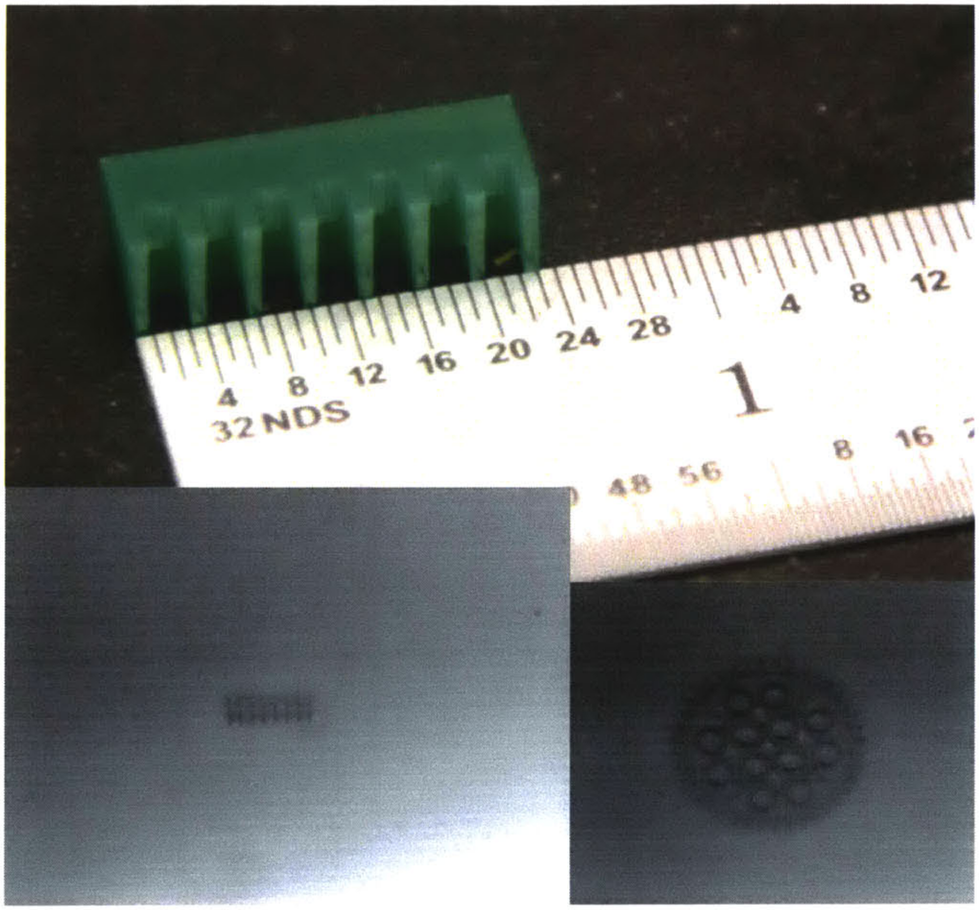


Figure 5-11: Top, Target used to measure accuracy. 0.8mm pins with 1.6mm spacing. Below, left target clearly resolved. Right, Lego gear clearly resolved

Figure 5-12: Tracking objects on deFORM. Left to right: Raw depth map of fingers pressed into gel; Background subtraction; Thresholded 2.5D image.

has a uniform durometer and so requires a relatively uniform force to deform it. By integrating the area bounded by the object in the depth map, we can estimate the relative force in the Z direction. This could be useful for determining the pressure applied to a stylus as opposed to a flat hand.

5.3.3 Tangible Tools

Our system can support input from both arbitrary objects and tagged objects such as tangible phicons (physical icons) [72]. Deformations from arbitrary objects can be mapped directly to input, while using special tagged tangible controllers to pre-form specific operations.

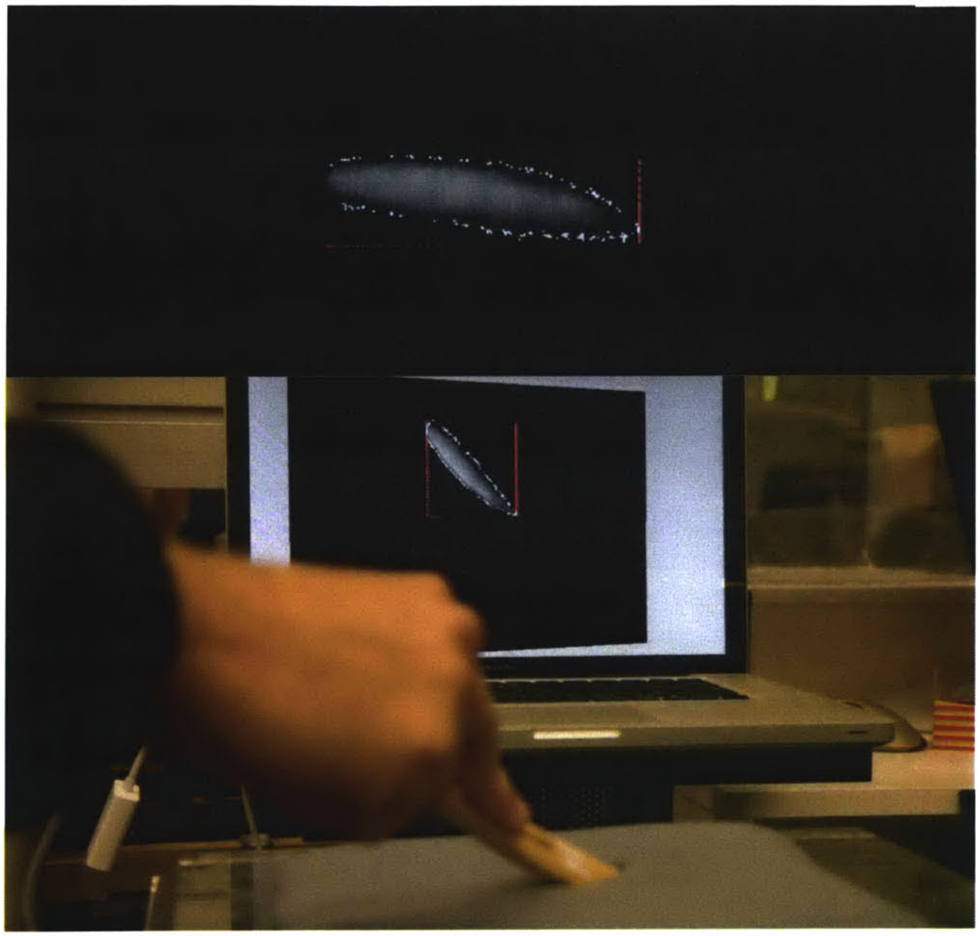


Figure 5-13: Tangible tools can be tracked as well from the depth map. Here a sculpting tool is background subtracted and labeled.

Arbitrary Objects/Tools

deForm can capture, in 2.5D, arbitrary objects pressed into the gel surface. We can use these 2.5D geometries to deform virtual meshes or to control 3D scenes. A wide variety of objects can be used to deform the surface, allowing for a diverse set of input means, beyond a single stylus. Multiple shapes can be captured at the same time. For example, traditional wooden sculpting tools could be used to deform digital models. Many projects have sought to use traditional paintbrushes with digital interfaces [120, 171] to capture particular properties and styles.

Since deForm can capture both 2.5D geometry and 2D grayscale texture information, the system can function as a fast 3D scanner. Optical multitouch systems have used scans of 2D graphics, such as real photographs and documents [183], to create an easy, direct way to input information. Our system adds another dimension to that intuitive approach. For example, a child could copy her toy by pressing it into the gel. The toy could then be modified in the digital world or uploaded to represent a digital avatar in a game. We discuss the concept of ÒremixingÓ toys in the application section below.

5.3.4 Tangible Controls

In some applications, developers may require specific tangible tools to perform predefined operations. Many systems for tangible interaction choose optical markers to track tangible tools quickly and easily [80].

Our system is able to use 2D optical markers by detecting objectsÕ 2D greyscale textures. We have used Reactivision markers with our system and tracked them when pressed into the gel surface and on the surface. In addition, our system can estimate the pitch and roll of the markers through the techniques described above.

Figure 5-14: Depth encoded markers. Left: two laser cut reactivision markers modified to encode pattern in height. Middle: depth map of depressed marker. Right: tracked and labeled depth marker.

deForm also encodes marker information in physical depth, which can be tracked in a depth map rather than in visible light. This approach allows for other information to be en-coded beyond a 2D pattern. In addition, the physical shape of a marker is easily changed, allowing for dynamic tags. This technique could also be applied to other depth-based input devices that do not capture 2D texture.

We encoded Reactivison information into depth markers by laser etching acrylic plastic, mapping black and white to height values. Using depth-encoded Reactivision markers, we are able to easily track these tags using just the depth map image, see Figure 5-14. As a result of the gel surface some error remains due to poor reconstruction of small, interior details. A modified Reactivision tag, with larger holes and fewer interior graphics, shown in Figure 5-14, allows for a recognition accuracy of 95% when directly pressed into the material. The adjustment limits the address space but greatly improves tracking performance.

Mechanical components, such as buttons and sliders, could be added to these tangible controllers, as implemented for Slap Widgets [181]. We could encode different information into the depth of a single mechanical pin. For example, instead of a single on/off button, we could have pressure sensitive buttons. Alternatively, rotation could be encoded in a pin by using a cam type system.

5.3.5 Touch Interactions

Our system supports traditional multitouch input, but due to its depth, it can also capture more complex hand interaction.

Into the Surface Touch interactions

Iconic Gestures

Using the 2.5D depth map deForm is able to support a number of different pressure-sensitive touch gestures, such as pinching and rotating, by tracking finger positions in 3D. We can extract finger locations from the threshold depth map through thresholding and blob detection.

Beyond simply detecting gestures by finger tracking, we are able to detect certain gestures from the displacement of the gel. When an object or finger is pressed into the gel, the gel deforms around the object, increasing in height surrounding the perimeter of the object. When pinching, the gel is displaced between the fingers greatly. This provides an easy way to detect pinching, by looking for areas in the depth map that have increased in height. This is just one example that highlights the differences between our system that captures the geometry of deformation and a system which merely senses pressure.

The friction that occurs when users articulate their fingers while pressed deeply into the gel necessitates a vocabulary of gestures based on mostly isometric relative change, rather than absolute positions. This approach would also benefit from the passive haptic feedback that the gel provides.

Beyond iconic gestures

Because our system can detect more complex hand poses than simple touch points, there is a large opportunity to support touch interactions beyond iconic gestures. We can use the 2.5D geometry of the hands to directly manipulate a mesh, much as one would manipulate clay. This type of interaction is explored in later discussion.

Touch Interactions on top the surface

We can use the reconstructed 2D texture image of the gel surface to do basic diffuse IR multitouch sensing. In the texture image we can clearly see finger-tips finely resolved even before they greatly deform the surface, as shown in Figure 5-15. We can use simple background subtraction and thresholding to find the finger or hand points in contact with the surface. This 2D image can then be compared to the background subtracted depth image to find touch points that are not pressing into the surface. This allows for touch interactions both on the surface and into the surface. For example, touch on the surface could be used as a hover mode, and pressing into the screen could select. Alternatively, touch gestures on the surface could change global application parameters, but touch gestures into the surface could change local parameters.

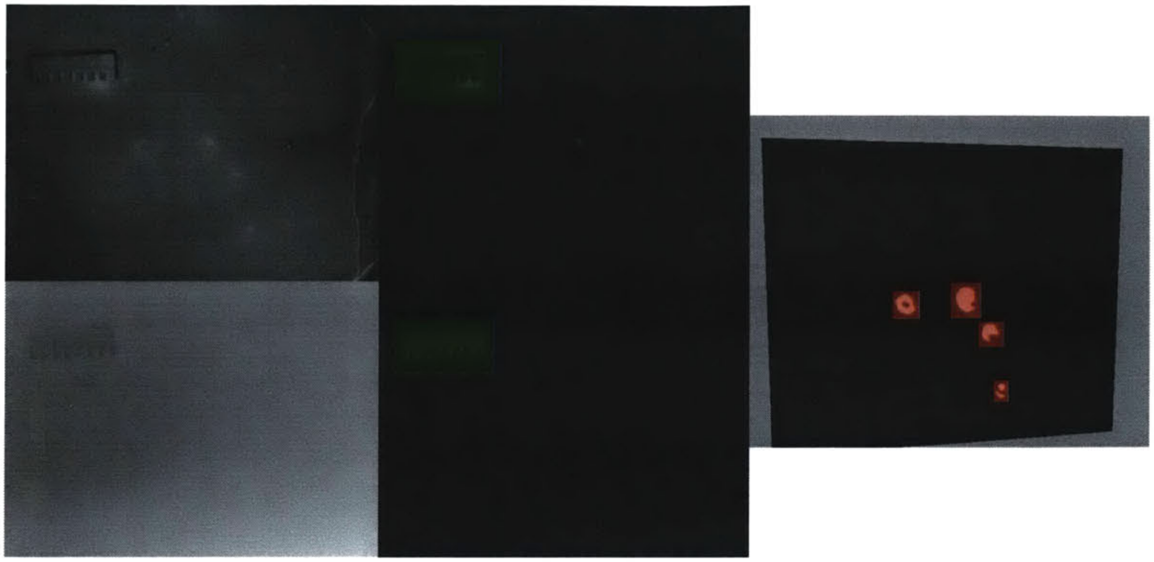


Figure 5-15: Using the reconstructed 2D grey scale texture to provide on the surface multitouch. Top: 2D greyscale and background subtracted greyscale images. Below: Depth information is subtracted from greyscale image to find only touches on the surface, as shown in the right picture.

5.3.6 Discerning Touch From Tools

Many optical systems that support multitouch interaction discern touch points from other objects by looking for the size of the blobs [52]. This method is fairly robust, but is not foolproof. Un-tagged tangible tools, such as a sculpting tool, may appear similar to a finger. To resolve this ambiguity, we propose the use of capacitive sensing in addition to optical sensing. Capacitive sensing relies on the change in capacitance between an electrode and the environment. Unlike human hands, non-conductive objects do not change the capacitance greatly. This allows deForm to distinguish between touch and tools.

Because our system relies on a very deformable and flexible surface, embedding traditional capacitive sensors on the surface is not ideal. Rather, we use conductive paint on the surface. A thin layer of silver-based conductive paint is applied to surface of the gel. With this setup, the system distinguishes between the presence of a hand and a non-conductive tool.

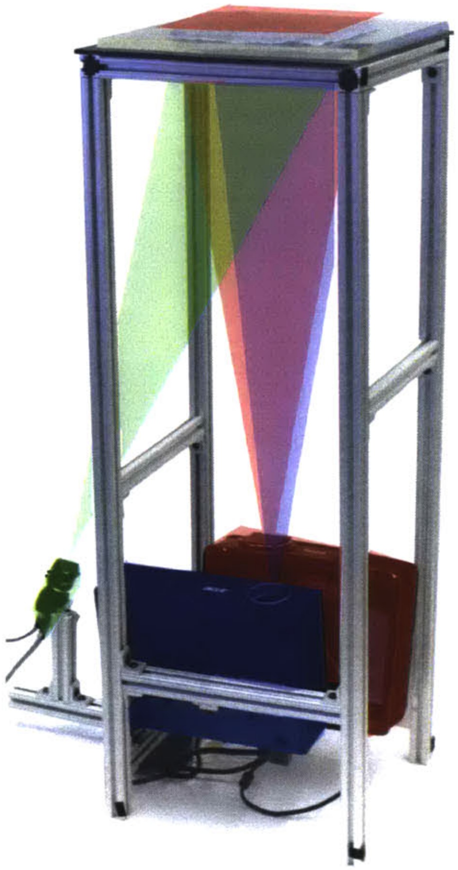


Figure 5-16: deForm System Configuration. IR (highlighted in red) and Visible Light (blue) projectors mounted in 80/20 box projecting upwards through glass to gel surface. An IR camera (green) off to the side captures deformations in the gel.

5.4 Technical Implementation

The gel structure is a soft, shor 00 durometer, thermo plastic elastomer called Ultraflex sold by Douglas and Sturges, which is heated and cast. We have explored different durometer gels and found a narrow range acceptable; if the gel is too stiff, it will be more difficult to use, if it is too loose, the gel surface will deform too easily and not retain its shape. Once painted, talc powder or cornstarch is applied to lessen the gel's stickiness.

In order to capture each projected fringe pattern frame we synchronized the camera with the vsync line of the VGA input of a projector. We used a DLP projector because the mirror arrays can update within the frame interval, unlike many LCD projectors. Using a DLP projector, we were able to achieve rates of reconstruction at 20 Hz, by projecting and capturing at 60 Hz. This technique should scale to much higher frame rates, as described in [195]. We calibrated the projector and cameras to correct for lens distortion using standard techniques [93].

To correct for gamma differences between projector and camera and phase errors, we implemented Zhang's calibration for phase error correction, which uses a look up table to match the recorded phase with the ideal phase [196].

To project IR patterns, we modified our DLP by removing the IR cut filter in front of the bulb and replacing it with a cold mirror that reflects visible light and allows IR to pass [20]. We attached a IR pass filter to our Point Grey grayscale camera so as to capture only IR light.

We mounted the two projectors, IR and visible light, on the inside of a box shown in Figure 5-16. We mounted the camera off to the side to observe deformations in the pattern projected on the gel surface. We placed the painted gel surface on top of the box on a piece of glass. One computer generates the patterns and another captures the geometry and displays interface elements. We created the software using C++, using the Open Frameworks and openCV libraries. We built our system on top of the Open Frameworks structured lighting library, ofxStructuredLighting.

5.5 System Limitations

The resolution of our reconstruction is dependent on both the camera and projector, which makes this system limited or unsuitable for reconstructing large surfaces. The trade-off between size of the reconstructed area and the PPI is quite clear, so a table size system would have a less appeal. However, the system could be combined with a digital SLR to capture single higher resolution scans, especially when combined with projector defocusing, which removes the constraint of projector resolution [99].

Currently we are using a time-multiplexed approach to capture the three required patterns to reconstruct the geometry. As a result of the time delay between each frame, large amounts of motion causes errors in reconstruction. This makes the current system ill-suited for applications such as gaming. However, smaller errors are corrected by replacing erroneous data points with information from the previous frame. Increasing frame rates could improve this problem. In addition, other phase-based structured lighting techniques have been developed to solve this problem. The 2 plus 1 phase approach is less sensitive to motion [195]. Another approach is to separate the patterns by color (often Red, Green and Blue channels), as opposed by time.

The current system requires a large total height due to the use of a camera and projector system, which can rarely be as thin as other approaches such as capacitive or FSR based input devices. It may be possible to reduce the height required by using wider field of view cameras and short throw projectors, or by introducing some sort of wave guide, such as [94].

Currently the system requires paint on the surface of the gel both to aid in reconstruction and as a projection surface. Heavy use degrades the paint over time, causing problems such as light leaks and lower quality reconstruction. Improving the robustness of the paint would lead to a more durable solution, and might also limit friction. Sliding and dragging are more difficult due to the friction caused by the gel and paint. Currently we apply a lubricant, but this is an insufficient solution outside of the lab setting.

5.6 Evaluating User Appropriated Affordances

We conducted a preliminary in-lab user study in order to evaluate the performance of User Appropriated Affordances and better understand how a system using deFORM could be used. We chose to evaluate User Appropriated Affordances in the context of 3D modeling for children. Particularly we were interested in understanding what patterns of use would emerge and how children would appropriate different tools for different tasks. We attempted to frame our observations and analysis around the design principles espoused in the chapter Design Principles for Tools to Support Creative Thinking [153] and suggested in the Creativity Support Tool Index [16].

Of those design principles, we chose to focus on Exploration, Expressiveness, and Supporting Many Paths, as we felt they most closely aligned with our design goals.

Thirteen children aged seven to ten years old, eight male and five female, participated in our preliminary study, in single child sessions in a lab based setting. Participants were self-selecting and found through an email message sent to a college campus mailing list, to which parents of participants responded.

The Sessions lasted 45 to 60 minutes. The study set up included the KidCAD system [39], built on the deFORM sensing platform, a second screen featuring a 3D perspective view of the model, and an assortment of toys and objects children could use with the system, shown in Figure 5-17. Each study session began with an introduction to the KidCAD system, and an explanation of its features. Next the participant had a warm-up task to get used to the system, and was free to play around for five to twenty minutes. In the second task the participant was asked to create two animals, an elephant and a rhinoceros, using the KidCAD system and the assorted toys and objects. The final task was for the participant to create a story with a character and design a toy of that character using the system, and then to tell the story to his or her parent. After the session, participants were asked a number of interview questions, pertaining to their experiences with the system. The sessions were video-taped and later transcribed and analyzed. The designs were not 3D printed in the session due to time constraints associated with 3D printing.

5.6.1 Findings

All children successfully completed our tasks, and many were pleased with their results. Children embraced the idea of "imprinting" shapes into the gel surface very quickly, as well as erasing and drawing new parts. It seemed easy for children to layout 2.5D designs, and there were almost no questions or need for clarification about the interaction. Children also remarked that they liked the feel of the gel. One participant, P13, explained how he liked it because "it was, like, squishy", and how it was "not hard" and that he would want one for his computer. Many explained that it reminded them of clay, and that the softness made it feel more natural.

5.6.2 Initial Use Patterns

One of our goals was to better understand what children would design using KidCAD without our supervision. During the unsupervised first session children were only given instruction on how to use KidCAD, but not on what to do with it. This session provide us with some insight into other uses for KidCAD beyond remixing toys.

One predominant theme we saw was children creating patterns and textures. These compositions allowed children to explore the accuracy of the system, but also seemed very expressive. Patterns were often dominated by repeated stamping of a few different toys, often to form very geometric shapes such as squares or crosses. Often one item would be a central fixture in the piece, and then many repeated items would surround it. Children also explored and played with texture, something that might be difficult with traditional CAD tools. Children created texture-scapes through a variety of different means, such as using their hands to imprint little dimples, using their entire arms and elbows to create deeper shapes, and rolling objects to get a repeated pattern.

Another emerging trend was to create pictorial scenes by copying a number of toys in their entirety. Children would imprint characters and also create settings, such as a tent or a tree, by combining multiple objects.

5.6.3 Exploration

We observed participants combining many different objects during the creation of a single model. For example, to design an elephant participants used an average of five different objects, often using these objects multiple times. Participants would often search for the object that fit their needs, and then try a few different locations with it above the gel surface before they pressed it in to copy it. This seemed to highlight the importance of having co-located projected feedback.

In addition, the flexibility of input choices provides users with many means for achieving the same goal. For example, to create a thick 2.5D line we observed children drawing, stamping lego blocks and plastic tubes, or even rotating a lego gear. We also observed children building things out of lego in the physical world and then stamping them to copy the new shape.

When they found that a part they had imprinted did not work as well as they had hoped, participants primarily used the erase tool to delete that part. If there was not that much progress on the model, they would often instead just clear the entire canvas. Other users found the clearing function to be liberating, and cited that as a large advantage over clay. One parent discussed with his son, P3, that the ability to clear things very easily, combined with the speed of copying objects enabled him to create many different scenes and test designs quickly.

5.6.4 Expressiveness

As documented in the objects created, users were able to create identifiable objects, and be satisfied with the results. Many of the users felt that the system was very expressive. When asked what she enjoyed about the system, one female participant, P5, remarked, "it was like sculpting with clay... I like how accurate it is, when I imprint the shape it is so accurate." She said she would use it at home to sculpt things instead of using clay. And users displayed a great deal of finesse while using the system: they were able to imprint portions of objects easily, as opposed to the entire object and seemed quite capable at combining objects together. Some users felt KidCAD would help them accomplish creative tasks that would have been difficult naturally. One user, P11, explained it could help people "draw something in 3D, if they weren't so good at drawing in 3D." Another thought that it was "easier than sculpting with clay. You don't have to cut it and wet the two parts to get them to stick together."

However, other participants found the system somewhat lacking in accuracy. One participant felt that it was better suited for roughing out shapes and then he would need to use something else later to get more detail. P3 added, "probably I could use the things I already have to make imprints of maybe a rough draft, of sort of the basic idea of what it would look like, but not all of the details." To him the pen tool did not provide enough accuracy to add the detail he wanted. P1 felt that it was difficult to use KidCAD to always "get exactly what you want" but that KidCAD was still useful because it allowed you to take more time and easily change things. Others missed the ability to fully feel the object they were creating, one user explaining that with clay you could "actually feel them."

Most children created fully fleshed out figures with 2.5D depth. However one child created designs that only consisted of outlines. This difference between line drawing and sculpting was also noted by some participants. One participant explained that kidCAD was like drawing but "its not lines" and that "it like absorbs [what you press in]."

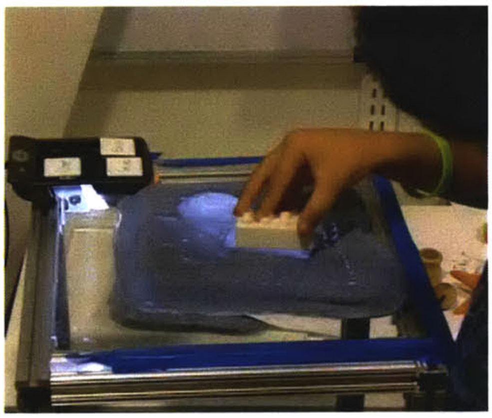
When participants did try to create full 3D structures they found the tools lacking, due to the limitations of 2.5D. One participant, P2, tried for around six minutes to create a DNA double helix with overlapping strands. The participant was unsatisfied with the fact that he could not create empty space between two of the strands when they overlapped. This seemed to highlight the limitations of 2.5D geometry vs true 3D geometry.

5.6.5 Supporting Many Paths

We observed many different styles of use during the KidCAD trials; however for the most part they fell into three categories: "stampers", "sculptors" and "sketchers".

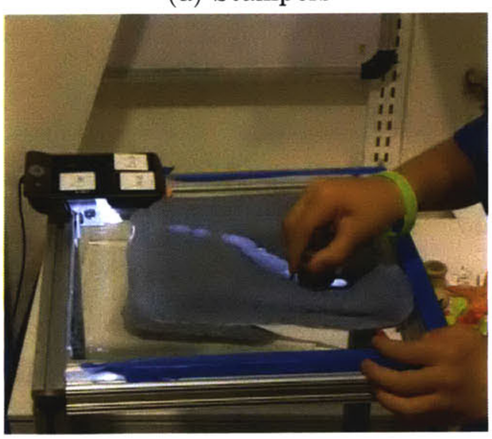
"Stampers" used KidCAD along the lines that we had created the system. These participants mostly used existing object, and copied them by stamping them down into KidCAD. They also used the drawing tool and erasing tool, as well as using their hands, but for the most part they were remixing existing objects.

"Sculptors" instead focused on using their hands or other tools to sculpt a 3D form. Even if they used objects, for the most part they were not copying the shape, but instead using it to deform the 2.5D geometry. These children treated the gel surface very closely to how one would sculpt with clay, often repeating the same basic path with their fingers or other tools to create more depth. One participant, P5, explained that the gel had a very similar feel to it to clay and that she liked the deformability. These users also seemed more concerned with the 3D form than many of the others,



(a) Stampers

(b) Sculptors



(c) Sketchers

(d) Some items used with KidCAD

Figure 5-17: a-c. Different ways of interacting with KidCAD. d. Some items provided for to children use as tools.

and looked at the 3D perspective view more.

The third group, "Sketchers," primarily used the drawing tool. They did not concern themselves with the 2.5D view and treated the canvas very pictographically. They created much of the content themselves and were less focused on remixing objects. P3 for example explained that he would use it for "exactly what I was doing, to make drawings for a story... like if I was telling a story to someone I would use this to make illustrations of what it would look like."

5.7 Discussion

Children found that it easy to copy geometry from physical objects using KidCAD and it was also clear to them what parts of the objects they were copying. One participant, P11, explained, she could "put 3D shapes on a rubber pad to make the same 3D shapes on the computer." Because the act of copying was embodied in the imprinting gesture children did not seem to perceive or talk about different modes, or the independent act of 3D scanning. It was clear to the users what the results should be because the deformation was an embodied process. It was also easy for the users to copy only parts of objects by only pressing those parts in, something that would be complicated with traditional 3D scanners. The inverted nature of imprinting to add material did not seem to bother the children.

Children seemed to be able to use KidCAD to remix objects through tangible imprinting very easily. One participant P8 identified this type of remix and enjoyed it, explaining her favorite part of KidCAD was that "you could turn every day things into a whole new idea." The advantage of using physical objects to design is speed; you don't have to make everything from scratch, and you can be inspired by the objects around you and create "new ideas" or designs that could be hard to think of. We believe that by grounding KidCAD in the physical world, it can help children to think about the process of design. The physicality and simplicity of creating geometry through stamping, sculpting and drawing allows children to design more quickly than with traditional digital modeling tools. And, in contrast to the physical world, children can easily create new toys with this system, without destroying or taking apart current ones.

Comparing the results of the of the user study with KidCAD to our initial user exploration with clay, we found children often used KidCAD in very similar ways to the clay experiments. Users of KidCAD relied less on drawing and created more sculptural forms, and were less reliant on thin lines. However KidCAD users did not frequently scale, rotate or move individual parts, instead focused on stamping techniques also observed with the clay experiments. Children, however, seemed to be able to more quickly explore alternatives, and more easily undo things with KidCAD. We observed children spend a longer time on individual designs with clay than with KidCAD, and more designs were explored with KidCAD when controlling for time. We believe further work could push KidCAD's modeling abilities further, especially beyond 2.5D sculpting.

In someways KidCAD is similar to existing building block type tangible interfaces. Users combine existing objects to create something new. However, we believe that KidCAD highlights a different type of design that is more improvisational. Instead of a fixed set of items that can be combined, KidCAD allows any physical object to be easily combined with other objects, in the digital world.

Instead of seeing the world the way it is, KidCAD encourages children to see objects in the world as tools to get what they want. Csikszentmihalyi explains that "every time we interact with an object the possibility of new learning is potentially there" and that artists and creative thinkers change their perception to see beyond what objects are "supposed" to mean [30]. KidCAD can also allow children to design with personal objects and many children reported that they had many objects at home they would like to use with KidCAD. Many physical objects can hold much more meaning and emotion than ink or clay, such as a shell you found on the beach. With KidCAD children can begin to explore expressing those meanings in new ways.

We observed that the 2.5D canvas has an effect on the way that children design and interact with KidCAD. The 2.5D canvas seemed to share more with the 2D page than with the full 3D space of traditional modeling. Children would layout scenes with many characters and a setting such as a tree, or a house. We did not originally envision this type of pictorial use, but it was an interesting emergent behavior. This could be due to a number of factors. Children are more used to the world of drawing, the 2D projection on the flat gel surface, and that relief sculptures historically have been more pictorial. Although we did not design KidCAD for all of these different patterns, they seemed to be well-liked by those who used them, regardless of which pattern they primarily used. All of these different paradigms are afforded to the user because of the wide variety of input supported by KidCAD. In addition there is no need for mode changes, instead users simply pick up different tools. It is easy for children to change styles quickly from one design to the next. This highlights the flexibility of KidCAD which in many ways mirrors that of clay; there are endless opportunities to modify clay; no one style is correct.

KidCAD focuses on tangible input, but not fully embodied interaction [37]. The system represents a hybrid approach with tangible, realtime 3D input but only real-time, co-located 2D output. Because of the co-located feedback children can engage in epistemic action [85], and we observed this type of interaction in our study. However, we found that in some cases children desired to have the physical object before it was 3D printed. They could not move or play with design until it is 3D printed, unlike the physical toys they used to design the object. Full 3D embodiment provides a great deal of advantages. However it is limited by the difficulty of computationally changing the physical model. For example, Sandscape stores the model in the physical world, therefore it is hard to computationally change the model. Because of its reliance on projected feedback on a 2D surface, KidCAD can easily change the model computationally, allowing for undo scale, reflection, etc. which are easy to implement in the digital world. We believe a hybrid approach, tangible input and co-located projected feedback, is more flexible and may come to be a more dominate method than fully embodied tangible design tools.

But it is not enough to merely replicate the fluidity and texture of clay sculpting and transport it to the digital, we need to consider how to provide the advantages of digital computation to these tools. KidCAD begins to explore some possibilities while remaining close to clay. We believe future work can push this boundary even further, while maintaining the ease and flexibility of interacting with clay.

We are currently in the planning stages of collaborating with an after-school arts program to do a longitudinal study with a whole class of children. This multi-week study will be useful for eliminating novelty bias, but is also more practical as the timescale for 3D printing is still quite slow. This will help us better understand how children would actually use KidCAD to make meaningful objects.

5.8 Conclusion

In this chapter we introduced a system that enables users to appropriate physical affordances of existing objects and tools in order manipulate digital information. By selecting and choosing from existing objects, a user has more freedom in the types of physical affordances and interaction styles than a single device can easily provide. In order to support these Appropriated Physical Affordances we created the deFORM malleable input device, which uses high resolution structured light scanning to capture expressive interactions using a wide variety of objects while providing passive haptic feedback. These Appropriated Physical Affordances are one type of Improvised Physical Affordance. In contrast to this chapter, in which users change affordances by selecting different objects, in the next chapter we examine how users can sculpt and create the affordances they require through direct deformation of a malleable user interface.

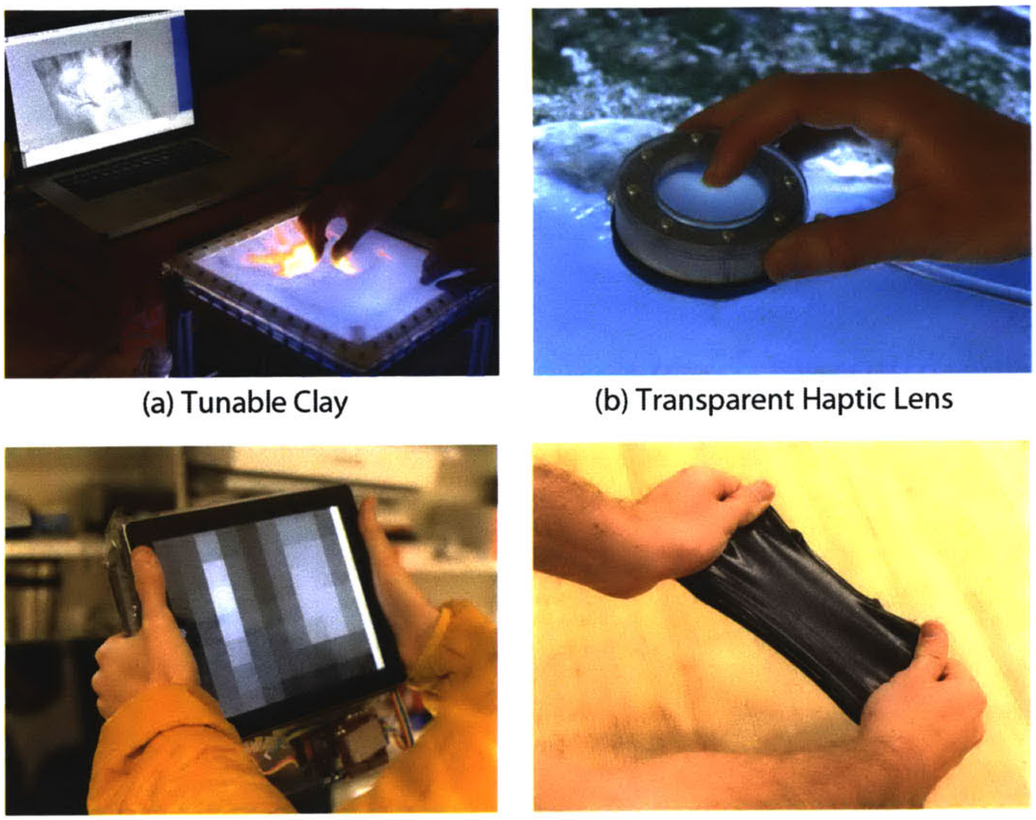
Chapter 6

User Defined Affordances

The previous chapter explored the idea of leveraging existing objects and tools for interacting with digital information by appropriating their physical affordances. This ability to pick and find objects can greatly expand the available affordances to interact with. However, users are still limited to what they have on hand. This chapter instead investigates how users can shape and form their own affordances on demand.

While other devices have allowed users to reconfigure interactive devices, to choose and arrange input elements [176], little research has focused on enabling users to shape the form of these devices and to radically change their physical affordances. Here we envision devices that users can bend, deform and shape like clay to create the buttons, grips, and other affordances they need to interact with different applications. Similar to The Bar of Soap device [164], these devices could understand how they are being held and touched, but also what shape configuration they are in, and respond by changing modes automatically.

This requires malleable and deformable devices with embedded shape sensing and touch sensing. Even more so these devices need to be highly flexible and malleable. However, users may not always want a flexible device - it is often quite useful to have rigid devices that provide more structure for holding and pressing. Our vision is to create a device that can change its stiffness on demand, allowing users to easily deform or stretch a device to change its shape, and then change the stiffness to lock it. Then a user can interact with the device as if it was a traditional rigid device, see



(a) Tunable Clay

(b) Transparent Haptic Lens

(c) Behind-the-Tablet Jamming

(d) ShapePhone

Figure 6-1: Jamming is a scalable technique for programmatic stiffness control, which can be applied to enable User Defined Affordances. Examples include: Tunable stiffness for malleable interfaces on tabletops (a, c), for haptic feedback (b, c), and for mobile shape-changing interfaces (d).

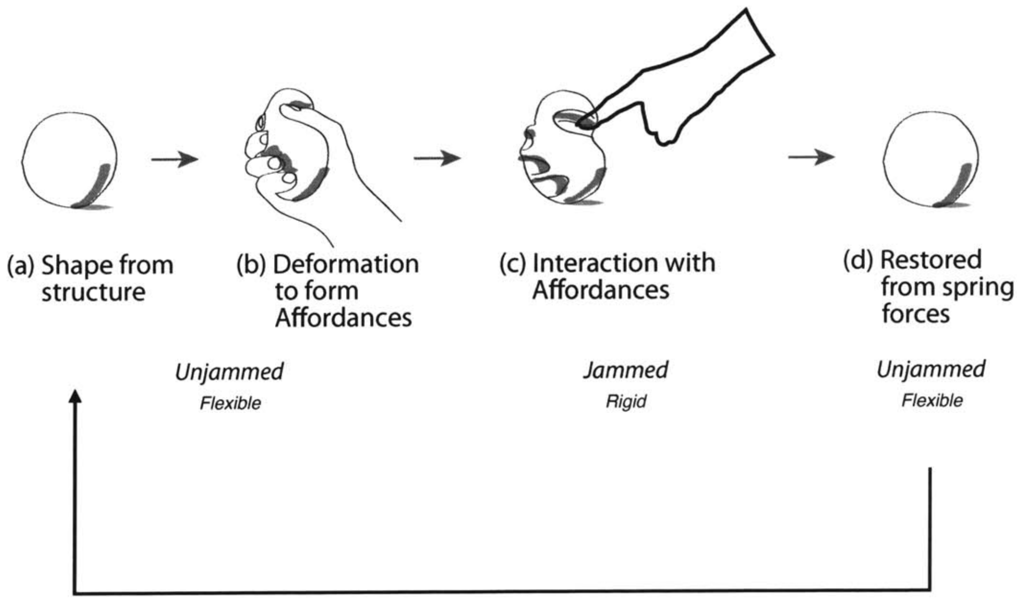


Figure 6-2: This diagram depicts interaction with User Defined Affordances. First a user can deform a flexible device to create the affordances she wants. Next the system changes from a flexible state to a rigid state by means of particle jamming. Now a user interacts with the device. Finally, a user can change the affordances by transitioning from a rigid state to a flexible one.

Figure 6-2. For example, a user can create a game controller with the buttons she wants by sculpting the Jamming User Interface.

The form, function and dynamics of user interface devices have traditionally been limited by the rigidity of materials used for sensing and display. Organic User Interfaces (OUIs) [174] embrace the advances of new technology and materials to enable deformable and actuated interfaces of arbitrary shapes. Major enabling technologies for such interfaces have included advances in sensing [136], display technology, and mechanical actuation [73, 125], but few projects investigate computationally controlled material properties, such as stiffness. In this chapter, we adapt particle jamming [14, 103] as a simple, effective method of stiffness control in HCI. The ability to switch between soft and rigid material states enables novel interactions for malleable, clay-like interfaces, haptic feedback and deformable devices.

Granular media can exhibit both fluid-like and solid-like states. Liquids typically flow freely due to external forces, while solids require certain applied stresses to deform plastically. Jamming describes a situation when granular media exhibits a yield stress, such that forces can be distributed through chains of grains as if each chain was a rigid object [19]. Thus, groups of particles as a whole can function as a compliant or stiff material, under computational control of this compliance level.

Engineers, architects, and designers have been utilizing these effective phase-change characteristics of granular media to develop devices, tools and systems that can transition between flexible and rigid states. Most applications induce jamming by enclosing grains in a non-porous, flexible membrane inside which a vacuum can be applied. The grains flow when excess interstitial fluid (typically air) is enclosed with the grains, but compact to form an effective solid when vacuum is applied inside the membrane to remove fluid. This enables drastic and reversible shape deformations using a single embedded actuator. Extremely flexible materials such as silicone can be utilized in jamming devices, allowing the system to be stretched, twisted, or bent with the jamming material flowing easily into these new shapes. However, when pressure is decreased these interfaces can become static. This makes jamming an ideal candidate for enabling malleable and shape changing user interfaces.

Jamming has become a popular research topic in the robotics community [14, 160, 161, 21], but its application to user interfaces has received less attention. A successful application of jamming to HCI requires advances in sensing to detect shape deformations and user input, as well as actuation for providing feedback to the user.

6.0.1 Contributions

We introduce techniques that exploit computer-controlled jamming of granular particles as a scalable method to programmatically control the stiffness of malleable devices. This, in combination with our embedded shape sensing, demonstrate the potential for novel interactions and increased expressiveness. Examples of how jamming can be utilized in HCI applications are illustrated in Figure 6-1 and 6-3. Our contributions include:

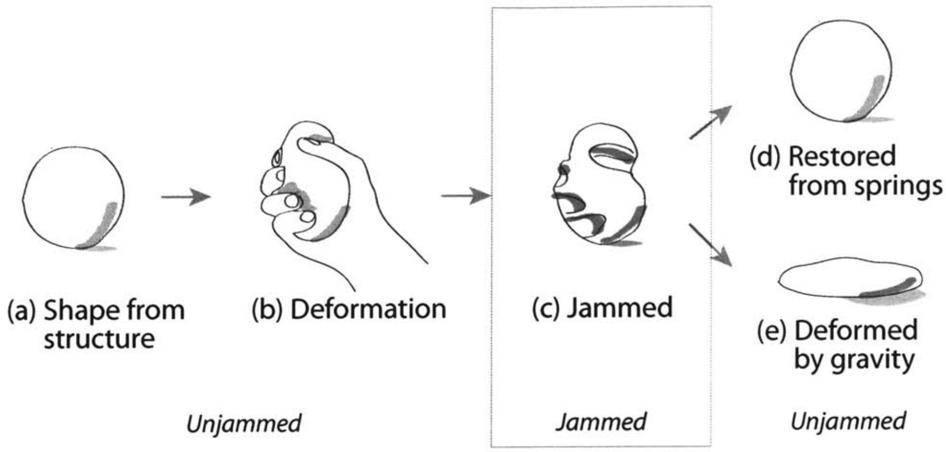


Figure 6-3: Jamming techniques enables new possibilities for shape state transitions. In this example, an object's shape is informed from structure (a), deformed by a user (b), and jammed to maintain the deformation (c). When unjammed, the object could return to its original shape if there are internal spring forces (d) or deform due to gravity (e).

- A review of the state of the art in jamming for use as a variable stiffness material from an HCI perspective.
- A novel hydraulic-based jamming technology, for rapid activation, silent actuation, and embedded optical sensing.
- Two techniques for high-resolution, integrated and embedded sensing for jamming interfaces: optical sensing, using index-matched fluids and particles; and electrical sensing, using capacitive and electric field sensing.
- A small, low-power jamming system for mobile and embedded organic user interfaces.
- Motivating prototypes to highlight how jamming can be applied to HCI

6.1 Background: Pneumatic Jamming Fundamentals

This section provides an overview of how jamming activation techniques enable the control of shape and material stiffness, and thus the degree to which a volume can be physically modified or actuated. This section includes a review and introductory discussion of jamming control to provide readers with the background for implementing their own systems; further details can be found in [160]. We also describe a platform for prototyping jamming user interfaces.

6.1.1 Pneumatic Jamming

Four main elements are required to control a jamming system: the jammable material and housing assembly (usually a non-porous, flexible membrane), a vacuum source or pump, a pressure-controlling valve, and a pressure sensor. We have implemented a closed-loop control system to achieve desired vacuum pressures as a test platform. While pressure relates to the magnitude of jamming, there is not necessarily a linear relationship between pressure and system stiffness [161].

Our system consists of an Atmel AVR microcontroller that interfaces with a 12V DC vacuum pump with a 20 cm^3/s maximum flow rate and a maximum vacuum pressure of 65 kPa, a 12V DC solenoid valve and an analog pressure sensor (see Figure 6-4). The vacuum pump, solenoid pressure-release valve and pressure sensor are connected in-line to the jammable module with 0.635-cm-diameter tubing. Coffee filters prevent particles from entering the air lines.

6.1.2 Differential Jamming Pressure and Activation Time

The differential jamming pressure is defined as the difference between atmospheric and internal volume pressure for the jammable module. For example, a balloon that is filled with jamming media and is open to atmospheric pressure, is near the jamming transition, since little fluid volume needs to be removed to induce jamming. The differential jamming pressure can, however, be raised to increase the mechanical stiffness of the system.

We can estimate the time it takes to cause a system to jam based on a pump's rated flow rate. Uniform spheres that are of random, close packing (as opposed to, for example, ordered in a lattice pattern) have a solid volume fraction of approxi-

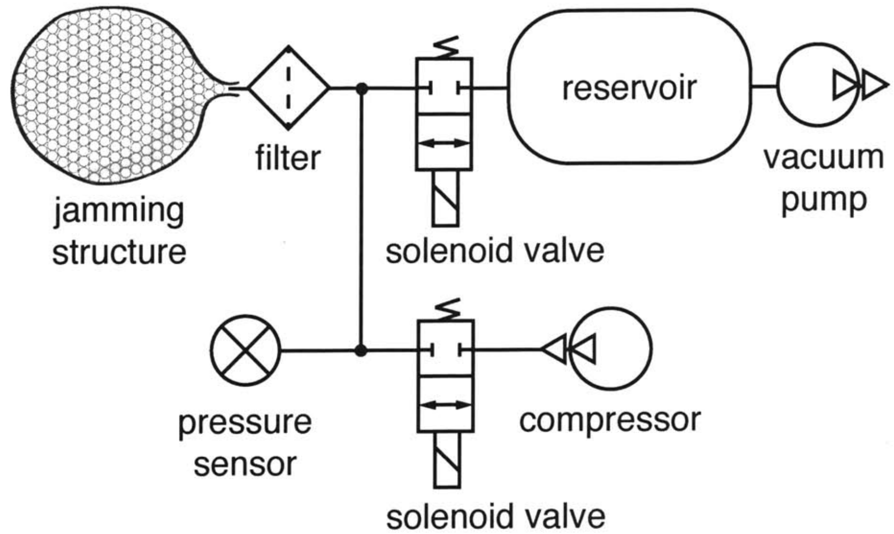


Figure 6-4: Pneumatic Jamming System. The system measures and controls the difference between atmospheric and internal volume pressure, such that particle jamming in the structure provides varying stiffness.

mately 0.64 [167]. Therefore, in a simple system in which a vacuum pump is directly connected to the jammable module, the amount of fluid volume, V_r, which needs to be removed to induce jamming can be approximated as:

$$V_r \approx V_b - V_g(1 + \frac{0.36}{0.64}) = V_b - \frac{m_g}{rho_g} \ (1.5625)$$

where V_b is current internal volume of the jammable segment (including excess fluid), and V_g, m_g, and rho_g are the solid volume, the mass, and material density of the particles, respectively. Therefore, the time to remove the excess volume, t_r, is:

$$t_r = \frac{V_r}{Q_p}$$

where Q_p is the pump's volumetric flow rate. Any additional vacuum that is applied to the system increases differential jamming pressure and system stiffness.

6.1.3 Accelerated Activation

While jamming speed is typically limited by the vacuum pump's and pressure-control valve's flow rates, it can be increased through the use of in-line reservoirs. For example, a PVC pipe can be added to build up vacuum pressure to increase jamming speeds. In addition, unjamming speeds can be increased by adding a positive pressure source [3].

6.2 Design Considerations For HCI

While actuated devices and displays have received extensive attention over the years, less emphasis has been placed on techniques for the control and modification of intrinsic material properties. The application of jamming has great potential to complement shortcomings of traditional shape-changing devices. In addition, due to its unique abilities to affect shape dynamics and kinetics, jamming is valuable as a standalone modality.

6.2.1 Facilitating Shape Deformation

Malleable interfaces typically need to both enable effortless deformation, and also provide mechanisms to stabilize resulting freeform shapes. Variable stiffness enables continuous transitions between compliant and solid objects. In addition to deformation in the unjammed state and solidification of the resulting shape in the maximally jammed state, there are interesting nuances related to expression and fidelity in the range of stiffness levels in-between. The type of deformation that is possible, and its effect on the overall shape, depends on material stiffness. It is thus possible to tune the control gain to tweak the precision and scale of user manipulations of the material shape.

User-applied forces or embedded spring elements can passively actuate shape change without the need for motors or active components, while passive spring elements may constrain motion or act as restoring forces. Adding a relatively stiff yet flexible plastic sheet to one side of a flexible bag with jamming media can, for example, constrain the otherwise extreme deformation possible with a loose bag. Besides limiting bending to certain axes, a restoring force can also help the system return to a specific shape when unjammed.

6.2.2 Augmenting Shape Actuation

Most actuation techniques for shape displays employ active elements to displace different types of media. While jamming does not provide actuation per se, it enables straightforward "locking" and "unlocking" of continuous freeform shapes with varying stiffness using a single actuator. The ability to maintain these states without the need to continuously power the jamming actuator is important for mobile, embedded and low-power devices. To change a jamming structure's shape dramatically, another source of actuation is necessary: either a passive source, such as the user's force or gravity, or an active source, such as a pneumatic air muscle. In addition to augmenting existing actuation techniques, novel actuators based on jamming structures could enable completely different shape-changing interfaces [160]. Granular particles can be combined with discrete element matrices as a hybrid approach to achieve smoother, higher-dimensional surfaces with variable stiffness. Passive, deformable shapes, with elastic or spring-loaded properties can also be added to the volume to provide restoring forces, so that when unjammed, the device returns to a certain shape.

The single actuator used to jam the particles may not only be used to accelerate the unjamming in reverse-operation, but could also be employed for inflation the jamming shape (similar to the technique described by Amend et al. [3]). By drastically changing the particle/medium ratio through inflation, we can allow the fluid jamming medium to dominate the shape volume and the user's experience of it.

6.2.3 Haptic Feedback Through Variable Stiffness

The ability to control material stiffness can be used as a degree-of-freedom (DOF) for an output device. The device stiffness can be directly mapped to represent object properties in simulation interfaces, such as various materials in a sculpting application. Stiffness can also be mapped to represent parameters, states and action in the user interface, as classical abstract haptic feedback.

6.2.4 Sensing Structure and Touch

It is often desirable to sense users' freeform deformations of malleable devices, including 3D shapes, as well as interaction on and above surfaces. Sensing proximity and touch allows 2D and 3D non-planar surface manipulations, which can be relevant and useful for a number of interactions [9]. Shape deformation can, besides the direct 1:1 manipulation of geometry representations, can also be used in pattern-matching of shapes. This could, for example, allow the embodiment of functionality, such that the device's behavior and interface would adapt to its form factor, or trigger different actions.

Jamming provides great flexibility for adapting the choice of particles and medium to a particular sensing approach, since there are no active electrical or mechanical elements that can cause interference in the volume.

6.2.5 Particle Types, Jamming Quality and Tactile Experience

The effect that different particle properties, such as size and shape, have on jamming performance has been extensively studied [104, 160, 21]. For user interfaces, the tactile experience is an additional important aspect.

For shape-changing interfaces, we are interested in particles that could achieve large changes in stiffness and jam in arbitrary freeform shapes. Ground coffee has previously been demonstrated as an effective material for systems that require large dynamic range in stiffness and strength [14, 21].

Glass beads provide a good balance of control and tactile stiffness response due to their smooth surfaces and low interparticle friction. This allows for a precise control over levels of stiffness for malleable manipulations, such as sculpting.

Other properties, such as particle weight or membrane thickness and elasticity, can

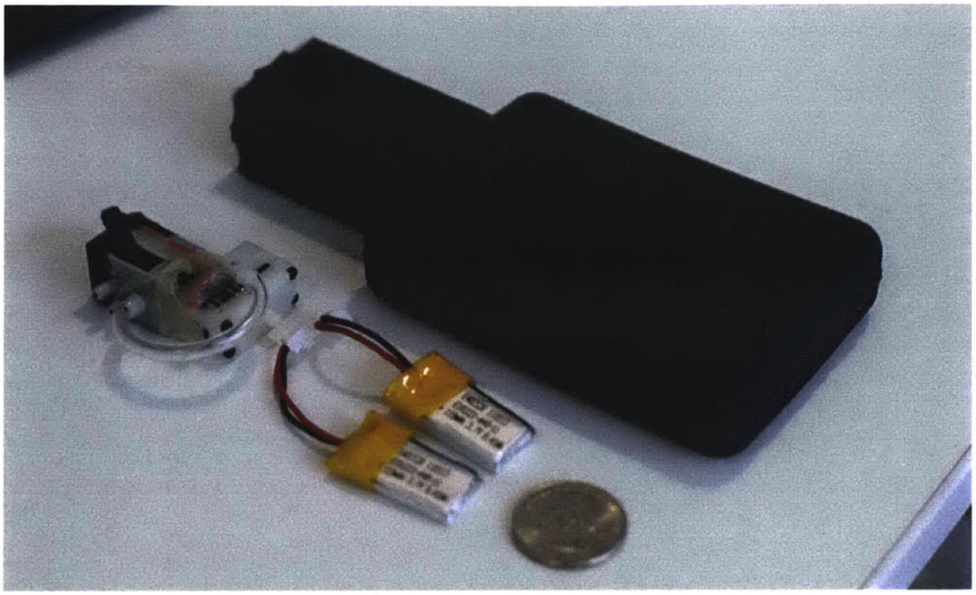


Figure 6-5: Our Mobile Jamming Platform (MJP) enables pneumatic jamming in a compact, low-power, battery-driven form factor. The MJP consists of a jamming volume, micro air pump, valve and LiPo batteries. (U.S. quarter for scale)

be optimized for a particular system design. The membrane qualities, for example, affect both the user's tactile experience and the jamming performance.

6.3 Novel JammingTechnique

6.3.1 Mobile Jamming Platform: Pneumatics for Portability

Jamming has great potential in enabling haptic feedback, malleable input, and shape-changing structures for flexible mobile devices, such as future tablets, e-readers, or mobile phones. Mobile jamming needs to be compact and self-contained, which introduces constraints on size, flow rate, maximum vacuum force, power consumption and sound level (e.g., due to the vacuum pump).

Our Mobile Jamming Platform (MJP) consists of a small vacuum pump, small solenoid valve, control circuit and battery pack, and measures $47 \times 27 \times 8$ mm^3 (see Figure 6-5). The pump draws 0.12A at 7.4V, and our current 100 mAh LiPo battery allows for one hour of continuous use of the pump, which means several hours in practice, as stiffness changes are rendered intermittently. Our MJP can currently

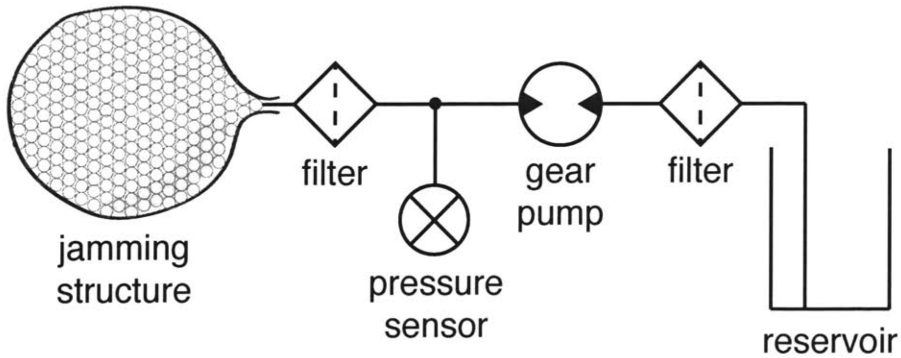


Figure 6-6: Hydraulic jamming system. Simila to a pneumatic system, a closed-loop control system measures and manages the differential jamming pressure. Hydraulics can, however, allow higher stiffness, silent operation, and faster actuation.

jam/unjam a cell-phone-sized volume of coffee particles in approximately one second.

6.3.2 Hydraulic Jamming: Fast, Silent and Transparent

Hydraulic jamming systems can be created by using liquids as the interstitial fluid between the particles, instead of air (a gas at room temperature). Since liquids are incompressible, hydraulic systems have higher efficiency, can be stiffer, quieter and can withstand more stress and load compared to pneumatic systems [67]. Hydraulic jamming can also enable optical sensing and transparency through the use of index-matched fluids and particles, which we describe in the next section.

We built several hydraulic jamming systems to investigate feasibility and performance compared to pneumatic systems. The system design is similar to a pneumatic system: a DC hydraulic pump, controlled by an H-bridge and microcontroller, moves liquid in and out of the system from a reservoir to change the differential jamming pressure. The pressure is digitally measured with a pressure sensor, and regulated by a control circuit and a hydraulic pump, as shown in Figure 6-6. Our hydraulic gear pump is $7.62 \times 10.16 \times 5.08$ cm^3, with a 2.3 liter/minute maximum flow rate and a maximum pressure of 151 kPa. Metal mesh filters prevent particles from entering the fluid line and the pump. The pressures required for jamming are significantly lower than pressures used in traditional hydraulic actuation systems because we are not

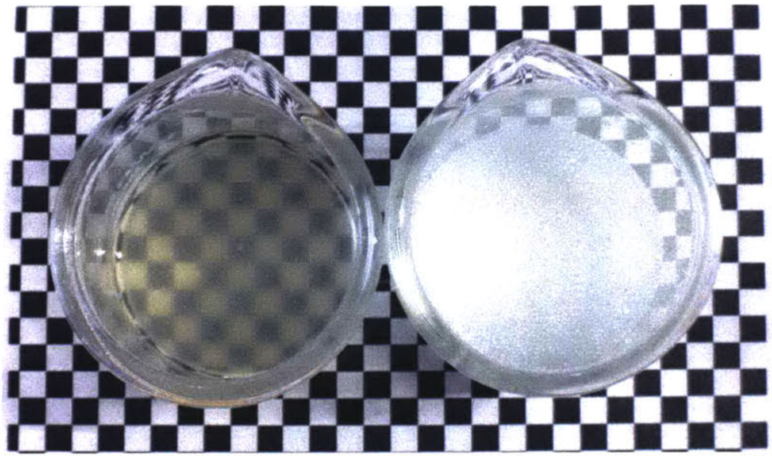


Figure 6-7: Transparency through index-matched fluid and particles. 3.5 cm of 1 mm Pyrex glass beads immersed in index-matched oil (left), and air (right). The oil reduces refraction as light enters and leaves each glass bead, with a drastic increase in transparency.

trying to transmit large forces; the goal is to change the interior pressure in reference to the external air pressure of 101.325 kPa.

6.4 Sensing For Jamming Interfaces

In this section, we discuss approaches that are particularly suitable to enable the shape and touch sensing that is necessary to leverage the flexibility and malleability of jamming structures for HCI.

6.4.1 Optical Sensing Through Transparent Jamming Volumes

To enable optical sensing of the interface's 3D shape, while avoiding user interference, occlusion and bulky system configurations, it is necessary to integrate cameras below the surface. This, however, requires thin [18] or optically transparent material [40].

Index-matched Hydraulic Jamming

A jamming system cannot provide optical transparency simply by using transparent particles, as each particle acts as a light-scattering lens, which makes the overall volume opaque. As light leaves the medium (e.g., air) and enters the particle (e.g., a glass bead), it refracts at an angle governed by Snell's law, due to the different refractive indices. However, by using a fluid that matches the refractive index (n) of the particle, we can suppress refraction and create an optically transparent volume.

Our hydraulic jamming system gives us flexibility to select fluids and particles with matching refractive indices. We chose to use the combination of boroscillicate (Pyrex) glass beads (n=1.474) and vegetable oil (n=1.4674–1.4736, depending on temperature and density). The volume is not completely transparent due to a slight deviation in the refractive indices.

However, our experiments show that the system is sufficiently transparent for optical sensing using projected reference patterns up to an 8 cm thickness of particles. The opacity was measured using a 2 mW red laser and a photometer at different reference thicknesses, and compared to glass beads alone. We determined that 4 cm of glass beads and oil provides 94% transmission, and virtually no transmission for glass beads alone, whereas 8 cm of glass beads and oil provides 47% transmission (see Figure 6-7). This configuration allows a rear-mounted camera to see through our transparent jammable volume, composed of index-matched fluid and glass beads with a transparent plastic bottom and an upper flexible opaque silicone skin, as shown in Figure 6-9. This device enables the use of different optical techniques for surface reconstruction, such as shape from shading, photometric stereo, embedded tracking markers in the skin [177], structured lighting, or other custom solutions [40].

Depth from Structured Light Through Transparent Volume

For 2.5D sensing through the optical jamming system we choose a custom IR structured light 2.5D scanning system, similar to the deForm system [40], due to its high resolution capture, ability to rear-project visible light content, and its flexibility with regards to changing cameras, projectors, and lenses. Three sequential fringe patterns are rear-projected in IR onto the deformable skin and are captured in 640×480 pixels at 60 frames/s by a side-mounted, synchronized IR camera. The 3 mm-thick silicone skin, with a durometer of 10 shore A, can be stretched and deformed 30 mm above and below its resting height. The deformations of the three patterns are used to re-

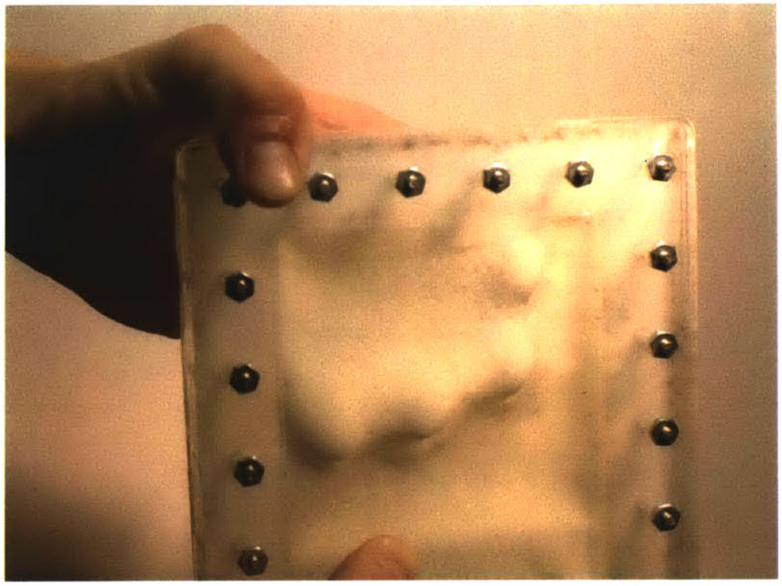


Figure 6-8: Jamming volume for optical sensing. 3 mm of Pyrex glass beads with index-matched fluid and particles enclosed between a flexible white membrane and a clear plastic sheet. Surface deformations are visible through 8 cm of jamming material.

construct 2.5D images at 20 frames/s from a 23×18 cm^2 region, at a spatial resolution of 28 pixels/cm, and 0–6 cm depth range, providing a 1–2 mm depth resolution.

Touch Sensing Using Structured Light

The greyscale surface image from our structured light capture system is also used to track touch points. The system works similarly to an IR diffuse illumination touch system, as the IR projector illuminates the silicone skin. By utilizing a thin, semi-transparent skin made of silicone, the camera captures reflections from fingers as they make contact. Other touch sensing techniques, such as FTIR, could also be explored with our transparent jamming system.

Limitations

While this approach provides high-resolution shape and deformation tracking combined with touch sensing, its use is limited to hydraulic jamming systems. Camera and optical sensor placement restricts the system's flexibility, and non-perfect index-matching complicates sensing at greater depths as transparency decreases.

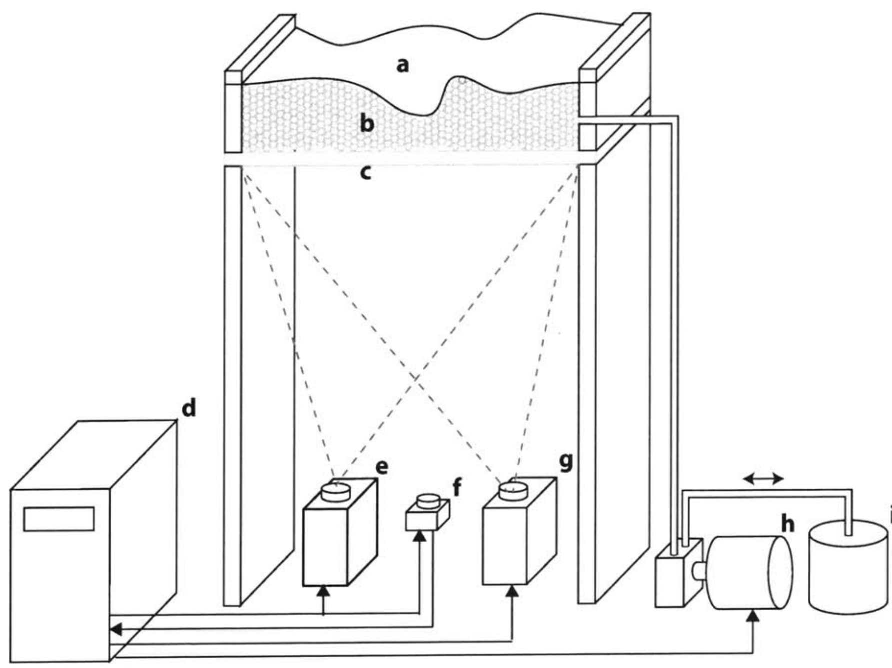


Figure 6-9: Structured Light Depth Sensing System with Index-Matched Jamming: (a) Silicone Membrane, (b) Pyrex glass beads and oil, (c) acrylic plate, (d) computer, (e) structured light IR projector, (f) IR camera, (g) graphics projector, (h) hydraulic pump, (i) reservoir

6.4.2 Capacitive Shape Sensing

Capacitive sensing can provide a scalable embedded approach to sensing shape in jamming interfaces, including deformations such as stretching, bending and twisting. In contrast to other techniques, such as resistive pressure sensors or electric impedance tomography, capacitive distance and shape sensing do not rely on a present applied force to the sensor. This makes it advantageous for both absolute and relative input.

Distance Sensing

The amount of known dielectric material between two electrodes can be measured through capacitance, and correlated with the distance between them. Pressure sensors have employed this principle [151], which also extends to larger distances and electrodes that can be used for flexible jamming volumes. In our system, an electrode transmits a reference square wave in the 100 kHz range to a receiving electrode, and the signal is sampled by a 12-bit A/D converter in an ARM microcontroller running at 72 MHz. We use synchronous under-sampling to demodulate the signal and recover the original amplitude, which is proportional to the capacitance between the electrodes. 32 samples are averaged to remove white noise.

Dielectric Properties and Sensitivity

Stretchable and bendable electrodes are needed for integration in the flexible jamming volume. We use silver-plated 76% nylon, 24% elastic fiber fabric, which has a low surface resistivity, and can be stretched up to twice its length. We insulate the fabric in a non-conductive silicone cast and use Pyrex glass beads as dielectric material. Pyrex glass beads have a dielectric constant of 4.6, whereas air has a dielectric constant of 1.00059. Assuming a random close-packing of glass spheres, 64% of the volume will be glass and 36% will be air [167], resulting in an overall average dielectric constant of 3.3. Hydraulic jamming greatly raises the possibility of increasing this dielectric constant. Using water with glass beads in the jamming volume could approach an average dielectric constant of 30 and increase sensing resolution at larger distances.

With simple two-electrode capacitive sensing through glass beads we are able to measure distances of 0–20 cm, with 5 mm or better accuracy (accuracy increases when the two plates are closer to each other).

Shape-sensing Prototype

Using rows of transmitting electrodes in a rigid back, and columns of receiving electrodes in a flexible skin, we sense the jammable volume's shape through time-division-multiplexing for each of the intersections in the sensing matrix and output a 2.5D depth map. Our prototype of the capacitive shape-sensing input device with jamming haptic feedback uses a 9×9 electrode grid. It measures $25\times17.5\times3$ cm^3 with an active sensing volume of $18\times11.5\times3$ cm^3. An overall 25-mm thickness filled with 2 mm glass beads are sealed within a highly flexible upper membrane and a bendable, yet relatively rigid, bottom surface. This device can be placed on a desk, or embedded in the back of a mobile phone or a tablet. Conductive fabric strips (9×1 cm^2 each) are embedded in the flexible skin as receiving electrodes, while strips of copper tape (also 9×1 cm^2 each) on the opposing, bottom surface act as transmitting electrodes, as shown in Figure 6-10. In addition, a layer of grounded conductive fabric on top of the flexible skin shields the system from the user. An analog multiplexer connects the receiver electrodes to our amplifier circuit and an ADC on an ARM micro controller running a custom C program, see Appendix C for the software implementation. The current prototype runs at 30 Hz and transmits data over USB serial or wirelessly using Bluetooth. The depth map is filtered and scaled by a factor of ten through bi-cubic interpolation (see Figure 6-11). The speed and resolution could be increased with dedicated hardware, and code-division-multiplexing could be applied for scalability.

Sensing Additional Dimensions

Separating transmitting and receiving electrodes into rows and columns for deformation sensing is only one approach to capacitive shape sensing electrode layouts; additionally, each electrode can act as both a transmitter and receiver. This can enable stretch, tilt or twist input to be quantified by measuring capacitance between

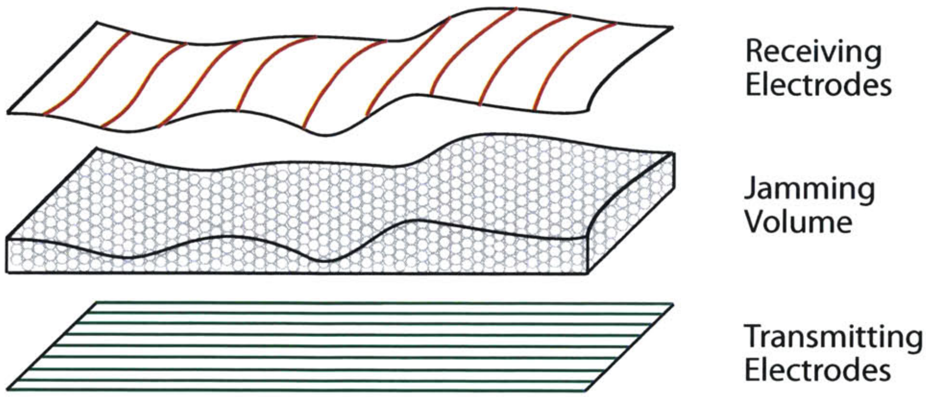


Figure 6-10: Capacitive shape sensing system. The jamming volume's shape is computed by measuring the capacitance at each transmitter-receiver electrode intersection.

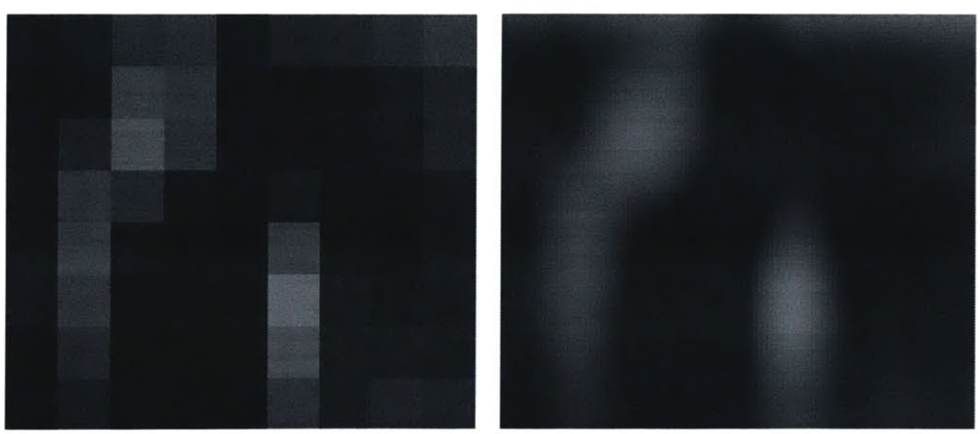


Figure 6-11: Left, Raw depth map output of 9×9 capacitive shape sensing system. Right, Bicubic interpolated depth map.

adjacent electrodes with different layouts.

Integrating Capacitive Touch Sensing

Our capacitive sensing also supports integrated multi-touch input. By replacing the flexible ground layer with lines of conductive thread that transmit the same reference signal, we enable mutual capacitance touch sensing [136]. When fingers or hands approach the conductive thread transmitting the reference signal, they capacitively couple with the system and decrease the signal. Time-division-multiplexing makes it possible to use the receiving electrodes both for shape-sensing electrodes below, and touch-sensing electrodes above, which reduces the total number of required electrodes for shape and touch sensing. To improve results, we use thin conductive thread (sewn in a zig-zag pattern for flexibility), instead of the thicker conductive stretch-fabric. The thread is more sensitive to capacitive coupling from the user, as its smaller size results in weaker coupling between transmitting and receiving electrode pairs. When not sensing touch, the conductive thread electrodes can be connected to ground to help shield the device. We built a small test system with a 3×3 touch-sensing grid for touch, pressure and hover sensing.

6.5 Applications and Prototypes

We built three prototypes that investigate the potential of variable material stiffness for different user interfaces.

6.5.1 ShapePhone: Shape-changing Devices

ShapePhone, depicted in Figure 6-12, is a user-defined mobile device that can be shaped into different forms and then locked into a rigid device for various forms of interaction. With our initial ShapePhone prototype users can transform the affordance of the device—from a phone, tablet (sheet), remote control, watch, game controller, or ball—by stretching, bending and molding ShapePhone when it is unjammed and thus extremely pliable, due to the stretchy silicone skin. The user can control the

(a) Phone

(b) Remote Control

(c) Watch

(d) Game Controller

Figure 6-12: The ShapePhone mobile device can be formed into different jammed shapes.

(a) Tablet Back. (b) Tablet Front.

Figure 6-13: The Behind-the-Tablet Jamming Interface enables malleable input with varying stiffness as haptic feedback, while avoiding occlusions with on-screen content.

jamming state using a small switch. When unjammed, ShapePhone returns to its normal state of a phone-sized rectangle, using the silicone skin as a restoring force.

Our prototype uses the Mobile Jamming Platform, described earlier, to control jamming in a small form factor, and enables ShapePhone to be entirely self-contained. The phone-shaped hollow silicone (Smooth-On EcoFlex 0030) body was cast from a 3D-printed three-part mold. This particular silicone is very flexible and can stretch up to four times its size. The skin is filled with coffee grounds and sealed with a tube for airflow connected to the MJP.

It would be relatively straightforward to add the previously described capacitive shape sensing techniques to ShapePhone to sense a variety of different shapes. These shapes could be used in addition to contextual information gathered through other sensors, or program state, to enable further functionality. Capacitive touch sensing could also be incorporated for user input and recognize how the user is holding the device to enable contextual information [164].

This same jamming phone device could also be used for interaction and haptic feedback while in a pocket. Changes in stiffness could convey battery life, for example, letting the ShapePhone "melt" when it runs out of battery, or allowing user input through the pocket using squeezes or deformations.

6.5.2 Behind-the-Tablet Jamming

In order to investigate malleable interaction and haptic feedback in the context of mobile devices, we created a jamming input device mounted on the back surface of a tablet, shown in Figure 6-13. A custom tablet case has an embedded jamming apparatus and shape deformation sensor for malleable interaction in the back of the tablet. A user can shape the different grips they need for certain applications. Or the user could create buttons on the back by sculpting them. In another scenario the tablet's rear interface allows users to navigate content on a tablet display by pressing into its malleable surface. This could, for example, be used for browsing information on the tablet using gestures, while receiving jamming-driven haptic feedback. A possible scenario could use kneading on either side of the tablet back to scroll content in that direction, or using both hands to zoom. When a limit is reached, the corresponding part of the tablet could turn stiff, preventing further kneading. This scenario could also allow deformations beyond what is possible in the Tunable Clay interface, since there is no occlusion by the user's hands. As in the previously described interfaces, changes in stiffness can enable different modes of user interaction.

The mobile jamming platform is pneumatically controlled with an on-board vacuum pump and uses capacitive shape sensing. We implemented two variations of the tablet. The first uses Bluetooth to communicate the capacitive shape sensing and jamming control to a tablet, which runs our Android application. The second used an iPad with screen-sharing software to view desktop applications that interface with the hardware over a serial cable.

6.5.3 Tunable Clay: Precision and Quality Through Stiffness

Tunable Clay (shown in Figure 6-14) is a malleable input device for 3D modeling, where material stiffness can be tuned to comply with different sculpting modes. The interface is inspired by our research group's previous work in 3D modeling in projects such as Illuminating Clay, SandScape [70] and deForm [40]; we are interested in observing how different materials influence the creative process. Tunable Clay is a

Figure 6-14: Tunable Clay uses material stiffness as an extra dimension for 3D modeling in its malleable interface.

30×33 cm^2 malleable tabletop, designed to mimic the malleability of clay, which is a continuous material that users can easily deform. Optical sensing—achieved using structured light through the back of the transparent, hydraulic-activated jamming volume—captures the shape in real-time and applies it to a virtual 3D model. The model is shown both on a separate display and through projected graphics on the malleable surface for direct feedback. The sensing and visible projection is integrated beneath the surface to avoid occlusions from user interactions.

Users can control the stiffness of the malleable surface using a potentiometer. This allows users to modify the resolution of manual input, thereby modifying the interface's control gain. One can increase the stiffness of the interface for detailed work, decrease it to increase malleability or to reset the shape. Tunable Clay highlights the potential benefits of controllable material properties to vary interaction style, precision and feel.

6.6 Discussion and Design Considerations

When designing a jamming system with shape and touch sensing, several design decisions are of importance, as demonstrated in our approaches for activation, sensing, and interactive applications and prototypes.

Jamming performance depends on activation technology and particle type, with hydraulic systems and high-friction particles offering wider dynamic stiffness range, while speed of activation can be accelerated using in-line reservoirs. Tactile experience, surface quality and malleability benefit from low-friction particles and thin, elastic membranes.

Hydraulic jamming enables optical shape and touch sensing through transparent volumes, and provides strong, rapid and silent operation. For mobile and embedded devices, pneumatic jamming has the advantages of being lightweight, simple and relatively small, as it can utilize the ambient air as the fluid reservoir. Our MJP demonstrates this with a combination of low-weight particles and compact elements for activation and capacitive sensing.

To address the loudness of most air compressors, such devices can be run at lower voltages if slower actuation speed is acceptable. The effect of gravity in a mobile jamming system can also be addressed using multiple compartments to constrain material placement.

6.7 Future Work

Currently our system does not provide a way to connect user input and the physical affordances a user created to generic applications, and is instead hard coded. This points towards future work in creating tools for end users to connect user defined affordances to certain application functions, which could build off of systems like Exemplar [56] or Voodoo IO [176]. One could imagine programing this functionality by example, first creating the affordances, then performing the input and linking it to a function.

Our prototypes utilize passive actuation either from the user or from restoring forces. There is, however, a large space to explore in actuation. Techniques, such as pneumatic artificial muscles, as well as other inflatable structures, could be used to quickly change state and help jamming enable an even wider array of shape-changing interfaces. Our next steps are also to explore integration of our jamming techniques with actuated displays and devices similar to those used in PneUI [189]. This could enable interfaces that support both Dynamic Physical Affordances and User Defined Affordances. In addition, this chapter investigated applying particle jamming to User Defined Affordances, but recent advances in layer jamming [121] (using friction between thin sheets of material) could also be applied to this domain.

Further work is required to explore other sensing techniques that can be integrated with flexible jamming devices. The conductive fabric we currently use is capable of only half the strain of that of the silicone used, and thus limits system flexibility. We plan to investigate other approaches to embedded electrodes and wiring, such as embedded liquid metal [124] and saltwater, for stretchable capacitive shape sensing.

Once flexible and stretchable displays are widely available, they will enable flexible mobile jamming devices with integrated displays. Until then, such future jamming devices and their related interactions can be prototyped using projection.

6.8 Conclusion

This work demonstrates how jamming of granular particles can be applied to malleable, flexible, and shape-changing user interfaces that allow users to define their own affordances. By embedding sensing through index-matched optical sensing or capacitive shape sensing, we enable jamming interfaces to become high-resolution input devices. We also show how jamming can be miniaturized for mobile applications. Through three prototypes and two activation technologies, we demonstrate a range of possibilities of jamming user interfaces applied to User Defined Physical Affordances, and point towards future work.

Chapter 7

Discussion

The previous chapters reviewed the underlying cognitive mechanisms that contribute to the benefits of physical affordances, and introduced methods to support rich physical affordances using shape-changing and deformable interfaces. In this chapter we look back at the design space of Dynamic Physical Affordances and consider when these new techniques are successful, how we can help designers prototype Dynamic Physical Affordances, and look forward to how Dynamic and Improvised Physical Affordances can impact a changing interface landscape.

7.1 Design Space of Dynamic Physical Affordances

This thesis has explored the design space of dynamic and improvised physical affordances through a number of examples; however, it can be useful to map a larger space. When considering the prior work on understanding affordances, Kaptelinin's handling and effector affordances may be the most central to this work [81], but we also must position these affordances in the context of their application to Tangible User Interfaces. Fishkin's taxonomy is useful for understanding different types of TUIs, and how these techniques can be applied to a broad set of application domains. But in this work, change (shape or stiffness) is the most central element, more so than form alone, and therefore it is also important to consider what changes (i.e. the handling or effector affordances) but also how it changes (i.e. what are the mechanisms of that

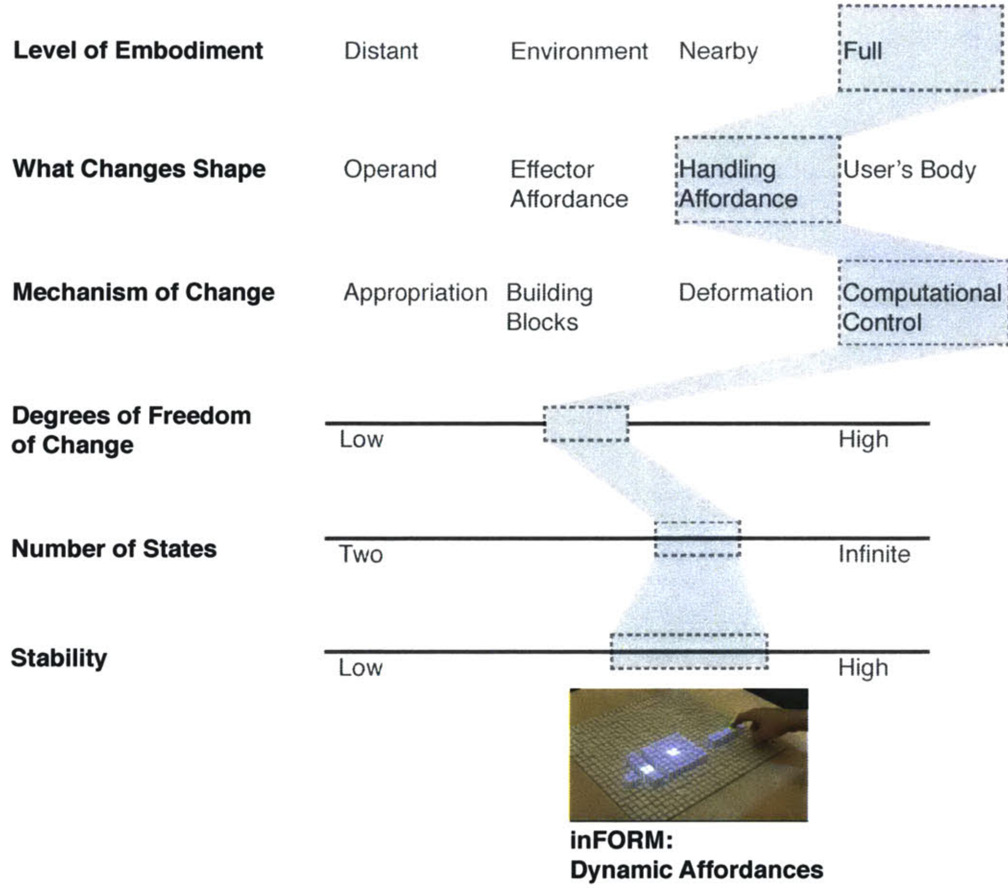


Figure 7-1: The design space of Dynamic Physical Affordances, with inFORM's Dynamic Affordances highlighted.

change). Figures 7-1 and 7-4 show the projects in this thesis are plotted on the design space, but other dynamic affordances can be plotted too, see Figure 7-4d, and most importantly open areas can be explored further.

7.1.1 Level of Embodiment

Fishkin's taxonomy of Tangible Interfaces describes the Level of Embodiment of different interfaces as one of two important axes to consider [37]. He suggests that on one end there are interfaces that are fully embodied, meaning that their input and output are completely collocated, and on the other interfaces where the physical controls map more loosely in a 'distant' configuration, see Figure 7-2. Here we can plot many different approaches to tangible interaction; on one end there is the self-contained interface, on the other end more of a controller based approach, and somewhere in the middle are tools for manipulating digital information through direct manipulation. These different types of interaction change how users will interact and the affordances that they need.

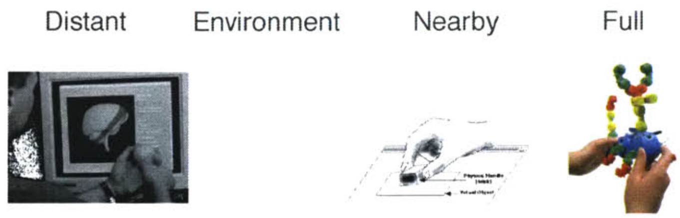


Figure 7-2: Three examples along Fishkin's Levels of Embodiment. Distant: Hinckley's props for neurosurgeons [64]. Nearby: Graspable Bricks [38]. Full Embodiment: Topobo [126]

7.1.2 What Changes Shape

Based on Kaptelinin's concept of handling and effector affordances, we can look at which part of the affordance is made dynamic, see Figure 7-3. However, as this thesis has shown we can use dynamic shape change to appropriate existing passive objects as dynamic affordances, so we must consider a wide range of points for shape change including the user, and the operand, (i.e. the object being modified).

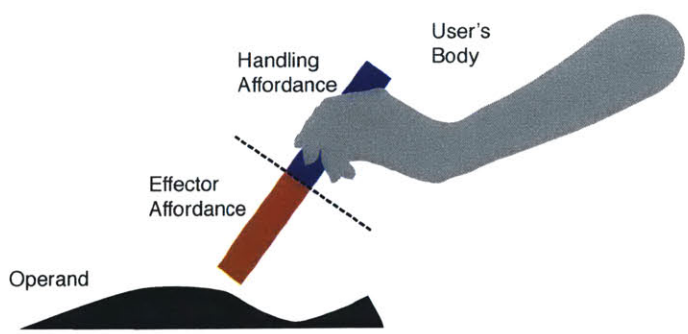


Figure 7-3: In a Dynamic Physical Affordance the user's body, Kaptelinin's Handling and Effector Affordances, as well as the operand or object, itself, can change shape.

7.1.3 What Is the Mechanism of Shape Change

The mechanism that provides the shape change can be the user appropriating other objects, the user selecting a number of parts and constructing them using a building block set, the user's own motion and deformation of a device, or a system's control through some type of actuator.

7.1.4 What are The Degrees of Freedom of That Change Shape

The degrees of freedom of the shape change contribute to the complexity of forms that the dynamic affordances can have. For example, a clam-shell style mobile phone, has one degree of freedom (a rotational hinge) that allows it to change from its closed state, which affords being placed in a pocket, to its open 'phone' state, which affords holding up against one's face to talk.

7.1.5 Number of States

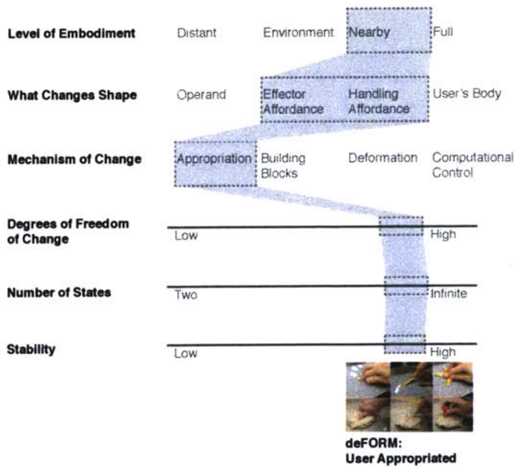
Some dynamic affordances will only have binary states, only able to change between two different states, such as Harrison's inflatable affordances [53]. Others like those displayed using inFORM will have near infinite number of states.

7.1.6 Stability

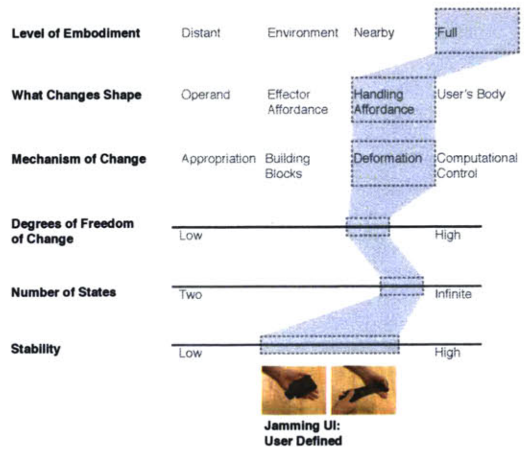
How stable is each state, how easily can it transition, and will it transition accidentally? The use of jamming to change stiffness makes it easy to computationally control this stability.

7.1.7 Areas Unexplored in This Design Space

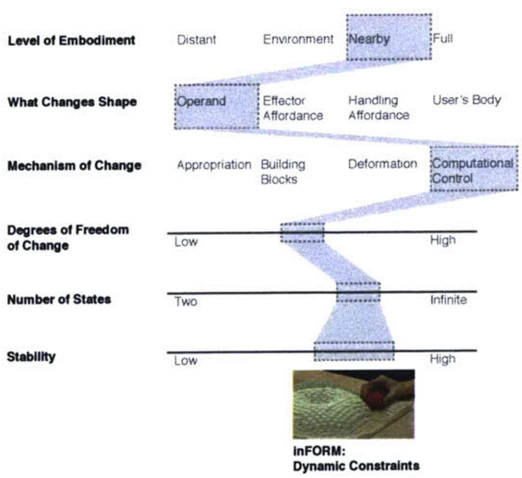
An interesting direction that this thesis did not explore that is uncovered by the design space is interfaces that can support both Dynamic Physical Affordances and User Defined Affordances. This could be enabled on a shape display or some other malleable input device that has actuation. Users could shape the affordances they



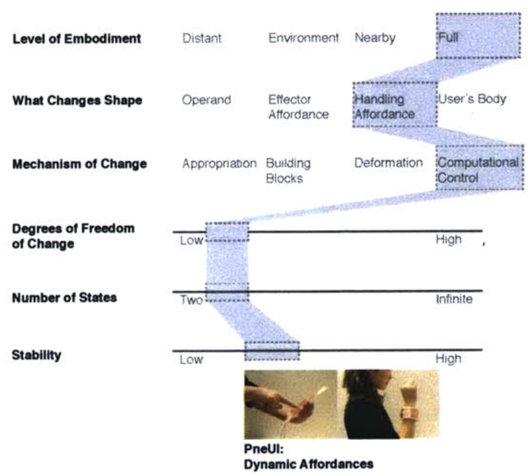
(a) deFORM - User Appropriated Affordances



(b) Jamming - User Defined Affordances



(c) inFORM - Dynamic Constraints



(d) PneUI - Dynamic Affordances

Figure 7-4: The design space of Dynamic Affordances for inFORM, deFORM, Jamming UI, and PneUI.

want and the system could record them and return to them at a later date. In addition while creating User Defined Affordances the system could provide feedback through the shape-output to guide them to certain 'hidden affordances,' or even apply a snap-to-grid type of interaction. This type of hybrid approach may combine the best of these two types of affordances, allowing for personalization while still transitioning between states quickly.

One area not explored in this thesis is the use of building blocks to change the affordances of a device, as shown by prior work like Villar's Voodoo IO [176]. However, this could be adapted to have dynamic blocks that could change form on demand to create different affordances, similar to Parkes's Bosu [126].

Another area that should be explored further is handheld tools that change shape and form, while interacting with a larger system. Tools could adapt both their handling affordances and their effector affordances to different operands they are working on. A simple example would be a screwdriver that could change its head from philips to flat head when a different screw is presented. But going further, one could imagine a hammer that turns into a screw driver, or a drill, or a putty knife all based on the context of the operand, i.e. the job at hand.

Further investigation of inter-material interaction, utilizing shape-change of both a tool and an operand should be considered. Here, one could imagine using multiple tools as well; one tool could rotate the operand as the user holds another tool in their hand. Finally, what if the user's body or a body worn glove or device, changed form? We could imagine having many hands at once, or making our hands smaller to manipulate a small object. Dynamic body modification in the context of affordance may allow for even richer interactions than limiting the dynamic shape-change to our tools.

7.2 When to Use Dynamic Physical Affordances

Bill Buxton's axiom, "Everything is best for something and worst for something else," is a reminder to us as HCI researchers and designers not to overstate our interaction techniques usefulness and instead to support a plethora of interaction modalities [15]. It is important to consider when Physical Affordances, Dynamic or Improvised, can make a difference, and to weigh the pros and cons of these techniques before advocating for their use.

In his thesis, Brygg Ullmer discusses when to make use of Tangible Interfaces over traditional GUIs [168]. Ullmer suggests that it is not as easy a question as it may seem, however he highlights a number of areas (Collocated Collaboration, Physically Situated tasks, and Spatial tasks) and application domains (Modeling and simulation, Visualization, Systems management and control, education, remote communication, entertainment, and artistic expression).

This raises the question when is it necessary to go one step further, to make those physical affordances provided by Tangibles dynamic or deformable? Ullmer and Ishii position the TUI specifically as special purpose, compared to the generality of the GUI [72]. This thesis instead suggests that we can have the physical affordances of TUIs, while also having the flexibility to support more types of applications.

7.2.1 Attending to Multiple Foci and Bi-manual Interaction

Dynamic and Improvised Physical Affordances leverage many of the same advantages as TUIs, and it is worth noting two areas where they are especially useful. Due to their inherent physical instantiation, Dynamic and Improvised Physical Affordances can be manipulated by touch, and potentially without visually attending to them. This means that a user can more easily focus their visual attention elsewhere, while still manipulating these physical affordances. These affordances also lend themselves well to bi-manual interaction, where one hand can manipulate one object and another hand another. These Dynamic and Improvised Physical Affordances also allow for manipulation and interaction with by a user's hands with more than just the finger tips.

7.2.2 Space Limited Applications

Tangible User Interfaces often require a great deal of physical space, and the more complex the application, often the larger it will be. Obviously there is a great advantage to making use of space for cognition, however there are times when this may not be feasible, such as mobile interaction. Here, like GUIs, shape-changing UI making use of Dynamic Physical Affordances may only display a small subset of the available interactive possibilities at once. But unlike a GUI it can still provide reach physical affordances.

7.2.3 Remote Tangible Interfaces

One limitation of un-actuated TUIs is that it is hard to update their physical representation to reflect changes in computational state. Actuated TUIs and shape-changing UI solve this problem. This becomes more of an issue when considering remote collaboration using shared tangible workspaces because the states must also be linked across distances. Dynamic Physical Affordances can easily reflect changes made by a remote user, while maintaining the advantages of physical affordances.

7.2.4 Switching Between Virtual and Physical Representations

Though this thesis argues for the importance of physical affordances, there are also times when virtual representations and virtual signifiers are better suited. As our study in Chapter 4 highlighted, sometimes physical objects get in the way, and virtual objects do not. Instead of choosing one or the other at the outset, a smarter approach may be to combine and quickly switch between physical and virtual renderings, and thus physical and perceived affordances. Leithinger et al showed there can be a great benefit for changing between virtual and representations using a shape display [100].

7.3 Authoring Dynamic Physical Affordances

This thesis has explored the design space of Dynamic Physical Affordances for shape-changing and deformable interfaces, however it has not discussed the means for prototyping and designing these affordances. Through our own efforts to explore this space and prototype interactions we have devised a number of methods and tools for authoring Dynamic Affordances: Procedurally, Key Frame Animation, Puppeteering, and Building Blocks. These techniques have advantages and disadvantages at different stages of the design process.

7.3.1 Early Stage: Key Frame Animation

The underlying model that the inFORM shape display uses to display shapes is based on 8bit 2D arrays, which are similar to greyscale images. This allows greyscale movies to be played on the Shape Display easily. Greyscale movies can be generated from a variety of sources quickly, such as 3D Animation packages like 3D Studio Max, Parametric 3D modeling applications such as Rhino, or even 2D Compositing tools like After Effects. This allows motion designers to quickly prototype physical interaction without the need for programming.

7.3.2 Early Stage: Puppeteering

The 2.5D nature of Shape Displays maps very closely to the output of 2.5D depth cameras. This close relationship makes it very easy to puppeteer a shape display using a depth camera. Designers can use a variety of means to puppeteer the system, by capturing different surfaces with the shape display. We previously explored directly capturing a user's hands, deformations in a flexible sheet of paper manipulated by a user, or deformations in a flexible, elastic membrane. These different media afford different styles of puppeteering and greatly change the recorded shape. However, using these methods users can very quickly prototype a wide variety of shape change. This is especially useful for quickly prototyping of motion and can be combined with Wizard of Oz prototyping.

7.3.3 Early Stage: Building Blocks

While we have not explored a Building Block approach to designing and exploring dynamic affordances we believe it is a rich area to explore. Parkes et al. have investigated using shape-changing building blocks to prototype interactive devices [126]. We believe a rich set of building blocks with embedded shape change could be created to allow for quick prototyping of Dynamic Physical Affordances.

7.3.4 Early/Mid Stage: Procedurally

Many of the Dynamic Affordances described in this thesis were prototyped procedurally using two methods. 1) A library we created that enables designers and programmers to layout different widgets (buttons, sliders, wells, etc.) in 2.5D and supports the use of greyscale heightmap images to display shape. 2) A OpenGL rendering pipeline that allows 3D models to be rendered on the shape display. These allow users to generate touch sensitive interactive elements, which can update their position, visibility, scale, and shape using different easing functions over a given timespan. Specifically, we found the ability to morph between two greyscale images very convenient. While procedurally creating Dynamic Affordances can be more work than the other earlier stage techniques, What You See Is What You Get editors could easily be created to lower the floor of development tools. Ultimately it may not be any more difficult to create interactive applications that utilize Dynamic Affordances than other graphical user interfaces.

7.4 Technical Considerations for Enabling Future Dynamic Physical Affordances

Advances in Shape-Changing technology can enable richer Dynamic Physical Affordances. Based on the work in this thesis we have come to see a number of important patterns and limitations in current dynamic physical affordances that could be mitigated by future technologies. Below we describe what we believe to be important areas for future research in enabling technologies for shape-changing interfaces.

7.4.1 Resolution

The spatial resolution of a shape-changing interface has a great impact on the information it can render, and the affordances it can create. So far we have built 4 different resolutions of shape displays in our research group. In moving from the 1.5 inch spacing of Relief to the 0.5 inch spacing of inFORM, many new possibilities for interaction have emerged. This increase in resolution not only allows for higher resolution models to be displayed, but more importantly it allowed us to render buttons and controls for user interaction. By creating higher resolution shape displays we believe will enable even more types of interaction.

7.4.2 Degrees of Freedom

Currently, the state of the art in shape displays is limited to one degree of freedom of output and input for each pixel. Roudaut's concept of Shape Resolution [142], discussed previously in Chapter 3, is of great interest here. However, her framework for Shape Resolution mostly considers surfaces, not volumes. I believe it is worth investigating a more volumetric approach to understanding the resolution of shape displays and shape-changing user interfaces. How can we consider more volumetric layouts of shape displays that can allow for richer interaction? Moving beyond 2.5D shape output will enable new affordances to be rendered, such as grips or handles, which are essential for certain types of interaction.

7.4.3 Speed

Previously to inFORM and Relief, shape displays were often much slower due to their type of actuation, a refresh rate in seconds in the case of Shame Memory Alloys in the case of Lumen, and minutes in the case of pneumatic actuation in Xenotran's Dynamic Matrix Display. inFORM and Relief have much higher refresh rates due to the faster motors used. This higher refresh rate has enabled many new interaction techniques and applications, and is perhaps more important than the increase in resolution.

What could faster shape-changing interfaces enable? One area is better haptic rendering, because our sense of touch has a much faster response rate than our visual system, with which we can get away with 15, 30, or 60hz. For true haptic rendering we need much higher refresh rates in the khz. With a faster refresh rate shape displays can simulate different material properties. Another avenue of research that could be enabled by higher refresh rates is the combination of shape displays with volumetric displays; a volumetric display enabled by persistence of vision, and LEDs mounted on the pins of a shape display, could create floating 3D graphics if each pin moved fast enough. However, this volumetric display could be 'frozen' so that users could touch and manipulate the shell of the volume (i.e. like a normal shape display).

To accomplish this shape displays can be made faster. Recent developments in optimization techniques have been shown to improve the refresh rate of pin arrays which use matrixed pneumatic actuation [186]. In other cases faster actuators can be used, such as linear motors.

The Jamming interfaces described in this thesis take longer to change state, and must be improved. Further work on hydraulic jamming may allow for much smaller and faster state transitions. That coupled with the use of layer jamming [121], where less fluid needs to displaced, could make for even faster state transitions.

7.4.4 Scale

The scale at which a shape display operates has a great effect on the interaction possibilities. The scale of a shape-changing interface is different than its resolution, and considers both the overall size of the interface, as well as the displacement or travel of the display. Large scale shape change has been prototyped on the building level, primarily for dynamic facades for advertising[50], but also for controlling the amount of light or air circulation through arrays of shades or vents. Micro-scale shape change may provide texture display, which can be useful for affording different types of grips, or cluing users into what should be touched, or increasing friction while touching some areas. In the future, it may be important to consider shape-displays that can change scales.

7.4.5 Sensing

Just as important as high-resolution, multi-dimensional shape change is the ability to sense a user's input in high resolution, and in the case of deformable user interfaces, the current shape of the device. Multitouch interfaces which are now standard can often only detect 10 points of contact, and often give little information about other features such as touch size. Now force sensing multitouch screens are being sold, however they have yet to gain wide use (although the coming Apple Watch may change this). But rich understanding of grasp, deformation, as well as touch has the power to provide for richer interaction, as well as more context dependent applications. True digital clay that can be easily deformed and report its high-resolution 3D shape as well as the forces (normal and tangential) acting on it, has yet to be developed even in the laboratory. New advances in shape sensing, tangential force input, and embedded capacitive gesture sensing will be as important to making shape-changing interfaces a reality as new types of actuation.

7.4.6 Force

The force that the shape-changing interface can apply is important not only for providing haptic feedback, but also for manipulating objects on its surface as shown in Chapter 4. By increasing the force, or more importantly the pressure (force over area), of a shape display can enable it to lift heavier objects, and appropriate them as part of the interface.

7.4.7 Compliance and Variable Stiffness

If the actuators in shape displays become faster and stronger it may become dangerous to interact with if there is an error in the software. Compliance is important when considering the safety of shape-changing interfaces. Pneumatic actuation is one area to explore here, as well as more complex controls, but much of the work on compliance and safety in robotics can be applied to shape-changing user interfaces.

7.4.8 Power Consumption

In order to make shape-changing interfaces function in mobile devices, we must rely heavily on user powered actuation, or have very efficient power consumption for actuation. New types of actuation techniques should be investigated as well as hybrid approaches which use both the user's power and computational control, building off of work in modular robotics. Wireless power may ultimately solve this problem, but this remains a technical challenge and an open research area.

7.5 Looking Forward: Roadblocks and Opportunities on the Road Towards Programmable Matter Interfaces

Today computation is truly everywhere and ubiquitous. We envision a future in which not only computation is cheap, but power and actuation as well. Advances in wireless power, battery technology, and MEMS actuators point to such a direction - and the robotics field is eager to see this become a reality. Robots will be everywhere soon, not in the form of anthropomorphic bipeds, instead invisible and as embedded as the computational power of today. Interactive devices will increasingly contain actuators, moving beyond the single vibration motor in mobile phones today. Already we are beginning to see this trend towards richer haptic display in mobile devices such as the Tactus haptic button displays [23] or Apple's Taptic Engine in the Apple Watch. Shape-changing interfaces have the potential to revolutionize the ways we can interact with digital information. This thesis has described new methods for interacting and understanding these types of interfaces, yet many questions remain and many questions are posed by the possibility of more ubiquitous shape-changing interfaces.

7.5.1 Understanding Affordances of Uncertain Objects

While Dynamic Physical Affordances and Improvised Physical Affordances can allow a single interface to have physical affordances that can adapt to different use scenarios, our prior research suggests that there may be a variety of issues that they cause. Firstly, Shape-changing interfaces can change shape very quickly, and the user may not be clear about how their actions will cause shape change. This can be jarring to users. Secondly, though the affordances of the current state are often made more clear by the use of Dynamic Physical affordances, because these affordances can change, the user may be unclear about what other affordances are hidden and how to uncover them. Thirdly, there is a lack of persistency with Dynamic Physical Affordances; they may change at any moment and be different from when a user last attended to it.

Addressing These Issues: Physical Feed Forward

Physical Feedforward maybe a means to address these issues. Feedforward informs the user what the result of a given action will be before preforming it [173]. Feedforward has yet to be explored in the context of Shape-Changing and Deformable UI. Different methods could be used to trigger the Feedforward, such as proximity to a signifier, or light touching of a signifier. In addition, a variety of different techniques to provide Physical Feedforward could be explored: animated shape rendering using motion cues that varies over time and space, sparse shape rendering previews, and graphical feedback.

Physical Feedforward could be used to give users a preview of the state/shape changes caused by a certain action. For example, imagine three buttons on a shape-changing interface; as the user approaches one of those buttons and hovers over it, the shape-changing interface could quickly transform between its current state and the one that would be caused by that button.

Feedforward has been used to guide users for pen gestures [7], 2D multitouch interactions [44], and 3D free hand gestures [158]. This research could be extended for uncovering hidden physical affordances and to guide users to uncover shapes for User-

Defined Affordances. For example, based on the ShapePhone prototype described in a Chapter 6, how could the system help guide the user to different possible shapes such as a phone or a game controller?

7.5.2 Adaptive Furniture

The world we live in is a dynamic place. People are in motion constantly, moving from activity to activity. Information changes state ever quicker, as streams of information flow in the cloud from news sources and sensors. Ubiquitous computing is here in our pockets and on our walls, allowing this information to be displayed anywhere, any time. And yet, we still find our interfaces to be separate from the physical world around us. We can interact everywhere, yet the richness of these interactions is limited to the devices we have on hand, and even more importantly the furniture and built environment with which we live. We envision a future in which information and interaction is everywhere, but it not only blends into the world around us, but also can reach out. We believe shape-changing interfaces can alter the way we work and live, and they can be integrated into the furniture around us.

Of course furniture has a long tradition of adaptability and shape change - a lazy boy reclines, a fold out bed triples in size, and a standing desk rises for a different task. These examples of furniture mix self-actuation through springs with user supplied actuation. We believe a new breed of devices can blur the boundary between traditional computer interfaces and task furniture. Instead of putting your computer on your desk, your desk is the computer when you need it, and an ambient display when you do not. It can change shape to support different tasks or collaboration and move objects on its surface to organize or notify.

But as these shape-changing interfaces begin to occupy the space of furniture it becomes even more important to understand how users will perceive them. Much work in HRI has explored perceptions of robots - however, the furniture we imagine is less of an agent and more of an appliance. Some researchers in Robotics have begun to explore this concept, such as the Room Bots project [159].

Shape-changing furniture may be able to support a wide variety of activities.

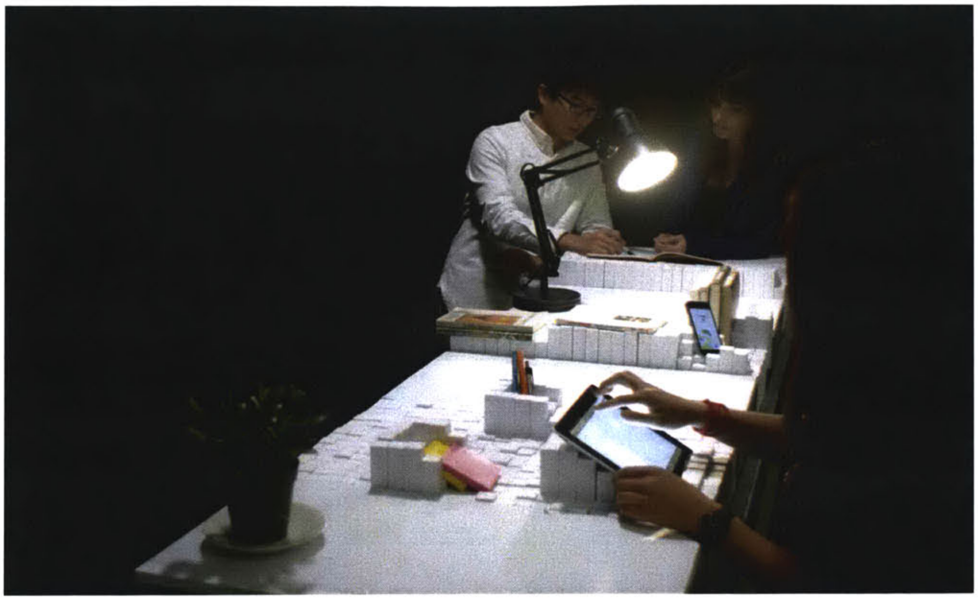


Figure 7-5: Transform, a shape-changing table, changes its shape to support different activities.

For example, it can create geometry and surfaces to provide interaction at different heights, or tilt up to provide more privacy. A desk could rearrange its contents to better support an activity, just as we have changing tool pallets for different modes in Microsoft Word. These interactions could be contextual - a user picks up a pen, and the surface changes to drafting table. Or the surface can be used to create different emotional patterns to set mood for different types of work, similar to how a user may change the lighting to match a task.

We believe shape-changing furniture can be used as an ambient display to convey information to users. These ambient display can be centered around objects on the table, and move to follow them. The shape-changing table can move objects, similarly to Chapter 4, and animate them to convey information. For example keys left on the table would be physically shaken when a user walks by to remind them.

We have begun to explore these interactions with Transform, a shape-changing table. The Transform hardware is based on a modular shape display platform. It utilizes 3 sets of shape display arrays with resolutions of 16 by 24 actuators, separated by static flat surfaces. This design exploration allows us to envision and prototype new ways of interacting with shape-changing interfaces.

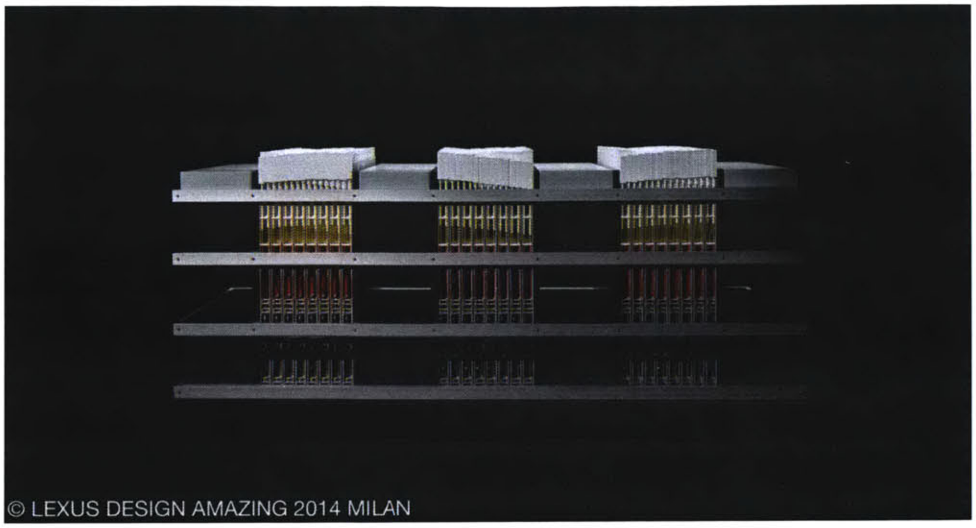


Figure 7-6: Transform a shape-changing table, as installed in Milan Design Week. Image courtesy Lexus.

The Transform system was installed in a design gallery for a period of five days where over 5000 visitors interacted with it. Our goal was to better understand how users would come to understand and interact with shape-changing furniture. This installation focused on using shape-changing furniture to convey information and notification.

During the installation, the Transform system was used to display different emotional motifs and convey story and information through pre-programmed animations. The system also had an interactive mode meant to draw visitors closer and create a jovial playful experience. This by no means was how shape-changing furniture would be installed in the home or the workplace, however our belief is that from these early installations much can be gained from observing interaction. There were some interesting patterns of use that emerged, but much of the lessons come in the form of new questions to help us understand how users consider shape change in different environments.

7.5.3 New Geometries: Edges, Chains, Snakes, Crusts and Swarms

As we push forward, away from considering shape change only in the context of desktop or mobile computing, but rather into our environments and homes, we need to consider new geometries and classes of shape-changing interfaces. Currently much of the research in shape-changing interfaces has been inspired by research in haptics and robotics. Researchers in Modular and Soft Robotics have been pushing boundaries in shape change. How can design and HCI lead the way in new types and classes of programmable robotic interfaces? Our belief is that the goals for shape change in robotics are fundamentally quite different than in HCI, and our hope is that research into user needs and new forms of interaction will drive advances in shape change technology. Geometries such as swarms, chains or snakes have their history in the field of Robotics. What new geometries will emerge driven by HCI researchers?

Edge Displays

While shape displays have primarily investigated creating 2.5D surfaces there may be a benefit to investigating linear arrangements that create 2D profiles. These 'Haptic Edge Displays', could be miniature shape displays around the edge of a traditional mobile device, which can allow for both haptic feedback as well as expressive input utilizing a non-dominant hand, see Figure 7-8. This allows for passive haptic exploration on the part of the user, in addition to active haptic output found in many current haptic interfaces. The Haptic Edge Display can work alone as a display for haptic notification or with a graphical user interface to augment interaction and provide haptic feedback.

Edge Displays can be used for a variety of application scenarios to provide: Dynamic Affordances (buttons and controls), Haptic Awareness for notification, Interpersonal Communication, and expressive haptic output for Gaming. Buttons and sliders can be rendered on the edge of the display to map to different functions and dynamic reconfigure when changing applications. Bi-stable buttons, such as radio

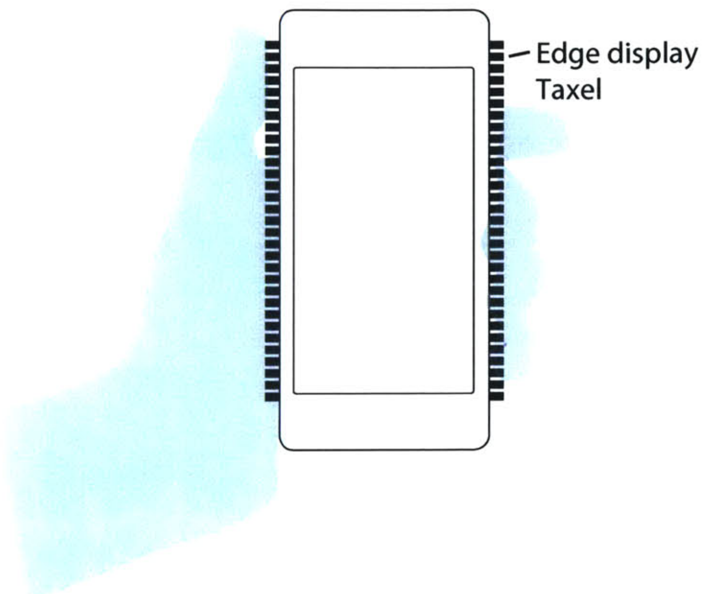


Figure 7-7: A sketch of a Haptic Edge Display for mobile devices. The edge of the display is made of small linear actuators.

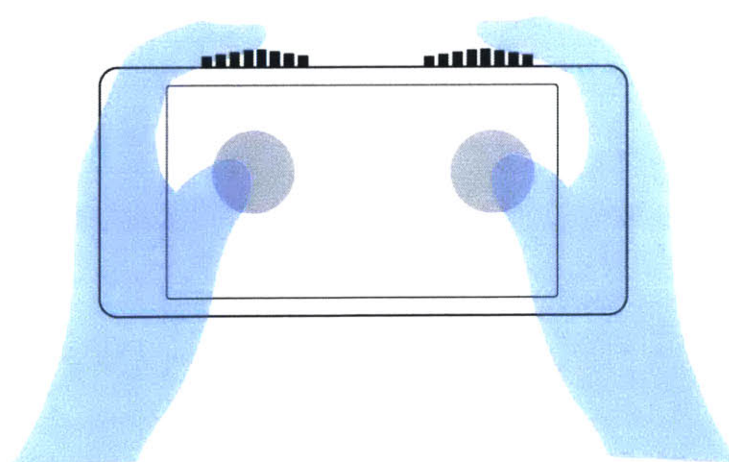


Figure 7-8: A sketch of a Haptic Edge Display used to play a game, where buttons are created by the Edge Display.

buttons, can be emulated with the Haptic Edge Display. Buttons can also have haptic feedback through vibration and detents.

These Dynamic Physical Affordances can be used to change the affordances for different applications. For example, when a user selects a certain game, shoulder buttons can be rendered on the Edge Display, allowing for more expressive control. However, when the user quits the game the buttons disappear. Another example would be for camera control: when a user is in camera mode a physical button is always rendered in the top right corner regardless of the orientation of the device.

Cord Displays

Although we live in an increasingly wireless world, we find ourselves surrounded by cables - cables for charging, cables for headphones, and cables for network connections. Other researchers have begun to explore how these could be used for input [149]. How could we appropriate these long surfaces for output and information display enabled by shape change? By adapting actuation techniques from snake and chain robotics [188, 133, 21], we could create long thin cables that could articulate 3D shapes to provide richer possibilities for interaction. For example, head phone cords could form different shapes to give users notifications or directions.

Swarm Displays

Swarm robots have long been investigated in robotics [36, 35], and draw from the elegant behavior of swarms found in nature, where simple individual elements, such as ants, can create complex behavior by working together from simple rules [33, 12]. Swarm robots have been preliminarily investigated in the context of HCI, being used both for interaction [89] and display [2], but it suggests many interesting new interactions and open research questions. Swarm-bots could combine to form complex shapes that afford different types of interaction. Currently, most swarm-bots are used on flat surfaces, but how would that change if we combine them with a shape display?

Slow Shape Displays

Much of the focus of this thesis is on creating interactive shape-changing interfaces, that can change shape very quickly. Arthur Ganson's work 'Machine with Concrete,' which has a motor attached to a long series of gears that will take over a trillion of years turn a block of concrete, points towards a different type of shape display, one that considers time on a different scale: generations. Long ago cathedrals were erected over the course of many generations, and the mason working on it at any given point was unlikely to see it finally constructed. We see information displayed through shape and form in nature at a slower pace: on a day long scale (a flower opening and closing), over the course of weeks (a seedling growing), on a year long time scale (blooming buds, flowering leafs, falls bright colors, and finally winter's bare trees). Taking inspiration from these settings, how can shape and form be used in more ambient interfaces, or even new interfaces we have never considered? The clock of the Long-Now is an example of what a new type of "Slow Shape Display" could look like, to encourage the public to think on a longer time scale [13].

Chapter 8

Conclusion

This thesis has explored the design space of Dynamic and Improvised Physical Affordances. With the introduction of Shape Changing and Deformable User Interfaces the tactile, physical properties of interfaces can now change state with the ease only previously afforded to visual content on screens. With these radically new interfaces, our understanding of affordances must be broadened to consider how we can leverage dynamic form to convey use and how dynamic form can be adapted for different physical affordances. This thesis takes a first step in that direction, and points to a space where general purpose computing tasks can easily be accomplished with the aid of rich physical affordances. An emerging trend in HCI has been towards gestural or 'natural' interaction, foregoing physical affordances and embracing 'invisible interfaces.' My hope is that this thesis is just one small stepping stone on a long journey towards a more embodied and physical style of interaction. This thesis presents a possible future where tangible interaction can clearly be weaved into domains that require general purpose interaction, providing great advantages in terms of expression and richness of input.

8.1 Restatement of Thesis Contributions

1. Techniques for providing Dynamic Physical Affordances through shape change.
   (a) An exploration of the design space of dynamic affordances and constraints.

   (b) Explorations in the use of motion and animation for physical affordances.
   (c) State-of-the-art system for fast, real-time 2.5D shape actuation, co-located projected graphics, object tracking, and direct manipulation.
   (d) Three applications that demonstrate the potential of these interaction techniques for HCI.
   (e) An evaluation of the performance of dynamic physical affordances.
   (f) An evaluation of the perceptual qualities of motion in shape change for physical affordances.

2. An Investigation of User Appropriated Physical Affordances.
   (a) An exploration of the design space of User Appropriated Physical Affordances.
   (b) A novel hardware system, deFORM, to support User Appropriated Physical Affordances through a real-time 2.5d deformable surface interface that uses infrared (IR) structured light 3D scanning and projected visual feedback.
   (c) Techniques for tracking arbitrary and tagged tangible tools (phicons), touch and hand gestures.
   (d) A number of application prototypes that make use of User Appropriated Physical Affordances.
   (e) A study exploring the use of User Appropriated Physical Affordances to support 3D modeling for children.

3. Techniques for supporting User-Defined Physical Affordances through direct deformation.
   (a) An exploration of the design space of User-Defined Physical Affordances.
   (b) Applying particle jamming for use as a variable stiffness material to enable User-Defined Physical Affordances.
   (c) A review of the state of the art in jamming from an HCI perspective.

(d) A novel hydraulic-based jamming technology, for rapid activation, silent actuation, and embedded optical sensing.

(e) Two techniques for high-resolution, integrated and embedded sensing for jamming interfaces: optical sensing, using index-matched fluids and particles; and electrical sensing, using capacitive and electric field sensing.

(f) A small, low-power jamming system for mobile and embedded organic user interfaces.

(g) Motivating prototypes to highlight how jamming can be applied to HCI, and particularly User-Defined Affordances.

Appendix A

inFORM Motor Control Boards

The motor control boards for inFORM were designed together with Daniel Leithinger and Ryan Wistor. The schematic diagram follows.

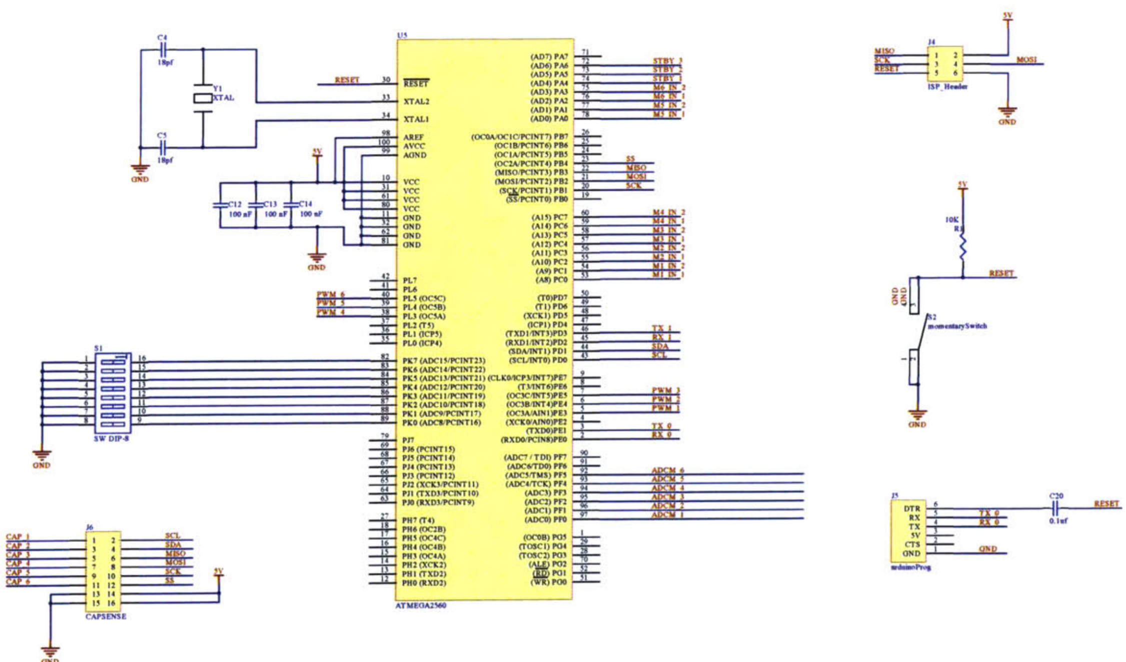


Figure A-1: inFORM Motor Control Board PCB Schematic.

A.1 Parts List

The following parts were used in the Motor Control Boards for inFORM:

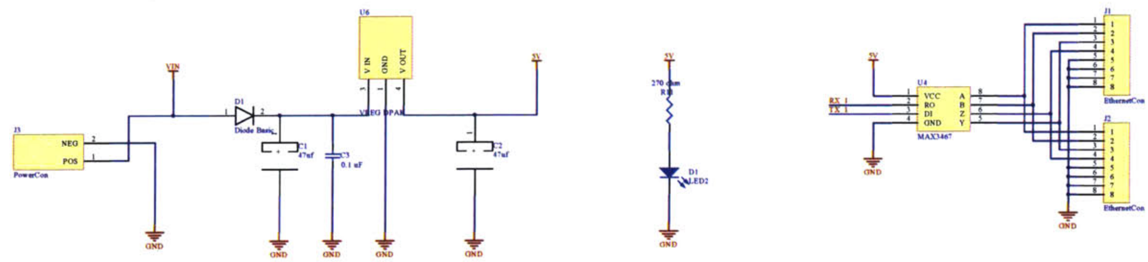


Figure A-2: inFORM Motor Control Board PCB Schematic.

Type	Description	Digikey Part Number	Number Needed	Board ID
MCU	ATMEGA2560	ATMEGA2560-16AU-ND	1	U5
Motor Power	Toshiba DRIVER DUAL DC MOTOR	TB6612FNGCT-ND	3	U1,U2,U3
MCU	Crystal HC-49USX 16 MHz	X1103-ND	1	Y1
Com	Max rs485 com chip MAX3467	MAX3467CSA+-ND	1	U4
Com	Ethernet connector	"A31444-ND "	2	J1,J2
Power	Power Connector	CP-037A-ND	1	J3
Switch	Reset Switch - basic push button	SW402-ND	1	S2
Power	DPACK 5v power reg	MC33269DT-5.0GOS-ND	1	U6
Cap	power cap 47 uF	718-1363-1-ND	2	C1,C2
Cap	general cap 1206 dim 0.1uf or 100nf	445-4008-1-ND	10	C3,C9-14
Cap	Crystal Cap 20 pf 1206 package	311-1153-1-ND	2	C4, C5
Switch	Dip switch for address	CT1938MS-ND	1	S1
Resistor	10k res pull up/down	P10KECT-ND	10	R1
Connector	ISP 6 pin Connector	609-3234-ND	1	J4
Connector	6 pin arduino prog header	"3M9471-ND "	1	J5
Power	General Diode for power	641-1018-1-ND	1	D1
Cap	electro cap 10 uF can	PCE3878CT-ND	3	C6, C7, C8
LED	Power LED	160-1456-1-ND	1	D1
Resistor	270 ohm for led	P270ECT-ND	1	R11
Diode	Motor kickback diode	MBRA140TRPBFCT-ND	24	D2-25

Table A.1: Parts list for inFORM Motor Control Boards

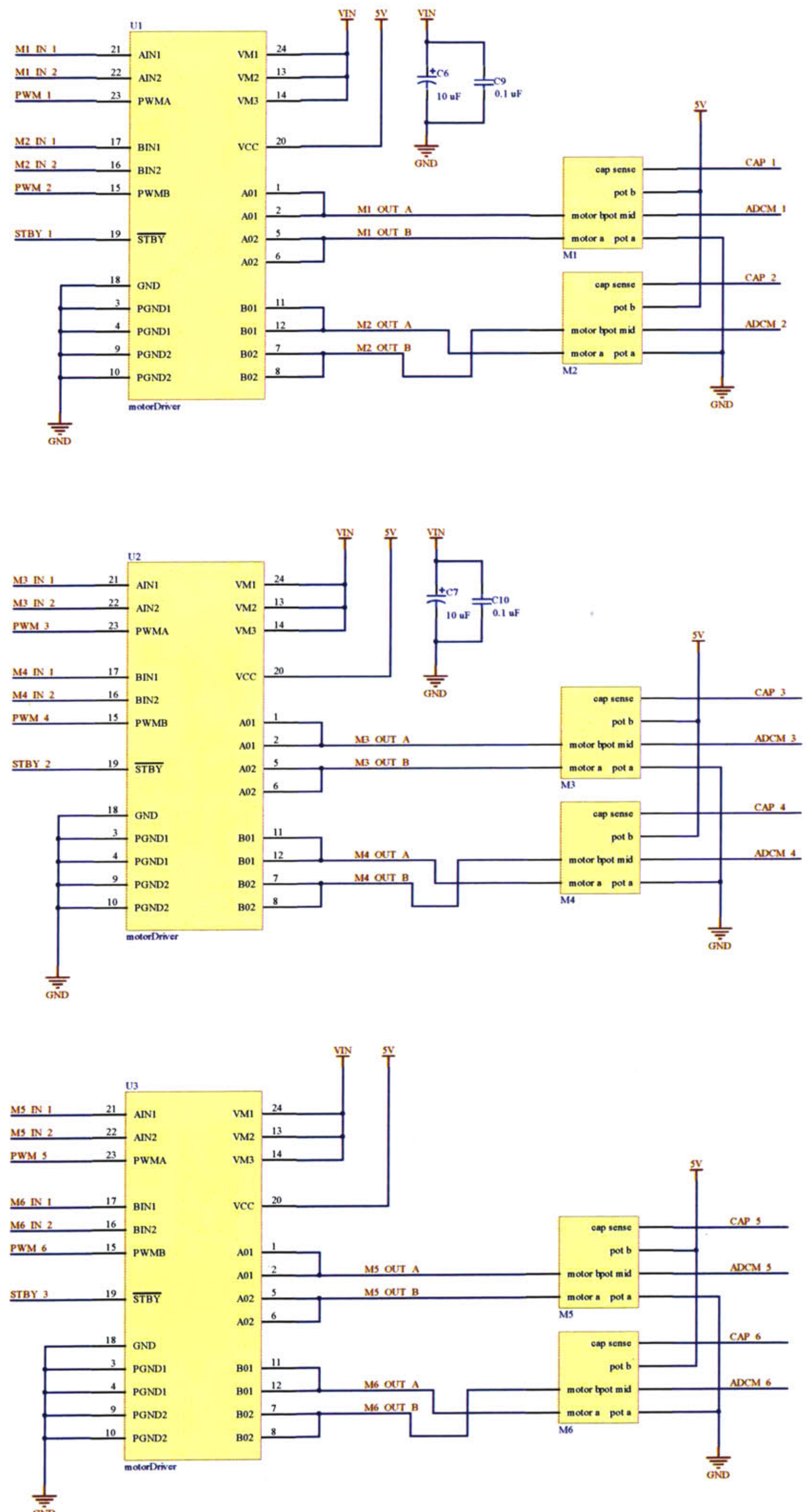


Figure A-3: inFORM Motor Control Board PCB Schematic.

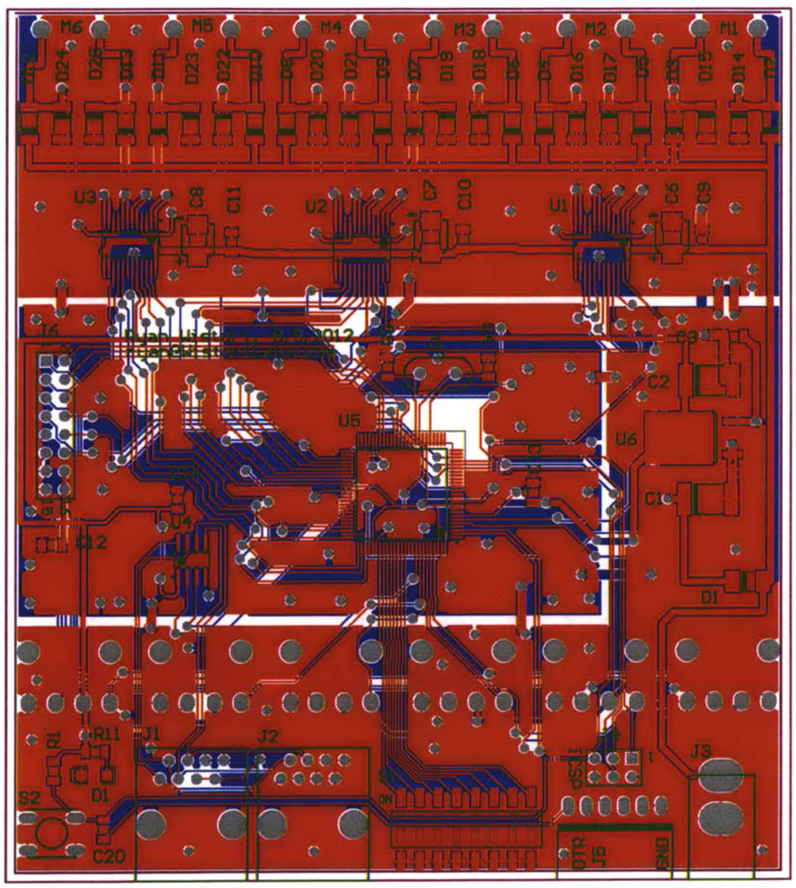


Figure A-4: inFORM Motor Control Board PCB Layout.

Appendix B

Motion Study

The following questions were asked of online participants in the Motion Study described in Chapter 4. Each participant viewed a pair of animations depicting shape change and were asked to select one of the videos in response to a single question from this list:

which video contains bigger shapes
which video changes shape more
which video has faster movement
which video has more sudden movements
which video is more energetic
which video contains more random motion
which video contains more organic motion
which video contains more mechanical motion
which video contains more rhythmic motion
which video contains more horizontal movement
which video contains more vertical movement
which video contains motion that is more circular

Appendix C

Code for Capacitive Shape Sensing

Listing C.1: Code for Capacitive Shape Sensing

```
/*
Code for measuring the shape of a jamming volume using capacitance. This code assumes 9 rows of flexible reciever electrodes, and 9 columns of copper transmitter electrodes on the back of the jamming volume.
The 9 flexible electrodes reciever electrodes are attached to a multiplexer, then to an amplifing circuit, and finaly to the analog input pin of a microcontroller. This code was written for the Maple ARM board.
*/
#include <stdio.h>

int rxSensorPin = 15;    // Input pin from multiplexer and amplifier
int s0Pin =0; //These pins are used to control the multiplexer
int s1Pin =1;
int s2Pin =2;
int s3Pin =3;
const int N_SAMPLES = 32; //The number of samples to average over
```

```
    for each measure of capacitance

int sensorValues[81]; //This assumes a 9 X 9 grid, thus 81 values
int currentSensor=0;


void setup() {
    // Declare the sensorPin as INPUT_ANALOG:
    pinMode(rxSensorPin, INPUT_ANALOG);


    //Set all of the output pins to output mode
    int outPutPin;
    for(outPutPin=0; outPutPin <15; outPutPin++) {

      pinMode(outPutPin, OUTPUT);
    }

    for(outPutPin=27; outPutPin <37; outPutPin++) {

      pinMode(outPutPin, OUTPUT);
    }

    // Sets the Multiplexer to the first position
    digitalWrite(s0Pin, LOW);
    digitalWrite(s1Pin, LOW);
    digitalWrite(s2Pin, LOW);
    digitalWrite(s3Pin, LOW);
}

// This function measures the capacitance for an entire row, one
    electrode at a time.
```

```
// It measure capacitance values for all 9 output electrodes from a single input electrode.
void readFromRow()
{

  int pinNum;
  for( pinNum = 27; pinNum < 36; pinNum++) //all of the output electrodes
  {
    int i;
    int accumulator = 0;
    int value;

    for ( i = 0; i < N_SAMPLES; i++)
    {
      digitalWrite(pinNum, HIGH);
      value = analogRead(rxSensorPin) - 2048;
      accumulator += value;

      digitalWrite(pinNum, LOW);
      value = analogRead(rxSensorPin) - 2048;
      accumulator -= value;
    }

    sensorValues[currentSensor] = accumulator; // record sum of all samples for a given electrode pair, can divide later
    currentSensor++;

  }

}

```

```
void loop() {

        // go through each receiver electrode, and measure
            capacitance from all transceiver electrodes

    //antena 0
    digitalWrite(s0Pin, LOW);
    digitalWrite(s1Pin, LOW);
    digitalWrite(s2Pin, LOW);
    digitalWrite(s3Pin, LOW);

    readFromRow();


    //antena 1
    digitalWrite(s0Pin, HIGH);
    digitalWrite(s1Pin, LOW);
    digitalWrite(s2Pin, LOW);
    digitalWrite(s3Pin, LOW);

    readFromRow();

    //antena 2
    digitalWrite(s0Pin, LOW);
    digitalWrite(s1Pin, HIGH);
    digitalWrite(s2Pin, LOW);
    digitalWrite(s3Pin, LOW);

    readFromRow();

    //antena 3
    digitalWrite(s0Pin, HIGH);
```

```
    digitalWrite(s1Pin, HIGH);
    digitalWrite(s2Pin, LOW);
    digitalWrite(s3Pin, LOW);

    readFromRow();

    //antena 4
    digitalWrite(s0Pin, LOW);
    digitalWrite(s1Pin, LOW);
    digitalWrite(s2Pin, HIGH);
    digitalWrite(s3Pin, LOW);

    readFromRow();

    //antena 5 - jumps over to c8
    digitalWrite(s0Pin, LOW);
    digitalWrite(s1Pin, LOW);
    digitalWrite(s2Pin, LOW);
    digitalWrite(s3Pin, HIGH);

    readFromRow();

    //antena 6 - c9
    digitalWrite(s0Pin, HIGH);
    digitalWrite(s1Pin, LOW);
    digitalWrite(s2Pin, LOW);
    digitalWrite(s3Pin, HIGH);

    readFromRow();

    //antena 7 - c10
    digitalWrite(s0Pin, LOW);
```

```
    digitalWrite(s1Pin, HIGH);
    digitalWrite(s2Pin, LOW);
    digitalWrite(s3Pin, HIGH);

    readFromRow();

     //antena 8  - c11
    digitalWrite(s0Pin, HIGH);
    digitalWrite(s1Pin, HIGH);
    digitalWrite(s2Pin, LOW);
    digitalWrite(s3Pin, HIGH);

    readFromRow();



        //Sends all of the data back over USB, in 3 64byte packets

    char tempBuffer[64];
    int currentIndex =0;
    int j;

    tempBuffer[0]=0;
    tempBuffer[1]=0;
    for(j =1; j <32; j++){
        uint16 tempVal = (sensorValues[currentIndex]/2);
        tempBuffer[j*2] = ((char *)(&tempVal))[0];
        tempBuffer[j*2 +1] = ((char *)(&tempVal))[1];
        currentIndex++;
    }
    SerialUSB.write(&tempBuffer, 64);

    for(j =0; j <32; j++){
```

```
            uint16 tempVal = (sensorValues[currentIndex]/2);
            tempBuffer[j*2] = ((char *)(&tempVal))[0];
            tempBuffer[j*2 +1] = ((char *)(&tempVal))[1];
            currentIndex++;
        }
        SerialUSB.write(&tempBuffer, 64);

        for(j =0; j <32; j++){
            if(currentIndex < 81){
              uint16 tempVal = (sensorValues[currentIndex]/2);
              tempBuffer[j*2] = ((char *)(&tempVal))[0];
              tempBuffer[j*2 +1] = ((char *)(&tempVal))[1];
            }
            else{
               tempBuffer[j*2] = 5;
               tempBuffer[j*2 +1] = 5;
            }
            currentIndex++;
        }
        SerialUSB.write(&tempBuffer, 64);



        currentSensor=0;



}

char * int2str( int num )
{
  static char retnum[21];        // enough for 20 digits plus NUL
     from a 64-bit uint
  sprintf( retnum, "%d", num );
```

```
    return retnum;
}
```

Bibliography

[1] Maneesh Agrawala, Andrew C Beers, Ian McDowall, Bernd Fröhlich, Mark Bolas, and Pat Hanrahan. The two-user responsive workbench: support for collaboration through individual views of a shared space. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 327–332. ACM Press/Addison-Wesley Publishing Co., 1997.

[2] Javier Alonso-Mora, Andreas Breitenmoser, Martin Rufli, Roland Siegwart, and Paul Beardsley. Multi-robot system for artistic pattern formation. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 4512–4517. IEEE, 2011.

[3] John R Amend, Eric M Brown, Nicholas Rodenberg, Heinrich M Jaeger, and Hod Lipson. A positive pressure universal gripper based on the jamming of granular material. *Robotics, IEEE Transactions on*, 28(2):341–350, 2012.

[4] Mahoro Anabuki and Hiroshi Ishii. AR-Jig: A Handheld Tangible User Interface for Modification of 3D Digital Form via 2D Physical Curve. In *Symposium on Mixed and Augmented Reality*, pages 1–10, 2007.

[5] S. Audet and M. Okutomi. A user-friendly method to geometrically calibrate projector-camera systems. In *IEEE CVPR 2009*, pages 47–54, 2009.

[6] Christoph Bartneck, Dana Kulić, Elizabeth Croft, and Susana Zoghbi. Measurement instruments for the anthropomorphism, animacy, likeability, perceived intelligence, and perceived safety of robots. *International journal of social robotics*, 1(1):71–81, 2009.

[7] Olivier Bau and Wendy E. Mackay. Octopocus: A dynamic guide for learning gesture-based command sets. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology*, UIST '08, pages 37–46, New York, NY, USA, 2008. ACM.

[8] Olivier Bau, Uros Petrevski, and Wendy Mackay. BubbleWrap: a textile-based electromagnetic haptic display. In *CHI EA '09*, page 3607. ACM Press, April 2009.

[9] Hrvoje Benko. Beyond flat surface computing. In *ACM MM '09*, page 935, New York, New York, USA, October 2009. ACM Press.

[10] Eric A. Bier, Maureen C. Stone, Ken Pier, William Buxton, and Tony D. DeRose. Toolglass and magic lenses: the see-through interface. In *SIGGRAPH '93*, pages 73–80. ACM, 1993.

[11] Sarah-Jayne Blakemore, Daniel Wolpert, and Chris Frith. Why can't you tickle yourself? *Neuroreport*, 11(11):R11–R16, 2000.

[12] Eric Bonabeau, Marco Dorigo, and Guy Theraulaz. *Swarm intelligence: from natural to artificial systems*. Number 1. Oxford university press, 1999.

[13] Stewart Brand. *Clock of the Long Now: Time and Responsibility*. Basic Books, 1999.

[14] E. Brown, N. Rodenberg, J. Amend, A. Mozeika, E. Steltz, M. R. Zakin, H. Lipson, and H. M. Jaeger. Universal robotic gripper based on the jamming of granular material. *Proceedings of the National Academy of Sciences*, 107(44):18809–18814, October 2010.

[15] Bill Buxton. Multi-touch systems that i have known and loved. *BillBuxton.com*, 2014. http://www.billbuxton.com/multitouchOverview.html. Accessed: 2014-12-24.

[16] Erin A. Carroll, Celine Latulipe, Richard Fung, and Michael Terry. Creativity factor evaluation: towards a standardized survey metric for creativity support. In *Proceeding of the seventh ACM conference on Creativity and cognition*, C&C '09, pages 127–136, New York, NY, USA, 2009. ACM.

[17] Alvaro Cassinelli and Masatoshi Ishikawa. Khronos projector. In *ACM SIGGRAPH 2005 Emerging technologies on - SIGGRAPH '05*, SIGGRAPH '05, page 10, New York, New York, USA, 2005. ACM Press.

[18] Alvaro Cassinelli and Masatoshi Ishikawa. Khronos projector. In *ACM SIGGRAPH 2005 Emerging Technologies*, SIGGRAPH '05, page 10, New York, New York, USA, 2005. ACM Press.

[19] M. Cates, J. Wittmer, J.-P. Bouchaud, and P. Claudin. Jamming, Force Chains, and Fragile Matter. *Physical Review Letters*, 81(9):1841–1844, August 1998.

[20] L.W. Chan, H.T. Wu, H.S. Kao, J.C. Ko, H.R. Lin, M.Y. Chen, Jane Hsu, and Y.P. Hung. Enabling beyond-surface interactions for interactive surface with an invisible projection. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*, pages 263–272. ACM, 2010.

[21] Nadia Cheng, Genya Ishigami, Stephan Hawthorne, Hao Chen, Malik Hansen, Maria Telleria, Robert Playter, and Karl Iagnemma. Design and Analysis of a Soft Mobile Robot Composed of Multiple Thermally Activated Joints Driven by a Single Actuator. *IEEE ICRA*, 2012.

[22] Keywon Chung. *OnObject: programming of physical objects for gestural interaction.* PhD thesis, MIT, 2010.

[23] Craig Michae Ciesla and Micah B. Yairi. User interface system and method, 07 2009.

[24] Andy Clark. *Being There: Putting Brain, Body, and World Together Again.* MIT Press, 1997.

[25] Manfred Clynes. Generalised emotion how it may be produced, and sentic cycle therapy. In *Emotions and psychopathology*, pages 107–170. Springer, 1988.

[26] Manfred Clynes and Yehudi Menuhin. *Sentics: The touch of emotions.* Anchor Press Garden City, NY, 1977.

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

[28] Gillian Crampton Smith. The hand that rocks the cradle. *ID magazine*, pages 60–65, 1995.

[29] Carolina Cruz-Neira, Daniel J Sandin, and Thomas A DeFanti. Surround-screen projection-based virtual reality: the design and implementation of the cave. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pages 135–142. ACM, 1993.

[30] Mihaly Csikszentmihalyi and Eugene Halton. *The meaning of things: Domestic symbols and the self.* Cambridge University Press, 1981.

[31] Mark Cutkosky and R. Howe. Human grasp choice and robot grasp analysis. *Dexterous Robot Hands*, 1990.

[32] Frédéric Dehais, Emrah Akin Sisbot, Rachid Alami, and Mickaël Causse. Physiological and subjective evaluation of a human–robot object hand-over task. *Applied ergonomics*, 42(6):785–791, 2011.

[33] Marco Dorigo. *Ant Colony Optimization and Swarm Intelligence: 5th International Workshop, ANTS 2006, Brussels, Belgium, September 4-7, 2006, Proceedings*, volume 4150. Springer, 2006.

[34] Paul Dourish. *Where the action is: the foundations of embodied interaction.* MIT press, 2004.

[35] F. Ducatelle, G.A. Di Caro, C. Pinciroli, and L. Gambardella. Self-organized cooperation between robotic swarms. *Swarm Intelligence Journal*, 5(2):73–96, 2011.

[36] Gregory Dudek, Michael Jenkin, Evangelos Milios, and David Wilkes. A taxonomy for swarm robots. In *Intelligent Robots and Systems' 93, IROS'93. Proceedings of the 1993 IEEE/RSJ International Conference on*, volume 1, pages 441–447. IEEE, 1993.

[37] Kenneth P. Fishkin. A taxonomy for and analysis of tangible interfaces. *Personal Ubiquitous Comput.*, 8:347–358, September 2004.

[38] George W. Fitzmaurice, Hiroshi Ishii, and William A. S. Buxton. Bricks: laying the foundations for graspable user interfaces. In *CHI '95*, pages 442–449. ACM Press/Addison-Wesley Publishing Co., 1995.

[39] Sean Follmer and Hiroshi Ishii. Kidcad: Digitally remixing toys through tangible tools. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 2401–2410, New York, NY, USA, 2012. ACM.

[40] Sean Follmer, Micah Johnson, Edward Adelson, and Hiroshi Ishii. deForm: an interactive malleable surface for capturing 2.5D arbitrary objects, tools and touch. In *UIST '11*. ACM Press, October 2011.

[41] Sean Follmer, Daniel Leithinger, Alex Olwal, Nadia Cheng, and Hiroshi Ishii. Jamming user interfaces: programmable particle stiffness and sensing for malleable and shape-changing devices. In *UIST '12*, pages 519–528. ACM, 2012.

[42] 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, pages 417–426, New York, NY, USA, 2013. ACM.

[43] John Frazer. *An Evolutionary Architecture (Themes)*. Architectural Association Publications, 1995.

[44] Dustin Freeman, Hrvoje Benko, Meredith Ringel Morris, and Daniel Wigdor. Shadowguides: Visualizations for in-situ learning of multi-touch and whole-hand gestures. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*, ITS '09, pages 165–172, New York, NY, USA, 2009. ACM.

[45] William W. Gaver. Technology affordances. In *CHI '91*, pages 79–84. ACM, 1991.

[46] James Jerome Gibson. *The Senses Considered as Perceptual Systems*. Houghton Mifflin Co., 1966.

[47] James Jerome Gibson. *The ecological approach to visual perception*. Psychology Press, 1986.

[48] Susan Goldin-Meadow, Howard Nusbaum, Spencer D Kelly, and Susan Wagner. Explaining math: Gesturing lightens the load. *Psychological Science*, 12(6):516–522, 2001.

[49] Nan-Wei Gong, Jürgen Steimle, Simon Olberding, Steve Hodges, Nicholas Edward Gillian, Yoshihiro Kawahara, and Joseph A. Paradiso. Printsense: A versatile sensing technique to support multimodal flexible surface interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pages 1407–1410, New York, NY, USA, 2014. ACM.

[50] M Goulthorpe and M Burry. Aegis hyposurface (c), p. *Maubant, Editor, Aegis Hyposurface©, Institut DÕart Contemporain*, 2002.

[51] Sidhant Gupta, Tim Campbell, Jeffrey R. Hightower, and Shwetak N. Patel. SqueezeBlock. In *UIST '10*, page 101. ACM Press, October 2010.

[52] Jefferson Y. Han. Low-cost multi-touch sensing through frustrated total internal reflection. In *Proceedings of the 18th annual ACM symposium on User interface software and technology - UIST '05*, UIST '05, page 115, New York, New York, USA, 2005. ACM Press.

[53] Chris Harrison and Scott E. Hudson. Providing dynamically changeable physical buttons on a visual display. In *CHI '09*. ACM Press, April 2009.

[54] Bjoern Hartmann, Maurizio Mancini, and Catherine Pelachaud. Towards affective agent action: Modelling expressive eca gestures. *International conference on Intelligent User Interfaces-Workshop on Affective Interaction, San Diego, CA*, 2005.

[55] 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*, pages 145–154. ACM, 2007.

[56] 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, pages 145–154, New York, NY, USA, 2007. ACM.

[57] Björn Hartmann, Maurizio Mancini, and Catherine Pelachaud. Implementing expressive gesture synthesis for embodied conversational agents. In *gesture in human-Computer Interaction and Simulation*, pages 188–199. Springer Berlin Heidelberg, 2006.

[58] Rex Hartson. Cognitive, physical, sensory, and functional affordances in interaction design. *Behaviour & Information Technology*, 22(5):315–338, 2003.

[59] Fabian Hemmert, Susann Hamann, Matthias Löwe, Anne Wohlauf, and Gesche Joost. Shape-changing mobiles. In *TEI '10*, page 249. ACM Press, January 2010.

[60] Ralf Herbrich, Tom Minka, and Thore Graepel. Trueskill?: A bayesian skill rating system. In *Advances in Neural Information Processing Systems*, pages 569–576, 2006.

[61] Gero Herkenrath, Thorsten Karrer, and Jan Borchers. Twend. In *CHI EA '08*, page 3819, New York, New York, USA, April 2008. ACM Press.

[62] O. Hilliges, S. Izadi, A.D. Wilson, S. Hodges, A. Garcia-Mendoza, and A. Butz. Interactions in the Air : Adding Further Depth to Interactive Tabletops. In *Proceedings of the 22nd annual ACM symposium on User interface software and technology*, pages 139–148. ACM, 2009.

[63] Otmar Hilliges, David Kim, and Shahram Izadi. Creating malleable interactive surfaces using liquid displacement sensing. In *Proc. Tabletop 2008*, pages 157–160. IEEE Press, October 2008.

[64] Ken Hinckley, Randy Pausch, John C. Goble, and Neal F. Kassell. Passive real-world interface props for neurosurgical visualization. *Conference on Human Factors in Computing Systems*, page 452, 1994.

[65] James Hollan, Edwin Hutchins, and David Kirsh. Distributed cognition: toward a new foundation for human-computer interaction research. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 7(2):174–196, 2000.

[66] Jonathan Hook, Stuart Taylor, Alex Butler, Nicolas Villar, and Shahram Izadi. A reconfigurable ferromagnetic input device. In *Proc. UIST 2009*, page 51, New York, New York, USA, 2009. ACM Press.

[67] J. E. Huber, N. a. Fleck, and M. F. Ashby. The selection of mechanical actuators based on performance indices. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 453(1965):2185–2205, October 1997.

[68] Edwin Hutchins. *Cognition in the Wild*, volume 262082314. MIT press Cambridge, MA, 1995.

[69] Atsushi Iriki, Michio Tanaka, and Yoshiaki Iwamura. Coding of modified body schema during tool use by macaque postcentral neurones. *Neuroreport*, 7(14):2325–2330, 1996.

[70] H Ishii, C Ratti, B Piper, Y Wang, A Biderman, and E Ben-Joseph. Bringing Clay and Sand into Digital Design âĂŤ Continuous Tangible user Interfaces. *BT Technology Journal*, 22(4):287–299, October 2004.

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

[72] Hiroshi Ishii and Brygg Ullmer. Tangible bits. In *Proc. CHI 1997*, pages 234–241, New York, New York, USA, March 1997. ACM Press.

[73] Hiroo Iwata, Hiroaki Yano, Fumitaka Nakaizumi, and Ryo Kawamura. Project FEELEX. In *SIGGRAPH '01*, pages 469–476. ACM Press, August 2001.

[74] Shahram Izadi, Steve Hodges, Stuart Taylor, Dan Rosenfeld, Nicolas Villar, Alex Butler, and Jonathan Westhues. Going beyond the display. In *Proceedings of the 21st annual ACM symposium on User interface software and technology - UIST '08*, page 269, New York, New York, USA, October 2008. ACM Press.

[75] Yvonne Jansen, Thorsten Karrer, and Jan Borchers. MudPad: tactile feedback and haptic texture overlay for touch surfaces. In *Proc. ITS 2010*, pages 11–14. ACM Press, 2010.

[76] Chujit Jeamsinkul and Sharon Poggenpohl. Methodology for uncovering motion affordance in interactive media. *Visible Language*, 36(3):254, 2002.

[77] M.K. Johnson and E.H. Adelson. Retrographic sensing for the measurement of surface texture and shape. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1070–1077. Ieee, June 2009.

[78] Edward G Jones. The sensory hand. *Brain*, 129(12):3413–3420, 2006.

[79] Lynette A. Jones and Susan J. Lederman. *Human Hand Function.* Oxford University Press, 2006.

[80] Martin Kaltenbrunner and Ross Bencina. reacTIVision. In *Proceedings of the 1st international conference on Tangible and embedded interaction - TEI '07*, page 69, New York, New York, USA, February 2007. ACM Press.

[81] Victor Kaptelinin and Bonnie Nardi. Affordances in hci: toward a mediated action perspective. In *CHI '12*, pages 967–976. ACM, 2012.

[82] David Kirsh. The intelligent use of space. *Artificial Intelligence*, 73(1-2):31–68, February 1995.

[83] David Kirsh. Embodied cognition and the magical future of interaction design. *ACM Transactions on Computer-Human Interaction*, 20(1):1–30, March 2013.

[84] David Kirsh and Paul Maglio. On distinguishing epistemic from pragmatic action. *Cognitive science*, 18(4):513–549, 1994.

[85] David Kirsh and Paul Maglio. On distinguishing epistemic from pragmatic action. *Cognitive Science*, 18(4):513–549, 1994.

[86] R.L. Klatzky and S.J. Lederman. The intelligent hand. *Psychology of learning and motivation: Advances in research and theory*, 21:121–151, 1987.

[87] Scott R. Klemmer, Björn Hartmann, and Leila Takayama. How bodies matter: five themes for interaction design. In *DIS '06*, pages 140–149. ACM, 2006.

[88] Minoru KOJIMA, Maki SUGIMOTO, Akihiro NAKAMURA, Masahiro TOMITA, Masahiko INAMI, and Hideaki NII. Augmented coliseum: An augmented game environment with small vehicles. In *TABLETOP '06*, pages 3–8. IEEE Computer Society, 2006.

[89] Minoru Kojima, Maki Sugimoto, Akihiro Nakamura, Masahiro Tomita, Masahiko Inami, and Hideaki Nii. Augmented coliseum: An augmented game environment with small vehicles. *Tabletop*, pages 3–8, 2006.

[90] Myron W Krueger, Thomas Gionfriddo, and Katrin Hinrichsen. VideoplaceÑan artificial reality. In *ACM SIGCHI Bulletin*, volume 16, pages 35–40. ACM, 1985.

[91] Dana Kulic and Elizabeth Croft. Physiological and subjective responses to articulated robot motion. *Robotica*, 25(01):13–27, 2007.

[92] Byron Lahey, Audrey Girouard, Winslow Burleson, and Roel Vertegaal. PaperPhone. In *CHI '11*, page 1303. ACM Press, May 2011.

[93] Douglas Lanman and Gabriel Taubin. Build your own 3D scanner. In *ACM SIGGRAPH 2009 Courses on - SIGGRAPH '09*, pages 1–94, New York, New York, USA, August 2009. ACM Press.

[94] Matthew. J. Large, Timothy Large, and A. R. L. Travis. Parallel Optics in Waveguide Displays: A Flat Panel Autostereoscopic Display. *Journal of Display Technology*, 6(10):431–437, October 2010.

[95] Vincent LeClerc, Amanda Parkes, and Hiroshi Ishii. Senspectra: a computationally augmented physical modeling toolkit for sensing and visualization of structural strain. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '07, pages 801–804, New York, NY, USA, 2007. ACM.

[96] A. Lecuyer, S. Coquillart, A. Kheddar, P. Richard, and P. Coiffet. Pseudo-haptic feedback: can isometric input devices simulate force feedback? In *Proceedings IEEE Virtual Reality 2000 (Cat. No.00CB37048)*, pages 83–90. IEEE Comput. Soc.

[97] Narae Lee, Juwhan Kim, Jungsoo Lee, Myeongsoo Shin, and Woohun Lee. Molebot: mole in a table. In *SIGGRAPH '11*, pages 9:1–9:1. ACM, 2011.

[98] Woohun Lee and Jun Park. Augmented Foam: A Tangible Augmented Reality for Product Design. *Symposium on Mixed and Augmented Reality*, 2005.

[99] Shuangyan Lei and Song Zhang. Flexible 3-D shape measurement using projector defocusing. *Optics Letters*, 34(20):3080, October 2009.

[100] Daniel Leithinger, Sean Follmer, Alex Olwal, Samuel Luescher, Akimitsu Hogge, Jinha Lee, and Hiroshi Ishii. Sublimate: State-changing virtual and physical rendering to augment interaction with shape displays. In *CHI '13*, pages 1441–1450. ACM, 2013.

[101] Daniel Leithinger and Hiroshi Ishii. Relief: a scalable actuated shape display. In *Proceedings of the fourth international conference on Tangible, embedded, and embodied interaction*, pages 221–222. ACM, 2010.

[102] Daniel Leithinger, David Lakatos, Anthony DeVincenzi, Matthew Blackshaw, and Hiroshi Ishii. Direct and gestural interaction with relief: a 2.5d shape display. In *UIST '11*, pages 541–548. ACM, 2011.

[103] AJ Liu. Jamming is not just cool any more. *Nature*, 396(November), 1998.

[104] Arjo J. Loeve, Oscar S. Ven, Johan G. Vogel, Paul Breedveld, and Jenny Dankelman. Vacuum packed particles as flexible endoscope guides with controllable rigidity. *Granular Matter*, 12(6):543–554, June 2010.

[105] Christine L MacKenzie and Thea Iberall. *The grasping hand*, volume 104. Elsevier, 1994.

[106] M.R. Marner and B.H. Thomas. Augmented foam sculpting for capturing 3d models. In *3D User Interfaces (3DUI), 2010 IEEE Symposium on*, pages 63 –70, march 2010.

[107] Nicolai Marquardt, Miguel A. Nacenta, James E. Young, Sheelagh Carpendale, Saul Greenberg, and Ehud Sharlin. The Haptic Tabletop Puck. In *ITS '09*. ACM Press, November 2009.

[108] Andrea Mazzone, Christian Spagno, and Andreas Kunz. The HoverMesh. In *ACE '04*. ACM Press, September 2004.

[109] Michael McGuffin and Ravin Balakrishnan. Acquisition of expanding targets. In *CHI '02*, pages 57–64. ACM, 2002.

[110] G. Michelitsch, J. Williams, M. Osen, B. Jimenez, and S. Rapp. Haptic chameleon: a new concept of shape-changing user interface controls with force feedback. In *CHI EA '04*, pages 1305–1308. ACM, 2004.

[111] T. Mitsuda, S. Kuge, M. Wakabayashi, and S. Kawamura. Wearable haptic display by the use of a Particle Mechanical Constraint. *Proceedings 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. HAPTICS 2002*, pages 153–158, 2002.

[112] AJung Moon, Chris AC Parker, Elizabeth A Croft, and HF Machiel Van der Loos. Design and impact of hesitation gestures during human-robot resource conflicts. *Journal of Human-Robot Interaction*, 2(3):18–40, 2013.

[113] Nikhil Naik, Jade Philipoom, Ramesh Raskar, and Cesar Hidalgo. Streetscore–predicting the perceived safety of one million streetscapes. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on*, pages 793–799. IEEE, 2014.

[114] Diego Nehab, Szymon Rusinkiewicz, James Davis, and Ravi Ramamoorthi. Efficiently combining positions and normals for precise 3D geometry. *ACM Transactions on Graphics*, 24(3):536, July 2005.

[115] Alva Noë. *Action in perception.* MIT press, 2004.

[116] Donald A. Norman. *The Psychology of Everyday Things.* Basic Books, Inc., 1988.

[117] Donald A. Norman. *The Design of Everyday Things: Revised and Expanded Edition.* Basic Books, 2013.

[118] Allison Okamura. *Haptic Exploration of Unknown Objects.* PhD thesis, 2000.

[119] Simon Olberding, Nan-Wei Gong, John Tiab, Joseph A. Paradiso, and Jürgen Steimle. A cuttable multi-touch sensor. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*, UIST '13, pages 245–254, New York, NY, USA, 2013. ACM.

[120] Mai Otsuki, Kenji Sugihara, Asako Kimura, Fumihisa Shibata, and Hideyuki Tamura. MAI painting brush: an interactive device that realizes the feeling of real painting. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*, pages 97–100. ACM, 2010.

[121] Jifei Ou, Lining Yao, Daniel Tauber, Jürgen Steimle, Ryuma Niiyama, and Hiroshi Ishii. jamsheets: Thin interfaces with tunable stiffness enabled by layer jamming. In *Proceedings of the 8th International Conference on Tangible, Embedded and Embodied Interaction*, TEI '14, pages 65–72, New York, NY, USA, 2013. ACM.

[122] Dan Overholt. The MATRIX: a novel controller for musical expression. In *Proc. NIME 2001*, NIME '01, pages 1–4, Singapore, 2001. AMC.

[123] Gian Pangaro, Dan Maynes-Aminzade, and Hiroshi Ishii. The actuated workbench: computer-controlled actuation in tabletop tangible interfaces. In *UIST '02*, pages 181–190. ACM, 2002.

[124] Yong-Lae Park, Carmel Majidi, Rebecca Kramer, Phillipe Bérard, and Robert J Wood. Hyperelastic pressure sensing with a liquid-embedded elastomer. *Journal of Micromechanics and Microengineering*, 20(12):125029, December 2010.

[125] Amanda Parkes, Ivan Poupyrev, and Hiroshi Ishii. Designing kinetic interactions for organic user interfaces. *Communications of the ACM*, 51(6):58, June 2008.

[126] Amanda Jane Parkes. *Phrases of the kinetic: dynamic physicality as a dimension of the design process.* PhD thesis, Cambridge, MA, USA, 2009. AAI0822076.

[127] James Patten and Hiroshi Ishii. Mechanical constraints as computational constraints in tabletop tangible interfaces. In *CHI '07*, pages 809–818. ACM, 2007.

[128] Esben W Pedersen, Sriram Subramanian, and Kasper Hornbæk. Is my phone alive?: a large-scale study of shape change in handheld devices using videos. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems*, pages 2579–2588. ACM, 2014.

[129] Sarah Perez. itunes app store now has 1.2 million apps, has seen 75 billion downloads to date. *TechCrunch.com.* http://techcrunch.com/2014/06/02/itunes-app-store-now-has-1-2-million-apps-has-seen-75-billion-downloads-to-date/. Accessed: 2014-12-24.

[130] Ben Piper, Carlo Ratti, and Hiroshi Ishii. Illuminating clay. In *Proc. CHI 2002*, page 355, New York, New York, USA, April 2002. ACM Press.

[131] Frank E Pollick, Helena M Paterson, Armin Bruderlin, and Anthony J Sanford. Perceiving affect from arm movement. *Cognition*, 82(2):B51–B61, 2001.

[132] Ivan Poupyrev, Tatsushi Nashida, and Makoto Okabe. Actuation and tangible user interfaces. In *Proceedings of the 1st international conference on Tangible and embedded interaction - TEI '07*, page 205, New York, New York, USA, February 2007. ACM Press.

[133] P Prautsch and T Mita. Control and analysis of the gait of snake robots. In *Control Applications, 1999. Proceedings of the 1999 IEEE International Conference on*, volume 1, pages 502–507. IEEE, 1999.

[134] Ramesh Raskar, Greg Welch, Kok-Lim Low, and Deepak Bandyopadhyay. Shader lamps: Animating real objects with image-based illumination. In *Proceedings of the 12th Eurographics Workshop on Rendering Techniques*, pages 89–102, London, UK, 2001. Springer-Verlag.

[135] Majken K. Rasmussen, Esben W. Pedersen, Marianne G. Petersen, and Kasper Hornbaek. Shape-changing interfaces: a review of the design space and open research questions. In *CHI '12*, pages 735–744. ACM, 2012.

[136] Jun Rekimoto. SmartSkin. In *Proc. CHI 2002*, CHI '02, page 113, New York, New York, USA, 2002. ACM Press.

[137] Jun Rekimoto, Brygg Ullmer, and Haruo Oba. Datatiles: a modular platform for mixed physical and graphical interactions. In *CHI '01*, pages 269–276. ACM, 2001.

[138] Dan Reznik and John Canny. A flat rigid plate is a universal planar manipulator. In *IEEE ICRA 1998*, volume 2, pages 1471–1477. IEEE, 1998.

[139] Howard Rheingold. *Virtual Reality: Exploring the Brave New Technologies.* Simon & Schuster Adult Publishing Group, 1991.

[140] Ilya Rosenberg and Ken Perlin. The UnMousePad. In *Proc. SIGGRAPH 2009*, volume 28, page 1, New York, New York, USA, July 2009. ACM Press.

[141] Jarek Rossignac, Mark Allen, Wayne J Book, Ari Glezer, Imme Ebert-Uphoff, Chris Shaw, David Rosen, Stephen Askins, Jing Bai, Paul Bosscher, et al. Finger sculpting with digital clay: 3d shape input and output through a computer-controlled real surface. In *Shape Modeling International, 2003*, pages 229–231. IEEE, 2003.

[142] Anne Roudaut, Abhijit Karnik, Markus Löchtefeld, and Sriram Subramanian. Morphees: toward high shape resolution in self-actuated flexible mobile devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 593–602. ACM, 2013.

[143] James A Russell. A circumplex model of affect. *Journal of personality and social psychology*, 39(6):1161, 1980.

[144] J.K. Salisbury and M.A. Srinivasan. Phantom-based haptic interaction with virtual objects. *Computer Graphics and Applications, IEEE*, 17(5):6 – 10, sept.-oct. 1997.

[145] K. Salisbury, F. Conti, and F. Barbagli. Haptic rendering: introductory concepts. *Computer Graphics and Applications, IEEE*, 24(2):24–32, 2004.

[146] Toshiki Sato, H. Mamiya, T. Tokui, H. Koike, and K. Fukuchi. PhotoelasticTouch: transparent rubbery interface using a LCD and photoelasticity. In *ACM SIGGRAPH 2009 Emerging Technologies*, pages 1–1. ACM, 2009.

[147] Toshiki Sato, Yasushi Matoba, Nobuhiro Takahashi, and Hideki Koike. ClaytricSurface: An Intetractive Surface that can Dynamically Change the Flexibility. Technical report, Laval Virtual Awards 2012, 2012.

[148] Valkyrie Savage, Colin Chang, and Björn Hartmann. Sauron: Embedded single-camera sensing of printed physical user interfaces. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*, UIST '13, pages 447–456, New York, NY, USA, 2013. ACM.

[149] Julia Schwarz, Chris Harrison, Scott Hudson, and Jennifer Mankoff. Cord input: an intuitive, high-accuracy, multi-degree-of-freedom input method for mobile devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1657–1660. ACM, 2010.

[150] Carsten Schwesig, Ivan Poupyrev, and Eijiro Mori. Gummi: a bendable computer. In *Proceedings of the 2004 conference on Human factors in computing systems - CHI '04*, pages 263–270, New York, New York, USA, April 2004. ACM Press.

[151] M Sergio, N Manaresi, and M Tartagni. A Textile Based Capacitive Pressure Sensor. *Sensors, 2002.*, 2002.

[152] Jia Sheng, Ravin Balakrishnan, and Karan Singh. An interface for virtual 3D sculpting via physical proxy. *Computer graphics and interactive techniques in Australasia and South East Asia*, page 213, 2006.

[153] Ben Shneiderman, Gerhard Fischer, Mary Czerwinski, Brad Myers, and Mitch Resnick. Creativity Support Tools: A workshop sponsored by the National Science Foundation, 2005.

[154] Sile O'Modhrain. *Playing by Feel: Incorporating Haptic Feedback into Computer-Based musical Instruments.* PhD thesis, Stanford University, 2000.

[155] Michael Sinclair. The haptic lens. In *ACM SIGGRAPH 97 Visual Proceedings: The art and interdisciplinary programs of SIGGRAPH '97 on - SIGGRAPH '97*, SIGGRAPH '97, page 179, New York, New York, USA, 1997. ACM Press.

[156] J D Smith, T C N Graham, D Holman, and J Borchers. Low-Cost Malleable Surfaces with Multi-Touch Pressure Sensitivity. In *TABLETOP 2007*, pages 205–208. IEEE, 2007.

[157] Ross T. Smith, Bruce H. Thomas, and Wayne Piekarski. Digital foam interaction techniques for 3D modeling. In *Virtual Reality Software and Technology*, pages 61–68, 2008.

[158] Rajinder Sodhi, Hrvoje Benko, and Andrew Wilson. Lightguide: Projected visualizations for hand movement guidance. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 179–188, New York, NY, USA, 2012. ACM.

[159] Alexander Sproewitz, Aude Billard, Pierre Dillenbourg, and Auke Jan Ijspeert. Roombots-mechanical design of self-reconfiguring modular robots for adaptive furniture. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 4259–4264. IEEE, 2009.

[160] E Steltz, A Mozeika, and J Rembisz. Jamming as an enabling technology for soft robotics. *Society of Photo-*, 2010.

[161] E. Steltz, A. Mozeika, N. Rodenberg, E. Brown, and H.M. Jaeger. JSEL: Jamming Skin Enabled Locomotion. *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5672–5677, October 2009.

[162] Ivan E. Sutherland. The ultimate display, 1965.

[163] Ivan E. Sutherland. A head-mounted three dimensional display. In *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*, AFIPS '68 (Fall, part I), pages 757–764, New York, NY, USA, 1968. ACM.

[164] Brandon T. Taylor and V. Michael Bove. Graspables. In *Proceedings of the 27th international conference on Human factors in computing systems - CHI '09*, page 917, New York, New York, USA, April 2009. ACM Press.

[165] Marco Tettamanti, Giovanni Buccino, Maria Cristina Saccuman, Vittorio Gallese, Massimo Danna, Paola Scifo, Ferruccio Fazio, Giacomo Rizzolatti, Stefano F Cappa, and Daniela Perani. Listening to action-related sentences activates fronto-parietal motor circuits. *Journal of cognitive neuroscience*, 17(2):273–281, 2005.

[166] Tom White. *Introducing Liquid Haptics in High Bandwidth Human Computer Interfaces*. PhD thesis, MIT, 1998.

[167] S Torquato, Tm Truskett, and Pg Debenedetti. Is random close packing of spheres well defined? *Physical review letters*, 84(10):2064–7, March 2000.

[168] Brygg Ullmer, Hiroshi Ishii, and Robert J. K. Jacob. Token+constraint systems for tangible interaction with digital information. *ACM Trans. Comput.-Hum. Interact.*, 12(1):81–118, March 2005.

[169] John Underkoffler and Hiroshi Ishii. Urp: a luminous-tangible workbench for urban planning and design. In *Proceedings of the SIGCHI conference on Human factors in computing systems: the CHI is the limit*, CHI '99, pages 386–393, New York, NY, USA, 1999. ACM.

[170] R. W. Van Boven and K. O. Johnson. The limit of tactile spatial resolution in humans: Grating orientation discrimination at the lip, tongue, and finger. *Neurology*, 44(12):2361–2361, December 1994.

[171] Peter Vandoren, Tom Van Laerhoven, Luc Claesen, Johannes Taelman, Chris Raymaekers, and Frank Van Reeth. IntuPaint: Bridging the gap between physical and digital painting. In *2008 3rd IEEE International Workshop on Horizontal Interactive Human Computer Systems*, pages 65–72. Ieee, October 2008.

[172] Leslie Carlson Vaughan. Understanding movement. In *CHI '97*, pages 548–549. ACM, 1997.

[173] Jo Vermeulen, Kris Luyten, Elise van den Hoven, and Karin Coninx. Crossing the bridge over norman's gulf of execution: revealing feedforward's true identity. In *CHI '13*, pages 1931–1940. ACM, 2013.

[174] Roel Vertegaal and Ivan Poupyrev. Organic User Interfaces. *Communications of the ACM*, 51(6):26, June 2008.

[175] Raquel Viciana-Abad, Arcadio Reyes Lecuona, and Matthieu Poyade. The Influence of Passive Haptic Feedback and Difference Interaction Metaphors on Presence and Task Performance. *Presence: Teleoperators and Virtual Environments*, 19(3):197–212, June 2010.

[176] Nicolas Villar and Hans Gellersen. A malleable control structure for softwired user interfaces. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction*, TEI '07, pages 49–56, New York, NY, USA, 2007. ACM.

[177] Kevin Vlack, T. Mizota, N. Kawakami, Kazuto Kamiyama, H. Kajimoto, and S. Tachi. Gelforce: a vision-based traction field computer interface. In *Ext. Abstracts CHI 2005*, pages 1154–1155. ACM Press, 2005.

[178] Malte Weiss, Christian Remy, and Jan Borchers. Rendering physical effects in tabletop controls. In *CHI '11*, pages 3009–3012. ACM, 2011.

[179] Malte Weiss, Florian Schwarz, Simon Jakubowski, and Jan Borchers. Madgets: actuating widgets on interactive tabletops. In *UIST '10*, pages 293–302. ACM, 2010.

[180] Malte Weiss, Chat Wacharamanotham, Simon Voelker, and Jan Borchers. Fingerflux: near-surface haptic feedback on tabletops. In *UIST '11*, pages 615–620. ACM, 2011.

[181] Malte Weiss, Julie Wagner, Yvonne Jansen, Roger Jennings, Ramsin Khoshabeh, James D. Hollan, and Jan Borchers. Slap widgets: bridging the gap between virtual and physical controls on tabletops. In *CHI '09*, pages 481–490. ACM, 2009.

[182] Pierre Wellner. Interacting with paper on the digitaldesk. *Commun. ACM*, 36:87–96, July 1993.

[183] Andrew D. Wilson. TouchLight. In *Proceedings of the 6th international conference on Multimodal interfaces - ICMI '04*, page 69, New York, New York, USA, October 2004. ACM Press.

[184] Frank R Wilson. *The hand: How its use shapes the brain, language, and human culture.* Random House LLC, 2010.

[185] Margaret Wilson. Six views of embodied cognition. *Psychonomic bulletin & review*, 9(4):625–636, 2002.

[186] Ryder Winck, Jingu Kim, Wayne J Book, and Haesun Park. Command generation techniques for a pin array using the svd and the snmf. In *Robot Control*, volume 10, pages 411–416, 2012.

[187] Terry Winograd and Fernando Flores. *Understanding computers and cognition: A new foundation for design.* Intellect Books, 1986.

[188] Cornell Wright, Aaron Johnson, Aaron Peck, Zachary McCord, Allison Naaktgeboren, Philip Gianfortoni, Manuel Gonzalez-Rivero, Ross Hatton, and Howie Choset. Design of a modular snake robot. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 2609–2614. IEEE, 2007.

[189] Lining Yao, Ryuma Niiyama, Jifei Ou, Sean Follmer, Clark Della Silva, and Hiroshi Ishii. Pneui: Pneumatically actuated soft composite materials for shape changing interfaces. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*, UIST '13, pages 13–22, New York, NY, USA, 2013. ACM.

[190] Haiyan Zhang. *Control Freaks.* PhD thesis, Interaction Design Institute Ivrea, 2006.

[191] Haiyan Zhang and Bjöern Hartmann. Building upon everyday play. In *CHI'07 extended abstracts on Human factors in computing systems*, pages 2019–2024. ACM, 2007.

[192] Jiaje Zhang and Donald A Norman. Representations in distributed cognitive tasks. *Cognitive science*, 18(1):87–122, 1994.

[193] Jiajie Zhang. The nature of external representations in problem solving. *Cognitive science*, 21(2):179–217, 1997.

[194] Jiajie Zhang and Vimla L Patel. Distributed cognition, representation, and affordance. *Pragmatics & Cognition*, 14(2), 2006.

[195] Song Zhang. Recent progresses on real-time 3D shape measurement using digital fringe projection techniques. *Optics and Lasers in Engineering*, 48(2):149–158, February 2010.

[196] Song Zhang and Shing-Tung Yau. Generic nonsinusoidal phase error correction for three-dimensional shape measurement using a digital video projector. *Applied Optics*, 46(1):36, January 2007.