# Sensing Through Structure

## Ronit Slyper

CMU-CS-12-115
April 30, 2012

School of Computer Science
Computer Science Department
Carnegie Mellon University
Pittsburgh, PA 15213

**Thesis Committee:**
Jessica Hodgins, chair
Ivan Poupyrev
Scott Hudson
Eric Paulos

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

**Abstract**

We present an approach to designing input devices that focuses on the structure of materials. We explore and visualize how a material reacts under manipulation, and harness the material's properties to design new movement sensors. Two benefits spring out of this approach. One, simpler sensing emerges from making use of existing structure in the material. Two, by working with the natural structure of the material, we create input devices with readily recognizable affordances. We present six projects using this approach. We use the natural structure (coordination) of the human body to enable a mapping from five clothing-mounted accelerometers to high-quality motion capture data, creating a low-cost performance animation system. We design silicone input devices with embedded texture allowing single-camera tracking. We study squishable, conformable materials such as foam and silicone, and create a vocabulary of unit structures (shaped cuts in the material) for harnessing patterns of compression/tension to capture particular manipulations. We use this vocabulary to build soft sensing skeletons for stuffed animals, making foam cores with e-textile versions of our unit structures. We also use this vocabulary to design a tongue input device for a collaboration with Disney Imagineering. Finally, we rethink this vocabulary and apply it to capturing, using air pressure sensors, manipulations of hollow 3D-printed rubber shapes, and 3D-print several interactive robots incorporating the new vocabulary.

# Contents

# Chapter 1

# Overview

Computer science rests on layers of building blocks, each set more elegant than the next. On one level, a grammar set of {if, else, print, =, . . .} designs a video game, manipulates a million images, simulates a piece of the world at the atomic level. Head a level down and the entire grammar reduces to the even simpler set of building blocks of {read symbol, write symbol, move tape, move state, . . . }. Our processors that implement Turing machines, in all their complexity, are built from transistors implementing gates of {and, or, not}.

When humans – the users of computer science — are added to the top level, however, all elegance seems to flee. Humans physically interact with computers using numerous devices with numerous input methods: mouse, keyboard, styluses, toys, game controllers, appliances, movement sensors, standard cameras, depth cameras, cellphone cameras . . . the interactions of a human with a computer seem not to be reducible to a few simple manipulations.

But although human interaction seems indefinably high-dimensional, structure exists in how we interact with objects. In this thesis I look at how we manipulate objects at the material level, and study what material properties I can harness to discover structure in the effects of those manipulations. The result is the lowering of the dimensionality of the problem; not always to the elegance of the described computing systems, but to the level where we can reduce the complexity of the sensing infrastructure for capturing an interaction to a manageable setup that is simpler to implement. I present six projects which each use this insight to address a different problem.

In my first project, I attacked the problem of capturing the movements of the human body and using the motion as input to control a real-time virtual avatar who will mimic the motion with high fidelity. Such a control problem, at the time, usually required a multi-thousand-dollar motion capture system to track the 3D position of dozens of markers attached to the body. The dimensionality of the problem, however, can be reduced. The material under consideration, the human body, is a hierarchical skeleton with joint limits. Although we have dozens of joints that can each move independently, they behave with coordination when performing an action. The body's anatomical constraints also limit the range of motion and therefore the actions it can perform. Some element of this anatomical structure – muscular layout, memory, or efficiency – also encourages the body to perform the same actions with the same acceleration profile each time. The structure I discovered was thus that human motion is lower-dimensional than it would otherwise appear. I

used this structure to conjecture a roughly one-to-one correspondence between the accelerations of a moving limb performing an action, and an actual, position-based representation of the action. Using this correspondence, I showed that I can use a pre-recorded, position-based motion capture dataset as the backend of an accelerometer-based performance animation system. Accelerations from a wired t-shirt were streamed in real time to a computer, which compared them to double-differentiated position values from the dataset, and smoothly blended to the best matching motion. I published this work in the ACM Symposium on Computer Animation (SCA) in 2008 [Slyper and Hodgins 2008].

In my second project, the goal was to capture, in high resolution, the deformations of a flexible silicone surface, and to design user interactions around this surface. The hard problem of capturing deformations was solved by building structure, in the form of texture, into the silicone. My role in this work was the design of the silicone input devices; credit for the original idea of a silicone input device, and the optical flow tracking implementation and 3D reconstruction, goes to my collaborators. My challenge was to optimize the tracking structure for reconstruction by a single camera. In the process, I harnessed several unique properties of silicone. The ability to easily embed objects and colored layers let me incorporate the tracking structure. With the use of additives to change the viscosity of silicone, I could experiment with different methods for incorporating texture. In addition, the final silicone-based textures stretched continuously instead of breaking apart, thus fulfilling the local affine movement assumption of the optical flow algorithm being used. In the course of my work, I showed that silicone input devices with texture could be created in any convex, or nearly convex, shape. In this chapter, I detail the various prototypes I built, and the lessons learned from each. I also describe how the properties of the prototypes affect the applications that could be developed around this technology, and I suggest several applications based on these guidelines. This work is in preparation for publication.

I turn for my third project, the core of this thesis, to capturing the manipulations of soft objects to be used as input devices. Conformable, squishable materials (foam, rubber, silicone) deform smoothly over their entire surfaces when bent, twisted, or pulled; capturing, e.g., the angle of continuous bend in a foam sheet is a hard problem. I observed how deformable materials behave: which parts are in compression, and which under tension, during a deformation. The continuous problem can be reduced to the binary question of the presence of compression versus tension. From there I created a vocabulary of atomic structures that, when placed in the material anywhere in the area undergoing deformation, discretize the problem to measuring a single linear displacement of that compression or tension. Much like movement (output) is built up from a vocabulary of six simple machines, I create a vocabulary of input. These structures have the advantage that they can be designed to be seen and/or felt, revealing the sensing affordances of the object. I developed a method of placing zigzag traces between layers of silicone while sandwiching in conductive fabric contact switches to create the structures. I used the method to create rugged, flexible sensors and game controllers. I published this work in Tangible, Embedded, and Embodied Interaction (TEI) in 2011 [Slyper et al. 2011a].

For my fourth project, I extended the application of the third, turning it into a methodology for solving a problem I see today: the increasing prevalence of stuffed-animal-type toys with hard sensors destroying their soft charm. I set forth a method of creating the structures in soft foam, using conductive fabric tape for the switches and conductive thread for the wiring. The conductive-thread wiring is sewed directly into the foam core. I showed that this method can be used to make nearly completely soft, sensing stuffed

animals. I created several stuffed animals with foam cores containing sensing structures. This work is not yet published.

My fifth project was a collaboration with the Walt Disney Company to build a system for an actor to hold a conversation using pieces of recorded audio. The actor is confined by an enclosing character costume, and must maintain the physical presence of the character. I attacked the problem on two fronts. On the first front, the actor needs an input device. Because the actor must gesticulate with his limbs, and cannot make noise inside the costume, the tongue is the only high degree of freedom muscle left that is versatile and quick. I prototyped a tongue input device using the vocabulary and construction methodologies from the third project, showing the vocabulary to be useful in thinking about the design of input devices. On the second front, the actor must hold a reasonable conversation using only pre-recorded input. I studied the progression of conversations held by costumed characters in the theme parks. I found such conversations to be rigidly structured, with predictable development. I developed a dialogue tree system and heads up display for the actor to hold conversations. I published this work in the ACM Symposium on User Interface Software and Technology (UIST) in 2011 [Slyper et al. 2011b].

For my sixth project I extended to another media, and another area of application, the thought process involved in creating a conformable-material sensing vocabulary. In this work, I showed that my ideas can be combined with properties of new rapid prototyping technology to iterate on the design of interactive robots. Objet 3D printers can now print a flexible rubber-like material which is airtight. The ability to 3D print the material means it can be built into complex shapes that would be difficult to cast, such as tight hollow areas. I created a set of structures for indicating and capturing various manipulations, built out of hollow flexible chambers and monitored using air pressure sensors. In my method, robot skins are designed in a CAD program, e.g. Solidworks, with the sensing structures and armature attachment points as an integral part of the skin. The models are then printed on an Objet 3D printer using its flexible rubber-like material. The air pressure sensors can be plugged in and easily removed. The entire process creates an easy way of prototyping robot skin, movement, and interaction in concert. This work is under review.

My thesis is about looking at the physical world of movement to find the elegance underlying it, and building sensing vocabularies and systems that harness that elegance. These projects demonstrate the following thesis statement:

> By discovering the natural material and human structure in an input problem, we reduce the problem's dimensionality, simplifying the sensing and creating intuitive input devices.

The accelerometer project showed that human motion operates on a lower dimension than the body's many degrees of freedom would suggest; I took advantage of this observation to reduce the sensing technology for performance capture to five accelerometers sewn onto the outside of a t-shirt. In the silicone input device project, I optimized structure, in the form of texture, to reduce tracking hardware requirements to a single camera. In the silicone sensors work, I created a vocabulary of atomic structures that sensed various manipulations, breaking down the problem of sensing an arbitrary configuration of a conformable material to an easier one of designing sensing for simple linear displacements. The atomic structures are a visible part of the input device, and provide tactile feedback when closed. I then took advantage of the idea of atomic structures in my next work, adding new materials to apply the method to the practical goal of soft interactive stuffed animals. The same structures were again used in the thought process for the design of

a tongue joystick, wherein I also discovered structure in human dialogue, and used that structure to guide the design of a dialogue tree for character interaction. Finally, designing robot skins with large swaths of sensing elements requires a large investment of time and money; I showed that a prototyping phase was possible, using flexible-rubber printing and only a few air pressure sensors to aggregate touch over areas of the robot skin. Similarly to the silicone devices work, I designed air pressure chambers to present physical affordances to the user and simplify the capture of various manipulations.

# Chapter 2

# Accelerometer Performance Animation System

As the ultimate conformable, unobtrusive user interface, augmented clothing has been the dream of science fiction writers and computer researchers for years (e.g. [Vinge 2006]). The realization of such a dream would result in a paradigm shift, removing the concept of user interfaces as external tools. With accelerometer prices having plummeted and etextiles on the rise, an ideal confluence of factors was present, at the time the work in this chapter was completed in 2008, to attempt to use clothing as a performance animation interface.

The motivation for this research was the question, "Are accelerometers accurate enough to allow the human body to be used as a real-time interface to a virtual avatar mimicking the human's motion?"

As it stands, the answer to the question is "No". Simply double-integrating accelerations to produce limb positions results in significant drift after a very short time. This fact can be quickly demonstrated by attempting to hold an accelerometer immobile on an outstretched hand; the computed position rapidly diverges.

We have, however, another hammer in our arsenal: data. Our data is motion capture data, consisting of frame-by-frame limb positions for a variety of human motions. Our key insight for this work was that, by using this data as a prior on limb positions, we could turn the problem into one of differentiation instead of error-accumulating double-integration. We double-differentiated the motion capture data to get frame-by-frame accelerations for each limb position; the data streaming in from accelerometers could be compared to these accelerations, the best match chosen, and the corresponding motion capture clip played.

This insight relies on the natural structure in the movement of the human body; our joints and muscles limit the range of actions we can perform, as well as make the actions we do perform consistently repeatable, to some degree. We are relying on this repeatability when we assume a $1:1$ mapping between the clothing's accelerations and the motion capture positions. A second aspect of the natural structure is the coordination among joints; we rely on this lower dimensionality when using only five accelerometers in our system.
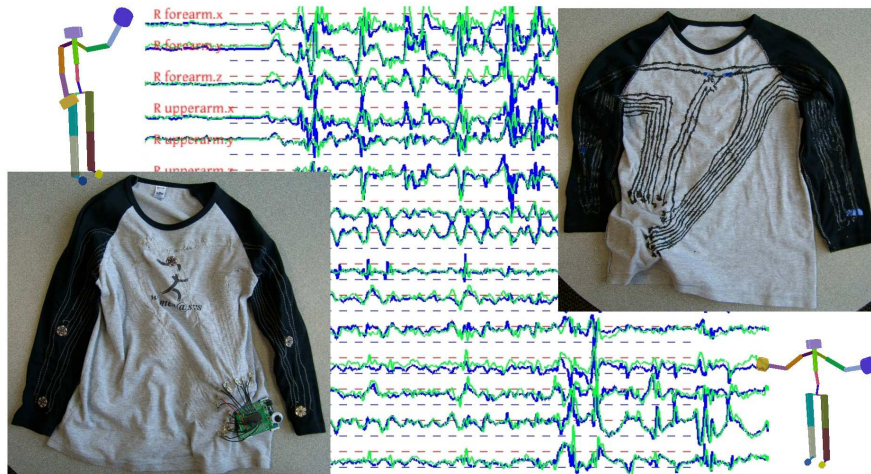
**Figure 2.1:** *Our research takes a step toward clothing as the user interface, creating a performance ani-mation system using low-cost e-textile materials and accelerometers, backed by a database of pre-recorded high-quality motion capture data.*

We call our system "action capture" to delimit its strengths and weaknesses: motion reconstruction works well when the user is performing repeatable, easily recognizable motions, such as martial arts moves or jumping jacks. It does less well on motions where the accelerations are small or do not have a consistent pattern, e.g. the casual gestures of a lecturer.

We used etextile materials to create the physical interface, shown in Figure 2.1. Washable accelerometers were sewn into a long-sleeve t-shirt with conductive thread, and a detachable microcontroller used for con-necting to the desktop computer. The action-capture shirt we created, with five accelerometers sewn in, was made for about $200 using only off-the-shelf parts.

We will next discuss work related to our performance-animation solution, and describe the techniques that we have borrowed from the field of e-textiles. We then detail our hardware construction and our soft-ware system built on a wavelet-matching algorithm. Our results are best demonstrated by video (please see http://graphics.cs.cmu.edu/projects/action_capture/), but we also provide a numerical comparison to simul-taneous motion capture accelerations. We conclude with a discussion of the limitations of our system, and possibilities for future work.

## 2.1   Related Work

The next section describes related work in new hardware for motion capture, real-time performance anima-tion interfaces, and accelerometer-based capture. The section following gives a brief overview of e-textiles.

## 2.2 Capture Interfaces

Researchers continue to build cheaper and more portable hardware for motion capture. Vlasic and colleagues created an impressive, portable system with acoustic-inertial trackers: accelerometers, gyroscopes, and an acoustic subsystem to measure distances between limbs [Vlasic et al. 2007]. The acoustic sensors are necessary to prevent the drift inherent in the double integration of accelerations. Our system avoids this problem by comparing against motion capture accelerations (i.e. not integrating). Raskar and colleagues used photosensing markers [Raskar et al. 2007]. The FootSee system used an XSensor pad to capture foot-ground pressure distributions, and matched these to an existing database of simultaneous motion capture and foot pressure [Yin and Pai 2003]. Our goal in this work is to test the capabilities of off-the-shelf hardware and accelerometers. Thus we use neither custom parts nor more expensive sensors, such as the inertial sensors used in commercial motion capture systems [Moven 2008].

The time required to prepare for a motion capture session has long motivated the idea of capturing motion from video, e.g. [Corazza et al. 2006], [Ren et al. 2005], and [OrganicMotion 2008]. Shiratori and colleagues developed a portable capture system using over a dozen action-sports cameras attached to the body [Shiratori et al. 2011]. Structure-from-motion was used to compute body movement. Chai and Hodgins used a small set of markers and two video cameras to reconstruct full body motion [Chai and Hodgins 2005]. Their performance animation system used a database of motion capture, and at runtime built a locally linear model of poses close to the control signals and to previous poses to find the best match to the current marker locations. We similarly use a database in our matching algorithm, but perform no interpolation.

Subsequent to publication of our original paper, our work was extended by Tautges and colleagues to include interpolation [Tautges et al. 2011]. They built a full-body motion controller using four accelerometers, placing them near the wrists and ankles. Their motion reconstruction system used a lazy neighborhood graph to look up a set of matching motion fragments, and optimized the resulting motion based on those fragments. Our work has also been combined with video: Conaire and colleagues extended our ideas to apply to tennis, using accelerometers and video to classify tennis strokes [Ó Conaire et al. 2010].

Our focus is on reconstructing human motion, but a related problem is mapping the motion of a manipulated object to the actions of a character on the screen. The Swamped! exhibit demonstrated a *sympathetic interface* – an inviting plush toy whose actions were mirrored in the virtual world [Johnson et al. 1999]. The toy was outfitted with gyroscopes, flex, squeeze, and other sensors, and communicated via a wireless transmitter. Dontcheva and colleagues created a capture system based on motion-captured Tinkertoys which were automatically mapped to DOFs (degrees of freedom) of the character being controlled [Dontcheva et al. 2003]. Numaguchi and colleagues built a system to retrieve motion capture using a puppet instrumented with potentiometers and an orientation sensor [Numaguchi et al. 2011].

Gaming with the Wii Remote can be regarded as a performance animation experience [Nintendo 2008]. The Wii Remote uses an off-the-shelf ADXL330 accelerometer, $\pm 3g$, the same used in our prototype. The Wii Remote allows the player to act *with* an object, performing and gesturing as if the object were an extension of his arm. For example, the player swings the Wii Remote as if he were playing tennis or launching a bowling ball. Our system is wearable and captures more degrees of freedom.

Gaming is not the only field to take advantage of the low cost of accelerometers. The medical community has used them to measure gait [Mayagoitia et al. 2002][Lee et al. 2007], recognize falls in the elderly [Boissy et al. 2007], and evaluate physical activity in obese children [Pietilinen et al. 2008].

Similarly, in Wearable Computing, accelerometers are used to identify discrete activities [Lukowicz et al. 2004] [Tapia et al. 2007]. When actions are performed slowly, the orientation reading, rather than the movement, dominates, and poses and gestures can be inferred from these readings (modulo rotation around the axis of gravitation) [Farella et al. 2007][Tiesel and Loviscach 2006][Fontaine et al. 2003]. Lee and colleagues propose a compensation technique for faster motions; they place two accelerometers at different points on a limb, and use geometry to subtract the acceleration due to motion [Lee and Ha 1999]. Realtime visual systems with accelerometers can also give feedback in motion training, in martial arts for example [Kwon and Gross 2005]. Our hardware framework would fit well in many of these applications.

### 2.2.1   E-textiles

E-textile research seeks to merge electronics with textiles to create comfortable and computational clothing. The field is closely allied with Wearable Computing, but takes the "wearable" one step further by not merely attaching computational devices, but *sewing* them into clothing with conductive thread and fabric. Post and colleagues describe projects from the MIT Media Lab, including a fabric keyboard and musical MIDI jacket [Post et al. 2000]. The paper describes the materials and technique of "E-broidery": embroidered textile circuits designed with CAD tools and sewn with conductive thread in embroidery machines. Buechley designed and made commercially available an e-textile kit called the "Lilypad" consisting of small sensors and a microcontroller with sewable leads [Buechley 2006] [SparkFunElectronics 2008]. Her innovations in e-textiles have made the field accessible; she uses the kit as an educational tool to teach electronics to children. We use the commercially available accelerometers and conductive thread from her kit, as well as her sewing construction techniques.

The Georgia Tech Wearable Motherboard project pioneered the paradigm of "fabric is the computer" [Park et al. 2002]. Their Smart Shirt is woven with optical fiber in a programmable network that can detect bullet wounds and transmit readings from other sensors for combat or medical monitoring. Other e-textiles platforms, e.g. PadNET, a hierarchical sensor network for wearables [Junker et al. 2003], and SMASH, an architecture for garments that monitor posture [Harms et al. 2008], have followed.

## 2.3   Methods

We have constructed a hardware and software system to demonstrate the feasibility of using accelerometers embedded in clothing for motion capture. The shirt streams data to the software system in real time, while the software does continuous searching of the motion capture database to control an avatar onscreen. We now describe our hardware construction and software matching algorithms.

**Figure 2.2:** Left. *The outside of the shirt shows the location of the accelerometers (circled in red).* Right. *The traces on the inside of the shirt are laid out to minimize crossings.*

### 2.3.1 Hardware

We used a popular electronics prototyping platform called Arduino, designed to be accessible to people with only minimal electronics experience. Arduino is open source, low cost, and easy to use. It consists of a microcontroller board with input and output pins, and a C++-like programming environment. Boards connect to a computer via USB, serial, or Bluetooth. Our project uses the $35 USB board (Sparkfun SKU: DEV-00666). One could easily substitute a board with wireless connectivity, at higher cost. The Lilypad project sells a sewable version of the Arduino.

Our accelerometer shirt is built out of an existing shirt, conductive thread, five sewable Lilypad accelerometers (built on triple-axis ADXL330, $\pm 3g$), and the microcontroller board. Our first prototype tested the assumption that accelerometers give noisy readings; it had two accelerometers per limb. We found that each accelerometer provided similar readings, and the redundancy was not necessary.

Our second prototype, shown in Figure 2.2, contains one accelerometer on each forearm and each upper arm, and one on the chest. The traces were sewn using conductive thread in the bobbin of a standard sewing machine. Extra thread was left at the start and end of each trace to make connections; on one end the thread was sewn to a standard snap, on the other to an accelerometer. The hand sewing of connections took the bulk of the time. The traces were coated with fabric paint to reduce erosion of the thread when washing the shirt and to prevent electrical shorts. The shirt is form-fitting but not unreasonably tight; the accelerometers are light – each about the weight of a small paperclip – and do not induce any pull on the sleeves.

We like to think of our "hard" hardware as a black box, where the current best hardware solution can be inserted. We used an Arduino USB board with a snapped-on protoshield and a 16:1 multiplexer. Wires are soldered to snaps on one end and header pins on the other which connect to sockets next to the multiplexer, as shown in Figure 2.3. The multiplexer's output is connected to the Arduino board's 10-bit analog-to-digital
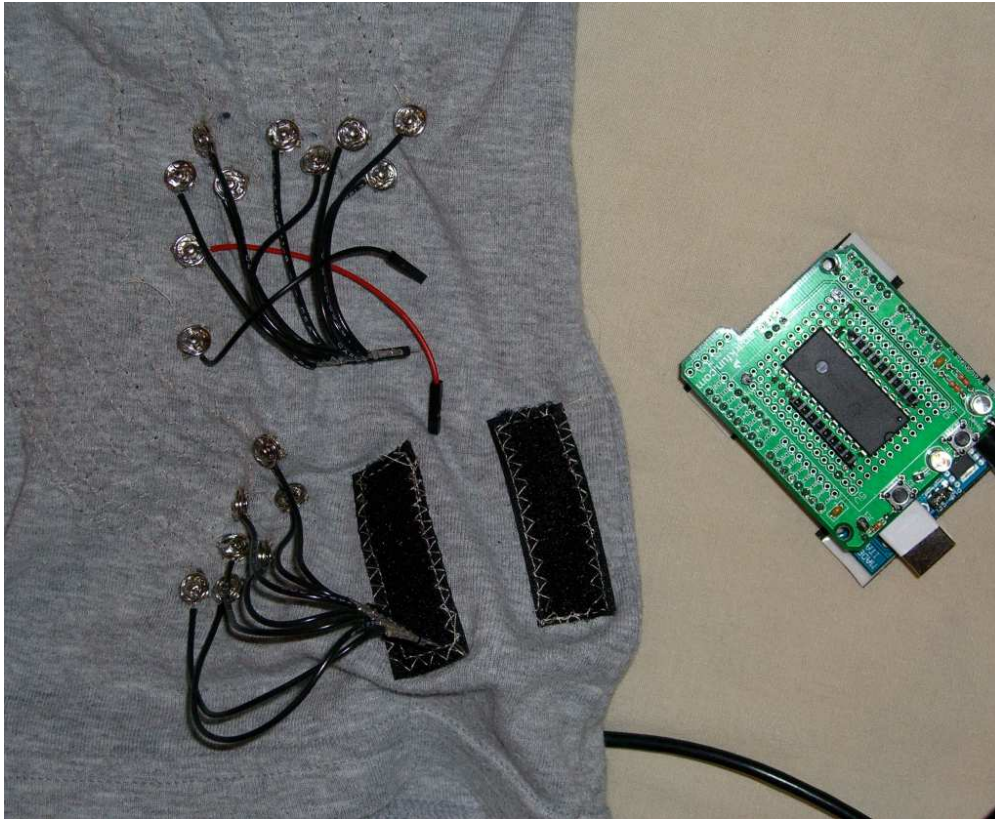
**Figure 2.3:** *A close-up of the "hard" hardware. Wires are snapped onto the shirt, and the board secured with black strips of velcro. The arrangement is secure and removable for washing.*

converter. Twenty lines of code loop over the multiplexer's inputs and send the results, together with a time stamp, to the emulated serial port at 115200 baud. A 5ms delay is inserted at the end of the loop to avoid overloading the desktop computer; the software gets a full sampling of the accelerometers about every 8ms.

The accelerometers are calibrated by pointing them upwards (registering -1g), then flipping them over (+1g). From these numbers we obtain the scaling factor per g, and the zero-g value. This step only needs to be done once for each sensor.

### 2.3.2   Software Matching

Our software system is written in C++ and will run on any Unix-like platform. It consists of three threads which run constantly throughout the life of the program: serial, search, and display. As accelerations stream into the serial thread, they are handed to the search thread, which finds the best match in our motion capture dataset. The dataset consists of $4.5$ minutes of motion, recorded and saved at $120$ frames/second from a Vicon system. The best matching motion is then passed to the display thread, which blends from the currently displayed motion to the new match.
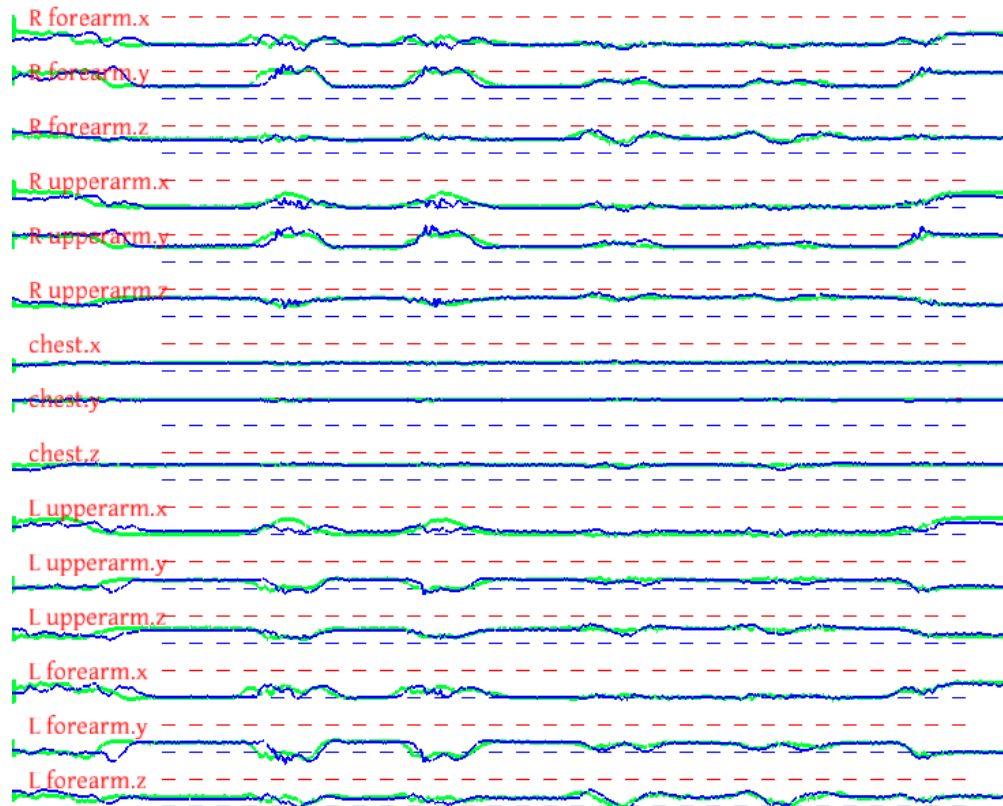
**Figure 2.4:** *Calibration of the sensor locations. Time runs from left to right across the screen, 14 seconds total; graphs of the accelerometer axes are stacked vertically, with the dashed red and blue lines showing +1g and -1g for each axis. Blue lines show shirt accelerations from the actor performing the calibration motion. Green lines show the accelerations of the virtual sensors. The software (or user) adjusts the virtual sensors on the motion capture figure until the green lines overlay the pink, indicating that the shirt's accelerometers and the virtual sensors are aligned.*

Motion capture data is provided in joint-angle space. To convert the data to accelerations suitable for direct comparison to the shirt's accelerometer data, we place "virtual accelerometers" on the motion capture figure and then use the joint-angle data to compute the (position-based) trajectories of these virtual accelerometers. Double-differentiating the trajectories gives the required accelerations.

Next we discuss the calibration process necessary to the correct placement of the virtual accelerometers. We then go into detail about the matching algorithm.

**Calibration**

For each motion capture clip, we compute accelerations at five point locations on the motion capture skeleton; these locations correspond to the locations of the accelerometers on the shirt. We think of this process
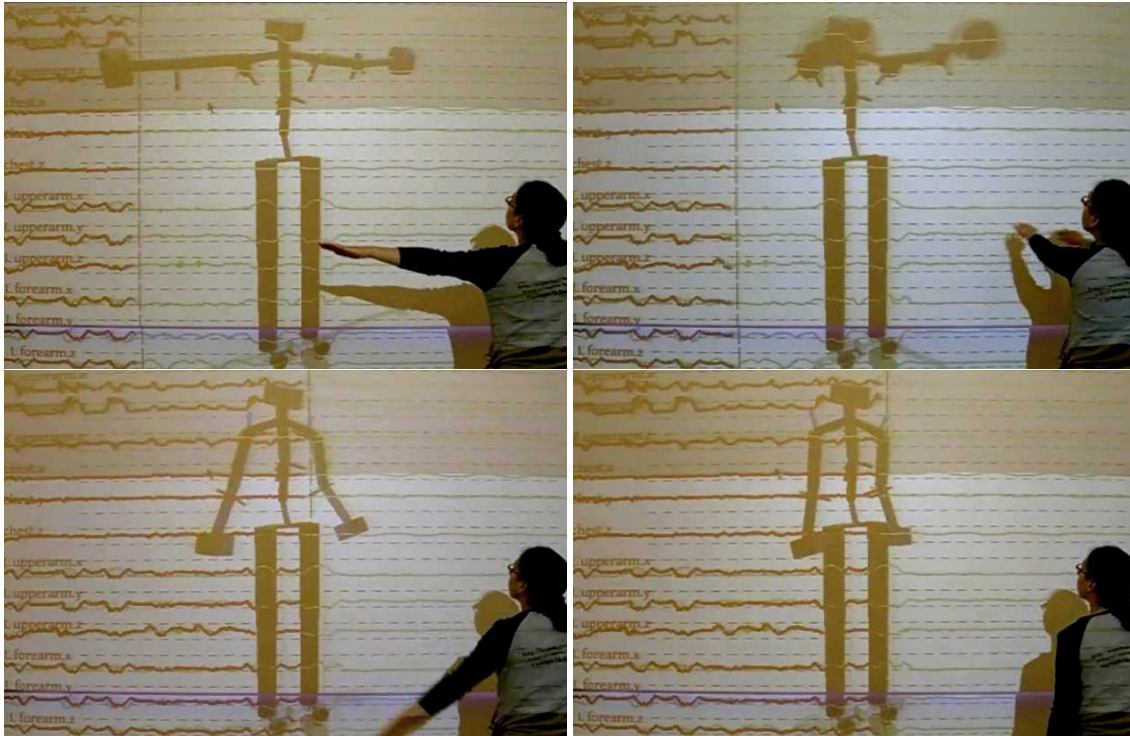
**Figure 2.5:** *Snapshots of a user performing a calibration motion.*

as placing "virtual accelerometers" on the motion capture skeleton, and reconstructing the data these virtual sensors would have generated.

The matching algorithm is sensitive to the placement and orientation of the virtual sensors – the axes of the virtual sensors, especially, *must* be well-aligned to those of the real accelerometers. We thus need to calibrate the locations of the virtual sensors based on the action-capture garment.

We perform this calibration by having the user wear the shirt while mimicking a prerecorded sequence (Figure 2.4 and Figure 2.5) from a displayed motion capture clip. Any motion may be used for calibration; several may be selected if the user wishes to calibrate the accelerometers individually.

An optimization process is then run. The user can manually adjust the virtual sensors until the acceleration graphs match, or set the process to run automatically. In the latter case, the user indicates which limbs contain accelerometers, and the software adjusts the positions and orientations of the virtual sensors until the computed motion capture accelerations line up well with the mimicked accelerations. Because this process needs to be done only once, we opt for a brute force approach, having the software automatically try all angles and positions and then select and refine the best configuration.

**Search**

The motion capture data is preprocessed to speed up searching. The software scans through all sliding windows in the dataset of accelerations, taking sets of length 128 frames – a total of $30,000$ records, each of length 128 x 15-dimensions from the five 3-axis accelerometers. For each set, a Haar wavelet transform is computed on each dimension separately and the first 15 coefficients retained. These 15 x 15 = 225 coefficients are concatenated into a vector along with the joint positions of the pose in the 128th frame. This coefficient and pose vector will be compared against others using the $L^2$-norm. We have found that including more than 15 coefficients from the wavelet transform yields little improvement in the search results. Including pose positions encourages continuity in the matching; we tuned the scaling factor on the positions so that continuity accounts for about 10% of the metric. The number 128 was selected based on fine-tuning; using too short a length loses the context of an action; too long limits the search. Preprocessing takes under a minute.

In the main program loop, the search takes the latest input accelerations, and computes the wavelet transform, as above, from the previous 128 frames. It concatenates onto this vector the joint positions of the last pose displayed, and then performs a linear scan through all of the preprocessed data to find the closest match. This computation completes in roughly 0.060 seconds with our dataset. Larger datasets would likely require a more sophisticated data structure and search algorithm to maintain performance.

Once the search has found a best match, it sleeps until the total time elapsed between searches amounts to 10 frames (0.083 seconds). A delay in searching is key to reducing jerkiness of the motion because it allows time for blending and prevents too frequent switching between matched activities.

When a new match is computed, the motion of the graphical character is blended across 10 frames to the new motion. The new motion clip is started at its 128th frame less 10 frames (the number that will play before the next search result comes in). The frame rate is set to accommodate the processing power of the computer; on a MacBook Pro 2.33 GHz, the system runs smoothly at 30 fps. When displaying the motion capture clips, we hold the legs fixed and zero the root position, as we are measuring and reconstructing only the torso and arm motion.

## 2.4 Results

Our system relies on the repeatability of accelerations, across motion capture sessions and across people. Figure 2.6 shows the motion-capture accelerations of two people doing jumping jacks, using the same virtual sensor calibration. The accelerations are similar, in both the left and right arms.

We used a Vicon motion capture system to test the accuracy of the accelerometer shirt. We performed a motion capture session while wearing the shirt; the results of overlaying the accelerations read from the shirt and computed from the motion capture database are shown in Figure 2.7.

The video on the project webpage shows our action capture system running with a database of $4.5$ minutes of motions such as waving, jumping, and stretching. This dataset corresponds to roughly $30,000$ records
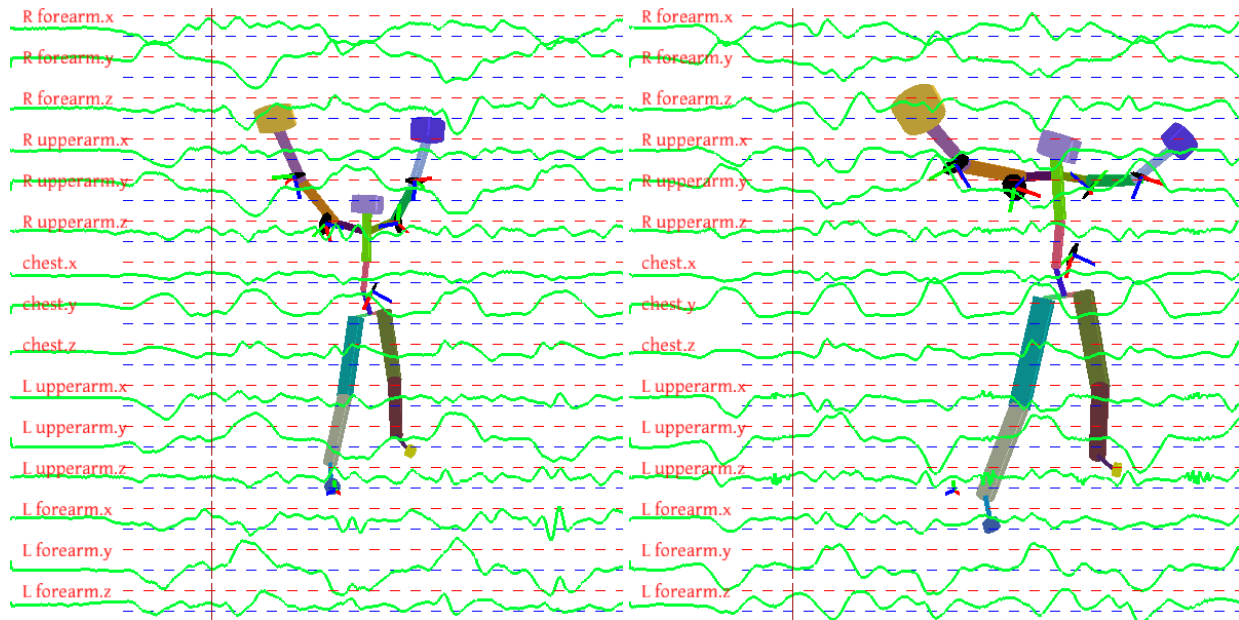
**Figure 2.6:** *Two different motion capture subjects performing jumping jacks. Our approach relies on similarity of accelerations across subjects.*

that the search thread searches each loop; on a MacBook Pro 2.33 GHz, this operation takes about 0.060 seconds.

We have provided an "action capture" system that runs smoothly with a 0.083 second delay, which feels sufficiently reactive. This delay could be reduced to roughly 0.06 seconds, at the expense of jerkier motion and less blending.

In the video, we show simultaneous motion capture and accelerometer-shirt capture of a longer sequence of 35 seconds. The RMS distance in acceleration between the accelerometer readings and the motion capture accelerations is shown in Table 2.1. Alongside is shown the distance between the accelerometer readings and the accelerations of the motion capture segments that our system matched to these readings. Both are in the same range; slight variations occur from the matching algorithm, and from occasional jumps (shoulder pops) in the motion capture data.

In the third column of Table 2.1, we reduce the number of accelerometers used in the reconstruction to the two end effectors, the left and right forearms. We compare the shirt accelerations to the accelerations of the motion capture clips used in this reconstruction. The quality of the motion suffers; as expected, the RMS distance of the end effectors is smaller, but most of the other acceleration errors grow.

As the video on the project webpage shows, our system is good at picking out common static poses, and well-defined actions whose accelerations are repeatable. It performs poorly when actions are performed with different timing or orientation than those contained in the database. Without strictly enforced continuity, occasional jumps occur when accelerations are ambiguous. Nonetheless, we think that our system shows the power of accelerometers for motion reconstruction.
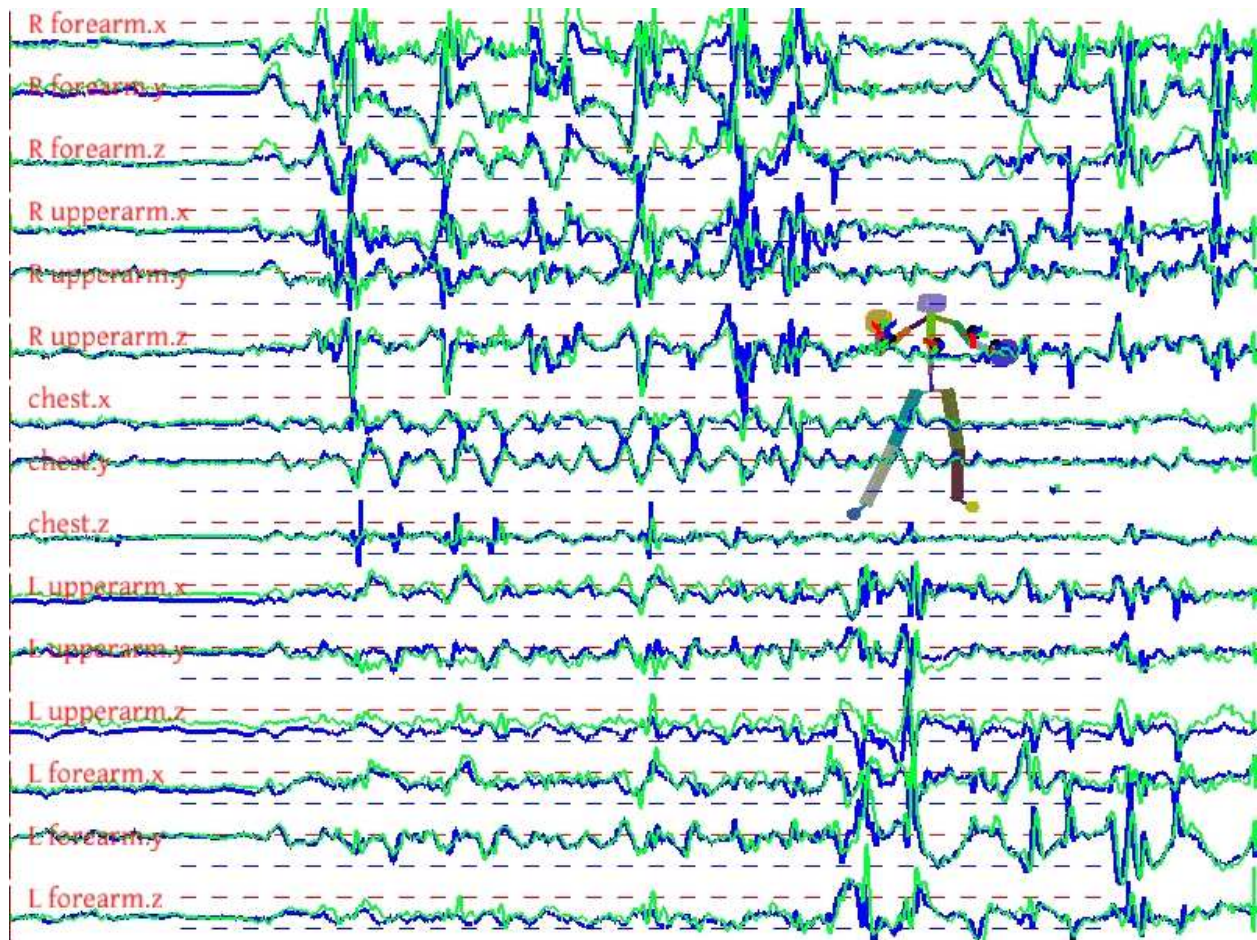
**Figure 2.7:** *Accelerometer readings (blue) are shown overlaid with accelerations computed from simultaneous motion capture (green). The match is close, though not exact.*

## 2.5 Conclusion

A natural question about this work is, will it extend to larger databases? We believe that as it stands, it will probably not, as adding more accelerations will muddle the search. Many poses and motions will have the same accelerations at the short timescale at which we are searching. For instance, when we added a few sitting motions to the dataset, flickering began to occur as poses from these motions were briefly chosen by the search. Tautges and colleagues, in their extension to our work, used more sophisticated algorithms to ensure continuity in position space, and a graph structure that allowed quick searching of a larger database of motion [Tautges et al. 2011].

Using a limited dataset may work to advantage in some situations, however. Video gaming is one application of our system. Limiting the database effectively limits the actions the player can perform to those appropriate to the game. Furthermore, by positioning the sensors on the limbs, we force the gamer to make the actual

RMS error (g's) comparison of accelerometer readings to:

| Axis | motion capture | standard reconstruction | end-effector reconstruction |
|---|---|---|---|
| *R forearm.x* | 0.31 | 0.30 | 0.28 |
| *R forearm.y* | 0.54 | 0.51 | 0.43 |
| *R forearm.z* | 0.22 | 0.24 | 0.22 |
| R upper arm.x | 0.22 | 0.24 | 0.27 |
| R upper arm.y | 0.15 | 0.21 | 0.32 |
| R upper arm.z | 0.19 | 0.22 | 0.28 |
| chest.x | 0.16 | 0.18 | 0.28 |
| chest.y | 0.19 | 0.19 | 0.31 |
| chest.z | 0.14 | 0.15 | 0.26 |
| L upper arm.x | 0.35 | 0.28 | 0.28 |
| L upper arm.y | 0.44 | 0.43 | 0.41 |
| L upper arm.z | 0.55 | 0.49 | 0.53 |
| *L forearm.x* | 0.30 | 0.29 | 0.29 |
| *L forearm.y* | 0.24 | 0.25 | 0.26 |
| *L forearm.z* | 0.25 | 0.28 | 0.28 |

**Table 2.1:**  *A comparison of the RMS error between the accelerometers and accelerations computed from simultaneously recorded motion capture (with manual calibration); accelerometers and the accelerations of motion capture clips used in its reconstruction; and accelerometers and the accelerations of motion capture clips used in its reconstruction, where only the end effectors are used in the matching. The first two columns are similar, as the visual quality of the matched motion is similar to the original motion capture. Matching with only the end effectors (italicized) causes a visually poorer result; the last column shows an increase in the error of the chest and right upper arm.*

full motions, preventing lower-effort "cheating" by make small quick motions with a hand-held controller such as the WiiMote. The limited dataset combined with the speed of our system makes it possible to control virtual avatars, opening it to applications such as online worlds and first-person shooters.

Our system could be used in performance animation, perhaps in laying down rough animation tracks for input into software such as Maya. One could limit the dataset to those types of actions one wished to see, then use our software to lay them out with physically realistic timing. This capability might be useful in the layout phase of creating an animation, for example. The hardware framework we have presented – promoting wearable comfort in addition to cost-effectiveness – would be useful in motion rehabilitation and training.

Our system does not reconstruct the root position and orientation of the character, except for the vertical axis; additional sensors would be needed to implement this. An IMU could be used to provide full root orientation. Position sensing would likely require either careful integration, with drift removed during known configurations, or an external sensor such as a set of cameras.

Several areas of future work present themselves. Additional sensors might allow for more accurate reconstruction of the user's motion. For example, flex sensors would provide approximate joint angles and pressure sensors would allow footplants to be identified. Our e-textile setup would readily extend to a full-body suit; searching may even benefit from the natural coordination of arms and legs to become more accurate. The data compression and search algorithms could be made more sophisticated, perhaps clustering the accelerations according to motion type to get a speed boost and increase continuity. Principal Components Analysis could reduce memory use by eliminating signal redundancy across dimensions.

In conclusion, we have shown that with simple hardware and simple searching, cheap accelerometers can create a workable action capture system by leveraging the power of existing motion capture databases.

# Chapter 3

# Silicone Input Devices

In this chapter, we describe the construction of a series of silicone input devices. The devices have a texture layer designed to allow single-camera tracking of deformations; we iterate on the design of this imposed structure to optimize it for tracking.

Our hands are remarkable tools: we can specify poses and velocities of the fingers precisely and compliantly in three dimensions; simultaneously, we get force feedback in all directions. Our current popular computer input methods – keyboard, mouse, touchscreen – however, make only partial use of the abilities of the hands. We move our fingers in two dimensions to create an input stroke (scrolling on a touchscreen, hitting a keyboard key) and receive feedback generally in a single dimension of click or other haptic sensation.

A significant thread of research in the computer graphics and animation community is higher-dimensional interfaces that harness the power of the hands. Digital analogues of such traditional sculpturing and animation tools as clay [Reed 2009], foam [Smith et al. 2008], and marionettes [Numaguchi et al. 2011] attempt to recapture the force feedback and multi-dimensional input natural in the real world but lost in our current input devices.
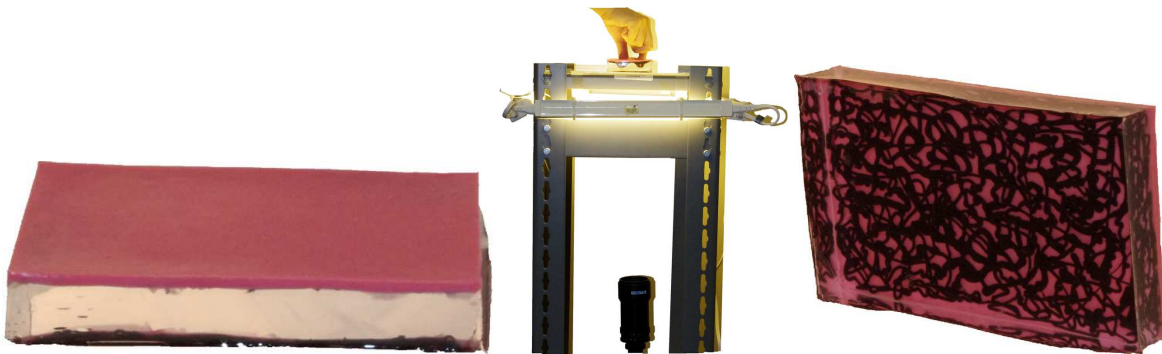


**Figure 3.1:** *Our silicone input device, top and bottom, with camera setup.*

Silicone provides the advantages of multi-finger input and force feedback found in traditional sculpting materials like clay and sand. We develop a new method of instrumenting silicone, embedding a texture inside it, that lets us track arbitrary convex shapes in high resolution using only a single camera. This instrumentation allows us to take advantage of the spatial layout of the silicone, and its ability to capture a range of hand gestures. We put forth several ideas and guidelines for using the shapes we built.

In the following section, we discuss related work in silicone and other physical interfaces. We next describe the system setup, including the tracking software of our collaborators, as the software informs the parameters of the imposed tracking structure. We then list the prototypes built, for each one describing the motivation and lessons learned. We conclude with a discussion of possible applications based on these prototypes' properties.

## 3.1   Related Work

Previous work has explored the various properties of silicone that can be used for tracking. DeForm uses structured light to track an opaque skin on a silicone block, and thus has the size, resolution, and configuration limits of structured light setups [Follmer et al. 2011]. The authors develop several applications, including modeling using sculpting tools. We build on this work, adding the ability to create more silicone shapes for modeling, and simplifying the hardware to a more easily deployable single-camera system.

GelForce [Vlack et al. 2005] and ForceTile [Kakehi et al. 2008] use the ability to embed objects in silicone, creating two layers of blue and red markers in a block of silicone to capture 3D force vectors; our construction method is less labor intensive and more scalable. A similar idea was previously implemented using a single layer of dots on a rubber sheet [Vogt et al. 2004].

Silicone can modify light transmission. PhotoelasticTouch uses silicone's ability to change the polarization of light when deformed to detect user interaction with the silicone; the authors create faces out of silicone and change the projected face image based on the position of the touch [Sato et al. 2009]. Silicone can also be placed atop a frustrated total internal reflection system to magnify the impact of users' touches [Smith et al. 2007].

Tracked deformable surfaces can also be created by using properties of fluids, rather than silicone, as the material. The displacement of black ink is easily tracked with a camera [Hilliges et al. 2008]. Changes in ferrofluid can be picked up with a 2D array of magnetic coils [Hook et al. 2009].

A range of materials have been instrumented or tracked for soft input with the goal of 3D modeling. Illuminating Clay scans a clay surface to create an interactive landscape [Piper et al. 2002]. Digital Clay tracks the deformations of clay using position sensors embedded in the surface [Reed 2009]. Digital Foam explores the interactions possible using a soft sphere of foam instrumented with conductive foam sensors [Smith et al. 2008].
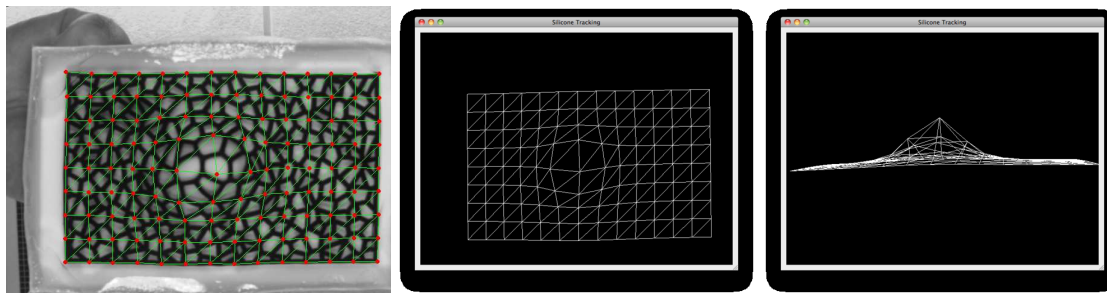
**Figure 3.2:** *Reconstruction of a single press on the silicone device.* Left, *bottom view of a press in the center of one of our silicone devices.* Center, *top view of the 3D reconstruction.* Right, *side view of the 3D construction.*

To summarize, our work is differentiated from previous work in silicone in that it combines, in one system, all of the following advantages:

- The minimal hardware of a single-camera system

- Easy mass manufacturing

- Arbitrary convex shapes

## 3.2 System Setup and Tracking

Our collaborators implemented an affine tracker, using the Lucas-Kanade method for estimating optical flow, to track a regular triangulation superimposed in software on the texture of the silicone device. An example tracked triangulation is shown in Figure 3.2 *left*.

A Point Gray camera mounted underneath the silicone device captures images of the deformation (Figure 3.1). Triangulation is done in near real-time, and fed to a 3D reconstruction algorithm which optimizes a 3D mass-spring system to fit the current, deformed triangulation. An example of this process is given in Figure 3.2 *center and right*.

## 3.3 Silicone Input Devices

The first prototype silicone input devices, built by our collaborators, were made with a rectangular grid for tracking. The grid was made by pumping silicone into a custom-made several-thousand-dollar aluminum mold, in a process conceptually similar to injection molding. Our contributions were to develop cheaper, more flexible prototyping methods, and to iterate on new types of texture for easier tracking.

We next detail the various prototypes we built, describing their motivations and introducing construction methods as they were developed. We divide the next sections into Voronoi Tessellations, Beads, Scribbles, Hemispheres, and Convex shapes.
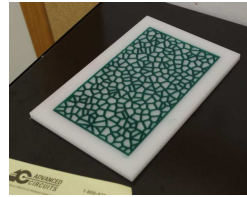
(a) A tessellation is piped into a laser-etched mold.
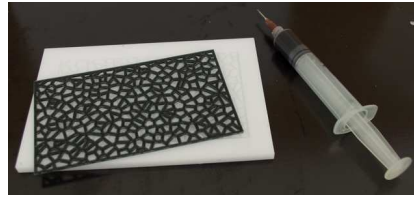


(b) A tessellation is removed from the mold.



(c) An acrylic box is prepared, with silicone painted up the sides. A thin layer of clear gel is poured in.



(d) A tessellation is placed on the clear gel. More gel will be poured in to fill the box.



(e) The acrylic box is cracked off to create the completed device.



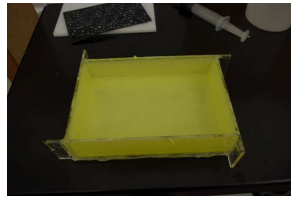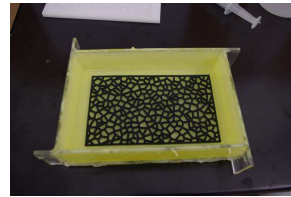(f) Finished device, top view.

**Figure 3.3:** *Making a Voronoi tessellation slab.*

### 3.3.1   Voronoi Tessellations

A tracking algorithm looking at local patches has a difficult time performing error-recovery on a rectangular grid: every local patch looks identical to its neighbor one grid cell over. Instead of a regular grid, we embedded a Voronoi tessellation in a block of silicone. The Voronoi tessellation presents a set of segments meeting at, effectively, random angles, providing easy differentiation from neighboring cells. The tessellation was created by taking a regular grid of points and randomly perturbing them within a small neighborhood, then using these perturbed points as the vertices of the tessellation. This construction simultaneously ensured dense coverage of the entire space, and minimized the presence of severely acute angles in the interior.

The process we developed for building the Voronoi tessellation into a silicone device is shown in Figure 3.3. We laser-cut the mold out of a piece of acrylic using a raster setting which cut roughly halfway through the acrylic. We added colored pigment to Psycho Paint (Smooth-On, Inc.), a soft, stretchy silicone, thinned it
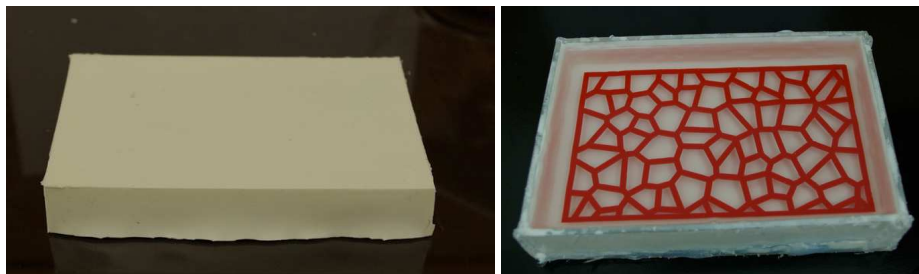
**Figure 3.4:** *Completed device with low-density Voronoi tessellation.*

with solvent, then used a syringe with a needle to pipe it neatly into the mold. Once cured, the tessellation was carefully peeled from the mold.

A $4in \times 6in$ mold box was prepared from acrylic, and assembled with hot glue. Psycho paint was painted in several layers onto the inside of the box, creating the outer surface of the finished device. A clear silicone gel, XP 429 (Silicones, Inc.), was poured into the bottom of the box in a thin layer. The tessellation was placed on the cured gel, and the box filled to the top with more gel ($15mm$ height), completely encasing the tessellation. When the gel was cured, the acrylic box was broken off, and the completed device flipped over onto a piece of glass.

The first-poured thin layer of gel between the texture and the top surface of the completed device acts as a low-pass filter for presses on the device. The thin layer can be skipped, and the texture placed directly on the inside surface of the top of the device; we do this for several of the Scribble devices in section 3.3.3. Omitting the thin layer provides higher fidelity movement of the texture, but makes the tracking problem harder because without the low pass filter, a user's press can affect a smaller area than the size of a tracking triangle.

A second consideration, in addition to filtering, is the density of the tessellation. A lower density, such as is shown in Figure 3.4, necessitates a lower-polygon-count tracking mesh. The lower density is simpler to track, however. Deep presses are less likely to cause neighboring line segments to merge together or overlap at different depths; these tricky situations are especially prevalent when a deep press has horizontal movement. A tradeoff thus exists between the tracking resolution, and the complexity of the tracking software.

### 3.3.2 Beads

A problem with having a stretchable silicone texture is that the lines and intersection angles deform when manipulated, violating the affine assumption of the tracker. We had the idea of using small plastic beads, of half a dozen colors, as texture points. Blob tracking could be done on the beads, with triangulation vertices at each blob's center.

The prototype, shown in Figure 3.5, was built using the same method as the Voronoi prototypes, with the silicone texture substituted with a poured layer of beads.

**Figure 3.5:** *Prototype with beads texture.* Left*, top view.* Center*, bottom view.* Right*, bottom view when being pressed.*

Unlike the silicone texture, which stretched, the beads separated when pushed, leaving large empty areas where a tracker would fail. Additionally, blob tracking proved to be nontrivial, as adjacent beads of the same color would overlap slightly, separate during manipulation, and rejoin.

### 3.3.3   Scribble

Our collaborators' tracking algorithm tracked the patch of texture under each triangle. Their algorithm was not making use of any of the properties of the Voronoi tessellation: straight lines, unique angles, Voronoi cells, etc. Additionally, the use of straight lines was a negative: a texture patch with, for example, a single horizontal segment, could drift horizontally.

We simplify construction and provide better texture by creating the texture as a random scribble. Scribbles have random curves, which prevent texture patch drift, and they can be made arbitrarily dense. A very dense Voronoi tessellation would have thin lines and be too fragile to peel out of the laser-cut mold.

The construction method for the scribble devices is shown in Figure 3.6. Similarly to the Voronoi tessellation, we built an acrylic mold box, and cast a thin layer of stretchy silicone into the bottom. We did not cover the sides, because we had realized by this time that they needed to be able to bulge outwards; otherwise, the stress would be transferred to the bottom of the device, which would lift off the glass plate that the device was resting on. Next we could optionally pour a thin layer of silicone gel to act as a low-pass filter (not shown in the figure). We then scribbled on the silicone using a syringe filled with a pigmented
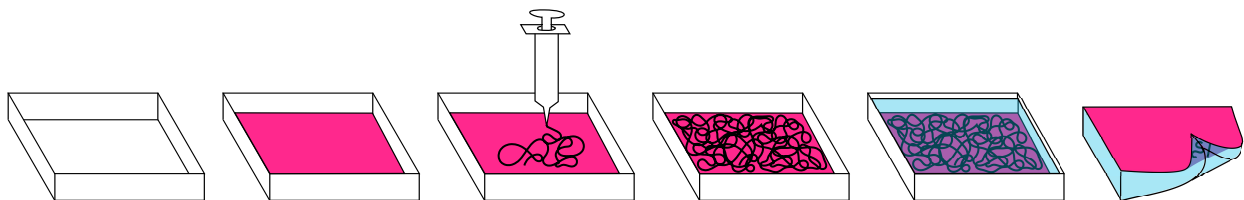


**Figure 3.6:** *The molding process for creating our silicone devices. An acrylic mold box is created, into which a thin layer of silicone is poured (pink). Next silicone is piped in to create a random 2D texture (black). The texture is covered with a thick layer of clear silicone (shown as transparent blue). The final device is taken out of the mold box and flipped over.*

**Figure 3.7:** *A scribble texture free-handed directly underneath the top layer.*



**Figure 3.8:** *A less dense scribble texture, suspended between gel layers.*
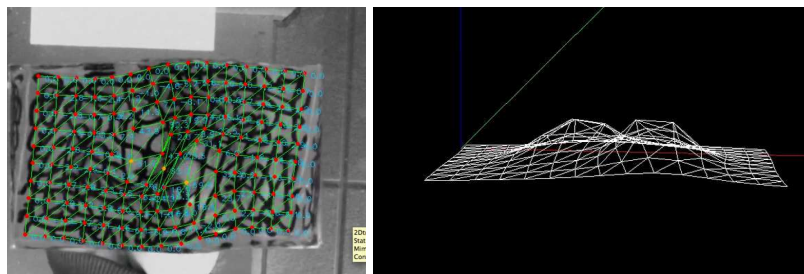


**Figure 3.9:** *Tracking and reconstruction of a pinch gesture, showing merging of the texture lines.*

silicone (Psycho Paint or Dragon Skin F/X from Smooth-On, Inc.) mixed with a thixotropic agent. With a thixotropic agent, the silicone dispenses easily but retains its shape when pressure ceases to be applied, not spreading out across the surface. Next the mold box is filled to the brim with clear silicone gel, cured, and the mold box removed.

Our prototypes are shown in Figure 3.7 and Figure 3.8. We again experienced the same trade-off between density of scribbles (thus, the highest theoretical tracking resolution), and the complexity of tracking as lines intersected. Figure 3.9 shows a pinching gesture with the 3D reconstruction; the merging of lines at the center of the pinch is visible in the camera image.

### 3.3.4 Hemispheres

Tasks in computer modeling such as facial sculpting and animation would more naturally use a hemispherical shape, rather than a flat slab. Additionally, as shown by the evolution of the computer mouse, the hand curves naturally around a curved surface. We thus made hemisphere prototypes, showing methods for regular as well as scribble texture.

**Figure 3.10:** *The filled hemisphere with grid lines.* Left, *blue thixotropic silicone has been piped into the pattern mold; the cover is shown beside.* Right and center, *bottom and side views of the finished prototype.*



(a) Top view of the pattern mold

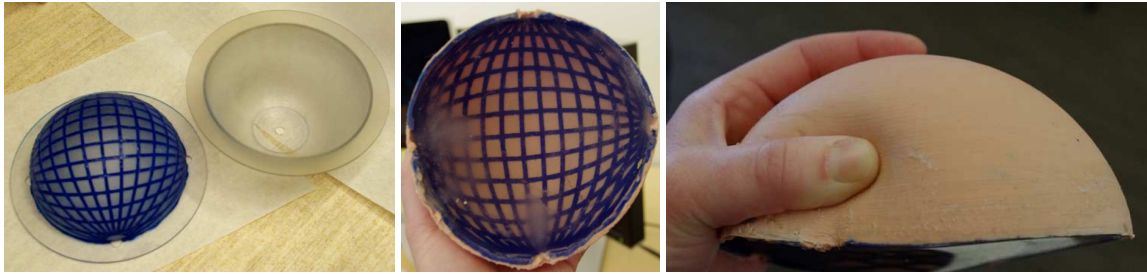(b) Bottom view of the pattern mold

(c) Cover with top hole for piping in the surface silicone

(d) Pattern and cover fitted over each other (vertically offset for clarity)

(e) Cutaway of the assembled mold pieces, showing the cavity for the surface silicone
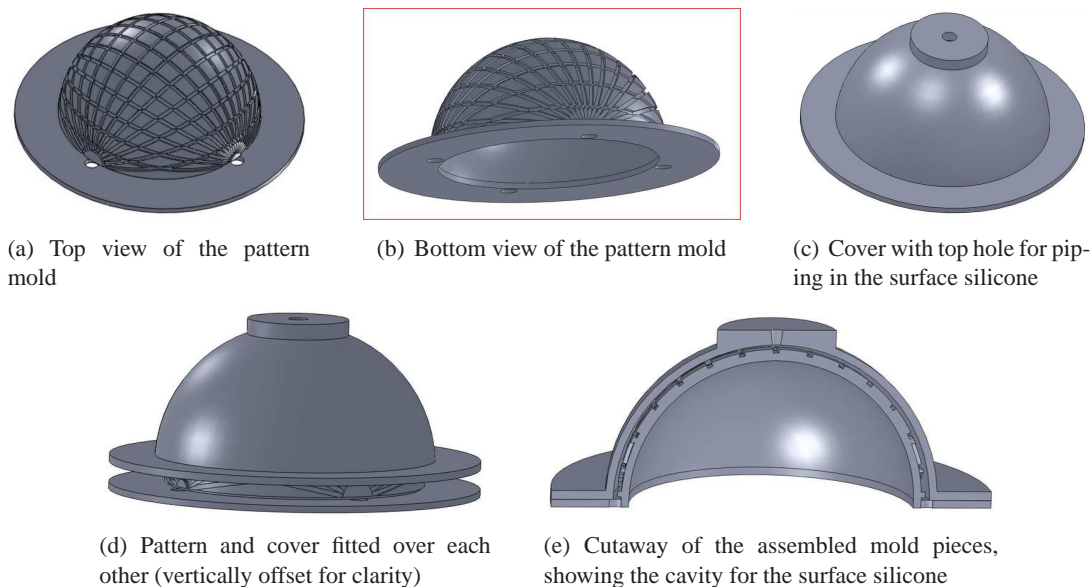
**Figure 3.11:** *Renders of the two mold pieces for the gridded hemisphere.*

The first method inscribes a pattern on the inside of the surface of a gel-filled hemisphere. The prototype is shown in Figure 3.10. To create this prototype, we designed two mold pieces in Solidworks for 3D printing, shown in Figure 3.11. The first piece is a hemispherical mold with the pattern embossed. Silicone with a thixotropic agent, blue in the photos, is piped into the grooves. A cover that fits onto the grooved piece, but slightly too large to fit snugly, is also printed. The cover is centered and clamped firmly on top of the embossed piece, and the surface silicone, shown in peach, is piped in through a hole in the top of the cover, filling the narrow space between cover and embossed piece. As this cures, the thixotropic silicone binds to the surface silicone, and we are left with a shell with surface color on the outside, and the pattern on the inside. This shell can then be filled with the clear gel.

Our second method places a scribble pattern on the inside of the surface, and allows the device to surround a hollow glass hemisphere. The prototype is shown in Figure 3.12. An $8mm$ thick layer of clear silicone

**Figure 3.12:** *The hollow hemisphere with scribble texture.* Left, *the snow globe held suspended while the clear gel cures.* Right and center, *bottom and top views of the finished prototype.*
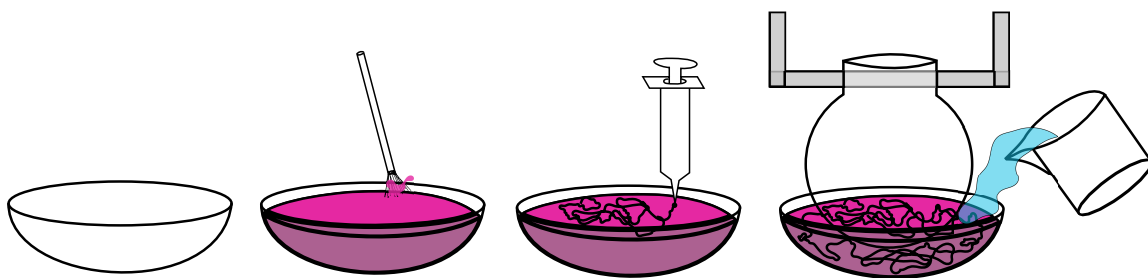


**Figure 3.13:** *Construction steps for the hollow hemisphere with scribble texture. The surface silicone (pink) is painted on the inside of a hemispherical bowl. Thixotropic silicone (black) is scribbled onto the pink silicone. A glass sphere is suspended inside the bowl, and clear gel (shown in transparent blue) poured between the bowl and sphere.*

gel is trapped between the surface scribble and a hollow glass hemisphere. The glass hemisphere allows the insertion of a fisheye lens camera to capture the sides of the hemisphere in higher resolution.

The prototype is made using the process shown in Figure 3.13. We paint the inside of a hemisphere – in our case, a cereal bowl – with the surface silicone, and then scribble onto that with a thixotropic silicone. An optically clear extra-large glass snow globe approximates a glass hemisphere. The snow globe is hung suspended over the bowl, and silicone gel poured into the space between. After the gel cures, the assembly is peeled out of the bowl and flipped over.

### 3.3.5 Convex Shapes

The scribbling process readily extends to making an input device in any nearly-convex shape with a flat bottom. For example, we created the princess-shaped input device shown in Figure 3.14. A plastic shell was printed from a bisected 3D model of the princess. Surface silicone was painted up the inside of the shell. Thixotropic silicone was then scribbled in, and the entire remaining concavity filled with gel to the top.

Because the devices are nearly-convex, if an object of the target shape exists, vacuum-forming would present a more economical method of creating the mold.

**Figure 3.14:** *Princess mold and bottom and top views of the finished device.*

## 3.4   Applications and Future Work

As mentioned in the introduction, the advantages of a silicone input devices are multi-finger input and force feedback. Our input devices lend themselves to applications which make use of these properties.

Computer modeling can make use of both multi-finger input and force feedback. The commercially available PHANTOM force-feedback haptic devices are commonly used for sculptural modeling and detailing [Sensable 2012]. Our devices are cheaper, and provide more natural force-feedback for a modeling task that would be done with the bare hands, such as rough clay sculpting; with the PHANTOM devices, the user gets force feedback only through the intermediary of the pen.

Both our flat and hemispherical devices would be appropriate for modeling. A flat surface might be chosen for terrain and texture modeling; a hemisphere would be more appropriate for curved surfaces such as faces. An interesting question is how specific we should make our devices: we could add in a cylinder shape for muscle modeling, a nose or ear shape for perfecting features, even an entire flat human shape. Further interviews and tests with artists on the production line would be necessary to determine the degree of specificity that would provide the most benefit without being a burden; both switching to a new device and mentally recalibrating to it would cost time and energy.

Unlike in clay sculpting, our devices do not, of course, maintain the shape of the model, but quickly rebound back when released. Experimentation will be needed to determine the best way to use our devices. Options for using them in 3D modeling include absolute positioning (the polygon surface is set to the current deformation), or additive and subtractive modeling (deformation adds or subtracts to the current surface), or mode switching between both. An artist's workflow might have one hand on the mouse and one on our device, using the x-y motion and scroll wheel of the mouse to position and scale the effects of our device, and the mouse click to capture and apply the device's current deformations to the model.

Our devices can detect not only multi-finger, but whole-hand gestures. Multi-finger touches can be used to modify a surface with coordination between fingers, e.g. pulling out the ridge of a mountaintop or smoothing a face model. Gestures could be used to do mode switching or perform actions – for example, the ridge of the hand to cut a model, the heel of the hand to extrude a surface uniformly, or a pinch to collapse a polygon face.

With improvement in the tracking algorithm, our devices could theoretically capture any hand position. Our devices might be used, for example, in a software program for teaching the alphabet in sign language, or for inputting a password set of gestures (thus also incorporating simple biometrics based on the size and shape of one's hand).

Our devices present a "feel" which is not appropriate for all uses, however. For the princess device, we had pictured a fashion application where a child redesigns the princess's dress by drawing on it with virtual colors. When dragging a finger across the dress, however, for it to trackable, one needs to press *into* the silicone. The silicone then induces a drag on the moving finger that feels unnatural for painting. In contrast, the hemisphere's feel makes it ideal for virtually shaping wheel-thrown pottery or playing whac-a-mole, or another application where the manipulation consists of moving an area on the surface of the device primarily downwards. The softness of the silicones used in the device, and the height of the gel, can also be adjusted to give slightly different feels.

The ability to create arbitrary shapes opens up fun possibilities for gaming input devices. Main characters with a flat surface (e.g. butterfly, iguana, or manta ray) could be created in silicone. Manipulating the silicone would manipulate the character on screen: e.g. pressing the iguana's legs alternately to walk, squeezing its mouth to stick out its tongue, etc. Characters could be rotated amongst during the game using the same camera setup.

If the tracking software were extended to do fusion of images from multiple cameras, the scribble method could be extended beyond convex shapes with flat bottoms. Silicone devices could be created to enclose an entire human hand; the low cost of the device would allow personalization for the user. We could also create rugged, water- and child-proof versions of current devices, such as the keyboard and mouse, as all electronics and expensive sensing are moved safely behind the glass or plastic panel holding the device.

We have presented a method for constructing silicone input devices that presents flexibility in construction, resolution, and shape. By incorporating texture in the silicone, we have provided the structure to solve a multi-dimensional capture problem using only a single camera. In the future we plan to continue to refine our prototypes, and to investigate applications.

# Chapter 4

# Sensing Through Structure: Designing Soft Silicone Sensors

Capturing the configuration of an arbitrarily scrunched up piece of soft, conformable material is a hard problem with today's technology. We find structure in the problem, simplifying it into the capture of distinct manipulations of such soft materials. We enumerate the individual types of manipulations a soft material undergoes (bend, stretch, etc.) and create a vocabulary of unit "sensing structures" that each react with a linear displacement to one of these manipulations.

Objects in the physical world are being linked to the cyber world with increasing frequency, whether the objects manifest as input devices to computers, game controllers, or computer-augmented toys. As these computing devices become more prevalent and more personal, users are expecting them to have both a softer look and a softer feel. Boxy desktop computers now sit alongside soft Chumbys [Chumby Industries 2010] and soon, flexible displays; hard actuated toys share aisle space with robotic plush animals; body-worn computing devices are moving from bulky calculator wrist watches to a range of soft e-textile materials; user input devices are pushing the bounds with attempts at soft input, from digital clay [Reed 2009] to reactive fur [Furukawa et al. 2010]. Consequently there is a growing need for new design techniques that allow easy and natural integration of sensing into these emerging categories of soft computing devices.
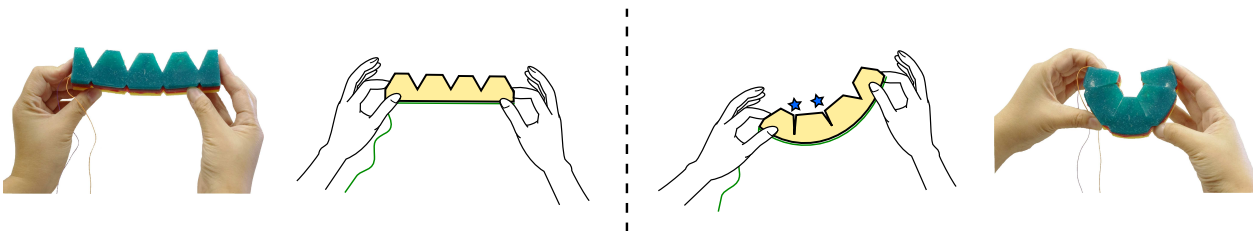


**Figure 4.1:** *Sensing structures, such as our bend structure, above, physically remap a continuous deformation into a set of discrete displacements.*
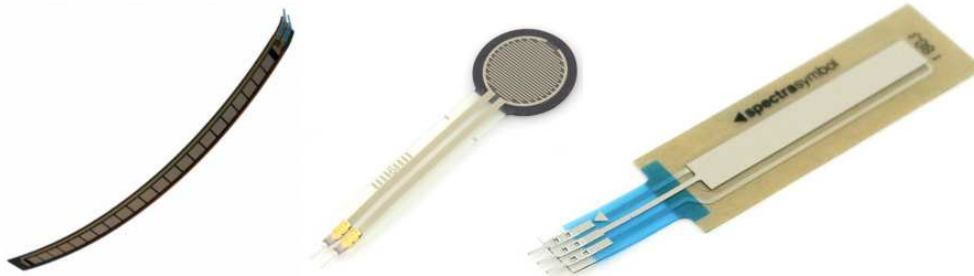
**Figure 4.2:** *Common commercially available sensors – bend, pressure, and position. (images copyright Spectra Symbol and Interlink Electronics)*

Our approach to sensor design, "sensing through structure", exploits the structure in the deformability of the materials that are used to design these soft, flexible computing objects.

We enumerate a vocabulary of manipulations (bend, stretch, etc.) that a material undergoes. Each of these manipulations causes the material to behave in a predictable way, with areas of tension and compression. We present a list of simple "structures" to capture these manipulations.

Figure 4.1 illustrates one such atomic structure, used for capturing bend. We can calculate the approximate curvature of an object by using a set of simple binary switches embedded in the valleys of the structure: when the user bends the object, the valleys are closed and the switches are triggered. We can easily embed such structures into a wide variety of objects.

Sensing through structure is a simple, scalable approach to designing and integrating sensors into soft objects. A sensor can be easily designed to match its application: various basic structural units can be selected in any combination, captured with either digital or analogue sensors, and constructed with shapes customized to respond to specific ranges of motion.

Our structures are easily manufacturable. In particular, we present a construction method of layering electronics between silicone pours to easily create sensors for arbitrary combinations of these deformations.

Sensors built using this approach have a nice physicality. One can see – and feel – the structural units deforming, and readily understand the inherent affordances and limitations of the sensor. The sensing structures can be designed to be clearly visible in the object where they are integrated, so the user can directly see the kinds of manipulation afforded by the object. Furthermore, the sensor's look and tactile feel can be controlled by selecting from a range of soft, pliable materials, and easily manufactured using methods we will describe in this chapter.

In the next section, we describe related work. We then discuss the concept of sensing through structure, and sketch example structures. We detail the construction of several of our sensors and present applications including toys, games, and therapy. We conclude the chapter with a discussion of future work.

## 4.1 Related Work

While sensing through structure focuses on the topological properties of materials, sensors for most common motions – position, twist, bend, stretch, and pressure – rely on monitoring changes in the electrical properties of materials. For example, common bend, stretch, and pressure sensors are often built with a piezoresistive material; changing the length or cross-sectional area changes the resistance of the material. The off-the-shelf sensors shown in Figure 4.2 are built using this method. Another technique for sensing is electro-optical sensing. For example, Zimmerman created an optical flex sensor that senses bending by measuring the amount of light able to pass through a flexible light guide [Zimmerman 1982]. He placed a light source and photodiode on opposite sides of the light guide; when the guide was bent, the amount of light reaching the photodiode decreased. This technology was used in early data gloves. Measurand's ShapeTape makes use of the same concept, building in multiple fiber optic sensors to measure bend and twist continuously along its length [Measurand 2010]. Kakehi and colleagues used reflected infrared light to measure the location and amount of compression in soft materials such as cotton stuffing [Kakehi et al. 2011].

While these sensors work well with traditional hard devices and objects, embedding them in soft and malleable objects presents a number of difficulties. Most of today's sensors are either rigidly encapsulated in metal or hard plastic, or built on a thin plastic backing. These latter flexible sensors can bend and twist, but not shear and stretch. Hence they do not conform well to soft materials such as human skin, textiles, and foam. For example, if the bend sensor is placed at a human joint, e.g. in a data glove, it will shift and slide, interfering with natural motion. These sensors are also somewhat fragile and cannot be creased, which limits their applications and reliability. Most importantly, embedding hard sensors into soft objects alters the tactile, malleable properties of these objects, which is their key distinguishing characteristic from traditional hard devices. Our work introduces sensing solutions using a foam or silicone base, which avoids these problems.

The e-textile community has built flexible analogues of some of these sensors, using the same principles [Buechley 2006]. Sturdy fabric is used instead of a plastic base, and resistive foam or thread is used as the piezoresistive conductor; soft bend, pressure, and stretch sensors have been built this way [Perner-Wilson and Buechley 2010]. For example, Shimojo et al. [Shimojo et al. 2004] created a pressure sensor grid from resistive foam and characterized its hysteresis. Our work provides a flexible and principled way to incorporate these materials.

An alternative approach to augmenting soft objects with sensing is to use external tracking, e.g. vision [Dontcheva et al. 2003], magnetic sensing [Corporation 2010], or RFID [Raskar et al. 2004]. Previous work has imbued silicone with structure to aid in tracking. GelForce embedded markers in silicone to track its deformation using a camera-based system [Vlack et al. 2005]. Cameras also track silicone in ForceTile using ID tags [Kakehi et al. 2008] and PhotoelasticTouch using polarization of light [Sato et al. 2009].

While tracking generally requires less modification of the object being deformed (in particular, the object can be unpowered), it limits use of the object to specific locations. In the sensing through structure approach, all sensing is localized to the device itself, thus making it compact and portable.

Ideas of exploiting structural properties in sensor design can be found in other work. Mannsfeld and colleagues built an extremely sensitive pressure sensor by molding an elastomer with microstructural pyramidal

holes in its surface; these holes squashed when pressed, changing capacitance [Mannsfeld et al. 2010]. Papakostas created a two-dimensional array of force-sensing elements on a polyester substrate with a spiral pattern cut around each element, restricting movement of the sensing elements to the perpendicular plane [Papakostas 2007].

Unlike previous approaches, sensing through structure does not separate the *object* that the user deforms and the *sensors* used to measure it. Instead of adding sensing to pre-existing objects, we start by designing objects with their sensing capability in mind. In a broad sense, our approach can be described as "form equals function" [OrganicMotion 2008][Coelho et al. 2009], where the physical input, interactions, and device embodiment are designed in tandem, in harmony with its material construction. We present the details of this approach in the rest of the chapter.

## 4.2   Sensing Through Structure

Sensing through structure uses the physical changes in the topology of a deforming material to suggest structures for simple multi-location sensing. These atomic sensing structures make up a sensing vocabulary, to be used as building blocks when a sensor is designed.

Figure 4.3 gives some example structures. Diagrams *A* and *B* show bend structure configurations; *A*'s contacts trigger when bent a certain amount; *B*'s trigger as soon as they are bent. Diagram *C* contains contacts on both sides, which trigger in pairs when it is twisted, but only singly when it is bent, thus allowing us to measure both twist and bend with a single structure. Diagram *D* contains a structure for sensing pressure. As the material is compressed, the two sides of the switch meet; the width and shape of the switch allows the activation pressure to be customized. In diagram *E*, the switch is rotated to measure stretch. When the left side of the material is stretched, the left switch triggers as its contacts are pulled apart.

Other structures or combinations are possible. For example, the bend structure in *A* or B may be mirrored to measure two-directional bend, or combined with position or stretch structures.

Our approach simplifies the problem of complex multidimensional sensing, transforming it to the easier and cheaper problem of sensing a set of one-dimensional displacements. The nature of the sensors used for measuring the displacement is not essential; we can choose from a range of sensors, from simple contact switches to analogue capacitive proximity sensors, depending on the application.

The physical structure, also, is customizable to a range of applications. For example, the angle of the valleys in the bend structure should be chosen to measure an amount of bend appropriate to the particular application. Small and large angles could be alternated to trigger at two angles of bend. The shape of the cut in the pressure and stretch units should be altered to fit the dynamics of the base material; for example, we found that a double-convex cut worked best for silicone.

Finally, our structures are designed to naturally guide the user by providing clear physical affordances and constraints so that he can see and feel what can be done with the sensor.

**A. Bend: normally open switches**
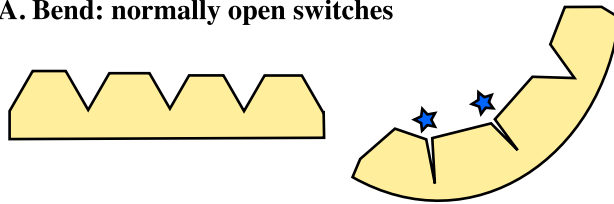
**B. Bend: normally closed switches**
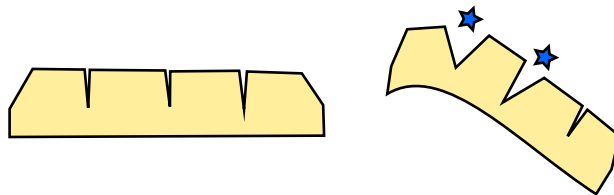
**C. Twist and Bend:**
  **normally closed switches**
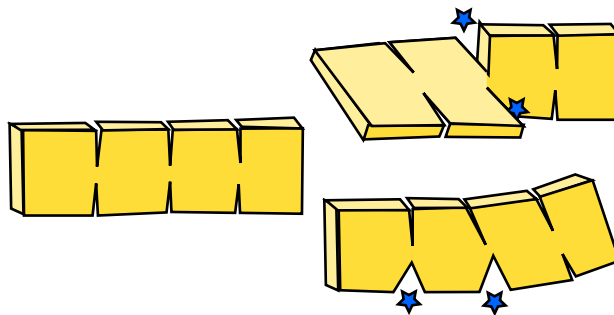
**D. Position/Thresholded Pressure:**
  **normally open switches**
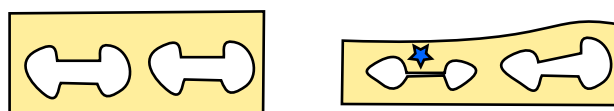
**E. Stretch: normally closed switches**
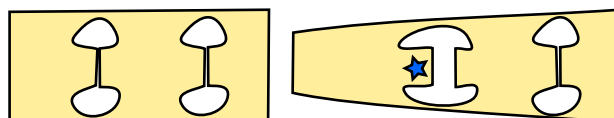
**Figure 4.3:** *Examples of sensing through structure. The structure is in its default configuration in the left column. Stars are shown on the right where switch contacts are toggled when the structure is manipulated.*

## 4.3    Designing Sensors Through Structure

In the following two sections, we describe the design of a bend sensor and a stretch/pressure sensor. To illustrate the generality of our approach, we build the bend sensor with bases of both silicone and foam, using fabric binary contact switches. We use a resistor network to minimize wiring. The stretch/pressure sensor we build out of silicone using analogue magnetic distance sensors whose values are transmitted over an I2C bus.

### 4.3.1    Multi-Location Bend Sensor

We built the bend sensor from the normally open bend structure in Figure 4.3A. We tested bases of both silicone and foam. The silicone is colorful and has a fun feel that demands to be touched and squished, and would work well in sensors that are handled directly by the user. The foam we used has easier overall compression, and thus is appropriate for applications such as stuffed-animal internals. We now describe the construction and wiring.

#### Construction

The silicone bend sensor was constructed from a stretchy silicone, Smooth-On Dragon Skin, with a hardness of Shore 10 and elongation break at 1000%. Stranded 32-gauge wire was connected to a copper polyester conductive fabric for the contact switches.

We laser-cut the mold layers from 4.6mm cast acrylic sheets. Figure 4.4 shows the molding process, with trace layers sandwiched between insulating layers.
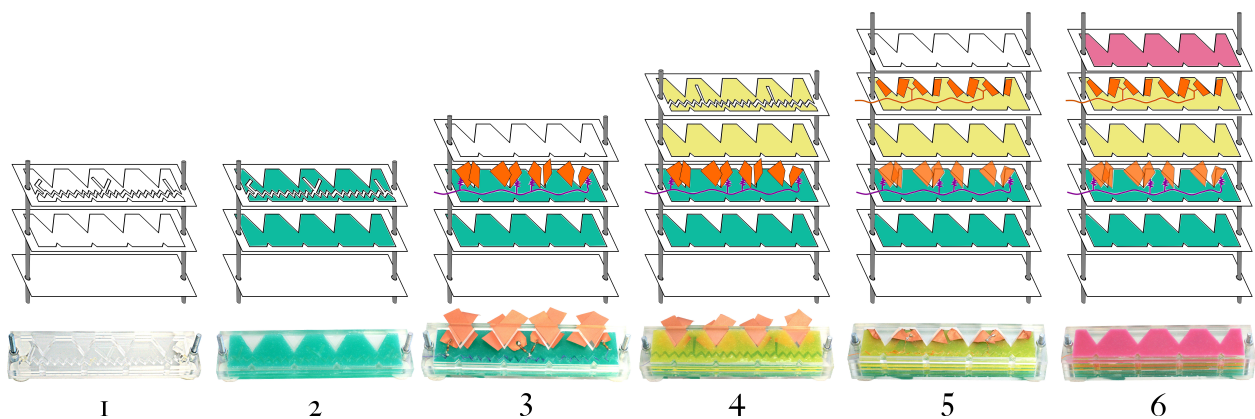


1    2    3    4    5    6

**Figure 4.4:**    *The silicone bend sensor was built with successive silicone pours from back to front, with alternating insulating layers and and trace layers. Wiring is shown with purple and orange lines; contact switches made from conductive fabric are orange polygons.*

In Step 1, the mold is set up for the outside insulating layer and the first trace layer. Trace layers create zigzag tunnels in which to place the wiring. In Step 2, these layers are poured.

In Step 3, the trace mold is removed and replaced with wiring and contact switches. We coil the wire before placing it in the tunnels so that it pushes against the tunnel walls. This friction lock anchors the wires and resistors while more silicone is being poured on top, stopping the wires from floating. The traces also guide the wires into a zigzag pattern, giving them slack for bending. Wires in an early prototype without this feature promptly snapped at the solder joints.

In Step 4, an insulating layer and the second trace layer are poured. In Step 5, the contact switches are threaded through the mold and folded over on top. The unconnected half are connected to the signal tracing being laid.

Step 6 creates a top insulating layer which also anchors the contact switches in place. Opposite ends of the switches are thus embedded between layers of silicone, using the self-stick property of silicone to anchor them firmly without glues or mounting hardware. The conductive fabric we chose does not shred and is tarnish resistant, so this construction creates rugged switches that cannot peel off. Figure 4.5 shows the result.

An important strain-relief feature in our design is the small valleys along the bottom of the sensor, visible in Figure 4.4. These valleys move the center of rotation away from the bottom of the sensor and towards the middle, in line with the wire. They also make the silicone easier to bend, and reduce the amount the wire has to stretch when the sensor is bent backwards. Additionally, all solder joints are moved away from the centers of rotation.

The foam bend sensor was made out of a base of soft foam with a firmness of 5 psi and 25% deflection. Conductive fabric tape with conductive adhesive backing was placed along the valleys to create contact switches; conductive thread was attached to the underside of the tape and then wired through the sensor. We isolated the ground layer and the signal layer between slices of foam. The sensor is shown in Figure 4.6.



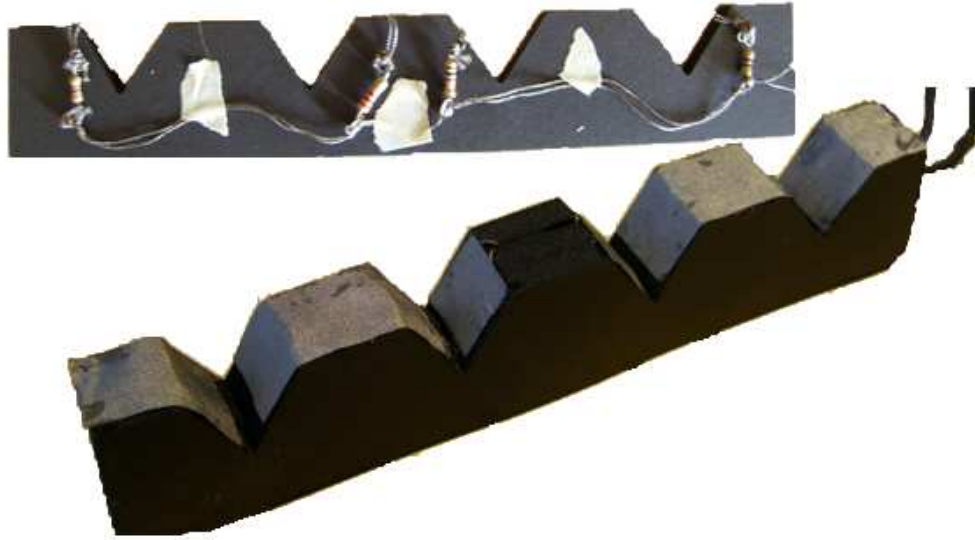**Figure 4.5:** *A completed silicone bend sensor.*

**Figure 4.6:** Top, *the foam sensor was built from back to front; shown is the ground plane wiring being laid on the first layer.* Bottom, *the completed sensor.*

**Electronics**

The challenge in sensor construction with many points of contact is to minimize the wiring as much as possible. The naive approach (Figure 4.7A), connecting each switch to a single microcontroller input, does not scale. We solved the problem by using a binary-weighted resistor Digital-to-Analogue Converter (DAC) technique (Figure 4.7B). With this technique, only two wires come out of the sensor body. Binary-weighted DACs and a similar idea, R-2R ladders, have been around since at least the 1920s in communication systems [Rainey 1926]. They can be found in toys and midi keyboards. Although R-2R ladders are easier to construct to a high accuracy, we chose to use a binary-weighted DAC because it requires fewer resistors in our otherwise soft sensor.

Every switch is connected to its own resistor and acts as a single bit; the DAC combines the bits into a single analogue resistance. Each resistor is a power of two larger than the previous, and thus all combinations can be differentiated by a microcontroller reading this analogue value. We used resistor values of 470 ohms, 1K, 2.2K, and 4.7K.

Using the circuit described above, we can track the state of the switches using the following algorithm. Let the indicator variables $S_i \in \{0, 1\}$ represent the state of the $N$ switches. Using the formula for sum of resistances in parallel, the resistance of the sensor, $R_{total}$, takes the following form:

$$\frac{1}{R_{total}} = \frac{S_0}{2^0 R} + \frac{S_1}{2^1 R} + \frac{S_2}{2^2 R} + \frac{S_3}{2^3 R} + \ldots = \sum_{i=0}^{N-1} \frac{S_i}{2^i R}$$
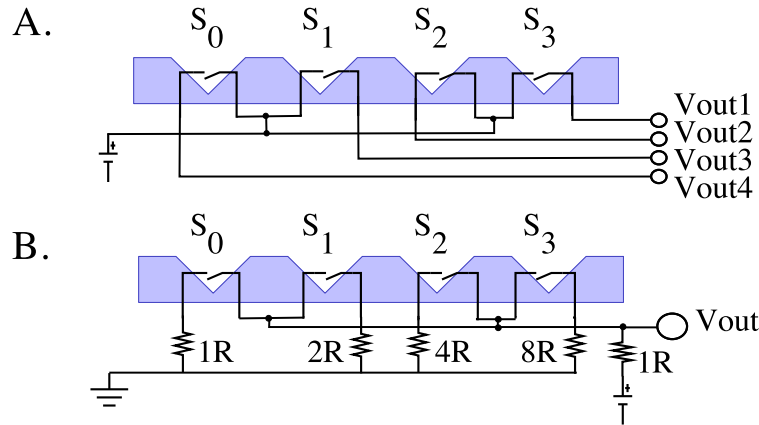
**Figure 4.7:** *Two methods for wiring switches.*

Then the following loop would continuously sample $(S_0, S_1, \ldots, S_{N-1})$, the configuration of the sensor:

1: $R_{remaining} \leftarrow \frac{1}{R_{total}}$
2: **for** $i = 0$ to $N - 1$ **do**
3:      **if** $R_{remaining} \geq \frac{1}{2^i R}$ **then**
4:          $S_i \leftarrow 1$
5:          $R_{remaining} = R_{remaining} - \frac{1}{2^i R}$
6:      **else**
7:          $S_i \leftarrow 0$
8:      **end if**
9: **end for**

This method, however, is subject to error; small variations in resistance could result in the "$\geq$" incorrectly evaluating false, and a completely wrong answer being returned. In practice we found it safer to precalculate the 16 resistance values expected from all different combinations of switches, store these as integer values in the microcontroller's memory, and in real time find the closest.

## 4.3.2 Analogue Stretch/Pressure Sensor

The concept of sensing through structure encourages any sensing mechanism to be used to measure the deformation in a structural unit. Up until now we have been using only binary contact switches, appreciating the elegance of using a resistor DAC for the wiring. It is possible, however, to use analogue sensors. We built a prototype stretch/pressure sensor which uses magnetic distance sensors to monitor the width of four oval holes along its body.
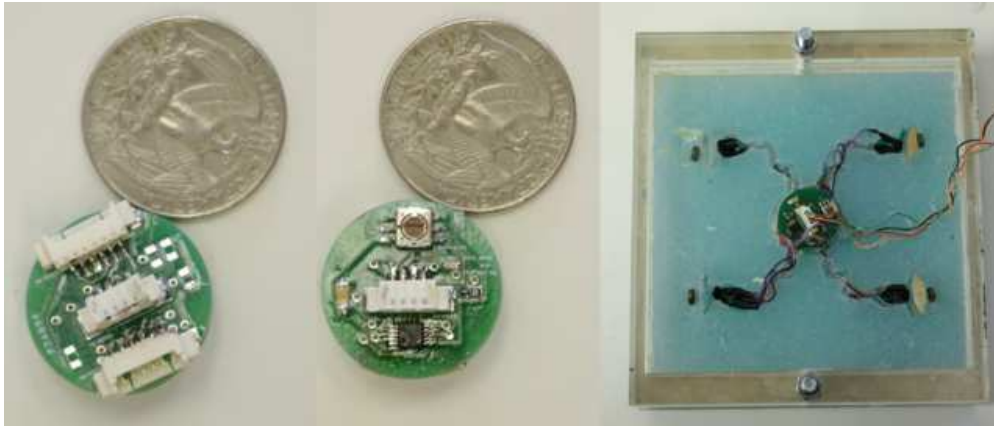
**Figure 4.8:** Left, *top and bottom views of our I2C-bus PCB.* Right, *a snapshot of the molding process with the PCB placed on the bottom layer of the analogue stretch/pressure sensor.*

**Construction**

For the silicone stretch/pressure sensor, we wired an SS49E magnetic hall sensor to each of the four ovals, and placed small rare earth magnets on the other sides. The setup with zigzag traces is shown on the right of Figure 4.8. Silicone was then poured over both the electronics and the magnets, encasing them completely.

The ovals allowed about 15mm of travel, a range trackable by the magnetic sensors. The sensor could thus monitor both stretch and pull, shown in Figure 4.9.

**Electronics**

To minimize wiring, we switched to a bus-based system. We designed a PCB, shown in Figure 4.8, which takes four analogue inputs and transmits them on an I2C bus using a TI ADS1015 chip. The boards are designed to be chained together. The four bus wires connect to the top of the chip (left image) and come out the bottom (center image). An I2C address selector switch is located on the bottom of the board. Any analogue sensors can be connected to the connectors on the top of the board.

## 4.4   Applications

We evaluated the sensing through structure approach by building a computer game, two toys, and an interactive cellphone case. We present these designs, and discuss the potential applications of our approach in other areas.

**Figure 4.9:** *With analogue magnetic distance sensors, both stretch and pressure can be tracked.*
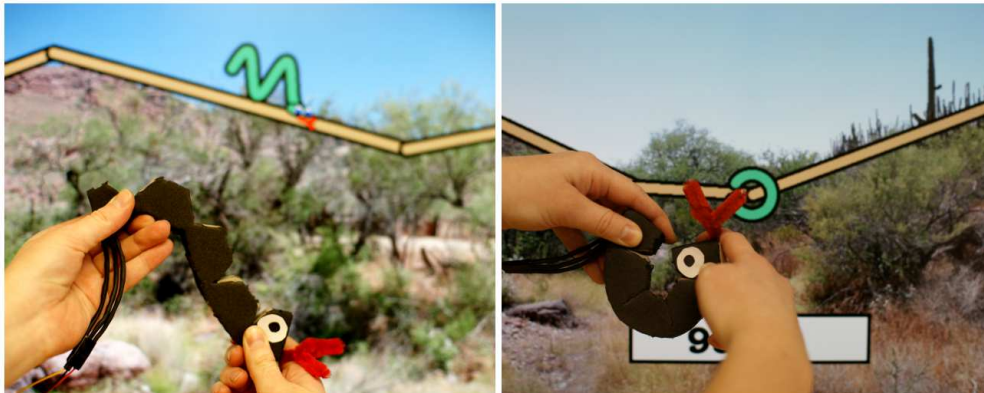


**Figure 4.10:** *The bend sensor as snake controller.*

### 4.4.1 Games

Sensing through structure can be used to design inexpensive interfaces to video games, in instances where they have the right tactile feel. Similar to the game BopIt! [Hasbro 2010], one could combine a multitude of structures to create a highly customized user interface.

As an example, we created a game based on the American folklore legend of the hoop snake. This snake would place its tongue in its mouth so it was shaped like a wagon wheel, then roll down hills chasing after its victims. In the game, moving the snake by undulating the snake sensor in an s-curve shows off our sensor's ability to recognize complex gestures using only binary sensing. The speed of undulation controls the speed of the snake onscreen. Folding the snake sensor into a hoop to roll the snake down hills demonstrates using our bend sensor as a configuration sensor. Figure 4.10 shows our game in action. We tried our game with
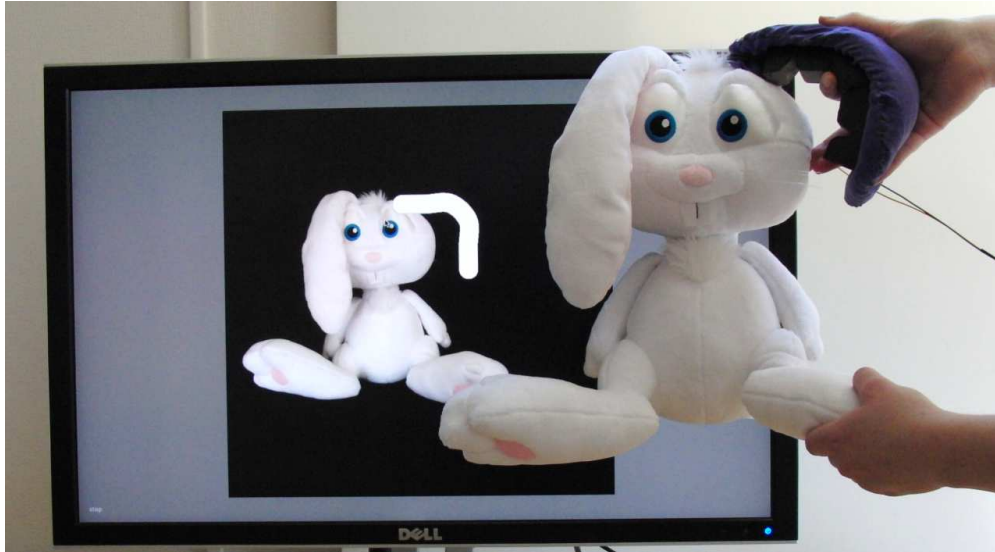
**Figure 4.11:** *A bend-sensor rabbit earmuff.*

both the foam and silicone bend sensor as the snake.  An informal user study showed that the bend sensor was easily understood as a controller, and very entertaining.

### 4.4.2   Toys

An increasing connection is being made between toys and computers.  A good example of this trend is the popular Webkinz [webkinz 2010], which shows how a commercial success can be created using even a nebulous connection between real and virtual toys.  Research projects such as Huggable [Stiehl et al. 2005] and Swamped! [Johnson et al. 1999], and commercial animatronics such as the Pleo [Innvo Labs 2010], show how sensors can make a toy more interactive and compelling.  In Chapter 5 we will separate out traditional stuffed animals for further analysis, arguing that our method should be used to combat the trend of adding hard electronics to stuffed animals.  Here we give two quick examples of other uses of sensing through structure in toys.

Our sensing structures fit perfectly sewn into fabric doll clothes, and achieve a good price / performance point. A proof of concept is shown in Figure 4.11. The rabbit's earmuff contains our bend sensor, allowing the rabbit's virtual double to mimic ear poses.

We also designed a custom toy, which we call "Cat Stretch". The cat-shaped toy has embedded in it four of the normally closed stretch structures from Figure 4.3E, two along its body and one each along its ear and tail.

In contrast to the bend sensor, the cat stretch sensor was built from the inside out. Trace layers were poured around a central insulating layer and small slits made through the layers with a sharp knife.  The contact switches were threaded through the slits and wiring laid around this core layer, as shown in Figure 4.12. As
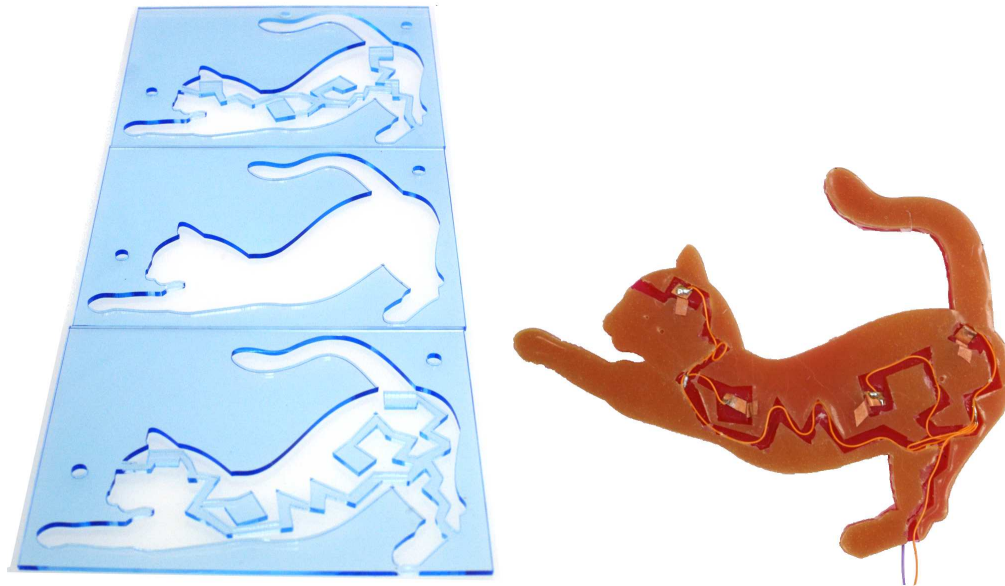
**Figure 4.12:** *Middle mold layers and central core of the cat stretch sensor.*

each outer insulating layer was poured, the cuts were continued with a sharp knife. Figure 4.13 shows the result. When the cat is stretched, the two sides of a given contact switch part, sending a unique resistance to the microcontroller.

Our cat-shaped stretch sensor is a toy simple and cheap enough to be a giveaway, and could be used as a stress toy, or for computer-mediated play. Sensing through structure works well with Cat Stretch: instrumenting the silicone to get complete knowledge of the forces on it would be a difficult problem, but here a few switches are enough to roughly encode the state of the system given the affordances of cat stretching.

These two example toys, both using binary switches, are also well suited to the interactions between toys and children. Children often push toys to the extremes of their (joint) limits; cheap binary sensing is thus appropriate. Also, our sensors provide more control by having tactile, built-in limits, thus eliminating frustration from ambiguity in sensor use.

### 4.4.3 Personal Electronics Accessories

To demonstrate the "form equals function" approach of sensing through structure, we designed an iPhone® case whose physical affordances embody the intended interactions with the cellphone.

The interactive cellphone case was built out of silicone in the shape of a guinea pig, using the same methods and materials as the previous sensors. To interface with the phone, we took apart a pair of iPhone® earphones with remote, soldering wires directly to the contact pads of the remote's three buttons. These wires were then connected to the switches inside the case.
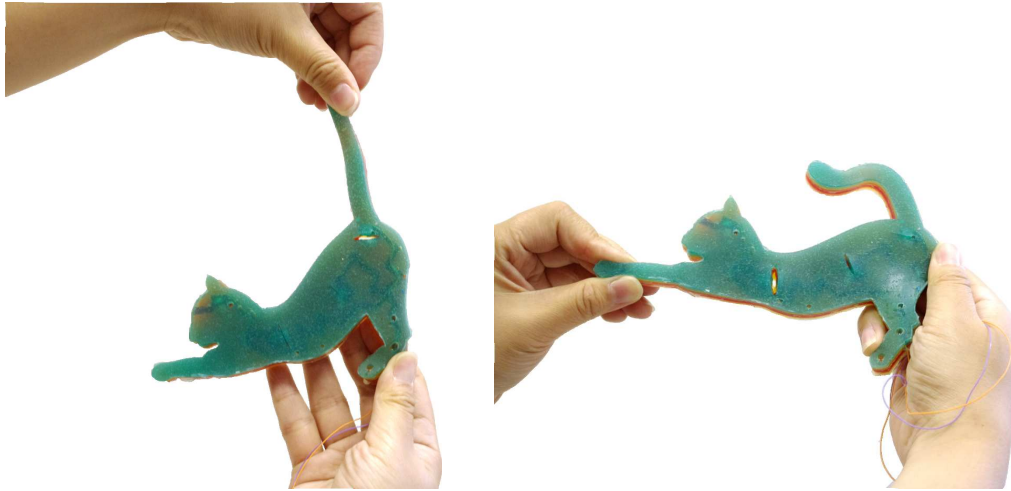
**Figure 4.13:** *Cat Stretch: The cat has stretch switches around its body.*

In the guinea pig case, a bend structure is located in each of the two sets of fur. The fur along the length of the phone demands to be stroked downward, as shown in Figure 4.14; doing so lowers the cellphone's volume. In contrast, stroking the upward-facing fur raises the phone's volume. A pressure structure is located in the guinea pig's button nose; pushing it controls play/pause.

### 4.4.4   Other Uses

Many other areas exist where sensing through structure can be effectively used. For example, sensors designed with our approach could be useful in wearable computing. Stretch and pressure formulations could measure movement of the torso. Our bend sensor nestles into the curve of a finger, and would work well in a data glove.

Computer modeling presents another possible application. An analogue version of the bend sensor could work well for curve editing. Alternatively, a squishable touchscreen made out of the stretch / pressure sensor could be used for 3D surface modeling. Because our sensors are cheap and quick to home-manufacture, custom versions could be created for control of individual animation riggings.

Our sensors have also generated interest from an autism therapist. She was excited by their potential ability to collect child play data while being non-threateningly soft and colorful. Play is used for both assessment and intervention in autism [Wulff 1985], but is hard to analyze [Baranek et al. 2005]. Recent work by Westeyn et al. [Westeyn et al. 2008] instrumented several plastic toys with sensors, but found the form factors of the toys too general to elicit specific actions. Our sensors' distinctive affordances could encourage a child into a particular action. The bend sensor could test pose mimicry; the stretch sensor, strength. We hope to explore work in this area in the near future.

**Figure 4.14:** *A user turning down the volume with the cellphone guinea pig case. The diagram of a mold slice shows the routing to the three switches.*

## 4.5 Discussion

People found our sensors to be engaging and liked their softness. The silicone bend sensor was particularly successful – the colors drew the eye, and everyone who picked it up spent a minute or two just bending it into different configurations, testing its limits and enjoying its tactile sensation.

In general, our game controllers, toys, and cellphone case were easily understandable. The snake controller proved amusing and its gestures masterable. The cat toy looked fun, but its interactions were not as easily grasped by people; designing the outer insulating layer to reveal more of the switch contacts would make its affordances more apparent. People enjoyed the natural motion of stroking the interactive cellphone case. At its current scale the case could be part of a speaker stand; with miniaturization and covered magnetic switches it could be a practical cellphone case.

The exposed contacts in our sensors beg the question of whether our sensors will remain rugged over time. The fabric contacts we used are tarnish resistant, and the silicone durable. Accumulated dirt on the contacts, however, would increase the resistance of the material, breaking the resistor DAC calculations. A calibration routine where the user closes (or opens) each switch individually would give the computer enough data to recalculate the resistances. If the sensor is being used in an application which expects certain configurations, changes in resistance could be tracked over time transparently to the user. As long as the contacts degrade somewhat uniformly, maintaining an order of magnitude difference in resistance, the sensor will continue

to work. Alternatively, the contacts could be protected by encasing the sensor in an outer layer of foam or stretchy fabric, possibly filling the holes in the structural units with a piezoresistive foam.

## 4.6   Conclusion

We have presented sensing through structure, an approach that uses the topological, rather than electrical, properties of objects and materials to design and construct sensing solutions for measuring motion. The approach is general: arbitrary combinations of sensing units can be constructed using our silicone method, and either analogue or binary measurements can be made. We have used only soft materials and low-cost manufacturing methods, which can be replicated by anyone interested in creating custom sensing interfaces.

Our sensors enable interactions ranging from dynamic, time-based gestures, to static poses and configurations. We have shown both types of input in the hoop snake game, and static poses in the toy proofs of concept. Dynamic gestures can be more complex with multi-location sensing. Onscreen visualizations can tween between configurations, covering for the lack of more expensive continuous sensing. In the static case, the conformability of the sensor lets the user hold it in a pose with a comfortable amount of tension. The sensor can also be designed, and the contacts placed, to capture poses with maximum robustness and efficiency.

Our approach emphasizes building multi-location sensors with discrete inputs. This approach works well for toys and games, where the physicality of the device can be more important than its accuracy. In our work, as switch contacts are made, the sensor provides tactile feedback, making gesturing more efficient. A traditional bend sensor, for example, reveals little about its limits.

In the future we would like to explore new applications of our sensors. We are particularly excited about applying our results to child therapy, customizing our sensors to children's needs. We would also like to exploit a particular facet of sensing through structure in games and toys: its transparency. Gross argues against the increasing trend of technology-enhanced children's toys being "black boxes" whose interior functioning is hidden from the user [Gross and Eisenberg 2007]. Our sensors, when made with clear silicone, completely reveal their inner workings. With unambiguous use and accessible construction, they present a viable alternative to the current trend.

# Chapter 5

# Soft Sensing Skeletons for Stuffed Animals

Stuffed animal toys are increasingly being instrumented with electronics, to the detriment of their traditional soft feel. In this chapter, we argue for solving this problem by taking the soft sensing vocabulary from Chapter 4 and incorporating the structures into sensing foam cores for the stuffed animals.

Nothing compares to the agelessness of a favorite stuffed animal – dragged around on the ground after a toddler, brought to imaginary life by a preschooler, kept as a companion through middle school, ignored through adulthood but then dug out of grandma's basement to show the next generation. The toy's fabric is patchable, stuffing replaceable, and charm indestructible.

A current trend is to add technology-supported interactivity to stuffed toys. Such toys follow one of two paths: creating a virtual double of the toy that is manipulated using a standard computer; or adding hard switches, motors, speakers, and batteries to the inside of the toy.

The first path, followed by, for example, the successful Webkinz[TM] brand, maintains the charm of the stuffed toy, but fails to create a story connection with the virtual version. The second path, found along toy shelves in department stores and including such hits as the Furby[TM] and Pleo[TM], creates an interactive experience, but with sacrifices: the animal is no longer lovably soft and rugged, and the electronics are breakable and quickly outdated.



**Figure 5.1:** *With a soft sensing skeleton, our frog has eyes that light up when "hopped", yet it can withstand the tough love given to a traditional stuffed animal.*
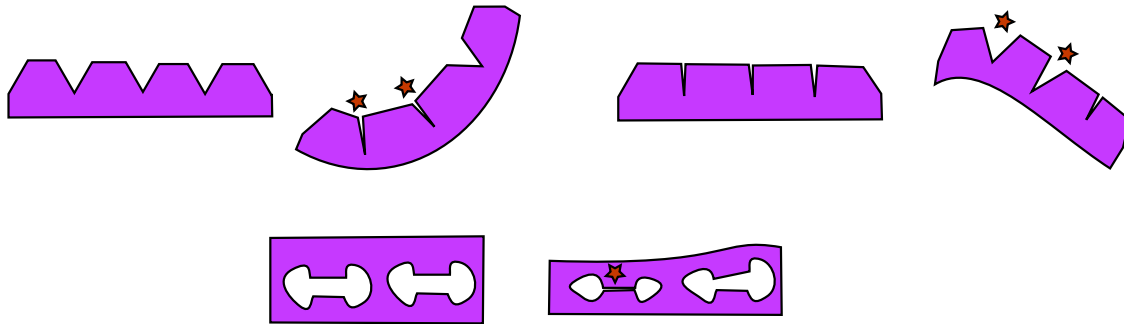
**Figure 5.2:** *Continuous bend and stretch of a conformable material can be recorded using discrete contact switches, with contact pads located at opposite sides of cuts in the material. The figure shows sensing of bend using both normally open and normally closed switches (top pairs), and sensing of pressure (bottom pair).*

We propose a third path, which capitalizes on the strengths of each: soft sensors inside the animal for interactivity, hard output and actuation in the virtual world to maintain softness and allow upgrades. We argue that until the technology exists for soft motors, power sources, and displays, such components should be transferred, as much as possible, out of the stuffed animal and into a computer. The one exception we note is stuffed animal eyes, which are traditionally hard, and thus ideal for replacing with a component like LEDs. In all cases, a stuffed animal should feel like a traditional stuffed animal.

In Chapter 4, we created a taxonomy of simple "structures" for sensing jointed movements in a squishable material (examples in Figure 5.2). The contribution of this chapter is to show that such structures, when built of e-textile materials and integrated into a foam core, make possible our vision of soft interactive toys. We build cores for a dolphin, an elephant, and a frog, incorporating structures and designing playtime applications based on the affordances of each stuffed animal. We instrument the structures with fabric contact switches, which are simple enough to be repairable (or even customized); in addition, the entire foam core can be cheaply replaced.

We next describe representative work in toys and soft sensing. We detail the construction and applications of three stuffed animal skeletons. We conclude with a discussion of the possibilities and limitations of our approach.

## 5.1   Related Work

Johnson and colleagues introduced the concept of a "sympathetic interface", building a plush doll that controlled a virtual character at the behavioral, rather than motor, level [Johnson et al. 1999]. We build on this work, but use e-textile materials to obviate the need for the hard armature found in their doll. The ActiMates[TM] Barney[TM] showed that children accept an interactive toy transitioning between different roles [Strommen 1998], encouraging the potential for acceptance of our method, where the animal becomes interactive / non-interactive based on context.

Posable puppets have been used as input for performance animation, from the 32 degree-of-freedom Monkey skeleton constructed by Esposito and colleagues [Esposito et al. 1995], to Mazalek and Nitsche's soft cactus marionette made from foam-filled cloth, whose motions are captured with hard embedded accelerometers [Mazalek and Nitsche 2007].

We build on the current work in the etextile community for creating soft sensors using conductive materials, primarily for application to light-up and responsive clothing and accessories [Buechley 2006]. Although in our work we chose to use contact switches, Perners-Wilson and Buechley describe other applicable soft sensors [Perner-Wilson and Buechley 2010]. Similar materials are used in Digital Foam, an innately soft input device using a grid of tubes of resistive foam [Smith et al. 2008]; and Shimojo and colleague's multipoint pressure sensor, using a stitched foam grid [Shimojo et al. 2004]. We chose not to use conductive foam because of its unreliability and slow recovery time, which make it inappropriate for the time-sensitive applications we envision. The FuwaFuwa sensor module, which tracks reflected IR light, is another way to instrument a soft object, but again introduces a hard element [Kakehi et al. 2011]; our goal is to introduce no hard pieces.

## 5.2 Soft Sensing Skeletons

We next describe our three sensing skeletons – dolphin, elephant, and frog – and the application each controls. Each skeleton example highlights a method of merging soft sensing into a traditional stuffed animal without destroying its softness. The dolphin with bend sensors in its tail demonstrates using offboard computation. The elephant hides a microcontroller in its head, because children usually squeeze the body. The frog with light-up eyes shows a self-contained system, where the LEDs replace the eyes, a traditionally hard part of a stuffed animal.

### 5.2.1 Dolphin: bend structures in the tail

Our first example of a soft sensing skeleton is a dolphin core with bend structures on each side of its tail. The tail is used for navigation in a side-scrolling game.

**Construction**

We carved the core out of an open cell polyurethane foam block, .32 psi (25% deflection). V-cuts were made at the top and bottom of the tail for the bend structures, as shown in Figure 5.3. The cuts created a weak joint; we reinforced it with a thin piece of rubber slotted through the center. Conductive fabric tape, which stuck well to the foam, formed the contact switches pads; traces were created using conductive thread, sticking it to the back of the tape and then sewing through the foam core, using the foam as an insulator. In production, such cores could be built up with traces between bonded foam layers.

With this construction, the dolphin fabric wrinkled into the v-cuts, preventing the switches from closing. However, prototypes had shown that the switches needed to be on the outside of the core, at the areas of
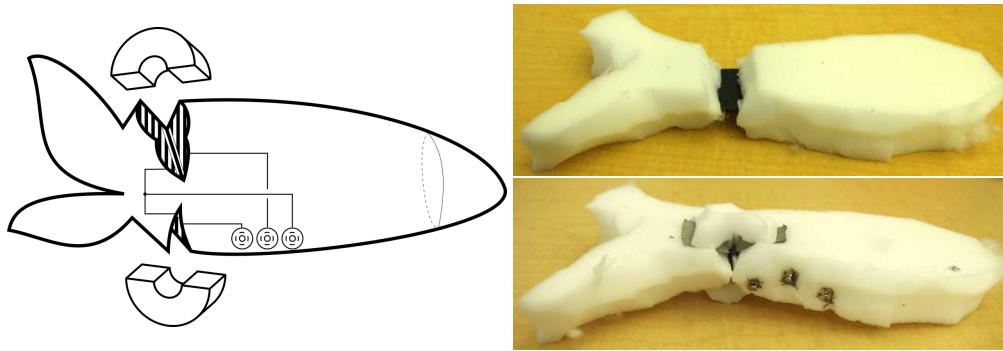
**Figure 5.3:**  *The foam core of the dolphin.*  Left, *striped patches show location of the contact switches, protected by arches. Wiring connects to snaps on the side.* Right, *the empty, then finished, core.*
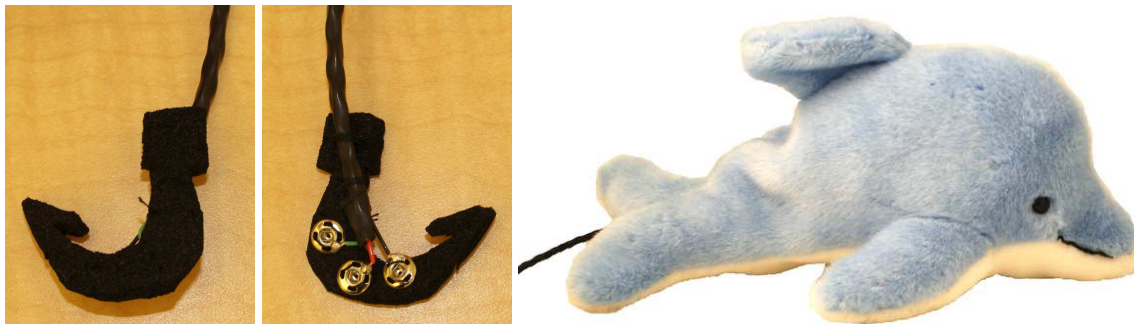


**Figure 5.4:** Left, *a marine-themed connecting cable for "hooking up" to the dolphin.* Right, *the completed dolphin.*

maximum compression and tension of the material during bend. If the switches were moved closer to the middle (the neutral mechanical plane), readings would be less reliable: due to the squishiness of foam, the setup would be highly sensitive to the child's grip, i.e. how much overall compression or tension he was placing on the joint. We noted that an arch was an optimal covering: on top of a v-cut, it is weak in compression from the side, and strongly resistant to bending inward. The arches are shown covering the v-cuts in Figure 5.3.

We connected the three traces from the contact switches to three snaps on the side of the dolphin. The connecting cable becomes less destructive of the charm of the stuffed animal if it is part of the story: a foam fishing hook attached to a piece of flexible cable, with its wires broken out into snaps, has become lodged in the dolphin's side (Figure 5.4). An Arduino Pro Mini is attached to the other end of the cable. One can imagine other plausible examples of the same technique: a "leash" that connects to a stuffed dog, a "bridle" for a stuffed horse, etc.
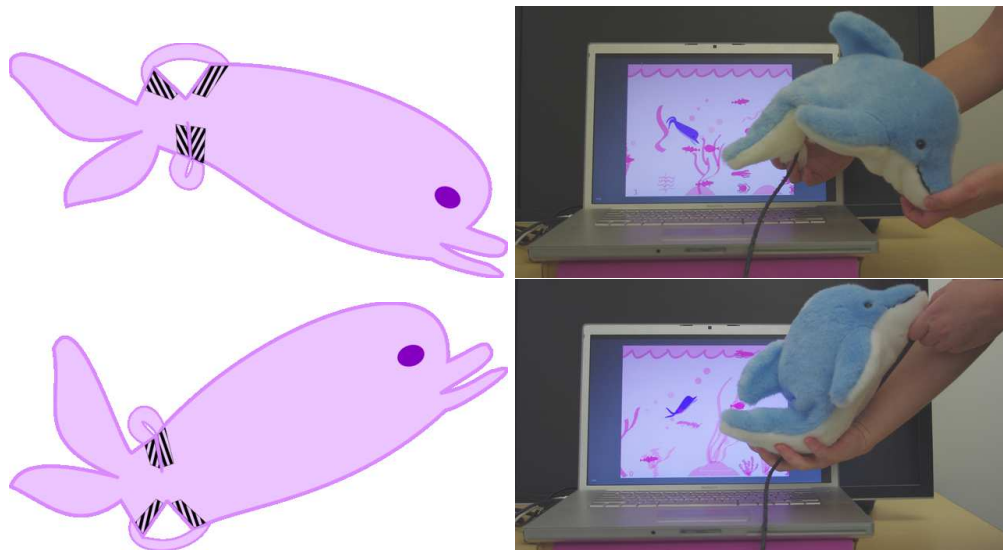
**Figure 5.5:** *The dolphin controlling a virtual double; each direction of bend corresponds to a different pattern of closed switches.*

### Application

The dolphin controls a virtual double in a side-scrolling fish-eating game, using tail motions similarly to a real dolphin. Figure 5.5 shows the patterns of open and closed switches for moving up and down in the game; to move straight, the child holds the dolphin unbent.

### 5.2.2 Elephant: bend structures in the limbs

Our second example is an elephant with bend structures to capture the bending of its four limbs. The limbs are moved in a Simon Says game to create patterns of motion as called out by the computer.

### Construction

We constructed the elephant core using the same methods and materials as the dolphin, but placed the microcontroller on-board the animal for compactness, as would be expected in a commercial product. The location of the microcontroller is shown in Figure 5.6. In play the head is made to nod and rotate, but is not squeezed. We propose that any hard electronics should be placed in the head, not the body, because humans have a learned aversion to squeezing heads. Although we used a wired microcontroller which could be plugged in through the back of the head, a wireless one with a battery could have been used.

Four slits on the elephant's foam core, one on each limb, were instrumented with fabric contact switches. Using normally closed bend structures, instead of valley cuts, gave a stiffer, less wobbly feel to the stuffed animal. The elephant's fur was tight enough that arches were not needed in this construction.
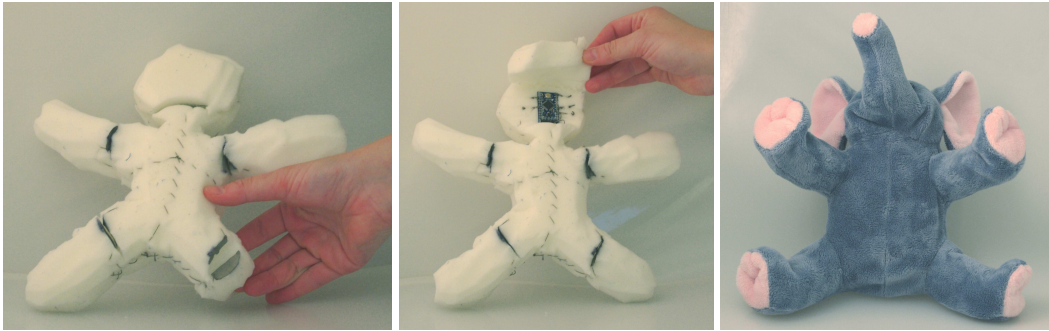
**Figure 5.6:**  *The foam core for the stuffed elephant contains four normally closed bend structures that trigger when bent (*left*), and a microcontroller in its head (*center*).  It fits snugly inside the elephant fur (*right*).*
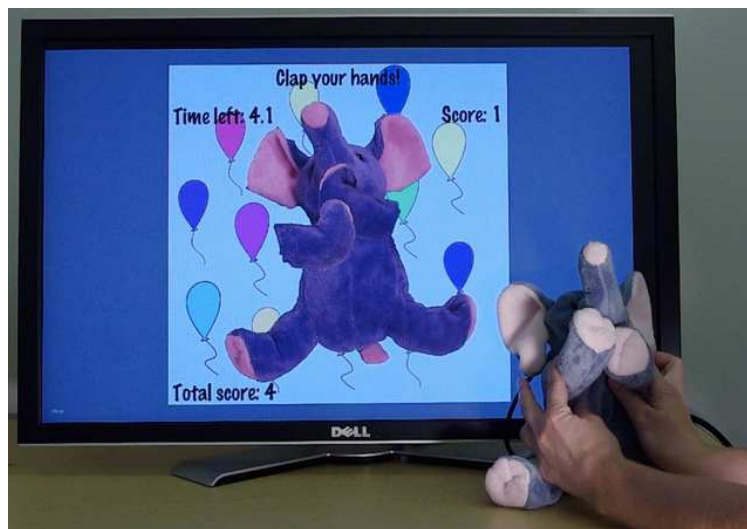


**Figure 5.7:**  *The user scores a point by clapping the elephant's hands; the figure onscreen mimics the motion.*

**Application**

We built a timed Simon Says game, shown in Figure 5.7, where the child tries to perform a given command with the elephant as many times as possible within five seconds.  The elephant's motions are mimicked onscreen in real time.  Simon Says commands include "wave your right hand" and "sit down".  Points are received for each correct motion; at the end of the game, the child views a "Dance Replay!" animation of the motions he has performed.

We used several techniques to compensate for the impoverished sensing and make the game fun. Exact limb position cannot be determined; if a limb is halfway bent at the start of a command, and the child fully bends it to complete the command, no switch changes state. We created a timed game, with repetitions, to cover

any lack of response from the first motion. We also chose commands that encouraged a full range of motion, e.g. "kick your leg" rather than "pick up your leg". We made the toy seem more sophisticated by clever use of commands: "wave your hand", "touch your nose", and "clap one-handed" all provided more challenges while using the same switch and thus an equivalent motion.

### 5.2.3 Frog: pressure sensor in the body

Our third example showcases a completely self-contained soft interactive toy. When the toy is pressed downward like a hopping frog, its eyes light up.

**Construction**

The eyes of stuffed animals are traditionally hard, thus introducing a place where we can substitute electronics without compromising the softness of a stuffed animal. We replaced the eyes of our stuffed frog with LEDs. The circuit diagram is shown in Figure 5.8. Conductive thread was sewn through loops bent from the LED leads.

We built a pressure switch into the frog's foam core, creating the core out of the three pieces shown in Figure 5.8: a bottom piece containing a contact pad; an offset ring; and a top piece containing a contact pad and the battery holder and LEDs.

We designed a soft 3V battery holder that would withstand being squished and sheared without losing contact. The battery holder is shown in Figure 5.9. In place of the metal spring or clip traditionally used to provide pressure, we introduce opposing cubes of foam covered with conductive fabric. The cubes compress around the battery. Conductive thread connects each cube to a circle of spandex; the circles are sewn together to form a pocket.
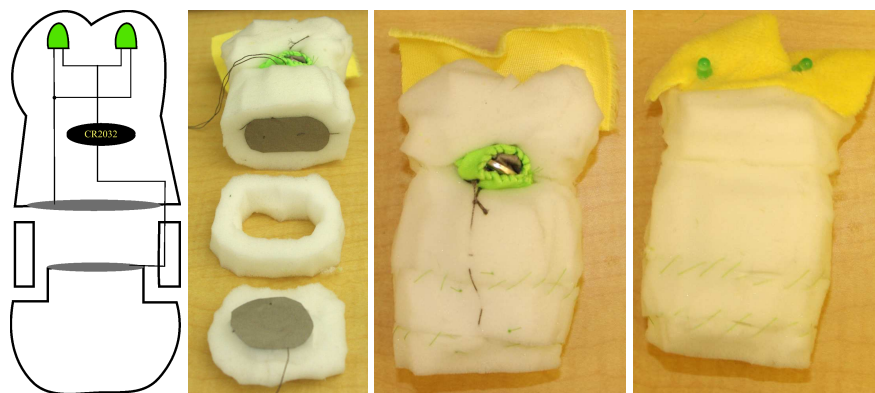


**Figure 5.8:** *Circuit diagram for the frog wiring; disassembled foam pieces; rear of core; front of core.*
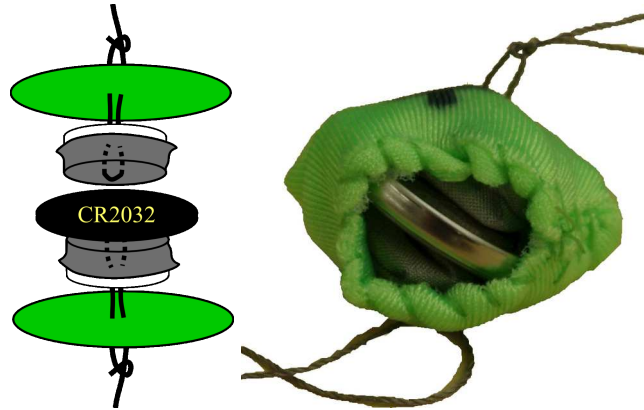
**Figure 5.9:**  Left, *a diagram of the battery holder.  Each half consists of a spandex circle (green) and a smaller cutout of foam (transparent) covered with fabric tape; conductive thread is sewn through.* Right, *the completed battery holder.*

**Application**

As shown in Figure 5.1, when the frog is hopped up and down, its eyes light up, adding another level of aliveness to the stuffed animal.

## 5.3   Discussion and Conclusions

We presented three examples of implementing soft sensing skeletons for stuffed animals, showing that if we move output to the computer, or to hard elements like stuffed animal eyes, we can use etextile materials to make these toys interactive without sacrificing their charming softness.

A foam core provides a simple base material for incorporating the structure vocabulary from Chapter 4. We added arches to the bend structure to prevent fabric from being pinched inside. A limitation of the arches – and the bend structure – in a soft foam is that the size of the joint being implemented cannot be much smaller than the size of the dolphin's tail joint; at smaller scales, the structure becomes too sensitive to overall pressure from the grip of the hand, and from the friction of the outer fur.

Another limitation of the softness of foam is that it is easily possible to force an incorrect reading, e.g. by gripping both sides of the dolphin and pushing to close both switches at once, or by pinching closed the area around a switch in the elephant. For our interaction scenarios, however, it is sufficient if the skeleton responds correctly to a purposeful interaction while connected to software. Switch state changes that do not reflect the expected pose will happen during the roughness of free play, e.g. between games, and should be discarded.

In this work, we considered various methods of incorporating necessarily hard elements without destroying the charm of the animal. The dolphin used a themed connector. A wireless version would remove connector blight, but hiding a microcontroller and battery inside would add hard elements and require a recharging

method. A detachable external wireless module, sharable between different stuffed animals, could provide the right compromise.

When hard elements are introduced, they should be placed in the head, the area least compressed during play. Another consideration is that stuffed animal eyes are traditionally hard; we capitalized on this to turn our frog's eyes into light-up LEDs. One could also take advantage of LEDs in this location for wireless optical communication, using a camera to capture either visible light or infrared communication.

# Chapter 6

# Tongue Interface Device

We use the materials and structures from Chapter 4, applying the methodology to design a reliable, flexible, waterproof tongue input device that fits inside the mouth.

The tongue interface is the solution to a highly constrained user interface problem from the Disney theme parks. Theme parks have the goal of bringing characters to life by having them act out and describe their stories. Many of the costumed characters, however, cannot converse with guests. Actors in these costumes wear large fur or plastic head pieces that prevent them from talking. Several of the characters also have highly recognizable accents and speech patterns, which actors cannot replicate accurately. These characters are reduced to using body language and gestures to communicate.

Articulation abilities have recently been added to some of these costumes. Motors in the eyes and mouth let them open and shut, as shown in Figure 6.1. This technology is currently used in shows in the Disney parks, where the mouth moves in sync with prerecorded audio coming out of loudspeakers.

We would like to use this articulation technology to allow these characters to converse directly with guests. Because these characters have such distinct voices, the actors inside the costumes will need to trigger context-appropriate prerecorded audio snippets. This situation suggests using a dialogue tree and a heads-up display inside the costume to show the current location in the tree. Although a dialogue tree cannot cover the full scope of conversation, the theme parks present a constrained situation. Actors for the current talking



**Figure 6.1:**  *An articulated character head shows the expressiveness added by just a single degree of freedom in each of the eyes and mouth.*

| Device | Direction | | | |
|---|---|---|---|---|
| | Left | Right | Up | Down |
| Sip/puff switch | soft puff | hard puff | soft sip | hard sip |
| Bite senor | two short bites | short long | long short | long long |
| Tongue joystick | press left | press right | press up | press down |

**Figure 6.2:** *All three devices, ready for testing. From left to right: sip/puff mouthpiece, bite sensor, and tongue joystick. Four-direction selection is accomplished using the actions in the table.*

characters in the park (e.g., princesses) tell us that conversations with guests tend to be character-driven: children frequently turn shy and let the actor lead the conversation; and when children actively interact, they ask the same questions (e.g., princesses are always asked where their prince is).

In this paper, we are primarily concerned with the input device that the actor inside the costume will use to navigate a dialogue tree with a branching factor of four. We consider four to be the minimum number of responses to hold a reasonable conversation: if a character asks a question, the guest's response can be categorized as 1) affirmative, 2) negative, 3) other (question-dependent), or 4) uncooperative; the character should have a response ready for any of these four cases.

Common input devices are not usable in this selection task, because they rely on the hands. A character's hands are visible at all times, waving, gesturing, and signing autographs. Instead, we seek an input method using the mouth. We need a device that has a low error rate and is fast enough to enable smooth conversations.

We compare three mouth input devices, using breath, teeth, and tongue. For breath, we use the leading portable mouth input device for people with quadriplegia, the sip/puff switch. For the teeth, we built a binary bite sensor. For the tongue, we built a new device. Its construction demonstrates using rapid prototyping and an unconventional combination of silicone and conductive thread as a flexible method for creating cheap, low-actuation-force devices.

We tested these devices with a user study that mimics aspects of our target application. The study included a speed test and a conversational interaction using a four-branch dialogue tree. Figure 6.2 summarizes how each device was used to select among the four directions. We found the tongue joystick to be both the fastest and most accurate. Dialogue latencies when using the tongue joystick ranged from 3 to 4 seconds, which is fast enough to hold a conversation with a child [Darves and Oviatt 2002]. No significant difference was found between the bite sensor and the sip/puff switch.

Our results have implications for the study of physical and situational impairments [Sears et al. 2008]. We show that the tongue is agile enough to select among multiple discrete locations, and thus should be further explored as an input device for people with quadriplegia or who otherwise cannot use their hands. Although preliminary work has been done in the area of capturing tongue gestures [Saponas et al. 2009;

Huo and Ghovanloo 2009], we are the first to test a form factor that requires no customization or attachment to the user.

In the following section, we describe less common input methods, and show why we converged on the *mouth* as the best high-degree-of-freedom input modality for this situation. Next we describe the three mouth devices. We divide the description of the devices into three sections. The first section describes the tongue joystick prototypes, showing how rethinking the design of the problem, and then applying our sensing through structure vocabulary, let us explore different ideas and ultimately led us to a nonstandard solution. The second section details the construction of the final tongue joystick design. The third section provides information on the remaining two mouth devices. We then detail the protocol of our user study to compared the joystick with two standard mouth devices, and show the results of the study. Based on the results, we select the tongue joystick to incorporate in a functional prototype system for an articulated character head. We conclude with possibilities for future work, both in our application and in assistive technology.

## 6.1   Related Work

Nearly all of our computer interfaces today – keyboard, mouse, touchscreen, driving wheel, button pad, gamepad – rely on fine dexterity in the hands. In our application, however, the hands cannot be used for input. An actor's hands must be free to interact with guests.

The human voice also has incredible degrees of freedom. Unfortunately, we found that sound transmits easily through our character head, ruling out standard voice recognition or nonverbal command inputs such as vowel [Bilmes et al. 2005] or duration of sound [Igarashi and Hughes 2001]. Whisper recognition or throat-microphone input would be possible, but to achieve high levels of accuracy, command phrases would need to be several syllables in length and easily differentiable, e.g. "show me A", "select option B", inducing a confusing lack of parallelism as well as slowing down the system.

Interfaces developed for people with physical disabilities show that other parts of the body can be used to communicate with computers. Any feature that can be tracked in two dimensions can serve as a substitute for the mouse. Head tracking can be done with an infrared camera [Natural Point 2011] or with an inexpensive webcam [Betke et al. 2002; Loewenich and Maire 2007]. Eye tracking, using infrared reflection [EnableMart 2009] or electrooculography [LaCourse and Hludik 1990; DiMattia and Gips 2005], requires less movement, but any type of eye-based input will monopolize the actor's attention. This compromise is acceptable for assistive technology, but not for our application, where the actor must attend to the child.

Any high-level voluntary movements that can be captured can be used for switch input. Examples include foot switches, eyebrow switches, and blink switches. These would be awkward or tiring to accomplish in costume.

Low-level signals from the body also present interesting opportunities. Electromyography (EMG) has been used to control artificial limbs [Kuiken et al. 2009]; attempts have been made to use it for mouse input [Kim et al. 2004] and wheelchair control [Felzer and Freisleben 2002]. Electrical skin potential has been tested as a switch input, but provides low accuracy [Masuda and Wada 2010; Moore and Dua 2004]. Brain-

computer interfaces continue to make progress, although they lack the speed and accuracy required for our application. Current research efforts involve techniques such as fMRI, electroencephalography (EEG), and electrocorticography (ECoG); see [Hochberg and Donoghue 2006] and [Wolpaw et al. 2002] for good overviews.

The mouth provides numerous methods of input; some of these can even provide direct spatial mapping. Sip/puff interfaces provide a single degree of freedom, but are portable and easy to use. BLUI [Patel and Abowd 2007] is an interface that extends the idea of using blowing as a means of input, but adds a spatial dimension, using a microphone to plot the location of a puff on a laptop screen. Breath can also be spatio-located using thermotransducers [Evreinov and Evreinova 2000] and piezo film sensors [Koichi 2010]; the latter work adds selection using tooth-touch recognition with a bone-conduction microphone.

Non-portable mouth devices commonly in use include joysticks, manipulated with the chin, lips, or tongue [Broadened Horizons 2011b], and mouth sticks, held in the mouth and used to tap directly on a keyboard or touch screen.

Our work was inspired by Huo and Ghovanloo's Tongue Drive System [Huo and Ghovanloo 2009]. The system consists of headgear containing magnetic sensors, and a small magnet affixed to the user's tongue. Voluntary motions of the tongue are classified and translated into powered-wheelchair control commands. We were drawn to this system for several reasons. First, as Huo and Ghovanloo point out, tongue muscle is similar to heart muscle, not tiring easily. In our application, we must be wary of repetitive stress injuries. Second, the tongue is very fast. In a similar setup, Struijk [Andreasen Struijk 2006] constructed a retainer with embedded coils inductively triggered by the proximity of a magnet attached to the tongue. Due to the stringent safety requirements of our application, we sought a self-contained device without the risk of swallowing a magnet.

Recent work by Saponas et al. [Saponas et al. 2009] uses optical infrared sensors to recognize four tongue gestures. The proximity sensors are placed in the right, left, front, and back of a dental retainer. Simple heuristics based on the pattern and timing of triggering are used to recognize one of four tongue gestures: left swipe, right swipe, tap up, and hold up. The 92% accuracy subjects achieved shows this approach to be promising. The Tongue-touch keypad [New Abilities 2009], a custom-molded dental plate containing nine buttons, was briefly on the market. The Tonguepoint [Salem and Zhai 1997], a mouthpiece containing an isometric joystick, showed that tongue dexterity improves with practice. These devices all require a custom retainer to achieve a low profile, but still present reasonable alternatives to our device.

## 6.2   Mouth Devices

Our target application, real-time conversational interaction, created a set of requirements for our input device:

*Speed:* The device must be fast and impose low cognitive load, as the actor will be simultaneously acting and conversing.
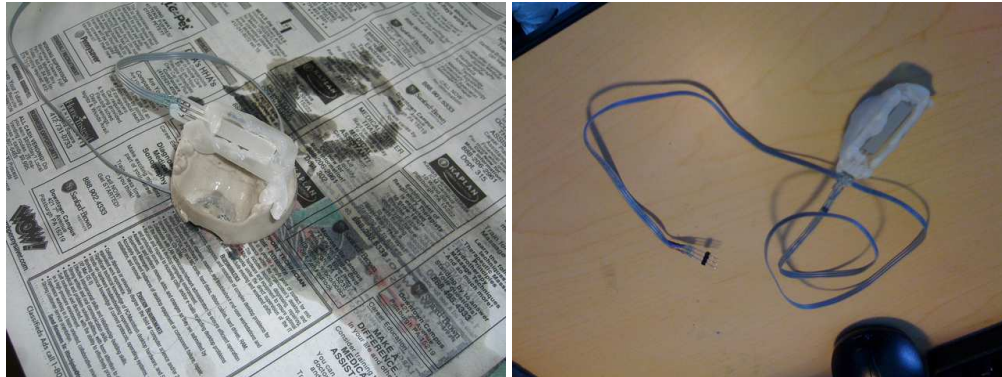
**Figure 6.3:** *A linear potentiometer is held in place by thermoplastic formed by hand over a model cast in permastone of the user's top row of teeth.*

*Accuracy:* Wrong selections would either trigger non sequiturs or slow down a conversation as the actor navigates backwards in the dialogue tree; the device must have near-perfect accuracy.

*Portability:* The device must be self-contained and fit inside the costume.

*User independence:* For convenient and low-cost testing and replacement, the device will preferably not be customized to fit a particular actor.

*Ruggedness and safety:* Failure of the device would destroy the magic of a performance; the device must be designed without weak points, able to last through many performances.

We explore which of the three input modalities of the mouth – the breath, teeth, and tongue – can be used to best meet these requirements. We represent these three modalities by portable input devices that attempt to make optimal use of their affordances: a sip/puff switch (breath), bite sensor (teeth), and tongue joystick (tongue). We present details of the devices in the following three sections.

## 6.3 Tongue Joystick Prototypes

The tongue joystick design went through several iterations, both before and after we had completed the sensing through structure vocabulary work. The next subsection shows the "before" work; the prototypes are straightforward, and the form factors similar to those used in previous work on mouth input devices. The following subsection presents a design tangent, thinking more about the structure of the problem. Armed with the knowledge gained from this section, and with the sensing through structure vocabulary work, the "after" iterations that follow benefit from a greater flexibility that brings the design of the device more in line with the affordances of the mouth and tongue.

**Figure 6.4:** *A 3D-printed holder for the joystick slots through one of the airholes in an off-the-shelf mouthguard. The joystick shaft slots in from the inside, poking through a silicone gasket glued to the inside of the holder.*

### 6.3.1   "Before" Solutions

Figure 6.3 shows our first prototype, consisting of a linear position sensor that rests on the roof of the mouth. We discovered that the tongue is strong enough to press the sensor (we are using the lowest-actuation-pressure sensor available), but repeatedly pressing hard with the tongue is uncomfortable.  In addition, having such a large bulk in the mouth was distressing.

Figure 6.4 shows our second prototype. We found a new joystick that uses magnetic (Hall) sensors, instead of mechanical linkages, for measurement, giving it a very low actuation pressure. We used CAD software and 3D-printing to create a holder for the joystick.  The holder slots into an off-the-shelf mouthguard, as shown. The tongue was able to move the joystick quickly from this position. We discovered, however, that the joystick head was not comfortable – the cup shape created suction with the tongue, a very unpleasant feeling – and the joystick was too far away, resulting in a very long joystick head that, confined along its length by the narrow mouthguard airhole, did not have enough range of motion. Additionally, airflow would be an issue in the Walt Disney World application, as actors are out in the hot sun in an enclosed plastic head, and blocking airflow to such a degree would not be acceptable.

### 6.3.2   Rethinking the Problem

The "before" prototypes suffered from a lack of thinking about the design space.  What objects do we hold in our mouths on a regular basis, and how do we manipulate these objects? Listing out the answers to this question gives a better understanding of the possibilities in this space, and provides ideas and guidelines for the creation of a tongue input device.
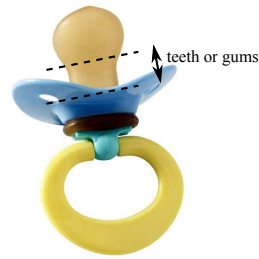
**Figure 6.5:** *The design of the pacifier has been refined to form the ideal shape for staying put after being lodged in a baby's mouth. If the tongue pushes, the shape resists by pressing against the inner teeth. If the baby sucks in, he avoids inhaling the device both by its size and the outer plate molded to his cheeks. As long as the baby is biting down, his (proto-)teeth are prevented from sliding outside the area between the dotted lines.*

| | |
|---|---|
| Food | teeth chew, tongue pushes into teeth |
| Food slivers | tongue pushes against teeth and levers sliver around |
| Gum | teeth chew |
| Piece of hay | teeth chew or hold, lips hold, tongue maneuvers |
| Drinking straw | teeth chew or hold, lips hold, tongue maneuvers |
| Pen or pencil | biting eraser, maneuvering around lips with tongue |
| Cigarette | lips maneuver, fingers hold |
| Lollipop | lips or biting provide counterforce against licking and pushing by the tongue |
| Thumb | suck while tongue cushions |
| Pacifier | suck while tongue cushions and teeth bite down to hold and cheeks provide counterforce |

The items in this list show that the key to any maneuvers in the mouth is the interaction between the tasks of the tongue, teeth, and lips. Our solution should harness this existing structure. We would like the tongue to do any maneuvering, because it has the most degrees of freedom. Our task in this respect is most similar to a piece of hay or drinking straw. The teeth and lips are often used to hold a device steady while the tongue manipulates it. Any jointed motion of a device will need to take advantage of this mechanism. For our system, we need something that is designed to stay in the mouth; the pacifier is designed for precisely this application, and adds another element: a plate located outside the mouth so that the cheeks also provide a stabilizing plane.

Our awkward prototypes from the previous section provided stability by effectively affixing the device to the teeth. The end result was that some part of the device would need to be customized to the user, i.e. the custom retainer or the mouthguard. A pacifier, in contrast, is not customized to a child. It conforms in a general way to the shape of the baby's mouth, but is held in place through its design, as shown in Figure 6.5.

### 6.3.3   "After" Solutions: Using Sensing Through Structure

With the pacifier form-factor in hand, we turn next to the structures outlined in our earlier sensing through structure work.



**Figure 6.6**

We selected a natural rubber pacifier with a flexible, hollow nib open to the outside. The pacifier contains four ribs on the inside, visualized in Figure 6.6, which encourage the four top sections to pop inwards individually when pushed with the tongue.

Our first attempt to capture this popping harnessed the constraints of the problem to minimize sensing. We glued a rare earth magnet to the inside of each of the four sections, alternating magnet polarity each adjacent section. Two hall sensors were placed back to back and poked through the nib of the pacifier. The setup is shown in Figure 6.7. The thought was that we could use two sensors, instead of four (one per magnet), because logically the effect of a single magnet push should be determinable (i.e. if one hall sensor reacted more strongly than another, then the push was on the side of the sensor with stronger response; the alternating polarity would determine which of the two magnets on a given side was pushed). The reality, though, was a lot messier: four overlapping magnetic fields combined with slight twists in a magnet as it was pushed, produced un-analyzable readings.
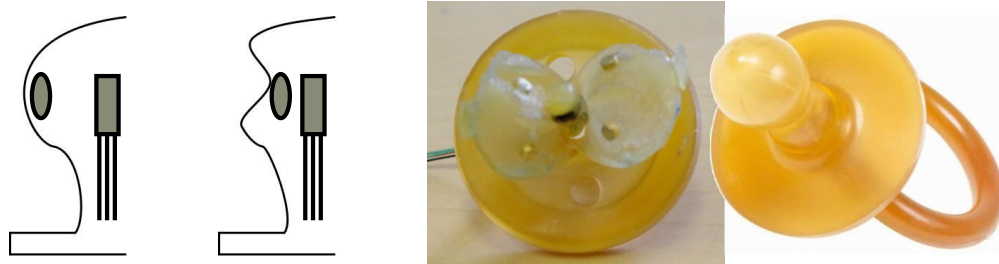


**Figure 6.7**

We also tried mounting a low-actuation joystick (2D Hall sensor based, from Austria Microsystems) on the base of the pacifier, with the joystick stick running the length of the inside of the nib. This attempt failed for several reasons:

- The joystick stick was very long, and thus would require a larger range of movement than is comfortable inside the mouth.

- The long stick required a pivot point at the same location as where the teeth bite on the pacifier. A hard tube inside the nib would thus be required surrounding the pivot area.

- The tongue is not strong at making a sideways push! When performing strong motions with the tongue, people tend to rotate the tip in the direction normal to the movement, and push from there.

Having learned that the tongue is better at pushing forwards rather than sideways, we attack again with a sensing through structure approach. Pushing near the top of the pacifier suggests several sensing structures, which we prototyped rapidly in silicone with conductive-thread contact switches.
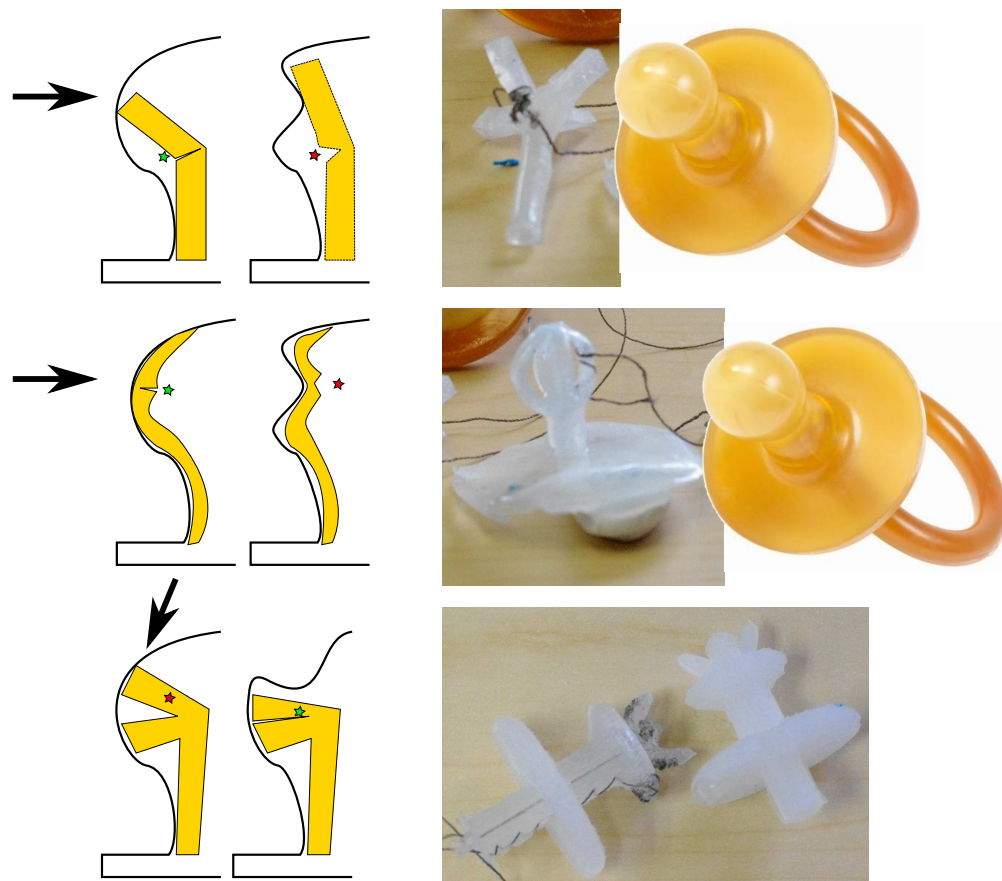


**Figure 6.8:** *Structures used to capture tongue presses on a pacifier. The left column shows half a cross section of the pacifier. Arrows show the direction of tongue force. The right column shows the corresponding prototype, with one direction's button fully instrumented.*

Figure 6.8 *top* uses the normally closed bend structure. A silicone framework containing this structure is pushed through the back of a pacifier and opens inside the nib like a stent. Figure 6.8 *middle* uses the normally closed bend structure again, but on the inside of the silicone. This put the structure closer to the area of movement and was thus more reliable, preventing other parts of the silicone from flexing and absorbing the motion. This prototype was also slotted into the pacifier.
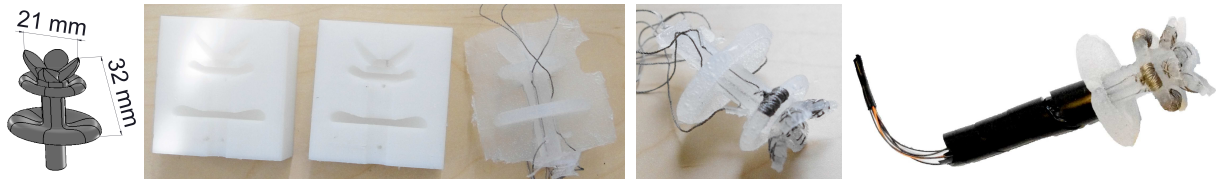
**Figure 6.9:** *The tongue joystick molding process. From left to right: 3D model; tongue joystick coming out of the printed mold; contact pads being sewn out of the embedded conductive thread; the finished product.*
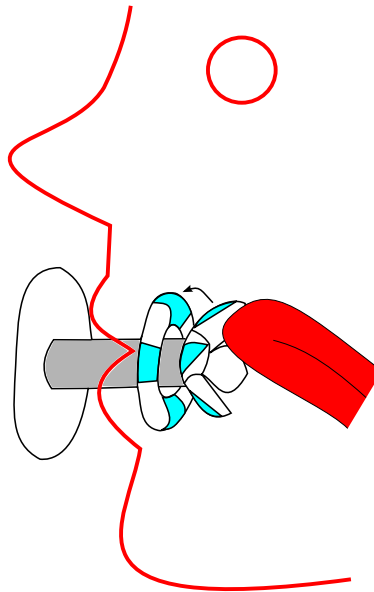


**Figure 6.10:** *The user bites on the stem (grey), then pushes one of the petals with his tongue (red). The petal contains a contact pad (cyan) which touches the ring beneath, also containing a contact pad (cyan).*

Finally, in Figure 6.8 *bottom*, we built our own "pacifier" using the normally open bend structure. While the "ship in a bottle" approach was viable, building our own device allowed us to lower the actuation pressure even further, and move the direction of push even more normal to the top of the pacifier, making it easier on the tongue. The prototype shown is design to be covered with a thin skin (e.g., plastic wrap or a shaped piece of silicone.). Details of this final device's construction are given in the next subsection.

## 6.4    Tongue Joystick Construction

Our device is shaped like a pacifier, but with the bulbous tip turned into a flower shape with a ring and four "petals". The tongue can easily align itself perpendicularly to a petal to push on it, as shown in Figure 6.10. (An early test using a four-directional joystick inserted in a pacifier had shown that though the tongue is dexterous, it actually cannot exert much sideways force.)

Figure 6.9 shows the steps in the construction of the tongue joystick. We designed the device in a 3D modeling program, and 3D-printed a plastic mold. To build the device, we fixed four strands of conductive thread to the inside edges of the mold, and one going through the center. We then poured Shore 40A RTV silicone into the mold. The device was cured and demolded. Then for each strand protruding from the silicone, we threaded it onto a needle, and "sewed" contact pads using the silicone as "fabric". The central conductive thread was sewn to the undersides of the top petals, forming the ground layer for the contact switches. The four outside threads were each sewn to a segment of ring beneath a petal, creating the signal layer.

This combination of materials – silicone and e-textile's conductive thread – creates a device which is completely soft and basically unbreakable. The silicone switches close with a low enough actuation pressure that they can be comfortably pressed by the tongue. Typical low-actuation pushbutton components close around 150 gram force; our switches close around 50 gram force.

Four-direction control is achieved through direct spatial mapping. In the study, signals less than 50 milliseconds long were dropped as transient. We covered the device in a new piece of plastic wrap for each subject in the user study; in production, the device would have a removable thin stretchy silicone skin in the shape of its convex hull.

## 6.5 Bite Sensor and Sip/puff Switch

We used an off-the-shelf USB sip/puff switch from Origin Instruments Corporation [Origin Instruments 2011]. Each subject in our study was given a new filtered mouthpiece when testing the device.

Four-direction selection with the device was done using soft puff, hard puff, soft sip, and hard sip to represent left, right, up, and down, respectively. This combination of breath force and direction is a common way to get four commands. Only the maximum (or minimum) value of the force signal was relevant, not the duration. Softness and hardness were assigned using pre-determined thresholds. Signals less than 80 milliseconds long were dropped as transient.

Commercial binary bite switches are used for skydiving photography and in cases of severe disability, e.g. ventilator dependence. These switches cost around $100 [Broadened Horizons 2011a]. Figure 6.11 shows the construction of our lower-cost switch. We use $\frac{1}{16}$"-thick polyurethane rubber, Shore 60A, and conductive adhesive tape. Thin top rectangles let the user locate the bite area by feel. For each study participant, the bite sensor was covered in a new piece of thin nitrile (a finger cut off an examination glove).

For four-direction control on this one-bit device, we used bite duration to add a second bit. The four directions were represented by the four combinations of short ($< 200$ms) or long ($\geq 200$ms) bite followed quickly (in $< 500$ms) by another short or long bite.
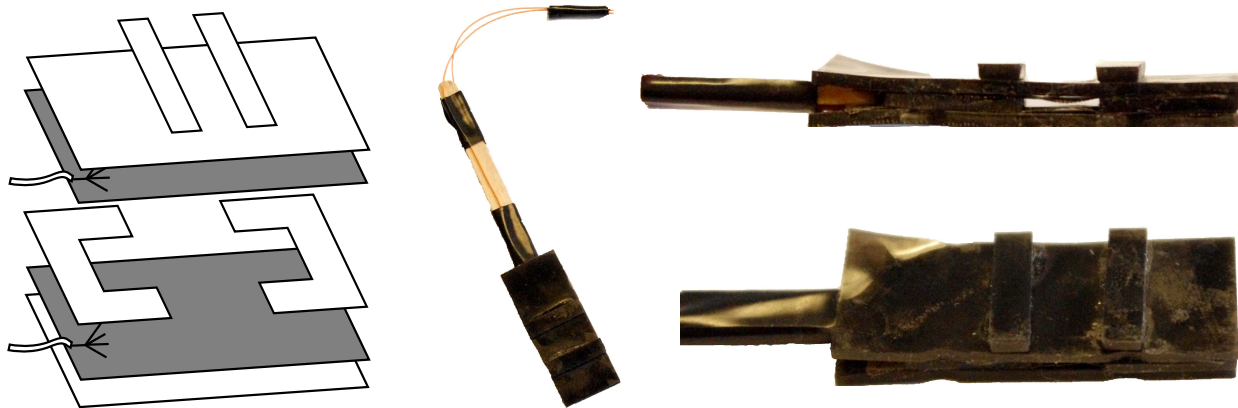
**Figure 6.11:**  Left*, an exploded view of the bite sensor's construction; white layers are insulating rubber, grey layers are conductive tape with attached wires.*  Center*, the finished bite sensor with mounting stick.* Right*, side and top view.*

## 6.6   User Study

We evaluate these three devices with tasks involving four-directional selection. In this setting, these devices represent different tradeoffs between cognitive load and motor complexity.

The tongue joystick has a direct spatial layout and thus the lowest cognitive load. While this spatial layout inherently biases our study in favor of the tongue joystick, it is the goal of the study to test whether the tongue has enough accuracy and agility to allow this lower cognitive load to dominate its performance.

The sip/puff switch, inversely, has the highest cognitive load. Its two-word prompts (e.g. "hard sip") must be combined along two dimensions (soft vs. hard, puff vs. sip) to produce a motor action.

The bite sensor also uses a combination prompt (e.g. "long short"), but each word maps to an independent motor action modulated only by time (short vs. long). Thus the bite sensor should have a slightly lower cognitive load. The bite sensor also has the simplest motor action, a bite, requiring little physical agility.

We have the following hypotheses:

1. The joystick will be the fastest, as it has a direct mapping from prompt to motor action.

2. The bite sensor, with a smaller cognitive load, will start out faster than the sip/puff switch. After a learning period where the sip/puff prompts are internalized, the results should flip, as sip/puff signals are shorter.

### 6.6.1   Experimental Design

The sip/puff switch, bite sensor, and tongue joystick were compared using a within-subjects design, counterbalanced for device order. Twenty-four users (11 women and 13 men, ranging in age from 18 to 58)

**Figure 6.12:** *A subject performs a forced-choice four-direction speed test of the tongue joystick.*

participated. Users were recruited from an on-campus behavioral research subject pool and were paid $15 for the one hour study.

Figure 6.12 shows the experimental setup. Devices were clamped at mouth height. Stimuli were presented on screen and consisted of four choices encircling a segmented central square with text reminders of the selection method. When a direction was chosen, feedback was provided by temporarily changing the color of the associated segment.

For each device in turn, subjects performed a Spelling and an Animal selection task. In the Spelling task, the device was explained and demonstrated to the subject, who then used it to spell six four-letter words. The correct letter occurred equally often in each of the four positions. Subjects were required to correct mistakes, thus ensuring a basic level of mastery of the device.

In the Animals task, we tested the response time of the device in combination with an audio prompt and changing selections, thus mimicking our target application. Subjects saw four new black and white animal pictures in each trial (Figure 6.13) and were given an audio prompt, 250ms later, of "bird", "horse", "dog", or "fish". Although subjects were asked to do as many trials as possible in ten minutes, all subjects received fifteen blocks of four trials, with a four second rest period between blocks.

After performing both tasks, subjects filled out a device questionnaire with a seven point scale for each of speed (slow/fast), accuracy (inaccurate/accurate), lack of fatigue (tiring/not tiring), and ease of use (confusing/easy to understand).

After testing each device, subjects performed a Dialogue task. Subjects were asked to choose one device to use for holding three conversations with a computer agent named Katie. Katie took four turns. For each of
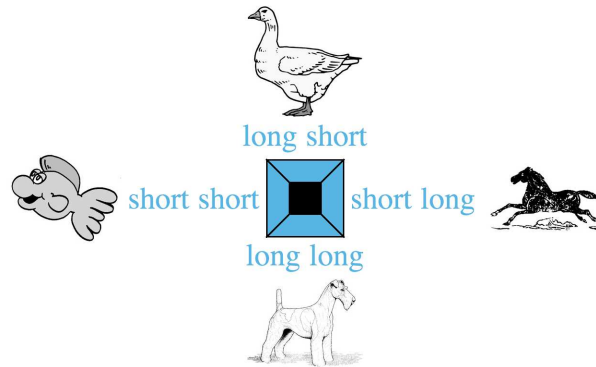
**Figure 6.13:**  *The Animals task with prompts for the bite sensor.*
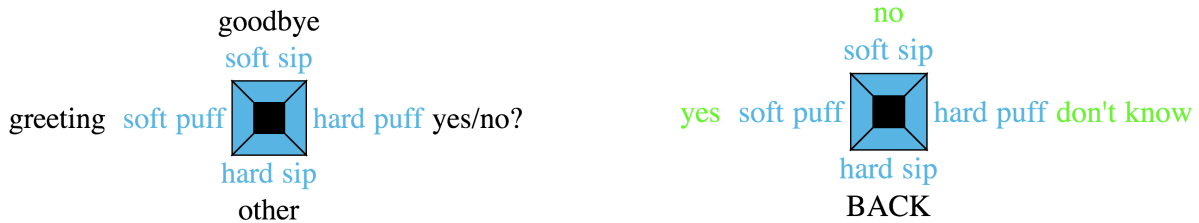


**Figure 6.14:**  *The two-level dialogue tree: category selection is shown at top; response selection for the category "yes/no?" is shown at bottom.*

Katie's turns, the subject first assigned Katie's speech to one of the categories "greeting", "yes/no question", "goodbye", or "other" (Figure 6.14, *top*). Subjects then saw three within-category options for their response and the "back" option in case they wished to re-categorize Katie's speech (Figure 6.14, *bottom*). When one of the response options was chosen, an audio file with that meaning was played aloud, and the screen reset to the category level.

A final questionnaire asked for ranked comparisons of the three devices using the superlative of the four scales (fastest, most accurate, least tiring, and easiest to understand). Subjects also provided general comments about their experiences and preferences.

## 6.7   Results

We give results of the Animals and Dialogue tasks, followed by summaries of the surveys. Tongue joystick performance was superior to the other two devices, and the tongue joystick was rated at or above the other devices on all scales. Results indicated, however, that subjects were still learning the bite sensor and sip/puff switch, suggesting the need for a longer follow-up study.

| Device | N | Reaction Time (ms) | | Error Rate |
|---|---|---|---|---|
| | | Mean | Std Dev | |
| Sip/puff | 24 | 3186 | 1721 | 23% |
| Bite sensor | 24 | 3016 | 1258 | 20% |
| Tongue joystick | 24 | 2065 | 880 | 10% |

**Table 6.1:** *Reaction times for the Animals task for all subjects.*



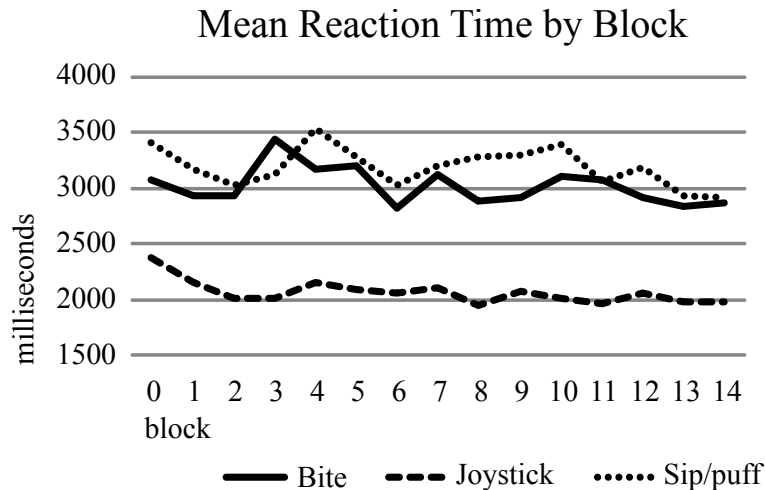**Figure 6.15:** *Average reaction times of the twenty-four subjects across the fifteen blocks of the Animals task.*

### 6.7.1 Animals Task

The tongue joystick proved the fastest and most accurate in the Animals task. Means and standard deviations for reaction time, measured from the onset of the visual prompt, are reported in Table 6.1. A repeated measures ANOVA showed that device significantly affected reaction time ($F(2, 22) = 42.27$, $p < .001$). Bonferroni-corrected pairwise comparisons showed the joystick to be significantly faster than the bite sensor ($p < .001$) and the sip/puff switch ($p < .001$). No significant difference was found between the latter two devices. Thus only the first of our hypotheses, not the second, was borne out. We conjecture that subjects were still mastering the bite sensor and the sip/puff switch during the fifteen blocks; the variability in reaction times for these two devices, seen in Figure 6.15, supports this view. In contrast, subjects learned the tongue joystick quickly and had consistently less variability. In future work, we plan to extend our study, looking for trends over a longer time period.

The joystick was the most accurate (Bonferroni-corrected, $p = .024$ vs. bite, $p = .001$ vs. sip/puff); no significant difference in error rates was found between the other two devices. Error rates by block are given in Figure 6.16. An analysis of the confusion matrices revealed that subjects had the most trouble with the "short long" and "long short" prompts for the bite sensor, and with thresholding soft versus hard for the sip/puff switch. This result supports our conjecture that subjects were still in transition for the two devices.
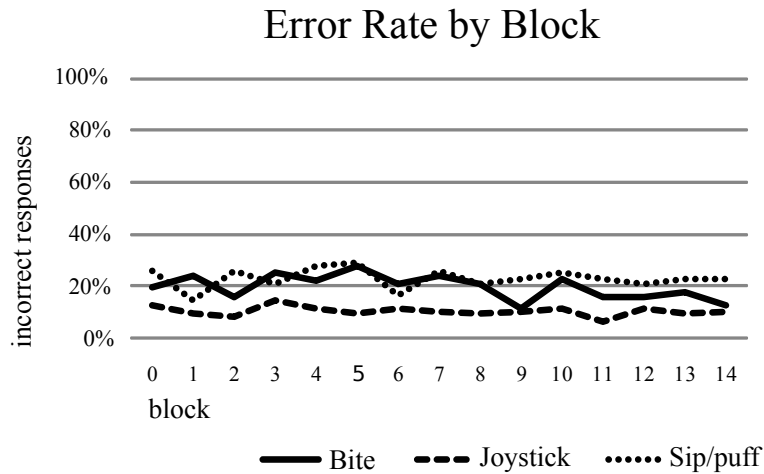
**Figure 6.16:** *Average error rates of the twenty-four subjects across the fifteen blocks of the Animals task.*

| | Category | Response Latency (ms) | | Error |
|---|---|---|---|---|
| | (action) | Mean | Std Dev | |
| Social | greeting (left) | 3552 | 1832 | 0% |
| | bye (up) | 3412 | 1582 | 6% |
| Content | yes/no (right) | 4129 | 2695 | 14% |
| | other (down) | 4198 | 2494 | 20% |

**Table 6.2:** *Average milliseconds taken by the twelve joystick subjects to navigate through the two-level tree, grouped by top level (category) and type of response (social vs. content).*

### 6.7.2   Dialogue Task

The Dialogue task recreates a language understanding component closer to our target application. Subjects were given their choice of device for this task; encouragingly, 17 (71%) of the subjects chose to use the joystick, 5 (21%) chose the sip/puff switch, and 2 (8%) chose the bite sensor.

Error rates were computed based only on category selection; Katie would reply cooperatively to any second-level response. Several of the subjects simply did not understand the task; because our interest is the viability of the device when used by a skilled performer, we eliminated from our analysis any subject whose error rate was above 25%. This elimination left 12 joystick users, 1 sip/puff user, and 2 bite sensor users. With such small numbers for the latter two devices, we analyze only the tongue joystick results.

Table 6.2 shows mean response latencies and error rates for the tongue joystick, grouped by category. Latency is defined as the time from the end of Katie's speech, through category selection, to final response selection.

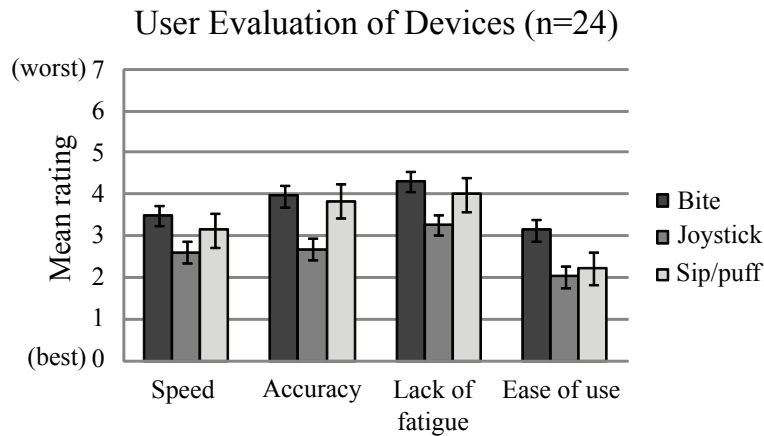**Figure 6.17:** *Average ratings given by subjects to each device after testing it in the Animals task.*

The difference in response latency between social and content categories was not significant. Error rates were significantly different between social and content categories ($F(1,11) = 12.78$, $p = .004$). This significance is not attributable to direction, which showed no comparable pattern in the Animals task.

Overall, subjects took about 3.8 seconds to compose a response (two selections at about 2 seconds per selection), with $90\%$ accuracy, roughly in line with their Animals task results.

### 6.7.3   User Surveys

Figure 6.17 shows the mean ratings for the individual surveys. On every dimension, the joystick was rated significantly better than the bite sensor (Bonferroni-corrected, $p < .05$ for each) and comparable to or significantly better than the sip/puff switch ($p < .05$ for accuracy only). Users did not rate the joystick significantly faster than the sip/puff switch, despite the better joystick timings.

After the Dialogue task, users provided a set of comparative rankings, tabulated in Table 6.3. Preferences at the end of the study mirror earlier judgments. The joystick was ranked better than the bite sensor for

| Ordering | Fastest | Most accurate | Least tiring | Easiest |
|---|---|---|---|---|
| joy > bite > sip | 8 | 10 | 4 | 9 |
| joy > sip > bite | 2 | 4 | 2 | 4 |
| bite > joy > sip | 2 | | 2 | 1 |
| bite > sip > joy | | 1 | | |
| sip > joy > bite | 3 | 2 | 7 | |
| sip > bite > joy | 3 | 1 | 3 | 4 |

**Table 6.3:** *Frequencies of the six device orderings for each category in the final comparative rankings (18 subjects completed the survey correctly).*

```
3 (chcknjoke) "I have a joke. Do you want to hear it?"
  1 (Yes:Xroad?) "Why did the dragon cross the road?"
    1 (punch) "Because...chickens weren't invented yet!"
    2 (answr:hey) "Hey, how did you know that?"
    3 (nope:punch) "Nope! 'Cause chickens weren't invented
  2 (No:Sad) "No? But it's such a funny joke. I'm sad."
```
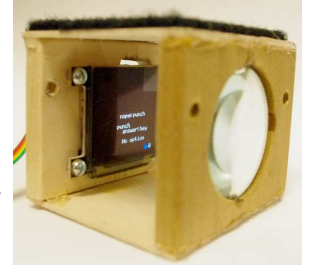
**Figure 6.18:**  *A snippet of dialogue tree, shown on the heads-up display normally lodged in the dragon's snout. The display contains a collimating lens so that the actor's eyes do not need to refocus between the screen and child.*

all characteristics (Wilcoxon Signed Rank, $p < .05$ for all), but was ranked significantly higher than the sip/puff device only for accuracy ($p < .01$).  No comparisons between the bite sensor and sip/puff switch were significant.

## 6.8   Articulated Character Head Prototype System

We incorporated the tongue joystick into a system prototype using the custom articulated-head dragon shown in Figure 6.1. We built a heads-up display that lodges in the character's snout, and designed a dialogue tree appropriate for interacting with a young child (Figure 6.18).  The actor holds the tongue joystick in her mouth while donning the dragon head. When she presses one of the petals in the tongue joystick, feedback is shown on the heads-up display; if a line of audio is triggered, it plays out of a speaker while the mouth moves automatically to prerecorded puppeteering. All hardware is connected through an external laptop.

As shown in the accompanying video, where the dragon meets a 6-year-old girl, the actor is able to use the tongue joystick quickly enough to engage in smooth, natural turntaking using prerecorded audio.  The dialogue-tree system excels at constrained situations where responses are predictable, such as knock-knock jokes and simple questions such as "What's your favorite color?" We thus optimized the tree for taking the initiative to control the conversation and guide it down one of these paths. This situation mirrors that of our target application, where conversations with characters in the theme parks are usually character driven. The tree also had a branch containing responses such as yes, no, and thanks.

From our trial sessions, we learned that we need to more thoroughly examine the dynamic of conversation with children.  We rapidly learned that one often needs to repeat a phrase until they focus enough to hear. Also, having a supply of giggles and other phatic expressions that could be inserted into a dialogue path without interrupting it would cover most awkwardnesses.

## 6.9    Discussion and Conclusion

We presented a new tongue input device, the tongue joystick, for maneuvering within a dialogue tree to select pieces of prerecorded audio. The device has a pacifier form factor which makes it both user-independent and firmly grippable for speed and accuracy. In building the tongue joystick, we developed a new method of constructing rugged, soft, low-actuation force devices by using both molding and sewing techniques to combine soft silicone and flexible conductive thread. Our hope is that this method will be useful to the assistive technology community in designing new devices.

The form factor and manufacturing method of our tongue joystick are amenable to easily incorporating other sensors: a bite switch could be located in the stem, for example, or a sip/puff tube embedded in the middle. Airflow and weight must be considered in any design, however. The device fit inside the dragon head, but the wires reduced the head mobility of the actor; we plan to build our next prototype with a wireless chip and small battery embedded in the cheek guard.

Our study design attempted to balance speed and accuracy. We urged subjects to work quickly, but did not penalize them for wrong answers. The error rate for the tongue joystick, 10%, was thus very high. In future work, we will run studies that isolate reaction time and error rate, giving a better lower bound for each while helping us better understand learning and fatigue effects. Future studies will also train users longer so that they will be closer to the "expert" user expected in our application.

The current user study showed that subjects were able to use the tongue joystick to respond to conversational turns with pauses around 3-4 seconds. Our articulated character head prototype system showed that a skilled actor could control conversational turns even faster. The maximum amount of latency that avoids conversational awkwardness is situation dependent; for instance, while cross-cultural analysis has shown that adults minimize the silence between turns [Stivers et al. 2009], at least one study has demonstrated that 7-10 year olds will converse cooperatively with an animated character with a 3-4 second response latency. Our application has more resilience than standard conversation, as the actor can cover pauses with physical acting. In our prototype testing, a case arose where the actor did not have audio for a particular situation; she switched from dialogue to solely physical acting for several minutes without the child noticing the transitions. Thus with a combination of lively acting, expressive costumes, and our tongue joystick, we hope to bring a new set of characters to life.

# Chapter 7

# Prototyping Robot Appearance, Movement, and Interactions Using Flexible 3D Printing and Air Pressure Sensors

We present a method for rapidly prototyping fully working interactive robot skins using flexible rubber-like 3D printing and analogue plug-in air pressure sensing. In Chapter 4, we presented a vocabulary for capturing manipulations in foam or silicone; here we analogously present a set of building blocks for using air pressure to register various manipulations in rubber. Our method takes advantage of the ability to print this air-tight flexible material in complex shapes.

The appearance of a robot is critical to its acceptance; studies have shown that humans will intuit factors such as personality [Goetz et al. 2003], intent [Woods 2006], and intelligence [Walters et al. 2009] solely from the external look of a robot. Our tendency to make these subjective judgments makes it crucial that all aspects of a robot's appearance, from the broad shape of the body to the subtle tilt of the eyes, are refined until they convey the image desired.



(a) Front view    (b) Twisting the handle    (c) Bending the flap    (d) Inside view    (e) Inside view with sensors
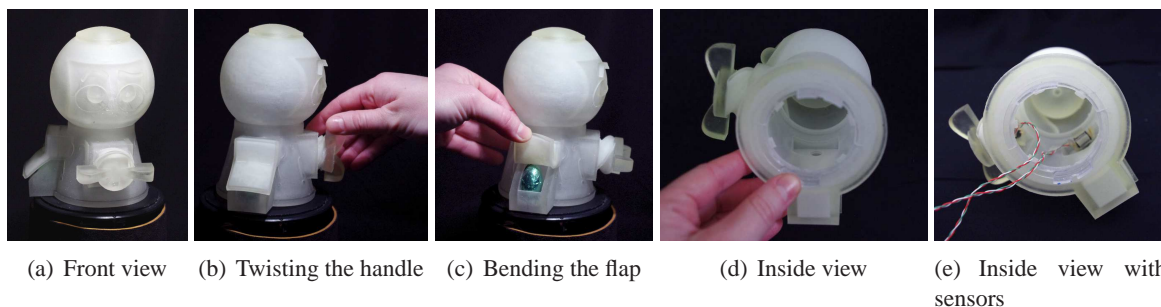
**Figure 7.1:** *Lucifer the Gumball Machine, 3D printed with a flexible rubber-like material. Air pressure sensors plugged into hollows in the twist handle and door flap capture interactions.*

In addition to appearance, the motion of the robot is critical to human perception; motion parameters such as acceleration and curvature [Saerbeck and Bartneck 2010], music synchrony [Avrunin et al. 2011], inclusion of gestures [Salem et al. 2011] or expressions [Blow et al. 2006], and even unintended cues like motor noise [Hegel et al. 2011] all affect a human's perception of the robot. Mori postulated that our acceptance of a robot form increases with its increasing similarity to humans, up to a point at which even slight deviations cause a sense of uncanniness [Mori 1970]. He further hypothesized that the uncanny valley's peaks are exaggerated for a moving robot, showing the importance of motion to robot acceptance.

Ishiguru extended Mori's uncanny valley graph with a third axis, similarity of behavior [Ishiguro 2006]. He argued that "humans expect balance between appearance and behavior when they recognize creatures". Maximum familiarity is achieved when a robot behaves congruently with expectations raised by its appearance. An important component of the behavior of a social robot is how it will respond to intended physical interactions, i.e. being pushed, squeezed, bent, or twisted. Where and how these interactions take place on the robot determine the affordances it will present to the world. Presentation of affordances is directly linked to robot appearance; and both appearance and interaction govern the space of movement. Our conclusion is that optimally, robot appearance, movement, and interaction would be designed in concert.

Creating flexible sensing skins for new robot actuation mechanisms is nontrivial, however. Casting a custom rubber skin requires a large investment in time and tooling; incorporating sensing often requires wiring swaths of point sensors over large areas. Thus studies on robot appearance and movement often rely on showing 2D images [Goetz et al. 2003] or testing on limited appearances in the real world [Saerbeck and Bartneck 2010].

In this chapter, we propose taking advantage of the power of rapid prototyping technology to open up new possibilities in robot design. With recent advances in 3D printing, robot parts can now be created out of a variety of materials, from hard to flexible to multi-colored, and intricate shapes and topologies constructed in a matter of hours.

We prototype interactive robot skins by 3D printing them in flexible rubber-like material. The power to print intricate shapes enables a new way of incorporating sensing. We propose a set of building blocks, consisting of hollow chambers, that each register a particular manipulation, such as twist, bend, or stretch. These chambers are printed as an integral part of the skin. Manipulating a chamber changes its physical volume, and hence, its internal air pressure. We use off-the-shelf air pressure sensors plugged into the chambers to sense user interaction. The sensors are reusable in future prototypes.

We demonstrate our prototyping method by developing, from scratch, three robot skins that fit on the Keepon Pro armature [Kozima et al. 2009]: an anthropomorphic gumball machine, an interactive devil, and a ghost. The first two characters demonstrate different examples of our building blocks; the third one focuses on iterating between movement and appearance.

Due to the current cost of 3D printing and the limited print area, our method is currently most appropriate for small robotics. Our method can be used to do the following:

- Design the affordances for interaction. Our building blocks provide an easy way to incorporate buttons and robot limbs with different affordances.

- Create iterations of a small robot to improve the appeal of its look combined with its movement.

- Prototype interactions combined with simultaneous movement (e.g. response to a robot being petted) and see the overall effect.

In the next section, we review related work in sensing methodologies and rapid prototyping for robotics. We then discuss our method in more detail, presenting our building blocks. We describe our three robot skins, and then conclude.

## 7.1 Related Work

Flexible robot skins with attached or embedded sensors have been fabricated from several different materials. Conductive thread and conductive fabric were used by Inaba and colleagues to create a robot sensing suit [Inaba et al. 1996]; Pan and colleagues quantified the accuracy of a textile position sensor [Pan et al. 2003]. Conductive textiles, while inexpensive, require labor-intensive hand-stitching for each new prototype; our method allows changes to be made in the 3D modeling program. Urethane foam was surrounded by a flexible circuit containing LEDs and phototransistors to create a multi-axis deformation sensor [Kadowaki et al. 2009].

Silicone rubber, with its skin-like feel, has been used widely in robotic skins. Silicone has been attached to optical reflectors [Yamaha et al. 1999], piezoelectric polymers [Miyashita et al. 2005], and piezoresistive polymers [Russell 1987]; and embedded with inductively coupled wireless sensors [Hakozaki et al. 1999], acoustic resonant tensor cells [Shinoda et al. 1997], neodymium magnets [Takenawa 2009], optical waveguides [Missinne et al. 2009], and microstructures for capacitive sensing [Mannsfeld et al. 2010]. Air pressure has been prototyped for use in robot skins by embedding a wireless air pressure sensor in a silicone cavity [Hakozaki et al. 2001]. Castable rubbers such as silicone are appropriate for use in the final product, with their ruggedness and skin-like feel, but the cost and effort required to create new molds and incorporate sensing elements makes them inconvenient for prototyping. As 3D printing becomes more widespread, the cost of the printing resin used in our method will continue to decline.

Rapid prototyping technology has been used for many years in designing jointed robots with rigid skeletons. "Rapid Prototyping for Robots" presents a good overview of previous work, with explanations of the various 3D printing processes and a database of moving joints made from rigid printable material [Ebert-Uphoff et al. 2005]. Current trends in rapid prototyping include 3D printing conductive materials [Malone and Lipson 2008], tissue scaffolds [Hollister 2005], unusual materials [Lipton et al. 2010], and embedded components [Periard et al. 2007].

## 7.2 Rapid Prototyping with Air Pressure Building Blocks

In our method, the designer builds a model of the robot in a 3D modeling program, designing portions that need to be touch-sensitive as hollow chambers, and prints the entire model using flexible rubber-like material. We built our models in SolidWorks and printed them using the TangoPlus material on an Objet Eden 260V.

As part of our method, we present a toolkit of building blocks, shown in Figure 7.3, for creating different affordances as part of the robot skin. Our toolkit builds on the ideas of Chapter 4, where we design a set of "sensing structures" for capturing deformations in a soft solid material such as foam or silicone. Our building blocks give the external shape of hollow chambers designed to be integrated into the robot skin. Each building block presents a given affordance, such as bend or twist, and deforms naturally when that manipulation is performed; some of the building blocks have stronger air pressure changes when the given motion is performed, and weaker response to other motions. Graphs of air pressure response for the accordion shape are given as an example in Figure 7.4. The list of building blocks is a first step to a more extensive vocabulary of building blocks, each with different properties.

These chambers are printed with a small hole for insertion of a 3D printed plug holding the air pressure sensor. The hole also allows removal of support material (a wax-like, water-soluble substance used in the printing process to maintain structural stability while printing).

The plug, shown in Figure 7.2, secures the sensor to a chamber wall. It is modeled in Solidworks and 3D printed with the same rubber-like material. The plug holds an off-the-shelf Freescale air pressure sensor, $0$ to $10$ kPa gauge. This range of pressure is suitable for registering typical light presses. Standard casings are used in Freescale's line of sensors, so sensors can be selected with a range appropriate to the scale of the prototype without modifying the plug. The air pressure sensors act as a sensing kit that can be re-plugged into each prototype robot, allowing quick, economical reuse of sensors.

Our plug's minimum diameter of $8.4mm$ fits tightly in $8mm$ holes in the robot skin. The air pressure sensor cannot be plugged directly into the building block, as the smaller $3mm$ hole for the sensor tip would make it difficult to remove support material.

The tactile feel of the chamber changes based on whether the seal between the plug and the robot skin is airtight. If the seal is leaky, the robot appendage is squishy, and requires recovery time to regain its shape. An airtight seal, made by gluing the plug to the chamber (while keeping the sensor removable), feels more like a firm balloon. The tactile feel can be customized to each application.

The rubber-like material is strong enough for attachment points (pockets, holes, etc.) to be printed as part of the prototype, allowing easy attachment to actuation armatures or rigid casings. We attach to the Keepon Pro armature, shown in Figure 7.5, by printing a socket in the head of our robots, and tabs along the bottom for aligning a rigid bracket.
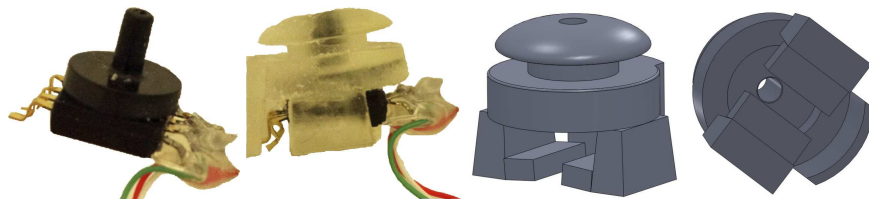


**Figure 7.2:**  *The plug we designed to hold the air pressure sensor.* Left*, air pressure sensor, with and without cover;* Right*, renders of the plug.*
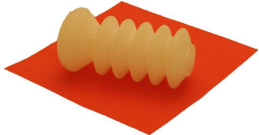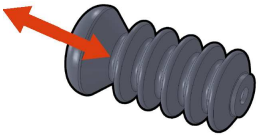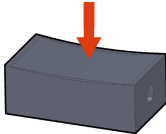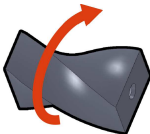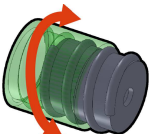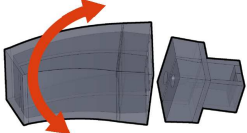
| Manipulation | Printed Form | Digital Model | Explanation |
|---|---|---|---|
| Pull and push | | | In an accordion shape, air pressure responds to changes in volume caused by pushing (positive pressure) and pulling (negative pressure); squeezes and bends do not change volume significantly |
| Pull | | | A shape with a grip suggesting pulling |
| Press | | | The top indentation suggests pressing or squeezing; images of buttons could also be embossed on the surface in the modeling program |
| Twist | | | Twisting of this shape happens easily in only one direction and causes a large difference in air pressure; the shape buckles when bent, thus small bends do not register |
| Twist | | | Screw-together twist pieces can be 3D printed; pressure changes with volume when twisted |
| Bend | | | Although this shape will respond to both pressure and bend, the appearance encourages bend; this suggestion could be further enhanced with surface creases |
| Bidirectional bend | | | Direction of bend can be measured by using two chambers, and a differential air pressure sensor with two input ports |

**Figure 7.3:** *Our set of building blocks for prototyping sensing robot skins with various affordances. Each model has a hole to plug in an air pressure sensor.*
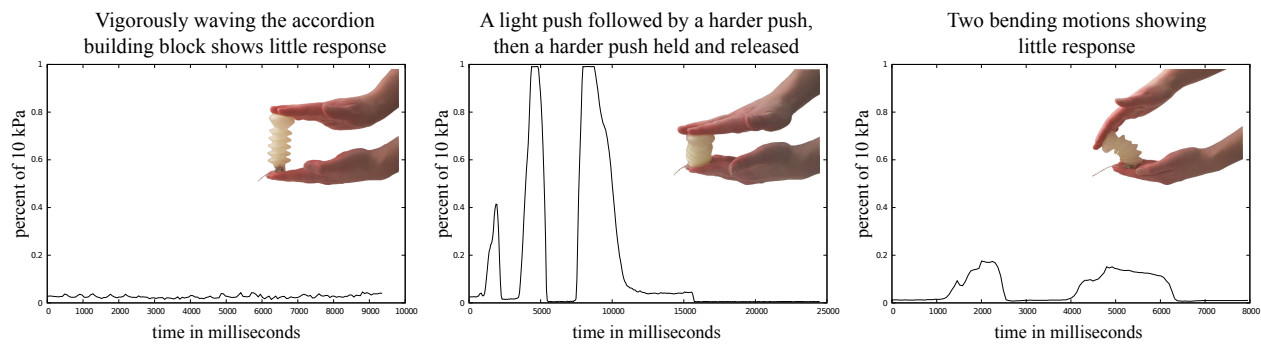
**Figure 7.4:** *Air pressure response to manipulations of the accordion building block.*
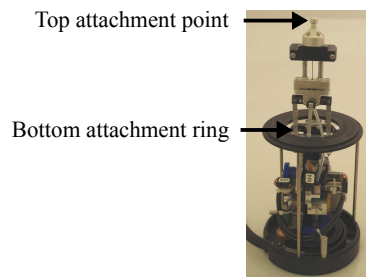


**Figure 7.5:** *The Keepon Pro armature.*

Future 3D-printable materials will present a range of different rubber properties, and will have the durability to allow our method to be used to construct and instrument finished skins. The rubber-like TangoPlus material is suitable for prototyping, and may not exactly mimic the properties of the final robot skin. Silicone, for example, has quicker rebound and more stretch. Nevertheless, TangoPlus gives a rough idea of the final movement, and its thickness can be varied across the model to encourage the desired dynamics (e.g. inserting a crease to encourage deformation in a given area).

Our method takes advantage of the strengths of 3D printing. With a 3D model of the device, designers can rapidly iterate on the design of the robot, and print skins with topologies that would be difficult to cast, such as the hollow chambers that make air-pressure sensing possible. When building our examples, we found it useful to print small-scale models to study the look of the character, before spending the printing resin on a full-scale prototype.

## 7.3  Robot Skins

We next describe the construction of our three robot skins. We show that by prototyping the skins using our method, we have an easy way to incorporate various affordances, as well as a platform to test questions that spring from the interdependence between appearance, movement, and affordances. In each robot, we give an example of how our method helped improve the original design.
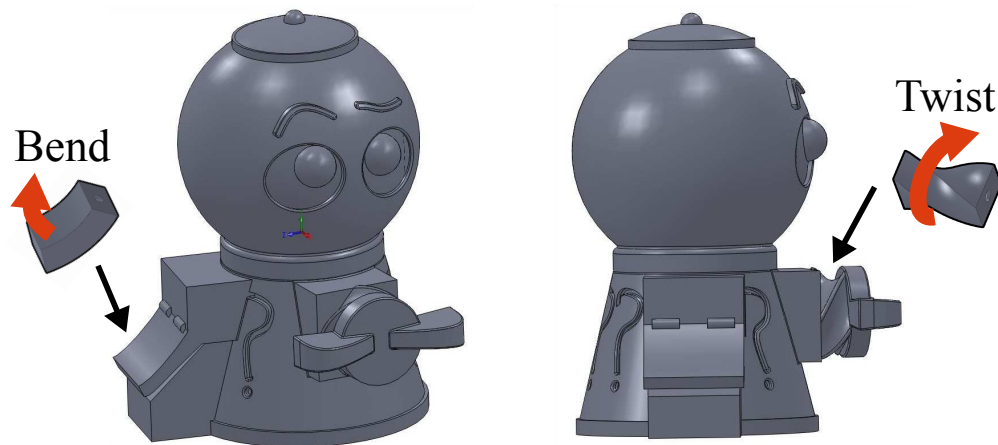
**Figure 7.6:** *Renders of the gumball machine model, showing our hollow building blocks.*
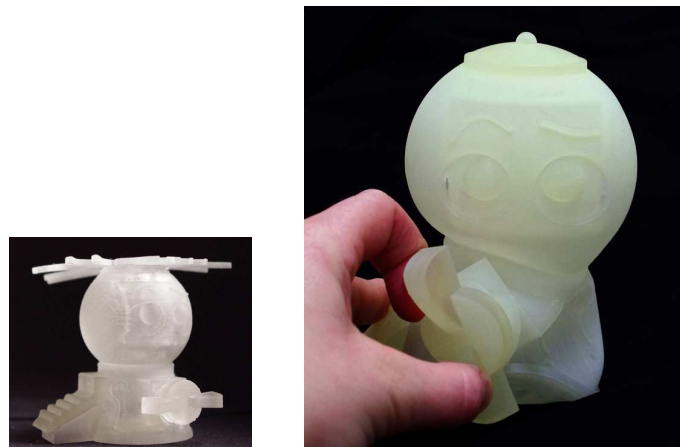


**Figure 7.7:** *Iterations revealed design flaws in appearance/affordance and affordance/motion.*

### 7.3.1 Robot Skin: Lucifer the Gumball Machine

Lucifer, an anthropomorphic old-fashioned gumball dispenser, uses our building blocks to sense twist in his coin handle and bend in his door flap, as shown in Figure 7.6. An air pressure sensor is plugged into each of the two building blocks in the final version, shown in Figure 7.1. Lucifer dances happily until a passerby twists his handle, whereupon he excitedly motions to them to lift the flap and take the concealed candy. If the passerby opens the flap first without "paying", Lucifer responds angrily, jerking the dispensing area away from the thief.

To conserve resources, we first printed hard-plastic miniatures to judge the look of each of our characters. Colleagues who saw our first gumball miniature, shown in Figure 7.7 *left*, thought the ridges intended to indicate bendability of the door flap were a staircase. The hair (it would drape when printed in rubber) detracted too much from the bubblegum-machine appearance; the eyes were sufficient anthropomorphism.

Our first full sized prototype, shown in Figure 7.7 *right*, revealed a flaw in the intended motion combined with the twist affordance. We had intended the entire body to be flexible when dancing, but with this prototype we could see that twisting the knob would case the entire lower body to buckle unnaturally. In the final version, we stiffened the lower body by making it thicker, and moved all motion to the robot's head.

If we desired to fully develop this character, we could move into user testing, and continue to iterate on the skin with questions such as the following:

- Will passersby be bold enough to twist the handle? Should the robot hold still, instead of dancing, until it is twisted, to appear less imposing? Will changing the friendliness of the appearance and/or motion better lure in bystanders?

- Does the handle need to be overly large, or have a "twist me!" sign on it, to make the affordance obvious? Will this constrain the movements? Are passersby willing to lift the flap, or should we actuate it?

Our method provides the flexibility both to refine the design, perfecting the current gumball dispenser's performance, or to easily broaden the scope of the questions. Our building blocks could be swapped out: for example, with a simple change in the CAD software, the twist handle could become a pull ring or push-button. If the entire concept is flawed (perhaps today's children no longer recognize bubblegum dispensers), the entire body could be changed, incorporating the knowledge learned from the current movement and affordances: for example, we could switch to a robot toy vending machine with flap and buttons.

### 7.3.2  Robot Skin: Creepon the Baby Devil

Creepon the Baby Devil, shown in Figure 7.8 and Figure 7.9, uses our air pressure method to make all its limbs sensitive to squeezing. As an example interaction, we test the feasibility of the piece of showmanship shown in the accompanying video: while Creepon tries to dance, a mischievous human repeatedly tweaks a limb and quickly hides, leaving Creepon to look around confusedly and get progressively angrier.

The interaction is entertaining, and the skin deforms significantly, enabling Creepon to dance fluidly and evince dejection and anger. One flaw with the concept was revealed, however: it took careful timing to grab hold of Creepon's small limbs when he was dancing, and mistiming it put stress on the Keepon Pro armature. In a future iteration, we could use this lesson to design bigger limbs that moved more flexibly where they joined the main body; or we could attack the movement and interaction, using a slower song with pauses. The solution could involve any of appearance, movement, and interaction; but the problem was only revealed when all three were combined in a working prototype.

### 7.3.3  Robot Skin: Gus the Talking Ghost

In our final example, we use our method to test a motion concept: designing a flexible robot whose mouth moves as if talking, although the actuation is transferred through the skin from the Keepon armature's top attachment point, as shown in Figure 7.12.

(a) Front view    (b) Rear view    (c) Deformation of the rubber-like material

(d) Inside view    (e) Inside view with plug and sensor installed in the wing
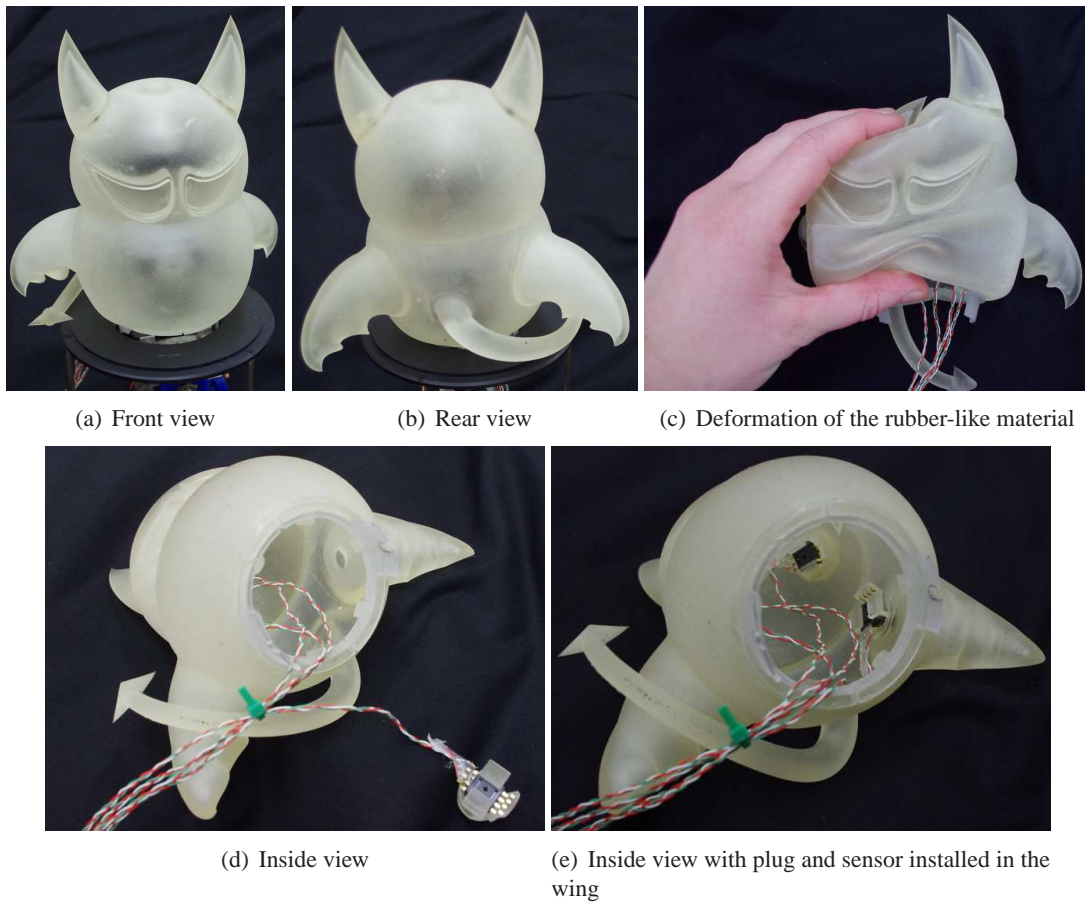
**Figure 7.8:** *Creepon the Baby Devil.*
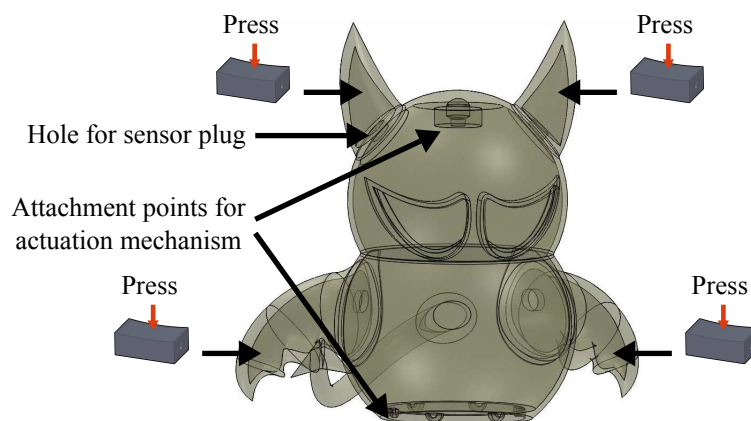


**Figure 7.9:** *A render of the Creepon model, with actuation attachment point, and hollows in the wings and horns visible.*

**Figure 7.10:** *Iterating on the mouth shape of the talking ghost.*
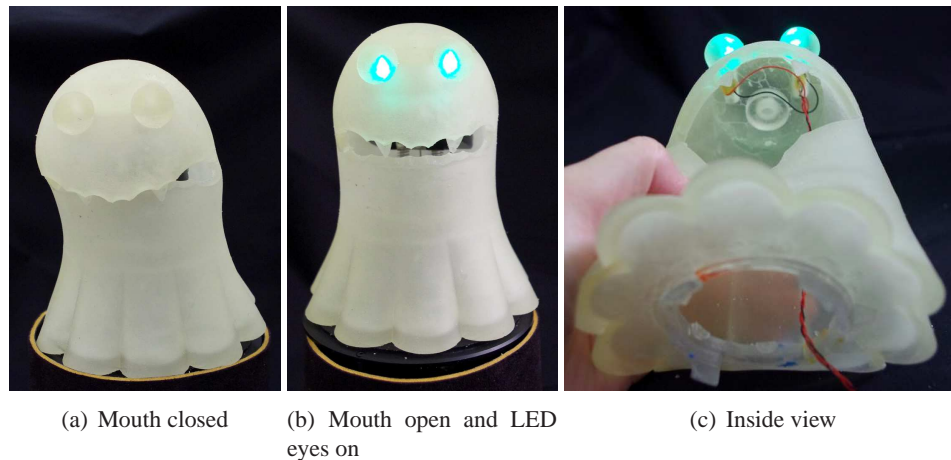


(a) Mouth closed        (b) Mouth open and LED        (c) Inside view
                        eyes on

**Figure 7.11:** *Gus the Ghost.*

Through iteration, we realized that the shape of the mouth was key to the illusion. Our first prototype of Gus the Ghost (Figure 7.10 *left*) buckled near the mouth; we tried a thinner curving mouth to distribute the stress (*center*) but the rigidity caused the head to assume an oval shape when bent; our final iteration (*right*) used a wide, empty mouth with circular arcs on the edges, which would bend easily under stress, minimizing buckling. The final Gus, with controllable LEDs inside his eyes, is shown in Figure 7.11.

## 7.4    Discussion and Conclusion

We have presented a method for rapidly prototyping flexible robot skins with easily incorporated sensing. The set of building blocks we developed provides guidelines for indicating and sensing various affordances.

**Figure 7.12:** *A render of the ghost model assembled with the Keepon Pro armature, to test the fit.*

Our method provides a full, working system throughout much of the prototyping process, thus allowing simultaneous iteration on appearance, movement, and interaction. We demonstrated this advantage by developing three robot characters, and describing needed modifications to the designs that only became apparent with a working system.

Our method has several practical limitations which currently restrict it to prototyping. The rubber-like material, currently available only in Objet's printers, tears more easily than silicone. Tearing can be minimized by rounding all corners and edges, and the quality of 3D printing materials will continue to improve. The air pressure sensor size limits the possible density of sensing regions (sensors would bump into each other). In the future, we could move the air pressure sensors outside the robot, routing the pressure using plastic tubing.

Our prototypes have practical uses besides iteration. We could control test conditions in robot appearance studies. Appearance changes could be made to a base model in software, and user interaction tested in the real world using the same robot mechanism and movement.

This chapter presents one of many ways to harness the exciting possibilities enabled by rapid prototyping. Rapid prototyping is leading a "personal manufacturing revolution" where anyone can design and create their own goods. Our work could be expanded into a robot kit that allows anyone to use 3D printing services to build their own flexible robots. We envision the kit consisting of a generic robot armature with a set of pre-wired air pressure sensors, and software that would help select from among our building blocks and merge them into an existing 3D character model for printing.

Multi-material printers, combined with the promise of printable conductive material, are getting us closer to 3D printing a complete, interactive robot. New research will need to explore what types of integrated sensing and actuation such a system makes possible.

# Chapter 8

# Conclusion

At a basic level, the contributions of this dissertation are the end products: a way to harness accelerometers to do motion capture, construction methods for any flat-bottomed convex silicone input device, rugged silicone sensors, soft sensing stuffed animals, a tongue joystick, and the ability to prototype working interactive flexible robots.

On a higher level, however, the contribution of my dissertation is providing a methodology for thinking about how movement in soft materials works. Throughout this thesis, I confront new soft materials – e-textiles, silicone, foam, 3D printed rubber-like material – and I use them to solve hard sensing problems in surprisingly simple ways. The core chapter in this thesis, soft silicone sensors, presents a vocabulary of sensing structures. In this chapter, I show that by thinking about manipulation and reducing manipulations' effects to their essence of tension/compression, it becomes apparent that the effects of a continuous deformation can be transmuted into simple linear displacements of the soft material. This vocabulary was used in sensing stuffed animals and the tongue joystick. I then used this sensing thought process in prototyping interactive flexible robots, again showing that various interactions could be captured with a one-dimensional signal, here air pressure level. The thinking process of looking for structure in motion is applied more broadly in the accelerometer shirt and silicone input devices, where I again harness structure to simplify sensing problems.

In Chapter 2, I showed that accelerometers can index into a motion capture dataset to control a performance animation system. This work has several points of fragility. To retrieve a motion, the orientation of the accelerometers on the t-shirt must exactly match the orientation of the virtual accelerometers from which the motion capture accelerations are computed. Even a slight twisting of a shirt sleeve markedly decreases the quality of the results. I can hypothesize several ways in which the problem might be overcome in future: continuous software recalibration, additional sensors, rethinking the t-shirt, or, perhaps, making orientation irrelevant.

*Continuous software recalibration:* With continuous software recalibration, a software application using the system would be designed to elicit predictable motions at given intervals (e.g., using a jumping jack gesture to switch levels in an exercise game). The system would then use this gesture to recalibrate the virtual accelerometers.

*Additional sensors:*  Additional sensors such as gyroscopes would separate orientation from movement; the readings could be used to recalibrate during play, or throw a warning when an accelerometer had slipped from its original position. The cheapness of accelerometers was one of the original motivations for this work; now, however, chips containing a combination of gyroscopes, accelerometers, and magnetometers are coming down in price.

*Re-thinking the t-shirt:*  Sliding of the fabric across the skin during movement, and bunching of the fabric near the elbows, are the causes of the accelerometers shifting position. Redesigning the t-shirt might solve these issues. The sleeves could be made tighter, or the whole shirt replaced with a less comfortable tight spandex version. The fabric could be make stickier in critical areas using the silicone tape used, e.g., to hold bicycle shorts in place around the legs. The elbow region of the shirt could have large holes at the joint to diffuse stress.

*Making orientation irrelevant:*  One avenue for future research is testing how well this entire system would work using just the magnitude of acceleration (the sum of the squares of the accelerations along the three axes of the accelerometer). Magnitude of acceleration from the accelerometers would be compared to magnitude of acceleration computed at the location of the virtual accelerometers. The system would be significantly less sensitive to both virtual accelerometer position and location. I hypothesize that a large body of motion capture clips would still be able to be differentiated, although the database might need to be more carefully curated to contain clips with different acceleration profiles. Perhaps this curating, too, could be optimized, with a user interface presenting the software designer with a graph of clips showing distances between each pair.

In addition to calibration sensitivity, a second point of fragility is the dependency on the composition of the database. The database composition needs to be optimal in several ways: motions being performed need to be in the database; motions of different actions need to be far apart in acceleration space; and motions need to be similar across people. The first restriction is one of the reasons I termed the project "action capture". The concept of the system works best in an application where the set of actions that are appropriate are clear to the user, rather than open-ended. If accelerations of two very different motions are similar, my current software will start flickering between the two motions. Although stronger continuity and context constraints can be imposed, the limitation is a fundamental drawback of using a sensor-impoverished system: two motions with the same accelerations but different positions will have the same accelerometer readings. In my work, the problem cropped up most when very little acceleration was occurring – the hands were dangling aimlessly, or the motion capture clip was holding a pose. The problem limits the scalability of my work, because adding more motions to the database will result in more clips with similar acceleration profiles. I also made the assumption in my work, based on comparing pilot data, that people perform the same actions using the same accelerations. Although the assumption is true for walking and running, and appears to hold true for constrained actions like jumping jacks, a useful area of future research would be conducting a study of a large multi-person dataset to see how far the similarity extends.

In Chapter 3, I built several prototype silicone input devices. As the tracking software for the silicone input device continues to be improved, the texture on the device will need to be further optimized, or changed completely, to make best use of the software. The current idea of a single layer of texture assumes that the tracking software, when completed, will be able to robustly handle areas where the texture merges into, or even overlaps, itself. If that assumption does not prove true, the resolution of the device will be

fundamentally limited: the texture will need to be sparse enough that it will never intersect itself, thus limiting the density of the tracking mesh. The scribble texture and software currently being used has this tradeoff; denser scribbles are tracked in higher resolution, but fail more quickly during larger deformations.

More optimal texture encodings could be implemented. One possibility to make the software task easier would be to create several textures in transparent, distinguishable colors: the texture layer could have a sparse blue texture with very thick lines, and a thinner, denser red texture. If the tracking began to fail on the red texture during, e.g., a large pinch gesture, the reconstruction could fall back on using the blue texture triangulation so that the system would not fail completely; the blue texture triangulation could also provide a check on the triangulation tracking of the denser texture. Although this seems an intuitively better idea, it would be worth mathematically analyzing the problem to enumerate the value of various texture schemes.

The silicone input device could also be expanded to incorporate other modalities: device as display and haptic feedback.

*Device as display:* The use of a colored texture prevents us from rear-projecting a display onto the silicone, as other projects have done. It may be possible to modify the texture's silicone to work around this problem. If the texture were visible only in infrared, it would not interfere with the display. Alternatively, a bright projector might transmit through a transparent texture well enough to be seen. Top projection is also an option, but brings the down side of occlusion.

Being able to project onto the hemisphere would allow interesting user interface investigations. For instance, an object being manipulated with the hemisphere device could appear to be on the surface of, or inside, the sphere; user testing would be necessary to see which paradigm is more natural. If the object were inside the sphere, users would need to imagine their fingers virtually extending through the surface of the sphere until they touch the object. In a traditional 3D modeling program, the mouse cursor, although moving in the 2D plane of the computer screen, effectively snaps its depth to the surface of the object. User testing would be needed to see whether a similar convention would appear natural for the fingers. The alternative, having the object appear on the surface, would deform the render of the model (e.g. a cube would render with curved lines on the surface of a sphere).

*Haptic feedback:* Silicone presents a three-dimensional canvas of possibilities for incorporating other elements. Haptic feedback (having the device actuate and move under your hand) could be achieved by embedding an actuation mechanism in the silicone. In the simplest case, this actuation could be a single rumble motor; I would need to test whether the motor would be more effective embedded in the silicone, which would probably dampen it to some extent, or outside the silicone but closer to the top surface. Another option is embedding air bladders during the casting process, and using air pressure to inflate them, although this option would be loud and require a fair amount of external hardware. Muscle wire is low-cost and low-power, but its installation and attachment points would need to be carefully thought out, as silicone tears easily. Piezoelectric elements, magnets, and commercial haptic actuation coils also present possibilities.

Chapter 4 began the core of my thesis with a presentation of silicone sensors built using elements of a vocabulary of sensing structures. The vocabulary is by no means complete or optimal; my hope is that by presenting it, the way of thinking that produced the vocabulary will allow it to be added to and refined over time.

The structures are clearly not applicable to all conformable-material sensing tasks: they are designed to capture discrete, purposeful motions, and fall far short of the goal of capturing the configuration of an arbitrarily scrunched-up piece of material.

In the chapter, I gave example uses that were mainly toys and games; the more serious application space has not been explored. Several possibilities exist, however.

*Replacing the hard accelerometers in the accelerometer t-shirt:*  One possibility is using sensors such as the bend sensor to modify the accelerometer t-shirt described earlier. If a very soft silicone was used, my silicone sensors could replace the hard accelerometers in the shirt, creating a system which was completely soft except for the microcontroller or connecting cable. The silicone sensors would not give global values such as the world-relative accelerations currently captured by the t-shirt, but they could provide local values such as the angle of bend at the elbow or the amount of stretch across a muscle.

*Skin-attached sensors:*  More drastically, by miniaturizing my sensors and using special-effects silicone, which adheres to the skin, I could create stretchy sensors which attach directly to the body. Having skin-adhering sensors would solve one of the main problems with the accelerometer shirt, fabric and sensor slippage. The idea has practical limitations (power and connectors), but the same sensing structures could be used to solve them. One avenue I did not have time to explore is power harvesting: anywhere we have a linear displacement, we could put a power-generator; and the structures I presented show how to turn many common movements into linear displacements.

*Physical rehabilitation tools:*  I have received interest in investigating the silicone sensors' practicality for use by stroke patients to practice and test their motor skills. Silicone comes in a range of hardness ratings, and the shape of the sensing structures can be easily modified, allowing the creation of a range of sensors for different motor strength levels.

In Chapter 5, I create several examples of soft stuffed animals with sensors as part of their foam cores. My claim of "completely soft" stuffed animals was mitigated by several qualifications – a hard connector, hard microcontroller, hard LED eyes. Nevertheless, I expect advances in polymer electronics to somewhat alleviate these issues; incorporating a flexible display into a soft toy might even make sense when the technology is further developed.

The foam-core construction method I demonstrated could be used to cheaply add sensing to other objects containing foam or soft material. Switches could be added to pillows and chair cushions to detect presence, or even posture; carpets could detect a person walking on them; martial arts pads could give a rough estimate of how hard they were hit; packaging materials could record tampering. The construction materials I selected and sensing structures I presented provide guidelines for implementing these and similar ideas.

In Chapter 6, I set forth a workable form factor and construction method for a tongue input device. Further work should test the limitations of the agility of the tongue for input tasks. Before putting this device into production, I would test how many petals the tongue could differentiate between, and analyze the speed/accuracy tradeoffs in using two, four, six, or more petals (perhaps also testing the feasibility of a center button).

Also needing further analysis is the right amount of triggering force for the petals, and the appropriate feel of them hitting the ring beneath. It should be possible to experiment with, and classify, different haptic profiles for the "click" made when a petal hits the silicone ring underneath it. The areas on the silicone ring beneath the petals would be shaped into cavities with flaps or hollows to present a more distinct haptic sensation when the petal makes contact.

In Chapter 7, I only scratch the surface of the possibilities for prototyping flexible robots using 3D printed rubber-like material. It would be worthwhile to consider the possibilities further, as I expect that in the future the material will be durable enough for production use.

I could consider appearance with the same categorical eye as I have considered sensing: How can we design the robot to wrinkle, crease, and bulge predictably? Experimentation, combined with studying subjects like origami folding, might lead to a vocabulary for creating different appearances from motion in flexible materials such as silicone and rubber.

I could study actuation of the robot skin with a similar goal. Local actuation would allow robot-specific movement for a robot mounted on a generic armature, e.g. mounting a robot with real moving lips to the Keepon Pro armature. The building blocks I created could be reversed, with some modification, from input to output: the accordion shape would extend and contract if air pressure were applied, the bend shape modified to unbend. A new vocabulary of air pressure actuation could be created. I could also experiment with mounting actuators in the robot skin, asking questions like what types of motions of the skin we could get out of, for example, restricting to linear actuators. Expanding the scope beyond robots, I could test which simple machines will work when made out of soft materials, and attempt further to create a vocabulary of soft motion to complement my vocabulary of soft sensing.

The development of new soft materials in the future will continue to present exciting opportunities for creating softer, more user-friendly electronic devices. Researchers should be excited, but at the same time, wary of imposing softness where hardness provides constraints that make users more efficient. Binary input buttons are simple, fast, and tactile; freely manipulatable 3D mice have never been close to superseding the standard mouse's (and touchscreen's) swift, easily controllable 2D input.

My dissertation work demonstrates that soft materials show their power when higher degrees of freedom than one or two dimensions are involved. A soft tracking t-shirt allows a human to involve multiple limbs in input with low cognitive load. Silicone input devices give multiple fingers force-feedback simultaneously. Silicone and foam allow broader, more flexible manipulations, whether of electronics accessories, the human tongue, or stuffed animals. And using 3D printed rubber-like material lets us make any arbitrarily large area of a robot skin touch sensitive using a single air pressure sensor.

Thus the soft materials of the future, however flexible and versatile, will continue to present sensing challenges. They will need to be instrumented with sensing infrastructure to capture the high-dimensional configurations that are their advantage. It will be worthwhile to study, at the development stage of a material, how it will be used and what unique properties it presents, or could present, and then build sensing structure as an integral part of the material. For example, a cloth could be constructed to let light pass through only when manipulated a certain way; the structure of a piece of foam could cause it to emit different audio profiles based on where it is being pressed. Having input from device designers at the material level could

open up a whole new range of collaborations. The sensing through structure thought process I demonstrated in my thesis provides a method of brainstorming about these possibilities.

This dissertation is situated within two current trends in computing. The first trend is a general increase in the ubiquity of computational devices. Ebooks, smartphones, electric cars – increasingly, objects which we used to consider passive tools have come into their own as interactive devices. The design focuses of input and interface have moved beyond ergonomics and speed, to intuitiveness, encouragement, and friendliness. I expect that the natural extension of these focuses will be a turn to softer interfaces, which are nonthreatening by their very nature. Flexible displays, stretchable circuits, and other research areas point in this direction. Input device designers will need to confront soft materials more rigorously than before; my dissertation presents one method of thinking about these materials.

The second trend is a resurgence of the do-it-yourself ethic in hardware construction. On the hardware side, drivers of this "personal manufacturing revolution" include affordable 3D printers, accessible microcontroller packages such as Arduino, and easy to work with materials such as e-textiles. The real driving force is less tangible, however: it is the social network that supports the current resurgence in hardware hacking. People are using online blogs, wikis, and websites to show their constructions, modify others', get help and training, and share ideas. The result is everyday people designing interactive devices at an unprecedented scale, and describing their creations online. Such people have fewer preconceptions about how materials and sensors were meant to be used, and readily combine them in unexpected, creative ways.

The descriptions of devices being posted online, on websites such as instructables.com, present a unique opportunity for a data-driven approach to sensing through structure. The vocabularies I have presented throughout this work are not "complete" in any mathematical sense of the word. We can use online data to start to build up a multidimensional dataset of what soft materials are manipulated, how they are manipulated, and what sensors, harnessing which properties, are used to capture those manipulations. From this dataset new vocabularies will emerge, and new guidelines for designing the next generation of soft computing devices.

As the personal manufacturing trend continues, 3D printers will follow the trickle-down path of previous technological innovations, transforming into tools usable by everyone. I see sensing vocabularies parlayed into design tools for 3D printing any multimaterial, soft input device. To create the input device, the user of the software would indicate materials, select from a library of interactions, and model the outer form. The sensing infrastructure and electronics would be computationally designed and optimized, then 3D printed as an integral part of the object. My vocabularies are currently generated through careful mental reduction and experimentation; each element of the vocabulary will work on its own, but I have not explored chaining together multiple elements in close proximity. To achieve this design-tool vision, we will need more rigorously developed vocabularies, with each element provably optimized and characterized with respect to its interactions with other elements.

In this dissertation I have presented six research projects which support the following thesis statement:

> By discovering the natural material and human structure in an input problem, we reduce the problem's dimensionality, simplifying the sensing and creating intuitive input devices.

# Bibliography

[Andreasen Struijk 2006]  ANDREASEN STRUIJK, L. 2006. An inductive tongue computer interface for control of computers and assistive devices. *IEEE Transactions on Biomedical Engineering 53*, 12, 2594–2597.

[Avrunin et al. 2011]  AVRUNIN, E., HART, J., DOUGLAS, A., AND SCASSELLATI, B. 2011. Effects related to synchrony and repertoire in perceptions of robot dance. In *ACM/IEEE Conference on Human-Robot Interaction*, ACM, 93–100.

[Baranek et al. 2005]  BARANEK, G., BARNETT, C., ADAMS, E., WOLCOTT, N., AND CRAIS, E. 2005. Object play in infants with autism: methodological issues in retrospective video analysis. *American Journal of Occupational Therapy 1*, 20–30.

[Betke et al. 2002]  BETKE, M., GIPS, J., AND FLEMING, P. 2002. The camera mouse: visual tracking of body features to provide computer access for people with severe disabilities. *IEEE Transactions on Neural Systems and Rehabilitation Engineering 10*, 1–10.

[Bilmes et al. 2005]  BILMES, J., LI, X., MALKIN, J., KILANSKI, K., WRIGHT, R., KIRCHHOFF, K., SUBRAMANYA, A., HARADA, S., LANDAY, J., DOWDEN, P., AND CHIZECK, H. 2005. The vocal joystick: a voice-based human-computer interface for individuals with motor impairments. In *Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, 995–1002.

[Blow et al. 2006]  BLOW, M., DAUTENHAHN, K., APPLEBY, A., NEHANIV, C., AND LEE, D. 2006. Perception of robot smiles and dimensions for human-robot interaction design. In *IEEE Symposium on Robot and Human Interactive Communication*, 469–474.

[Boissy et al. 2007]  BOISSY, P., CHOQUETTE, S., HAMEL, M., AND NOURY, N. 2007. User-based motion sensing and fuzzy logic for automated fall detection in older adults. *Telemedicine and e-Health 13*, 6, 683–694.

95

[Broadened Horizons 2011a]       BROADENED      HORIZONS,      2011.      Bite      Switch. broadenedhorizons.com.

[Broadened Horizons 2011b]       BROADENED      HORIZONS,      2011.      QuadMouse. broadenedhorizons.com.

[Buechley 2006]       BUECHLEY, L. 2006. A construction kit for electronic textiles. In *IEEE International Symposium on Wearable Computers*, 83–90.

[Chai and Hodgins 2005]       CHAI, J., AND HODGINS, J. K. 2005. Performance animation from low-dimensional control signals. *ACM Transactions on Graphics 24*, 3, 686–696.

[Chumby Industries 2010]       CHUMBY INDUSTRIES, 2010. Chumby. www.chumby.com.

[Coelho et al. 2009]       COELHO, M., POUPYREV, I., SADI, S., VERTEGAAL, R., BERZOWSKA, J., BUECHLEY, L., MAES, P., AND OXMAN, N. 2009. Programming reality: from transitive materials to organic user interfaces. In *ACM Conference on Human Factors in Computing Systems Extended Abstracts*, 4759–4762.

[Corazza et al. 2006]       CORAZZA, S., MUNDERMANN, L., CHAUDHARI, A., DEMATTIO, T., COBELLI, C., AND ANDRIACCHI, T. 2006. A markerless motion capture system to study musculoskeletal biomechanics: visual hull and simulated annealing approach. *Annals of Biomedical Engineering 34*, 6.

[Corporation 2010]       CORPORATION, A. T., 2010. www.ascension-tech.com.

[Darves and Oviatt 2002]       DARVES, C., AND OVIATT, S. 2002. Adaptation of users' spoken dialogue patterns in a conversational interface. In *International Conference on Spoken Language Processing*, 561–564.

[DiMattia and Gips 2005]       DIMATTIA, P., AND GIPS, J. 2005. Eagleeyes: Technologies for non-verbal persons. In *Handbook of Human Development for Health Care Professionals*, K. Thies and J. Travers, Eds. Jones and Bartlett, 429–448.

[Dontcheva et al. 2003]       DONTCHEVA, M., YNGVE, G., AND POPOVIC, Z. 2003. Layered acting for character animation. *ACM Transactions on Graphics 22*, 3.

[Ebert-Uphoff et al. 2005]       EBERT-UPHOFF, I., GOSSELIN, C., ROSEN, D., AND LALIBERTE, T. 2005. Rapid prototyping for robotics. In *Cutting Edge Robotics*. Pro Literatur Verlag, 17–46.

[EnableMart 2009]       ENABLEMART, 2009. EyeTech TM2. enablemart.com.

[Esposito et al. 1995]        ESPOSITO, C., PALEY, W. B., AND ONG, J. 1995. Of mice and monkeys: a specialized input device for virtual body animation. In *ACM Symposium on Interactive 3D Graphics*.

[Evreinov and Evreinova 2000] EVREINOV, G., AND EVREINOVA, T. 2000. "Breath-joystick" – graphical manipulator for physically disabled users. In *International Conference on Computers Helping People with Special Needs*, 193–200.

[Farella et al. 2007]         FARELLA, E., BENINI, L., RICC, B., AND ACQUAVIVA, A. 2007. Moca: A low-power, low-cost motion capture system based on integrated accelerometers. *Advances in Multimedia*, 1–1.

[Felzer and Freisleben 2002]  FELZER, T., AND FREISLEBEN, B. 2002. HaWCoS: the "hands-free" wheelchair control system. In *ACM Conference on Assistive Technologies*, 127–134.

[Follmer et al. 2011]         FOLLMER, S., JOHNSON, M., ADELSON, E., AND ISHII, H. 2011. deForm: an interactive malleable surface for capturing 2.5d arbitrary objects, tools and touch. In *ACM Symposium on User Interface Software and Technology*, 527–536.

[Fontaine et al. 2003]        FONTAINE, D., DAVID, D., AND CARITU, Y. 2003. Sourceless human body motion capture. In *Proceedings of Smart Objects Conference*.

[Furukawa et al. 2010]        FURUKAWA, M., UEMA, Y., SUGIMOTO, M., AND INAMI, M. 2010. Fur interface with bristling effect induced by vibration. In *ACM Augmented Human International Conference*, 1–6.

[Goetz et al. 2003]           GOETZ, J., KIESLER, S., AND POWERS, A. 2003. Matching robot appearance and behavior to tasks to improve human-robot cooperation. In *IEEE Symposium on Robot and Human Interactive Communication*, 55–60.

[Gross and Eisenberg 2007]    GROSS, M., AND EISENBERG, M. 2007. Why toys shouldn't work "like magic": Children's technology and the values of construction and control. In *IEEE International Workshop on Digital Game and Intelligent Toy Enhanced Learning*, 25–32.

[Hakozaki et al. 1999]        HAKOZAKI, M., NAKAMURA, K., AND SHINODA, H. 1999. Telemetric artificial skin for soft robot. *Structure 99*, 844847.

[Hakozaki et al. 2001]        HAKOZAKI, M., HATORI, A., AND SHINODA, H. 2001. A sensitive skin using wireless tactile sensing elements. In *Technical Digest of the 18th Sensor Symposium*, 147–150.

[Harms et al. 2008]          HARMS, H., AMFT, O., ROGGEN, D., AND TROSTER, G. 2008. Smash: A distributed sensing and processing garment for the classification of upper body postures. In *Third International Conference on Body Area Networks*.

[Hasbro 2010]               HASBRO, 2010. BopIt! www.playbopit.com.

[Hegel et al. 2011]         HEGEL, F., GIESELMANN, S., PETERS, A., HOLTHAUS, P., AND WREDE, B. 2011. Towards a typology of meaningful signals and cues in social robotics. In *IEEE Symposium on Robot and Human Interactive Communication*, 72–78.

[Hilliges et al. 2008]      HILLIGES, O., KIM, D., AND IZADI, S. 2008. Creating malleable interactive surfaces using liquid displacement sensing. In *IEEE Tabletop and Interactive Surfaces*, 157 –160.

[Hochberg and Donoghue 2006]  HOCHBERG, L., AND DONOGHUE, J. 2006. Sensors for brain-computer interfaces. *IEEE Engineering in Medicine and Biology Magazine 25*, 5 (Sept.-Oct.), 32–38.

[Hollister 2005]            HOLLISTER, S. 2005. Porous scaffold design for tissue engineering. *Nature Materials 4*, 518–524.

[Hook et al. 2009]          HOOK, J., TAYLOR, S., BUTLER, A., VILLAR, N., AND IZADI, S. 2009. A reconfigurable ferromagnetic input device. In *ACM Symposium on User Interface Software and Technology*, 51–54.

[Huo and Ghovanloo 2009]    HUO, X., AND GHOVANLOO, M. 2009. Using constrained tongue motion as an alternative control surface for wheeled mobility. *IEEE Transactions on Biomedical Engineering 56*, 6, 1719–1726.

[Igarashi and Hughes 2001]  IGARASHI, T., AND HUGHES, J. 2001. Voice as sound: using nonverbal voice input for interactive control. In *ACM Symposium on User Interface Software and Technology*, 155–156.

[Inaba et al. 1996]         INABA, M., HOSHINO, Y., NAGASAKA, K., NINOMIYA, T., KAGAMI, S., AND INOUE, H. 1996. A full-body tactile sensor suit using electrically conductive fabric and strings. In *IEEE/RSJ Intelligent Robots and Systems*, vol. 2, 450–457.

[Innvo Labs 2010]           INNVO LABS, 2010. Pleo. www.pleoworld.com.

[Ishiguro 2006]             ISHIGURO, H. 2006. Interactive humanoids and androids as ideal interfaces for humans. In *ACM Conference on Intelligent User Interfaces*, 2–9.

[Johnson et al. 1999]       JOHNSON, M., WILSON, A., BLUMBERG, B., KLINE, C., AND BOBICK, A. 1999. Sympathetic interfaces: using a plush toy to direct

synthetic characters. In *ACM Conference on Human Factors in Computing Systems*, 152–158.

[Junker et al. 2003] JUNKER, H., LUKOWICZ, P., AND TROSTER, G. 2003. Padnet: Wearable physical activity detection network. In *IEEE International Symposium on Wearable Computers*, 244–245.

[Kadowaki et al. 2009] KADOWAKI, A., YOSHIKAI, T., HAYASHI, M., AND INABA, M. 2009. Development of soft sensor exterior embedded with multi-axis deformable tactile sensor system. In *IEEE Symposium on Robot and Human Interactive Communication*, 1093 –1098.

[Kakehi et al. 2008] KAKEHI, Y., JO, K., SATO, K., MINAMIZAWA, K., NII, H., KAWAKAMI, N., NAEMURA, T., AND TACHI, S. 2008. ForceTile: tabletop tangible interface with vision-based force distribution sensing. In *ACM SIGGRAPH New Tech Demos*.

[Kakehi et al. 2011] KAKEHI, G., SUGIURA, Y., WITHANA, A., LEE, C., NAGAYA, N., SAKOMOTO, D., SUGIMOTO, M., INAMI, M., AND IGARASHI, T. 2011. Fuwafuwa: Detecting shape deformation on soft objects using directional photoreflectivity measurements. In *ACM SIGGRAPH Emerging Technology*.

[Kim et al. 2004] KIM, J. S., JEONG, H., AND SON, W. 2004. A new means of HCI: EMG-MOUSE. In *IEEE Conference on Systems, Man and Cybernetics*, vol. 1, 100–104.

[Koichi 2010] KOICHI, K. 2010. Input device for disabled persons using expiration and tooth-touch sound signals. In *ACM Symposium on Applied Computing*, 1159–1164.

[Kozima et al. 2009] KOZIMA, H., MICHALOWSKI, M., AND NAKAGAWA, C. 2009. Keepon: A playful robot for research, therapy, and entertainment. *International Journal of Social Robotics 1*, 1, 3–18.

[Kuiken et al. 2009] KUIKEN, T., LI, G., LOCK, B., LIPSCHUTZ, R., MILLER, L., STUBBLEFIELD, K., AND ENGLEHART, K. 2009. Targeted muscle reinnervation for real-time myoelectric control of multifunction artificial arms. *Journal of the American Medical Association 301*, 6, 619–628.

[Kwon and Gross 2005] KWON, D. Y., AND GROSS, M. 2005. Combining body sensors and visual sensors for motion training. In *ACM SIGCHI International Conference on Advances in Computer Entertainment Technology*, 94–101.

[LaCourse and Hludik 1990]   LACOURSE, J., AND HLUDIK, F.   1990.   An eye movement communication-control system for the disabled. *IEEE Transactions on Biomedical Engineering 37*, 12, 1215 –1220.

[Lee and Ha 1999]   LEE, J., AND HA, I. 1999. Sensor fusion and calibration for motion captures using accelerometers. In *IEEE Conference on Robotics and Automation*, vol. 3, 1954–1959.

[Lee et al. 2007]   LEE, J. A., CHO, S. H., LEE, J. W., LEE, K. H., AND YANG, H. K. 2007. Wearable accelerometer system for measuring the temporal parameters of gait. In *IEEE Engineering in Medicine and Biology Society*, 483–486.

[Lipton et al. 2010]   LIPTON, J., ARNOLD, D., NIGL, F., LOPEZ, N., COHEN, D., NOREN, N., AND LIPSON, H. 2010. Multi-material food printing with complex internal structure suitable for conventional post-processing. In *Solid Freeform Fabrication Symposium*.

[Loewenich and Maire 2007]   LOEWENICH, F., AND MAIRE, F.   2007.   Hands-free mouse-pointer manipulation using motion-tracking and speech recognition. In *ACM Australasian Conference on Computer-Human Interaction: Entertaining User Interfaces*, 295–302.

[Lukowicz et al. 2004]   LUKOWICZ, P., WARD, J., JUNKER, H., STAGER, M., TROSTER, G., ATRASH, A., AND STARNER, T. 2004. Recognizing workshop activity using body worn microphones and accelerometers. In *Pervasive Computing*, vol. 3001. Springer Berlin, 18–32.

[Malone and Lipson 2008]   MALONE, E., AND LIPSON, H.   2008.   Multi-material freeform fabrication of active systems. *ASME Conference on Engineering Systems Design and Analysis 1*, 59313, 345–353.

[Mannsfeld et al. 2010]   MANNSFELD, S., TEE, B., STOLTENBERG, R., CHEN, C., BARMAN, S., MUIR, B., SOKOLOV, A., REESE, C., AND BAO, Z. 2010. Highly sensitive flexible pressure sensors with microstructured rubber dielectric layers. *Nature Materials 9*, 859–864.

[Masuda and Wada 2010]   MASUDA, F., AND WADA, C.   2010.   Effects of visual stimuli on a communication assistive method using sympathetic skin response. In *International Conference on Computers Helping People with Special Needs*, Springer-Verlag, 189–192.

[Mayagoitia et al. 2002]   MAYAGOITIA, R., NENE, A., AND VELTINK, P.   2002.   Accelerometer and rate gyroscope measurement of kinematics: an inexpensive alternative to optical motion analysis systems. *Journal of Biomechanics 35*, 4, 537 – 42.

[Mazalek and Nitsche 2007] MAZALEK, A., AND NITSCHE, M. 2007. Tangible interfaces for real-time 3d virtual environments. In *ACM Conference on Advances in Computer Entertainment Technology*, 155–162.

[Measurand 2010] MEASURAND, 2010. Shapetape. www.measurand.com.

[Missinne et al. 2009] MISSINNE, J., VAN STEENBERGE, G., VAN HOE, B., VAN COILLIE, K., VAN GIJSEGHEM, T., DUBRUEL, P., VANFLETEREN, J., AND VAN DAELE, P. 2009. An array waveguide sensor for artificial optical skins. In *SPIE Photonics Packaging, Integration, and Interconnects*, vol. 7221.

[Miyashita et al. 2005] MIYASHITA, T., TAJIKA, T., ISHIGURO, H., KOGURE, K., AND HAGITA, N. 2005. Haptic communication between humans and robots. In *International Symposium of Robotics Research*, 525–536.

[Moore and Dua 2004] MOORE, M., AND DUA, U. 2004. A galvanic skin response interface for people with severe motor disabilities. In *ACM Conference on Computers and Accessibility*, 48–54.

[Mori 1970] MORI, M. 1970. Bukimi no tani (the uncanny valley). *Energy 7*, 33–35.

[Moven 2008] MOVEN, 2008. moven.com.

[Natural Point 2011] NATURAL POINT, 2011. SmartNav. naturalpoint.com.

[New Abilities 2009] NEW ABILITIES, 2009. Toungue-touch keypad. newabilities.com.

[Nintendo 2008] NINTENDO, 2008. nintendo.com.

[Numaguchi et al. 2011] NUMAGUCHI, N., NAKAZAWA, A., SHIRATORI, T., AND HODGINS, J. 2011. A puppet interface for retrieval of motion capture data. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 157–166.

[Ó Conaire et al. 2010] Ó CONAIRE, C., CONNAGHAN, D., KELLY, P., O'CONNOR, N., GAFFNEY, M., AND BUCKLEY, J. 2010. Combining inertial and visual sensing for human action recognition in tennis. In *ACM International Workshop on Analysis and Retrieval of Tracked Events and Motion in Imagery Streams*, 51–56.

[OrganicMotion 2008] ORGANICMOTION, 2008. organicmotion.com.

[Origin Instruments 2011] ORIGIN INSTRUMENTS, 2011. Sip/puff Breeze. orin.com.

[Pan et al. 2003] PAN, Z., CUI, H., AND ZHU, Z. 2003. A flexible full-body tactile sensor of low cost and minimal connections. In *IEEE Systems, Man and Cybernetics*, vol. 3, 2368–2373.

[Papakostas 2007]            PAPAKOSTAS, T. V. 2007. Tactile sensor: Stretching the limits. In
                             *International Conference on Intelligent Environments*, 472–476.

[Park et al. 2002]           PARK, S., MACKENZIE, K., AND JAYARAMAN, S. 2002. The wear-
                             able motherboard: a framework for personalized mobile information
                             processing. In *ACM Conference on Design Automation*, 170–174.

[Patel and Abowd 2007]       PATEL, S., AND ABOWD, G. 2007. BLUI: low-cost localized blow-
                             able user interfaces. In *ACM Symposium on User Interface Software
                             and Technology*, 217–220.

[Periard et al. 2007]        PERIARD, D., MALONE, E., AND LIPSON, H. 2007. Printing em-
                             bedded circuits. In *Solid Freeform Fabrication Symposium*.

[Perner-Wilson and Buechley 2010] PERNER-WILSON, H., AND BUECHLEY, L. 2010. Making textile
                             sensors from scratch. In *ACM Conference on Tangible, Embedded,
                             and Embodied Interaction*, 349–352.

[Pietilinen et al. 2008]     PIETILINEN, K., KAPRIO, J., BORG, P., PLASQUI, G., YKI-
                             JRVINEN, H., KUJALA, U. M., ROSE, R. J., WESTERTERP, K. R.,
                             AND RISSANEN, A. 2008. Physical inactivity and obesity: A vicious
                             circle. *Obesity 16*, 409–414.

[Piper et al. 2002]          PIPER, B., RATTI, C., AND ISHII, H. 2002. Illuminating clay: a
                             3-d tangible interface for landscape analysis. In *ACM Conference on
                             Human Factors in Computing Systems*, 355–362.

[Post et al. 2000]           POST, E. R., ORTH, M., RUSSO, P. R., AND GERSHENFELD, N.
                             2000. E-broidery: design and fabrication of textile-based computing.
                             *IBM Systems Journal 39*, 3-4, 840–860.

[Rainey 1926]                RAINEY, P. M., 1926. US Patent 1608527.

[Raskar et al. 2004]         RASKAR, R., BEARDSLEY, P., VAN BAAR, J., WANG, Y., DIETZ,
                             P., LEE, J., LEIGH, D., AND WILLWACHER, T. 2004. RFIG lamps:
                             interacting with a self-describing world via photosensing wireless tags
                             and projectors. *ACM Transactions on Graphics 23*, 406–415.

[Raskar et al. 2007]         RASKAR, R., NII, H., DEDECKER, B., HASHIMOTO, Y., SUMMET,
                             J., MOORE, D., ZHAO, Y., WESTHUES, J., DIETZ, P., BARNWELL,
                             J., NAYAR, S., INAMI, M., BEKAERT, P., NOLAND, M., BRANZOI,
                             V., AND BRUNS, E. 2007. Prakash: lighting aware motion capture us-
                             ing photosensing markers and multiplexed illuminators. *ACM Trans-
                             actions on Graphics 26*.

[Reed 2009]                  REED, M. 2009. Prototyping digital clay as an active material. In
                             *ACM Tangible, Embedded, and Embodied Interaction*, 339–342.

[Ren et al. 2005]        REN, L., SHAKHNAROVICH, G., HODGINS, J., PFISTER, H., AND VIOLA, P. 2005. Learning silhouette features for control of human motion. *ACM Transactions on Graphics 24*, 4, 1303–1331.

[Russell 1987]        RUSSELL, R. 1987. Compliant-skin tactile sensor. In *IEEE Conference on Robotics and Automation*, 1645–1648.

[Saerbeck and Bartneck 2010]        SAERBECK, M., AND BARTNECK, C. 2010. Perception of affect elicited by robot motion. In *ACM/IEEE Conference on Human-Robot Interaction*, ACM, 53–60.

[Salem and Zhai 1997]        SALEM, C., AND ZHAI, S. 1997. An isometric tongue pointing device. In *ACM Conference on Human Factors in Computing Systems*, 538–539.

[Salem et al. 2011]        SALEM, M., ROHLFING, K., KOPP, S., AND JOUBLIN, F. 2011. A friendly gesture: Investigating the effect of multimodal robot behavior in human-robot interaction. In *IEEE Symposium on Robot and Human Interactive Communication*, 247–252.

[Saponas et al. 2009]        SAPONAS, T. S., KELLY, D., PARVIZ, B., AND TAN, D. 2009. Optically sensing tongue gestures for computer input. In *ACM Symposium on User Interface Software and Technology*, 177–180.

[Sato et al. 2009]        SATO, T., MAMIYA, H., KOIKE, H., AND FUKUCHI, K. 2009. PhotoelasticTouch: transparent rubbery tangible interface using an LCD and photoelasticity. In *ACM Symposium on User Interface Software and Technology*, 43–50.

[Sears et al. 2008]        SEARS, A., YOUNG, M., AND FENG, J. 2008. Physical disabilities and computing technologies: An analysis of impairments. In *The Human-Computer Interaction Handbook*, A. Sears and J. Jacko, Eds. 829–852.

[Sensable 2012]        SENSABLE, 2012. PHANTOM Device. sensable.com.

[Shimojo et al. 2004]        SHIMOJO, M., NAMIKI, A., ISHIKAWA, M., MAKINO, R., AND MABUCHI, K. 2004. A tactile sensor sheet using pressure conductive rubber with electrical-wires stitched method. *Sensors Journal 4*, 5, 589–596.

[Shinoda et al. 1997]        SHINODA, H., MATSUMOTO, K., AND ANDO, S. 1997. Acoustic resonant tensor cell for tactile sensing. In *IEEE Conference on Robotics and Automation*, vol. 4, 3087–3092.

[Shiratori et al. 2011]        SHIRATORI, T., PARK, H. S., SIGAL, L., SHEIKH, Y., AND HODGINS, J. 2011. Motion capture from body-mounted cameras. In *ACM Siggraph*, 31:1–31:10.

[Slyper and Hodgins 2008]     SLYPER, R., AND HODGINS, J. 2008. Action capture with accelerometers. In *ACM SIGGRAPH / Eurographics Symposium on Computer Animation*.

[Slyper et al. 2011a]     SLYPER, R., POUPYREV, I., AND HODGINS, J. 2011. Sensing through structure: designing soft silicone sensors. In *ACM Conference on Tangible, Embedded, and Embodied Interaction*, 213–220.

[Slyper et al. 2011b]     SLYPER, R., LEHMAN, J., FORLIZZI, J., AND HODGINS, J. 2011. A tongue input device for creating conversations. In *ACM Symposium on User Interface Software and Technology*, 117–126.

[Smith et al. 2007]     SMITH, J., GRAHAM, T., HOLMAN, D., AND BORCHERS, J. 2007. Low-cost malleable surfaces with multi-touch pressure sensitivity. In *IEEE Tabletop and Interactive Surfaces*, 205–208.

[Smith et al. 2008]     SMITH, R., THOMAS, B., AND PIEKARSKI, W. 2008. Digital foam interaction techniques for 3d modeling. In *ACM Symposium on Virtual Reality Software and Technology*, 61–68.

[SparkFunElectronics 2008]     SPARKFUNELECTRONICS, 2008. sparkfun.com.

[Stiehl et al. 2005]     STIEHL, W., LIEBERMAN, J., BREAZEAL, C., BASEL, L., LALLA, L., AND WOLF, M. 2005. Design of a therapeutic robotic companion for relational, affective touch. In *IEEE International Workshop on Robots and Human Interactive Communication*, 408–415.

[Stivers et al. 2009]     STIVERS, T., ENFIELD, N., BROWN, P., ENGLERT, C., HAYASHI, M., HEINEMANN, T., HOYMANN, G., ROSSANO, F., DE RUITER, J. P., YOON, K. E., AND LEVINSON, S. 2009. Universals and cultural variation in turn-taking in conversation. *Proceedings of the National Academy of Sciences 106*, 26, 10587–10592.

[Strommen 1998]     STROMMEN, E. 1998. When the interface is a talking dinosaur: learning across media with ActiMates Barney. In *ACM Conference on Human Factors in Computing Systems*, 288–295.

[Takenawa 2009]     TAKENAWA, S. 2009. A magnetic type tactile sensor using a two-dimensional array of inductors. In *IEEE Conference on Robotics and Automation*, 3295–3300.

[Tapia et al. 2007]     TAPIA, E. M., INTILLE, S., , AND LARSON, K. 2007. Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart rate monitor. In *IEEE International Symposium on Wearable Computers*.

[Tautges et al. 2011]     TAUTGES, J., ZINKE, A., KRÜGER, B., BAUMANN, J., WEBER, A., HELTEN, T., MÜLLER, M., SEIDEL, H.-P., AND EBERHARDT, B.

2011. Motion reconstruction using sparse accelerometer data. *ACM Transactions on Graphics 30*, 18:1–18:12.

[Tiesel and Loviscach 2006] TIESEL, J.-P., AND LOVISCACH, J. 2006. A mobile low-cost motion capture system based on accelerometers. *Advances in Visual Computing 4292*, 437–446.

[Vinge 2006] VINGE, V. 2006. *Rainbows End*. Tor Books.

[Vlack et al. 2005] VLACK, K., MIZOTA, T., KAWAKAMI, N., KAMIYAMA, K., KAJIMOTO, H., AND TACHI, S. 2005. GelForce: a vision-based traction field computer interface. In *ACM Extended Abstracts on Human Factors in Computing Systems*, 1154–1155.

[Vlasic et al. 2007] VLASIC, D., ADELSBERGER, R., VANNUCCI, G., BARNWELL, J., GROSS, M., MATUSIK, W., AND POPOVIĆ, J. 2007. Practical motion capture in everyday surroundings. *ACM Transactions on Graphics 26*, 3, 35.

[Vogt et al. 2004] VOGT, F., CHEN, T., HOSKINSON, R., AND FELS, S. 2004. A malleable surface touch interface. In *ACM SIGGRAPH Sketches*, 36–36.

[Walters et al. 2009] WALTERS, M., KOAY, K., SYRDAL, D., DAUTENHAHN, K., AND TE BOEKHORST, R. 2009. Preferences and perceptions of robot appearance and embodiment in human-robot interaction trials. *Artificial Intelligence and Simulation of Behaviour*, 136–143.

[webkinz 2010] WEBKINZ, 2010. www.webkinz.com.

[Westeyn et al. 2008] WESTEYN, T., KIENTZ, J., STARNER, T., AND ABOWD, G. 2008. Designing toys with automatic play characterization for supporting the assessment of a child's development. In *ACM International Conference on Interaction Design and Children*, 89–92.

[Wolpaw et al. 2002] WOLPAW, J., BIRBAUMER, N., MCFARLAND, D., PFURTSCHELLER, G., AND VAUGHAN, T. 2002. Brain-computer interfaces for communication and control. *Clinical Neurophysiology 113*, 6, 767–791.

[Woods 2006] WOODS, S. 2006. Exploring the design space of robots: Children's perspectives. *Interacting with Computers 18*, 1390–1418.

[Wulff 1985] WULFF, S. 1985. The symbolic and object play of children with autism: A review. *Journal of Autism and Developmental Disorders 15*, 2.

[Yamaha et al. 1999] YAMAHA, Y., IWANAGA, Y., FUKUNAGA, M., FUJIMOTO, N., OHTA, E., MORIZONO, T., AND UMETANI, Y. 1999. Soft viscoelas-

tic robot skin capable of accurately sensing contact location of object. In *IEEE/SICE/RSJ Multisensor Fusion and Integration for Intelligent Systems*, 105 –110.

[Yin and Pai 2003]    YIN, K., AND PAI, D. K.  2003.  Footsee: an interactive animation system.  In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 329–338.

[Zimmerman 1982]    ZIMMERMAN, T., 1982. US Patent 4542291.