SCALE: Exploring Human-Object Interaction Through Force Vector Measurement

by

Takatoshi Yoshida

M.S., University of Tokyo (2017)

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

Master of Science in Media Arts and Sciences

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2019

Signature redacted

Author ....
Program in Media Arts and Sciences, School of Architecture and Planning
Aug 9th, 2019

Signature redacted

Certified by.. ....
Hiroshi Ishii
Professor, MIT MediaLab
Thesis Supervisor

Signature redacted

Accepted by . ...
Tod Machover
Academic Head, Program in Media Arts and Sciences

SCALE: Exploring Human-Object Interaction Through Force Vector Measurement

by

Takatoshi Yoshida

Submitted to the Program in Media Arts and Sciences, School of Architecture and Planning
on Aug 9th, 2019, in partial fulfillment of the
requirements for the degree of
Master of Science in Media Arts and Sciences

Abstract

I introduce SCALE, a project aiming to further understand Human-Object Interaction through the real-time analysis of force vector signals, which I have defined as "Force-based Interaction" in this thesis. Force conveys fundamental information in Force-based Interaction, including force intensity, its direction, and object weight - information otherwise difficult to be accessed or inferred from other sensing modalities. To explore the design space of force-based interaction, I have developed the SCALE toolkit, which is composed of modularized 3d-axis force sensors and application APIs. In collaboration with big industry companies, this system has been applied to a variety of application domains and settings, including a retail store, a smart home and a famers market. In this thesis, I have proposed a base system SCALE, and two additional advanced projects titled KI/OSK and DepthTouch, which build upon the SCALE project.

Thesis Supervisor: Hiroshi Ishii
Title: Professor, MIT MediaLab

SCALE: Exploring Human-Object Interaction Through Force Vector Measurement

by

Takatoshi Yoshida

This master thesis has been reviewed and approved by the following committee members:

Signature redacted

Hiroshi Ishii ....

Professor, MIT MediaLab

Signature redacted

Fadel Adib.... ....

Assistant Professor, MIT MediaLab & MIT EECS

Signature redacted

Wojciech Matusik..... ....

Associate Professor, MIT EECS

Contents

Acknowledgement

I would first like to thank my thesis advisor Professor Hiroshi Ishii of MIT Media Lab. The door to Prof. Ishii office was always open for the in-depth discussion of ontological research questions, beyond a mere technical arguments.

I would also like to thank the experts who were involved in the validation survey for this research project: Ken Nakagaki, Xiaoyan Shen, Tal Achituv, Koichi Yoshino, Junichi Ogawa, Michel Bove and Yo Sasaki. Without their passionate participation and input, the validation survey could not have been successfully conducted.

I am also grateful for the colleagues in Tangible Media Group, who were supporting me in explicit and implicit ways: Kyung Yun Choi, Jifei Ou, Amos Golan, Daniel Levin, Hila Mor, Joao Wilbert, Joanne Leon, Alice Hong and Deema Qashet. Without their assistive attitude and open-minded conversation, my two-year exploration of master degree in a foreign country could not have been completed.

I would also like to acknowledge Associate Professor Fadel Adib at MIT MediaLab and Associate Professor Wojciech Matusik at MIT EECS, as the second readers of this thesis, and I am gratefully indebted for his very valuable comments on the thesis.

Finally, I must express my very profound gratitude to my parents and to my partner Ikuko Kanamori for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

Takatoshi Yoshida

Chapter 1

Introduction

1.1 Motivation

Force conveys fundamental information in Human-Object Interaction, including force intensity, its direction, and object weight [28] - information otherwise difficult to be accessed or inferred from other sensing modalities. When force is captured during interaction, a wide range of activities can be reconstructed such as way of touch, movement of objects and patterns of body motion. Therefore, it is important to explore the design space of *Force-Based Interaction*, which we define here as contact based dynamic interaction between two objects or between an object and the human body based on force vector direction and amount.

Force-based interaction is involved at different scales in terms of the intensity of loaded force and the size of the interaction area. For instance, force-based interaction can range from actions such as drawing minute letters on a piece of paper (~1g, 1mm), to handling tools on a workbench (~1kg, 10cm), to dancing in a room (~100kg, 10m). Even though researchers have already tackled each respective task, [47, 60], it is ideal if interaction designers are able to explore the wide range of force-based interactions within a single integrated framework.

To show the broad area in HCI, covered by force-based measurement method, I've demonstrated a variety of use cases in this master thesis, including prototyping a tangible interface, recognizing human behaiviors in a residential room, augmenting customer experience in a retail store and achieving seamless interaction with 3d displays. I believe 'Force' could be an overarching approach to capturing Human-Object Interaction.

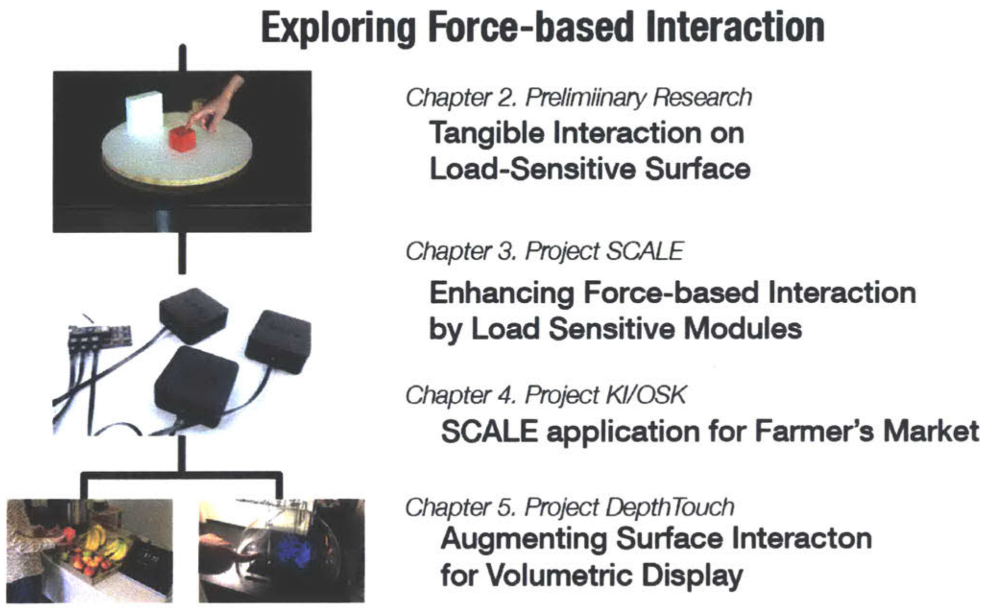

Figure 1-1: Overview of Thesis Framework

1.2 Thesis Outline

In this master thesis, I will discuss on the Human-Object Interaction enabled by Force Sensing method. The overview flow of my thesis is illustrated in Fig. 1-1.

In the chapter 2, I introduce that a preliminary research of the tangible system that localizes the object position and recognizes a simple set of touch inputs. This pilot research has a less impact on the academic community, however reveals the potential of weight information for capturing Human-Object Interaction.

In the following chapter 3, I showed that the SCALE system, the modular load sensors for enhancing force-based interaction. With the significant technical advancement including 3d touch point localization with our patented algorithm, modularized hardware and a variety set of applications. This work is also published at the renowned venue of UIST 2019 conference.

In the chapter 4 and 5, I discuss on the industry-oriented applications of the SCALE system, developed through the project mentioned in chapter 3. With the enormous support from TOPPAN company, a MediaLab sponsor, we explored the application for farmer's market toward augmenting customers UX. Also, in collaboration with Yokogawa Electric.inc we developed an interative system for volumetric display.

1.3 List of Accomplishments

This thesis is composed from many source of accomplishments, conducted in the two years master course, including the publications in the renowned conferences in HCI community, patents currently under preparing with an attorney and industrial collaborations.

1.3.1 Project SCALE

For SCALE project:

- SIGGRAPH ASIA 2018 poster session, 2018.12 @ Tokyo, Japan (reviewed)
- MIT College of Computating Reception 2019 poster session, 2019.02 @ Cambridge, MA (reviewed)
- Media Lab Sponsor Workshop in Panasonic Beta, 2019.03 @ CA, USA
- UIST 2019 full-track publication, 2019.09 (reviewed)

1.3.2 Project KI/OSK

For KI/OSK project:

- Sponsor Collaboration with TOPPAN company, 2018.09 - current
- User Study in an Actual Farmer's Market, 2019.07 @ Tokyo, Japan
- (preparing) CHI 2020 full-track

1.3.3 Project DepthTouch

For DepthTouch project:

- Sponsor Collaboration with Yokogawa Electric company, 2019.05 - current
- (submitted) TEI 2020 full-track

Chapter 2

Preliminary Exploration of Load-based Interaction

2.1 Introduction

In this chapter, we introduce a System for Characterization And Localization of Elements for touch and object location recognition by sensing weight.

In the advent of touch screens, and ever-increasing interweaving of interactive surfaces across scales, it is apparent that more than ever, our ambient spaces are becoming portholes to digital experiences. Phones, tablets, books and now wearables are all providing interactive extensions to our surfaces. These surfaces are designed to be able to track location of touch or tangible objects for physical and spatial interactions beyond 2D surfaces [28]. Researchers have presented interactive touch and object sensing systems that can be enabled by various technical implementation such as computer vision[31], capacitive surface[56], magnetic sensors grid[42] or acoustics sensors[29]. While these previous systems presented rich capabilities for detecting a variety of interactions, weight of objects and force of touch is has remained out of reach for these systems. Additionally, such instrumentation also have limitations for deployability, for example, computer vision requires bulky, expensive systems, that are not always scalable, could suffer from occlusions, and while it is able to detect touch visually, it is unable to sense force or weight. Alternative approaches for tracking and recognizing surface interactions are beneficial for enhancement of user interaction, and offer other advantages such as enhanced privacy since they do not use cameras. Several approaches have been published recently [59, 60, 47, 19] for such sensing at large scales, a comparison of our system to these implementations shows an improvement in resolution and speed, while lowering costs and simplifying implementation. We have constructed several implementations of the system to detect various interaction modalities beyond simple force touch or object location, to include touch on objects, object stacking and weight change within objects (such as the addition of fluid or material). We have demonstrated the functionality and specifications of our system by introducing several applications for tangible and touch based interactions such as drawing, story-telling and tangible interaction prototyping. Testing has shown that even in large scales such as instrumentation of a one square meter surface made of solid wood weighing 20 kg, weight force could be measured down to an accuracy of 0.03 Newtons, with spacial accuracy of under 1 cm, at a sample rate close to 100Hz, with the system as shown in Figures 2-1 and 2-1. Due to these surprisingly sensitive results we were able to fathom applications without the hindrance of performance being as limiting of a factor as we expected from experience with traditional weight-based systems.

In particular, these performance figures allow for the implementation of sophisticated algorithms for real-time interaction. Our initial explorations of these prototypes has been very positive, and users have found them easy to use. Limitations of the system when compared to more traditional approaches, such as capacitive touch, include the systems inability to perform true multi-touch detection, as only a single center-of-mass can be detected at any moment. This limitation can be overcome in many situations where there is temporal difference between the multiple touch interactions. In addition, since the instrumentation of a surface with our system is fairly cheap and straightforward, a combined system can easily benefit from both sensors' capabilities. These hybrids can be implemented through direct integration into a product, or through a temporary link between a product and a surface, such as when placing a tablet on top of our system instrumented table, applications running on the tablet

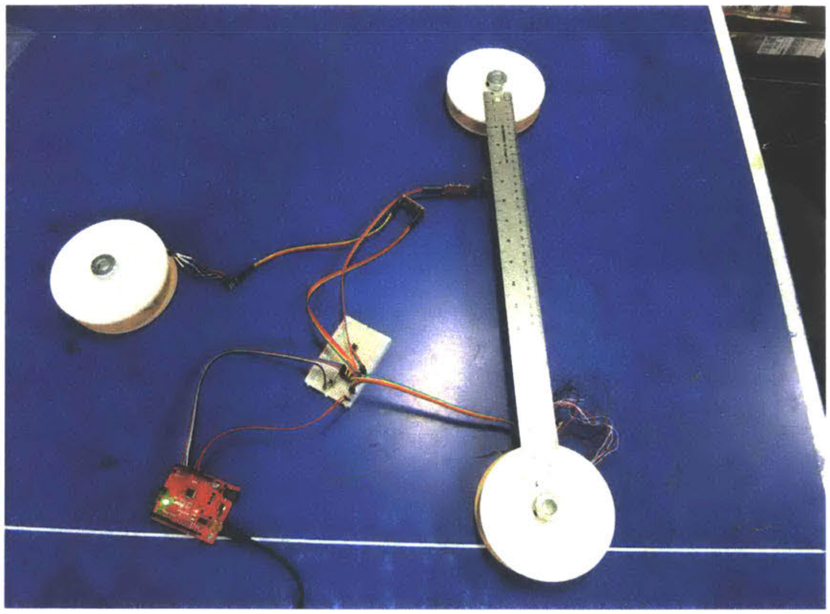

Figure 2-1: Preliminary prototype of the load sensors and acquisition system

can respond to touch and force from both sensors at once. In this chapter we focus on the interactions achievable through our system alone, to demonstrate its versatility and the capabilities achievable through high resolution, high speed, sampling of load, with a focus on tabletop and horizontal surface interactions.

2.2 Related Work

2.2.1 Touch and Object Sensing Tabletop Interfaces

In the field of HCI, tabletop or plane-based interfaces have been one of the major form factors that allows for a variety of touch interaction and token-based tangible interfaces [28]. One of the primary research agendas for the tabletop interface has been on the sensing technology for detecting the physical interactions, which has been proposed in the past few years with a variety of techniques.

One of the popular object recognition techniques is computer vision systems with camera and markers. reacTable introduced musical control Tangible Interface with a marker-recognition technique [31]. ForceTile utilized deformable gel with optically trackable dots to detect the object's position and touch (force and position) interaction to the object [32]. Lumino introduced tokens that incorporate glass fibers for a tabletop system to detect stacking interactions [7]. While these system were widely implemented in research purposes, they are usually expensive, bulky, and require special stick-on tokens according to expected interactions.

Using piezo microphones, acoustic sensing techniques have been introduced for cheap and space efficient techniques for detecting touch on arbitrarily shaped objects[51] and balls hitting on a ping-pong table [29]. While these systems are generally low cost and easy to instrument, they lack the sensitivity and speed to generate more rich interaction.

With capacitive sensing surfaces, Rekimoto demonstrated a touch and object sensing interactive system [56]. This technique has been widely explored especially in last decade as capacitive touch screens have been widely commercialized [35, 13]. Apple's force touch (or 3D touch) added extra dimension on 2D touch interactions for Trackpads and touch screens [8]. Force touch is currently only available on small surfaces and responds to force but not to object based interactions.

Liang et al. developed GaussSense which is a magnetic sensor grid to detect 2.5D motion of magnet-enclosed stylus and tangible tokens [41, 40]. The pcb-based system was able to be attached to any display to enrich the interaction beyond touching the surfaces. While their series of research cleverly utilized the characteristics of magnets to identify various interactions such as stacking and assembling [39, 42], scalability of the system is a primary issue. The size of interaction surface is limited to the number of sensor grids, the larger the surface the higher the cost, while with load-cell based sensing such as our system the costs remain relatively fixed with respect to size, and scale with weight of the instrumented object.

2.2.2 Load cell based interactive system

Lastly, multiple load sensors have been utilized to detect touch and objects on a surface. This approach is easily implemented by placing the sensors under a rigid surfaces and spatially efficient. This technique is also scalable that same number of sensors can be used to sense activity at the scale of a tabletop or an entire room.

Previously, Schmidt et al. presented load cell-based sensing system for object and human behavior tracking with a room scaled surface [59]. They also presented to use similar system for pointing device that is incorporated into ubiquitous environment (e.g. tables) [60]. Murao et al. improved the such system for multiple objects detection and accurate sensing algorithm based on machine learning [47]. In our work, we expand on that approach and achieve improvements in accuracy while making the system simpler, customizable, and cheaper. The faster sampling rates and the simultaneous sampling of all sensors enable novel interactions not previously shown on simple load cell based systems.

Our system is also capable of sensing advanced interactions such as touch on placed objects and stacking objects which were partially explored in recent research using a proprietary industrial multi-directional force sensor which is two orders of magnitude (nearly x300) more expensive than our system [24].

2.2.3 Original Contribution

From having studied related work, our contribution builds upon the use of force measurement contact sensing with increased accuracy and decreased latency, thus improving the overall user experience for HCI applications. With the system we have developed we are able to implement applications that require fast response times and accurate position sensing.

By testing multiple weight and location-specific applications, we are able to provide a platform upon which users can execute a broad range of activities that include weight as a measurable parameter, a unique quality that is not possible with standard tabletop touch screens and surfaces.

The design of our system is such that it can be made with low cost, off-the-shelf hardware making deployment economic and applicable to a diverse user-base. With the applications we have proposed herein, we have made possible the prototyping of use-cases for accurate, force-inclusive interactions with limited hardware and simple instrumentation. Often an existing instrumented surface can be used for multiple prototyping explorations simply through placement of simple constructions on top of the surface. For example existing architecture foam-based models can come alive though placement on top of an instrumented table. Interaction with the model through placement of objects and touching the model is enabled seamlessly and without any modifications to the model itself.

In this paper we focus on two types of evaluation: (1) a quantitative evaluation of the system's resolution in time, space, and (2) a qualitative user experience evaluation of prototypes utilizing the system.

The improved resolution demonstrated in the quantitative evaluation leads us to believe that the system could be used for user identification tasks and other similar tasks that have been shown to be possible based on load sensor data as discussed in the related works section above. We expect that increased resolution should lead to same-or-better performance for these existing tasks. Instead we choose to focus in a new direction towards the strengths derived from the capabilities of the our system and evaluate its compatibility as an implementation of a TUI prototyping device - enhancing existing objects and sensors merely by placing them on top of the surface installed with our system, where the properties of force sensitivity and localization can be applied to any item, for example any laptop can be enhanced to include force sensitivity touch-pads, and even the entire surface of the laptop can become an input device. Under certain restrictions, and with additional calibration, force sensitivity can even be applied to the screen on a laptop placed on top of our system. We think of this approach as Rapid Prototyping for tangible interactions, this could be thought of as a TUI equivalent of GUI paper-prototyping.

2.3 Implementation

2.3.1 Approach

Our approach utilizes inexpensive off-the-shelf components, with minor modifications in hardware and software, which achieve an order of magnitude improvement over previously published results. Improvements are shown in weight sensing resolution, weight sensing accuracy, and response times. These material improvement, in turn, translate into improvements in response time, object detection capabilities, noise avoidance and rejection, and dramatically reduce and even eliminate (as some of our studies results suggest) false object detection. At its current performance characteristics the system has been demonstrated to be an effective tool for HCI. We primarily explored its usability as an interaction prototyping platform, as well as a platform which enhances existing interfaces with force sensing capabilities.

2.3.2 Design Goal

Our main design goal was to construct a system that could be realistically easily deployed in many environments, thus cost and flexibility of integration were heavily favored. Minimizing cost while maximizing performance is always desirable. To achieve a good balance of cost and performance we optimized key performance bottlenecks across both software and hardware, as described in more detail later.

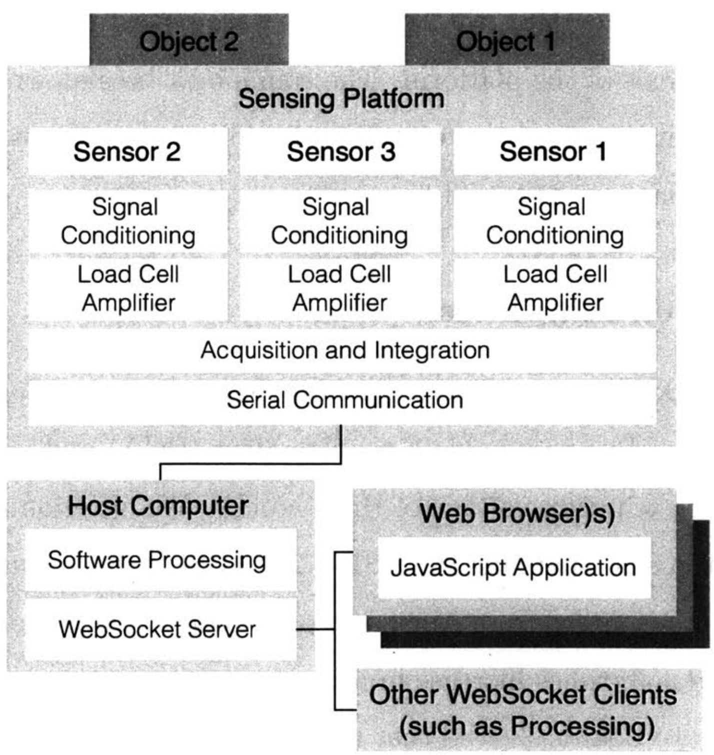

Figure 2-2: Sensing platform diagram.

2.4 Architecture

The system consists of three layers: The hardware sensing platform, an intermediary software processing layer, and the final client layer. This architecture enables concurrent applications to run at the same time and/or on multiple devices.

The system is made using three force sensing units, implemented using load-cells, signal acquisition (amplification and conditioning), data integration and processing, and an application layer which can be implemented on the host computer or distributed over a network, see figure 2-2. For reasons detailed below, use of three load sensing units improves performance over four (or more) based systems, while obviously also lowering cost.

2.4.1 Hardware Design

While the system can operate with any number of sensors greater than 3, construction using 3 sensors greatly simplifies processing and keeps costs to a minimum. To further simplify processing of the data, the load sensors are placed in a symmetric manner around the central axis of the platform. The component breakdown and architecture can be seen above in Figure 2-2, while the selection of the various components and related considerations are explained below.

2.4.2 Load Cell Amplifiers

HX711 **-** The HX711 is essentially a 24 bit Sigma-Delta Analog-to-Digital Converter (ADC) with a Programmable Gain Amplifier (PGA) and dedicated circuitry for controlling the excitation voltage of the load cell. We have found several sources for HX711 circuit boards spanning the USD 1.20 to USD 9.99 per unit. A comparison of the cheap circuits (sourced from Amazon) to the more expensive ones (sourced from SparkFun) showed a 40-times increase in noise levels. Comparison of the components and layout of the two boards showed that the cheaper board is missing a 3.3uH inductor and a 10uF capacitor in a low-pass filter configuration on the excitation circuitry compared to the more expensive, less noisy boards. With the load-cells in our tests, adding the capacitor alone reduced noise levels such that the cheap board's performance was indistinguishable from the more expensive one. These boards both lack any electromagnetic shielding present on boards from other manufacturers. In any mass manufacturing design, shielding these circuits from electromagnetic interference is highly recommended due to their sensitivity to noise. In our tests described here no shielding was applied. The HX711 has an internal oscillator and supports sampling at either 10 Hz or 80 Hz. In all our experiments and implementations, we used it in the 80 Hz mode, since we found that the added temporal resolution contributes greatly to the responsiveness of our system, while the added noise from the faster sampling averages at lower noise overall after software filtering, as compared to the 10 Hz mode. We have found that the actual number of samples per second achievable is 90 sps.

Sampling Library - The HX711 is digitally controlled through a serial interface consisting of a clock pin and a data pin. Many libraries exist for interfacing with the HX711, and some even support sampling of multiple HX711 units. However, all the libraries we could find implemented the concurrent sampling internally as sequential sampling, meaning that any benefits of concurrency were merely for the sake of a clean software design rather than any real benefit in hardware and timing. To improve on this bottleneck we published a new library[2] which supports simultaneous reading of multiple LCAs. The maximum number that could be read simultaneously is limited by the speed of the microcontroller used, as well as timing variations within each LCAs internal oscillator - even though the data being read is clocked by the reader, timing between completed reading and the next sample being ready varies. On an Arduino Uno, we have been able to read up to 5 LCAs simultaneously, and on other system were able to extend that number further with successful attempts of reading 6 and 12 sensors, and theoretically at least a few dozens should be easily possible.

2.4.3 Load Cells

For our load cells we picked the readily available bar-style load cells of model TAL220 rated at 10 kg, measuring 80mm (L) x 12.7mm (W) x 12.7mm (H), as shown in Figure

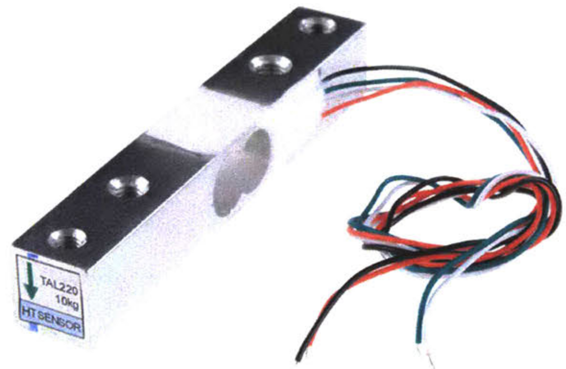

Figure 2-3: TAL220 10Kg Full-bridge Load Cell

2-3. The load cells are fixed to a shared rigid platform on the bottom, and their other end is attached under the surface to be measured. Force applied to these cells causes deformation of the cell which in turn changes the resistance in a wheatstone bridge. Load cells in this form are extremely common and can be acquired for between 1 and 10 USD from a variety of sources at the 10 kg capacity. Load capacities available span from grams to tons, and cost generally tends to increase with capacity and, of course, production quality. The achievable resolution with a given amplifier and environmental conditions decreases as the load cell capacity increases. Another important consideration is the load cell's ability to measure both negative and positive forces, to allow measurement of force outside the bounding polygon of the load cells, as described below. When choosing, placing, and measuring the load cells, their characteristics should be taken into careful consideration. The important factors include:

Ultimate Capacity - The load cell is rated for a certain maximum force, in our case 10kg. Application of force above this limit can damage the load cell permanently. Usual values for load cells are around 150 percent of the rated maximum weight for a safe overload, and around 200 percent for an ultimate overload. While our standard overall capacity is 30kg in the small-prototype(Figure 2-5) due to the use of three such load cells, it is important to ensure no single load-cell is overloaded, as there are places on the platform where the majority of the force would be applied to only a single cell. These capacities should be observed even when the system is offline, such as during transport or storage. As described in more detail below, our large-prototype (table-top) (Figure 2-6) has an increased capacity of 90kg overall through a hierarchical design.

Off axis forces - When a force is applied to a cell off of its primary axis, it will gradually be dampened. Designing the interconnect between the surfaces to transfer force as much as possible on the axis is desired, especially in cases where shear forces / torque are expected, and not only push/pull forces. If such forces are expected, their adverse effects can be minimized by mechanical conversion, or by measurement and compensation. If cost is not an object, 6-degree load cells exist which can measure forces in all axes, including torque forces. Such a cell has been used in the INTACT system [24]. In our large prototype we utilize a hierarchical implementation which also shows promise for measuring off-axis forces while still maintaining a low cost. We expect to be able to report on the results of such experimentation in the near future.

Hysteresis - The cell's measurement stabilizes slightly differently depending on the direction from which the load cell approaches the final value. This means adding a certain weight will result in a different displacement magnitude than removing the same weight. Consequently, adding and then removing a fixed weight could result in a shift from the previous stable measurement even though the weight on the load cell is the same. This translates into inaccuracy in the ability to measure precise weight, but can be taken into account when measuring weight differentials, especially when the recent history of force applied to the cell is known.

Temperature coefficient - The relationship between the measurement and the force applied on the load cell is temperature dependent. This could have an effect on the order of 1 percent of full range per 1 degree Celcius. This means that the wider the workable range of the load cell the worse the temperature effect would be. This could be compensated for, and is the appropriate coefficients usually are included in

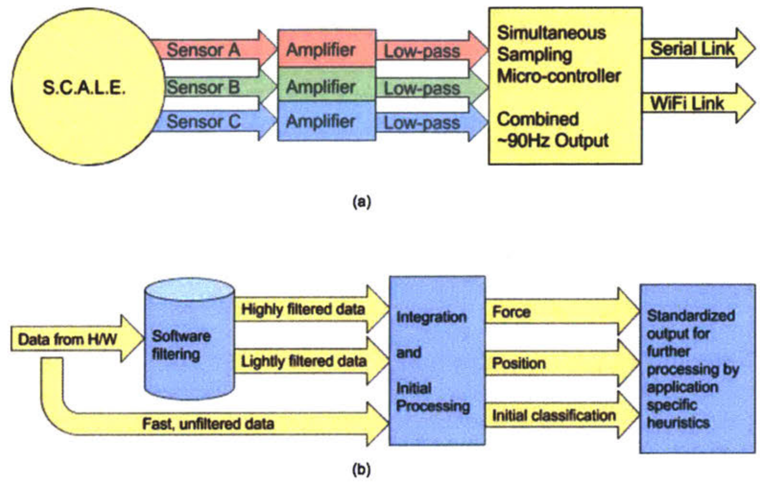

Figure 2-4: (a) Hardware data flow (b) Software data flow

the data sheets, especially for industrial, expensive, brand name load cells.

Non-linearity **-** The response of the load cell, especially if a cheap load cell, or if the cell is being utilized close to the edges of its workable/sensitive range should not be expected to be linear. This can be compensated for by measuring the load cell's response over the desired range with known weights and calibrating for linearity by inverse interpolation across the known measurements. It is important to note that non-linearity and hysteresis interact, meaning that for a truly accurate calibration, linearity should be characterized both by adding and removing weights.

Creep **-** Creep is the difference between the first stable measurement after a change in force, and a later measurement without any change in force. There are two categories: Creep Response - following a increase in force, and Creep Recovery Response - following a reduction of force. Mostly, creep is derived from thermoelastic effects in the metal from which the cell is made. The time until final settling of the measurement can range from minutes to hours, for load cells that are sufficiently well made to even reach a stable measurement. Creep magnitude is typically around 0.02 percent of the full scale load, while creep-like effects can appear to be greater than this figure due to other components in the system, especially with cheap LCAs that lack stability and compensation mechanisms.

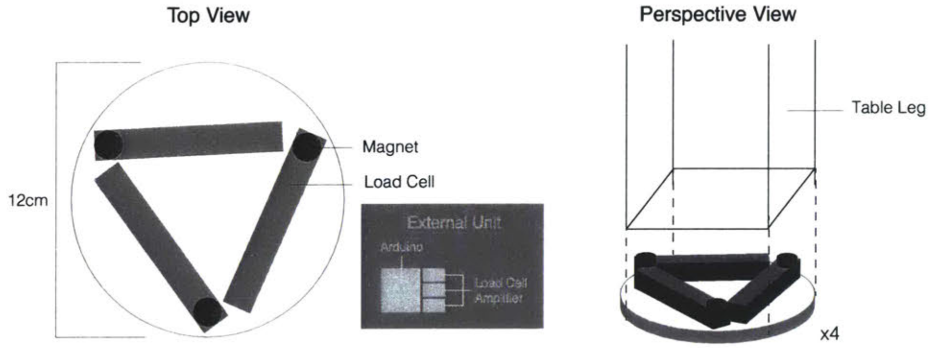

Figure 2-5: Small load cell platform diagram.

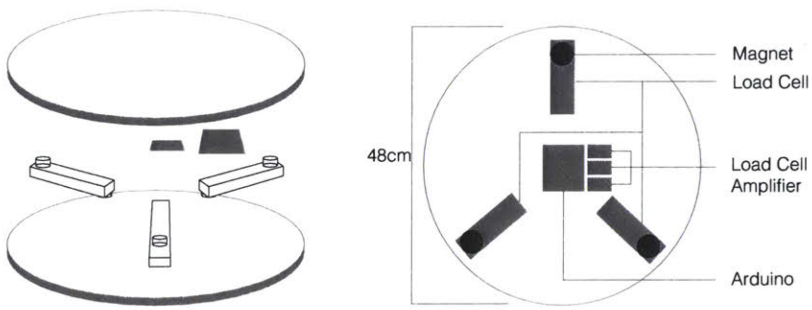

Figure 2-6: Large load cell platform diagram.

2.4.4 Sensor Placement

Three Point Contact - Many of the household human scales utilize four half-bridge cells, to achieve both a robust base for supporting the significant weight of a human as well as lower cost. For these reasons, it might be considered intuitive to implement systems such as these with four load cells. However, there is a big disadvantage to using four contact points for measuring the position of force on a plane. If modeling the surface as completely rigid and non-compressive, application of force to such a surface suspended over flexible load sensors inherently means tilting the surface. Since a plane is defined by three points, measuring such tilt using four points is over constraining the problem, this means that the data acquired has an unknown distribution across the sensors, greatly limiting the achievable accuracy. We found the result through the preliminary experiments, however, this could be numerically explained in future works.

Reducing the sensor count to three sensors resolve this issue. In addition lowering the sensor count lowers the overall cost of the system, however at the cost of increased complexity in sensor placement selection. The results presented in this chapter strongly suggest that if accuracy is desired investing in properly connecting the units to the surface such that exactly three sensors can be used pays dividends. For these reason, we chose to implement a three point contact mechanism and the improvement in accuracy in force localization are shown in the evaluation section to be an order of magnitude better (up to x40) more sensitive.

Surface attachment - In some situations, it could be tricky to implement a three point contact system as it requires bidirectional force coupling in order to cover the entire surface, and is not as intuitive as using 4 sensors, as described above. In our prototypes we attached the sensors to a strong base with screws, and the sensed (top) surface to the load cells using several approaches, for permanent installations we used either screws or epoxy glue, and for temporary installations or when we wanted interchangeable surfaces we used strong (1" x 1/8", N52, round) neodymium magnets, this allowed for quick change of the surface for evaluation purposes. We used square tops as well as round ones, utilizing a variety of materials including stretched canvas, aluminum, wood, and acrylic. The evaluations for performance in this paper were conducted on acrylic and wood surfaces. Further and more rigorous evaluation is required to characterize the relationship between rigidity of the surface and accuracy.

2.4.5 Scaling up

3x3 hierarchy - To scale our system up from a personal work-area sized system to a full table-top system, and beyond, we could have replaced the load cells with ones with a higher load rating. However, we chose to implement a hierarchical system utilizing the same TAL220 load cells for a number of reasons: keeping costs down, examining flexibility of the design in stretching the same component, and for enabling future examination into off-axis force measurement as might be enabled by having each load sensor be a node capable of measuring the distribution of force and not just it's total sum. This exploration has so far presented positive results as a cost-effective method for extending the capabilities of futuredesigns.

2.4.6 Software Design

The software architecture for our current prototypes is entirely implemented in Processing, a Java based programming environment suitable for such experimental platforms, and includes easy debugging and visualization tools showing the raw data as well as filtered results.

We further implemented general heuristics in the software for detection of objects and user interactions in support of more application specific developments. Low-pass filtering of sensor data, computed force position and weight are all provided.

Low pass filters are used both to easily configure for an optimal experience, depending on the goal to optimize for, as well as to simplify heuristics - as in comparing a fast and slow low-pass of sensed weight to determine 'stability' [stable/unstable]. We used a simple implementation of a single-pole infinite impulse response filter, and added to it noise rejection capabilities tailored for load cell driven data, rejecting samples that would not make sense in the context of this system.

Figure 2-4 above shows the data flow in hardware and software. As can be seen, the raw data from the hardware is split into three streams with different levels of low pass filtering. The raw data, lightly filtered but somewhat delayed (milliseconds) and highly filtered data (delayed by up to a second) is integrated and processed in a generic fashion on top of which all of our applications were later implemented. This generic process produces the force and position data as well as an initial classification of the action performed amongst the basic set: object addition, removal, stacking, and touch. Detection of movement, pouring action, etc. is handled by application specific heuristics, in order to keep the generic platform simple, and widely applicable.

On top of the generic system different software decisions can use different types of filters to properly balance response time and accuracy. In many interaction situations decisions can be *updated*, a quick decision can be made using the raw (or lightly filtered) data, and later on updated given the newer more robust data. This can be performed in the millisecond scale, and above, as required. We implemented object recognition, dragging, and interaction (touch, pressure, on or outside objects) in our

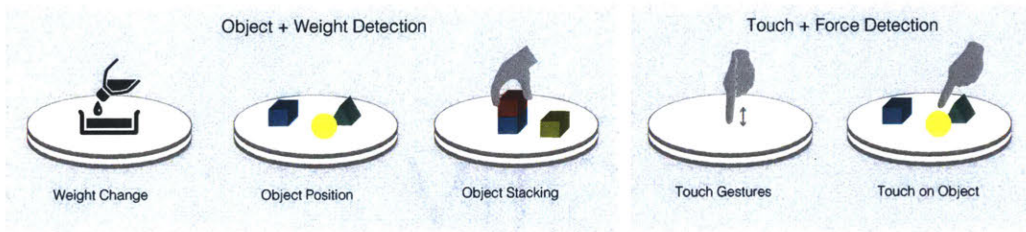

Figure 2-7: Interactions the system can identify

prototypes, as well as several basic event-handlers for integrations with visual or auditory extensions - such as playing audio in response to placement or touch of objects.

2.4.7 Auto-calibration and simultaneous forces

A limitation of the system, as previously discussed, includes the lack of ability to detect simultaneous forces. As such, if a user leans against the platform the calibration could be thrown off. For the weight signal, we have dealt with these issues through relative processing of weight rather than absolute processing, while for the position signals a more application-specific approach is required. For example if one would like to detect typing on a keyboard, the structure of the typing (frequency, position, length) can be easily filtered out from the slower moving, heavier actions of the user's body changing posture, or a second user pushing on the platform, etc.

2.4.8 Interactive functionality

The system allows for two main interactive modes: object detection though specific weight sensing, and touch detection through force sensing based on quick fluctuations of sensor data as shown in Figure 2-7 below. The sensing for each allows for a range of activities to be performed such as object detection with items placed upon the platform, and human touch gestures making contact with the platform, not limited to one or the other.

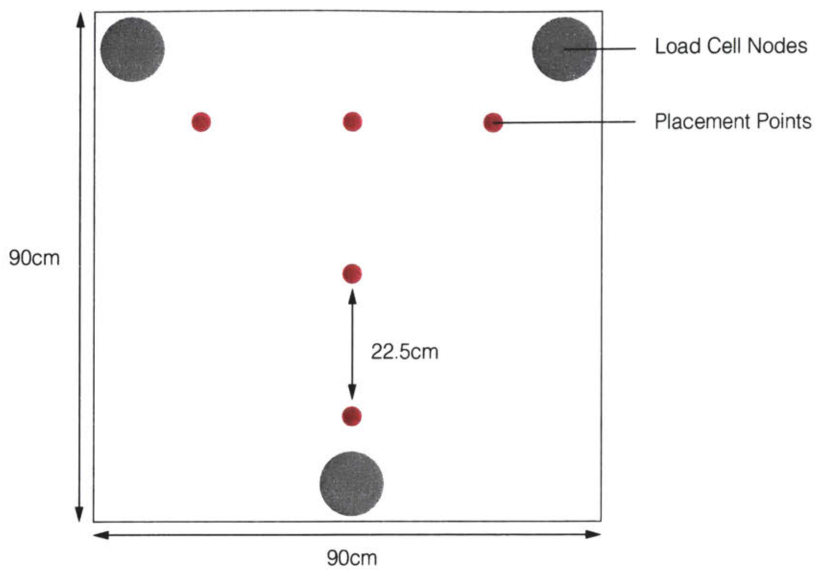

Figure 2-8: Experiment Setup

2.5 Technical Evaluation

2.5.1 Quantitative Evaluation

For our evaluation we followed the procedure used by Murao et al. [47] wherein objects of different weights are placed repeatedly on several different points on the surface and their recognition and position were recorded. In our experiment we placed three different objects weighing 300, 600, and 900 grams each on five different points on the table, in a T-shaped layout as can be seen in Figure 2-8.

The objects were each placed on each point five times, in total 75 object placements were performed and the detected weight and position were recorded for each placement. The overall position and weight errors per object can be seen in Table 2.1.

Using the same size platform and objects of the same weight as Murao et al. [47] used our system performed a full order of magnitude (x10 to x40) better on both positional accuracy as well as force accuracy.

	Position		Weight	
Object (weight)	Avg [cm]	Max [cm]	Avg [g]	Max [g]
300 g	0.77	1.50	0.20	5.20
600 g	0.40	1.20	0.80	11.8
900 g	0.55	1.41	2.00	5.00

Table 2.1: Position error and weight error

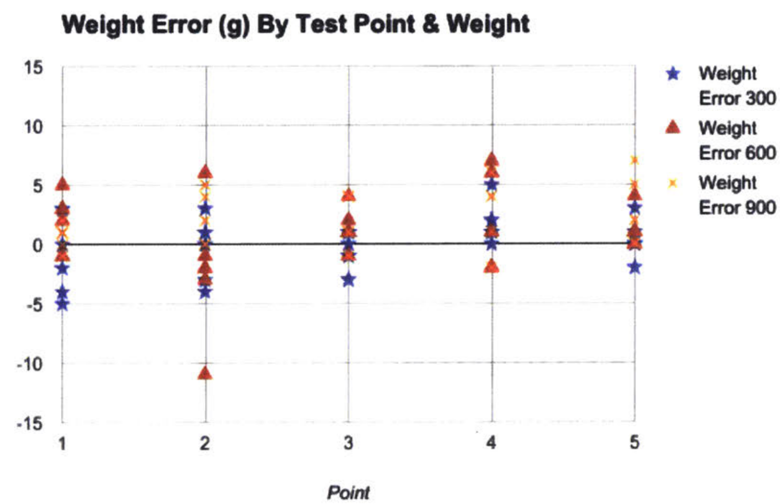

Figure 2-9: Distribution of weight error

2.5.2 Position Detection

The overall error in position detection was 0.57cm on average, and 1.5cm maximum error. This amounts for an error on the same order of magnitude as the human error in placement, which was observed during the experiment. While this is not sufficient evidence to determine that the error in sensing is at any lower order of magnitude, the stability of the signal suggests strongly that this is the case. This can be shown experimentally, but is out of the scope of this paper. We plan further examination of the accuracy for position detection on a variety of surface types, as previously mentioned. For this purpose a robotic placement device will be utilized to eliminate the human factor.

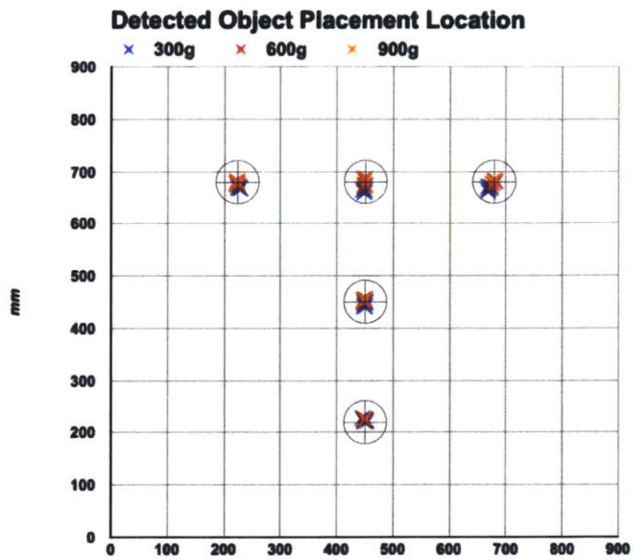

Figure 2-10: Plots of detected points of object placement. Cross-hairs show ground truth.

2.5.3 Weight Detection

The overall error in weight detection was 0.87g on average, and 11.8g maximum error. If removing the single outlier of 11.8g out of the 75 placements, the worst case in all other 74 placements was 7g only. This suggests that better filtering on the software side can keep the error to within 7 grams, which is well within the capabilities of the load cells.

2.5.4 Qualitative Evaluation

We constructed a dozen prototypes including several custom software layers on top of the generic one previously described. The qualitative evaluations were mostly done in-house, and we are gearing up production of more prototypes for more rigorous qualitative testing in the wild. In these evaluations of prototypes, which are further described in the applications section below, we observed that the users have had no trouble interacting with the system, that interaction was natural, and that there were no complaints of latency or confusion. This is far from sufficient testing, but is suggestive of the positive results we expect to achieve from the next stage of testing these surfaces outside of the lab.

2.5.5 Discussion

The accuracy of the system, especially given its low cost, is considerably better than other systems recently published. We attribute these improvements, as previously discussed, primarily to using three (rather than four or more) sensor design, the intelligent data filtering (in both software and hardware), and the simultaneous reading of the load cells.

The experiment results suggest that placement error can be further improved, as the error direction seems to correlate with weight, as can be observed in Figure 2-10. This might be achievable with a linear weight-dependent factor, though further experimentation is required to clarify. It is conceivable that the bias in placement detection is not really weight dependent but rather stems from experimenter introduced bias, since the experimenter might hold the different objects differently, etc. However we conclude this to be unlikely, especially since the bias is consistent across all five points, albeit at different magnitudes. Further experimentation could clarify these points.

While the number of placements used in the quantitative experiment described here is rather low (75), the experimental data has a very low standard deviation, and we have observed the same accuracy across all the prototypes we constructed. The limited evaluation was performed at a larger scale than our other prototypes to enable a more direct comparison with current state of the art systems which tend to be larger than the prototypes we created, as those were more focused on TUI interaction prototyping. We hope that the proof of concept for such interactions shown here will give rise to more prototypes, which would enable a more rigorous comparison across different implementations.

2.6 Applications

By developing a multi-purpose surface that detects location, movement and weight, we are able to prototype numerous applications, as well as utilizing everyday found objects and applying interactive qualities to them, as shown in Figure 2-11.

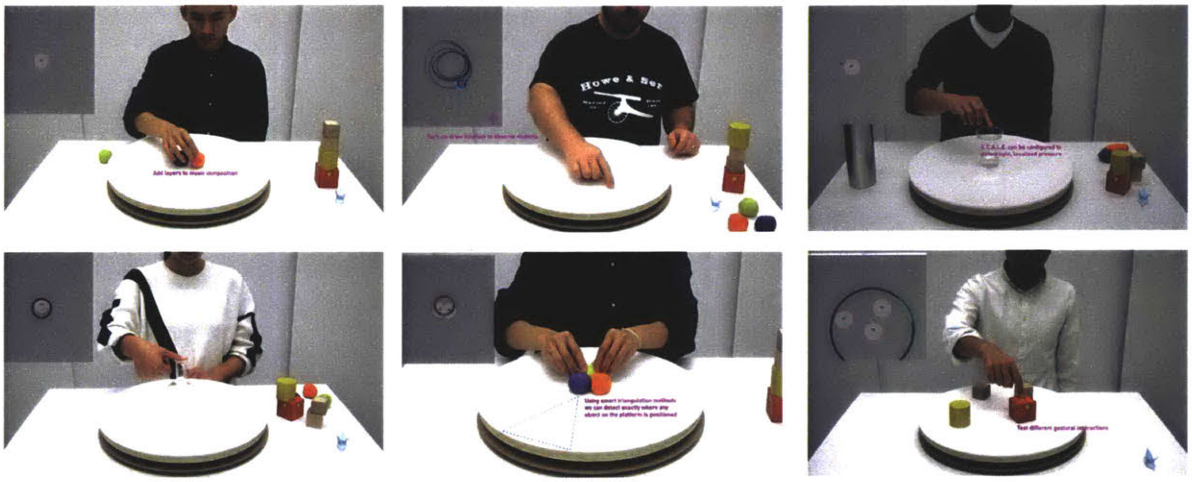

Figure 2-11: Images of applications using found objects as controls. Including software screenshots (top left of each image)

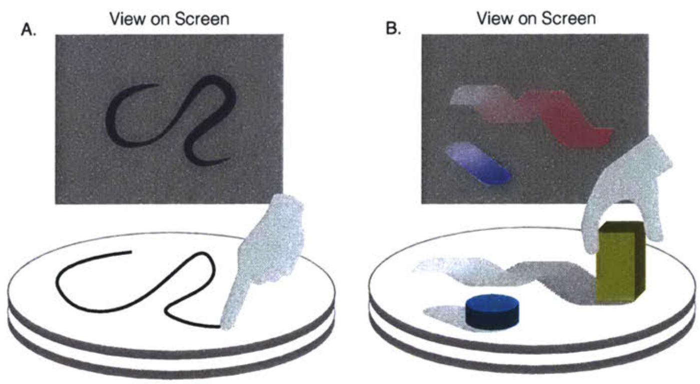

Figure 2-12: Drawing applications (A. Force-based drawing affecting stroke width. B. Object-based drawing with assigned visual brushes.)

2.6.1 Drawing platform

Our first application builds upon the localization functionality of the system by providing a drawing platform that responds to movement and pressure by accurately translating the location point of the user's hand placement. Users can draw using a finger with variable force for variable thickness stroke (see Figure 2-12A). Also, by training the system to recognize an object, we can assign different virtual brushes (size, color or shape) to any found object to create a range of visual feedback for painting and drawing applications, letting the affordance of the object guide the interaction (See Figure 2-12B).

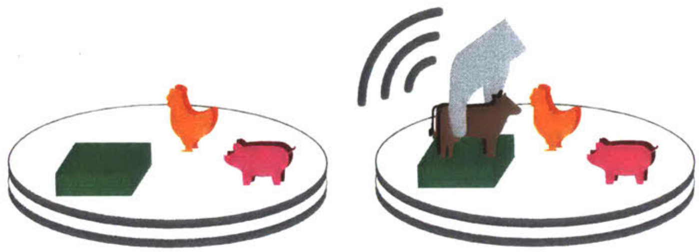

Figure 2-13: Farm animals placed on platform, including stacking functionality.

2.6.2 Interactive storytelling

By applying distinctly weighted characters, for example wooden farm animals, upon the surface, the system can identify the animals and its position. Accordingly, sounds or visual feedback from separate speaker or display can be provided for enhancing imagination of children for storytelling. Stacking functionality can also be utilized to detect specific characters being on top of a specific block (e.g. a cow on grass), to add a layer to the storytelling (Figure 2-13). As long as there is a difference in the weight, any objects can be identified in the system so that children can bring there favorite toys in the application.

2.6.3 Interaction prototyping for User Interface Design

The use of our system for interaction-design prototyping such as interactive wireframes, allows for feedback such as sounds or vibrations to be incorporated into the early phase of the design process in GUI based prototype (see Figure 2-14A). This idea can be extended to TUI design, by the system recognizing physical object of the weight. A good example utilizing the characteristics of our system can be MusicBottles by Ishii et al. [27]. The stacking functionality lets the system identify if a bottle is being opened by detecting the cap's weight (see Figure 2-14B).

2.6.4 Force-based Touch Gesture Controls

By applying various forces within a gestural interaction, controls can be trained to understand what gesture is being made, as well as tracking the dynamics of the movement to provide extra intuition about a user's behaviors (see Figure 2-15). For

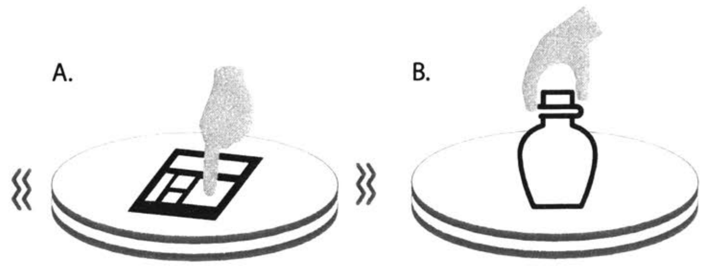

Figure 2-14: A: Touch interaction with paper-based wireframe prototype, and B: opening of MusicBottles-like TUI on the system [27].

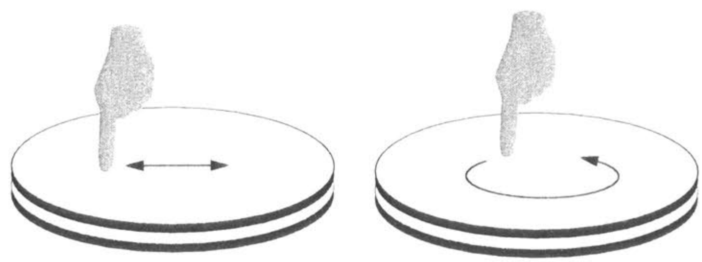

Figure 2-15: Force-based gestures as input control

instance temperament, intent (specific or accidental), engagement, command, and so on. Having this data gives the opportunity for assessment of factors that are otherwise unobtainable without obtrusive hardware.

2.6.5 Weight Tracking of Multiple Items

Using the functionality of multiple weight tracking, the system can be used within a kitchen or lab table setting to aid in material measurement, inventory management, etc. By coupling this with real-time feedback functionality, users can be guided through the making process. In the case of cooking, they can precisely follow recipes. Also, having the system in restaurant tables, customer behavior can be tracked by measuring the weight of each dish so that restaurant owners can make use of the data in an inherently anonymous way, as opposed to systems utilizing video.

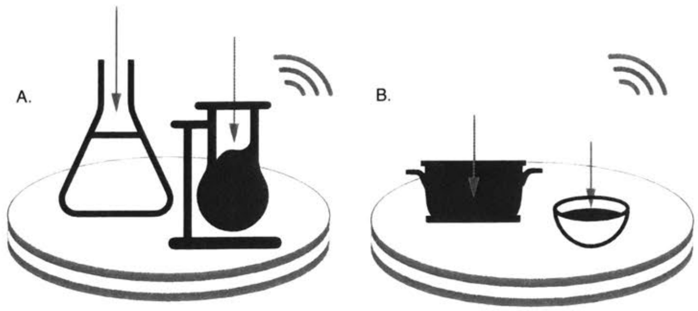

Figure 2-16: A.Lab items on platform. B.Kitchen items on platform.

2.7 Limitation and Future Work

The main limitation of the system stems from its inability to track multiple simultaneous forces. Being a non-multitouch system puts significant limitations when compared to other multitouch systems. Additionally, detection of object orientation, such as rotation or tilt can enhance the interaction even within the single-force domain. Luckily, hybrid systems can resolve issues such as these while maintaining all of the benefits of the system. For example placement of an iPad device on-top of an augmented surface produces a multi-touch, force sensitive surface, albeit only on the system aware applications (through websockets, for example) on the iPad. Tighter integration can be envisioned whereby the operating system itself could communicate directly with the device and provide the data to all applications natively.

By combining with other sensing modalities, interaction could be further improved. For example, adding acoustic sensing [51] could add a dimension for identifying different touches, objects, materials, etc.

Further improvement could be made to the heuristics and filtering in software, to tracking speed and accuracy, and to the detection of object dragging. Currently all of these are implemented such that they perform well separately, and only the needed modules are used in each interaction type. In principle, a more intelligent/cognitive system could automatically adapt to the user's current type of interaction.

Tracking speed could be improved by sampling the load cells even faster, current industrial systems support data acquisition rates in the kHz range, such improvements could allow detection of additional features such as minute vibrations, and perhaps even sounds.

Regarding the easily customizable, deployable and scalable hardware design, we are curious to conduct a user study for wide range of target audiences such as students, designers, artists and scientists. We have already begun to distribute units, and are currently lending several prototypes to students and researchers across the U.S. and Europe. We hope to receive feedback that would help improve the software architecture, and enhance the libraries currently available for the system as well as explore integrations into other existing complementary platforms and uses.

2.8 Conclusion

Through the development of the load-sensitive system, we have shown that it is possible to easily add ubiquitous interactivity to surfaces using minimal hardware that is low-cost, durable and accurate. As weight is the main sensing factor, we allow for interactivity to take place upon any material, thus diversifying the design space in which the platform can be used. Due to its cost-effectiveness, our system is easily deployable, allowing for numerous applications in multiple fields, as well as simple integration into existing products. This chapter has presented the steps necessary to produce such a system for other practitioners in the field. The next step in this research will involve expanding the software applications as well as minimizing the load-cell sensors to enable better integration with existing environmental elements such as furniture and countertops.

Chapter 3

SCALE: Load Sensitive Modules for Force-based Interaction

3.1 Introduction

As stated in the introduction of this thesis, Force conveys fundamental information in Human-Object Interaction, including force intensity, its direction, and object weight [28] - information otherwise difficult to be accessed or inferred from other sensing modalities. When force is captured during interaction, a wide range of activities can be reconstructed such as way of touch, movement of objects and patterns of body motion.

Force-based interaction is involved at different scales in terms of the intensity of loaded force and the size of the interaction area. For instance, force-based interaction can range from actions such as drawing minute letters on a piece of paper (~1g, 1mm), to handling tools on a workbench (~1kg, 10cm), to dancing in a room (~100kg, 10m). Even though researchers have already tackled each respective task, [47, 60], it is ideal if interaction designers are able to explore the wide range of force-based interactions within a single integrated framework.

In this work, the most essential technical underpinning of my thesis , we propose a framework of processing load data from load sensitive modules to cover the three main categories of force-based interaction, including *Touch Interaction*, *Object Status*

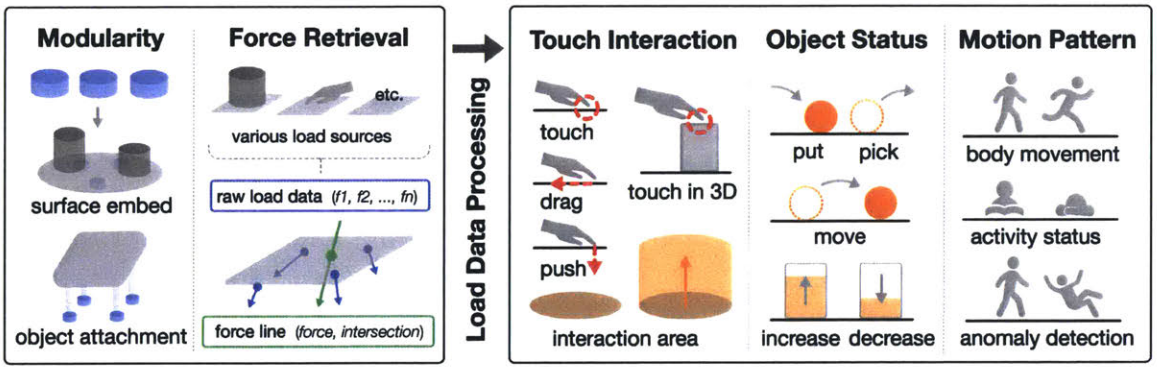

Figure 3-1: The concept of SCALE: (left) Modularity and Force Retrieval (right) Load Data Processing enables three types of functions: Touch Interaction, Object Status Tracking and Motion Pattern Recognition.

Tracking, and *Motion Pattern Recognition.* The modularity of our system expands two key aspects of load sensitive applications: ***scalability*** in weight tolerance by adding a number of modules to fit the target load capacity on demand, and ***variability*** in spatial configuration by reconfiguring the spatial placement depending on their respective objectives.

Specifically for *Touch Interaction*, we have expanded the interaction area from a flat 2D surface to 3D volume by developing a new algorithm freed from the geometric shape information of an object, which is required in the previous method [24]. With a broadened set of applicable objects, this function allows us to utilize the information of a touch point in 3D space for further analysis, telling us which part of an object is currently being touched, or what kind of shape outline the object has.

In addition to the algorithm improvements, we have implemented this framework into a physical prototyping kit with compact hardware and a GUI designed for novices. We additionally conducted a workshop with corporate designers and engineers to explore the application space enabled by the system, and evaluated its utility.

Our contributions described in this chapter include;

- An architecture and design space for load sensitive modules to allow a range of force-based interactions, including touch interaction, object status tracking and motion pattern recognition.

- A new algorithm expanding interaction range from 2D to 3D above a load sensitive surface based on an inverse-matrix framework without prior shape knowledge.
- Technical implementation of hardware and GUI, and summarized findings from our workshop with corporate practitioners to explore the application space and to evaluate its utility.

3.2 Related Work and Approaches

The sensing technology for detecting the physical interactions between humans and objects is one of the primary research agendas in HCI. A number of contact sensing techniques using non force-based methods have been introduced, including vision-based [7, 31, 32], IR based [22], capacitive sensing [13, 35, 56, 61], swept frequency capacitive sensing [23, 58], EM based [72], microphone [29] and acoustic based method[51].

Among the techniques stated above, force-based sensing methods have the notable advantage of direct capturing of the contact force [68]. In the context of HCI, several force-based methods have been investigated, including Piezoelectric [17], and force-sensitive registers [12, 52]. In terms of *scalability* in weight, the methods with load cells show a wide range of applicability due to its high tolerance in maximum force [9, 46].

For the load-based methods, we have categorized the functionality into three parts, including *Touch*, *Object* and *Activity*. On the load-based approach, many systems have been proposed for touch detection purposes [47, 59, 67]. This approach naturally expands to variations of touch, including tap, press, drag and draw, however, the interaction area of these systems is constrained onto a 2D surface. Notably, INTACT pushes the interaction area to a 2D surface in 3D volume by assuming prior shape knowledge of the object on the geometrically-constrained surface [24]. Preliminary formulation of our approach is proposed previously [69], and we improved the algorithm in terms of mathematical stability with regularization terms, together with added design framework and workshop study.

Detection, or identifying objects, is another critical domain in the load-sensitive method. As the foundation of this category is regarding objects, the concept of *Weight as ID* conveys an essence that precise measurement of weight can be useful for identifying objects due to its occurrence in daily life [10]. Localization of the target has been a hot topic from fields such as Biology [57, 74] and Robotics [4, 38].

In addition to *Touch* and *Object*, load-based activity recognition has been investigated for many years. Context-aware systems have developed in combination with the algorithms of classifying signals [47, 59, 60]. Especially, the pose estimation for the human body has been a growing field [18, 66].

Among such broad applications on the load-based approach, our system as a prototyping tool kit unifies all the three application domains, including *Touch*, *Object* and *Activity*, into a single framework of load data processing. With the technical breakthrough being for detecting 3D touch, we expand the application field to everyday objects, freed from the requirement of having the geometric shape model in advance.

3.3 SCALE: A tool kit for force-based interaction

3.3.1 Design Space

SCALE is a prototyping tool kit to encourage interaction designers and engineers to explore *force-based interaction*, which is uniquely enabled by capturing direct force information, with the architecture composed of load sensitive modules and a framework of load data processing. The key feature of SCALE is its modularity, aiming at *scalability* and *variability*, so that the users can increase the number of modules to be capable of accepting heavier load on demand, and place modules to reconfigure the spatial arrangement to fit their objectives, as shown in Fig.3-1 (left).

Furthermore, the modularity enables the system to cover a wide range of *force-based interaction* with the support with three functions in the load data processing, including *Touch Interaction*, *Object Status Tracking* and *Motion Pattern Recognition*, as shown in Fig.3-1 (right). Here we describe the design requirements for each process as following:

Touch Interaction

The system should be capable of capturing the interaction between a human and objects, and particularly *touch* is the common interaction seen in a wide range of situations. If the system captures both of the force intensity of a touch and the position of the touch, this information can be utilized for further analysis. For example the system could infer which part of an object is currently being touched. Furthermore, if the system has less constraints on an object, such as restrictions on a shape, the system could be applicable to many purposes. Therefore *Touch Interaction* of SCALE is designed to capture various types of touch interactions happening on 2D surfaces or in 3D volumes, freed from the shape constraint.

Object Status Tracking

The system should be capable of handling a large set of light and heavy objects in a single manner. A pen with 10 grams and an adult with 60 kg would represent the scalability seen around our life. Therefore *Object Status Tracking* has a function to track the object position and weight. By calculating total weight and center of mass, the five different status of an object can be classified: *pick*, *put*, *move*, *increase* and *decrease*.

Motion Pattern Recognition

The system should be capable of capturing what people are doing on a table, or how people are moving their body on a floor. When people walk or stretch, it causes different signal patterns on load sensors. So we designed *Motion Pattern Recognition* as a framework for recognizing different activities based on the signal pattern. Our simplest scheme is composed of feature extraction, and the support vector machine can distinguish between different user-defined activities.

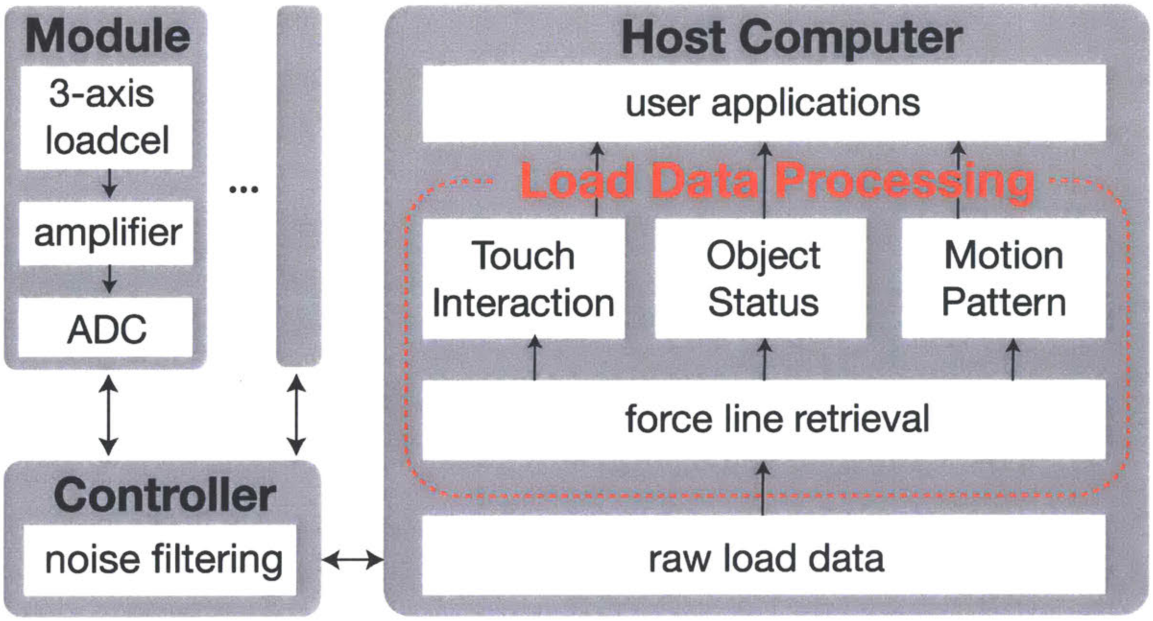

Figure 3-2: Architecture of SCALE: (left) Load sensitive modules and its controller (right) Applications on top of the *Load Data Processing* architecture on a host computer

On top of these processes, the user can develop their own applications in accordance to their purposes. Since this application space, uniquely enabled by *force-based interaction*, is thought to be broad, it is useful if the scope of the application space is being disclosed as a list of potential scenarios. Therefore we figured out the scope by having a workshop with corporate designers and engineers, as we describe the detail in the latter part of this paper.

3.3.2 System Architecture

The system architecture of SCALE is illustrated in the block diagram shown in Fig.3-2. There are three hardware components: *modules*, *controller* and *host computer*. Each module contains three-axis load cells and its peripheral circuits to transmit load data to controllers. All module data in the system is sent to one unified controller and pre-processed with a simple noise filtering. Since the raw data from load cells are sometimes polluted with sporadic saturated signals, we eliminate the outliers by applying a simple threshold on the absolute value of the raw data.

The host computer receives raw load data from a controller through a USB serial bus. If we have N modules, s.t. $N \geq 3$, we receive an array of $3N$ load data. This load data is sent to the signal processing core called *Load Data Processing* and the system retrieves the force and its intersection as shown in Fig.3-1. This force information is exploited by following three different pipelines: *Touch Detection*, *Object Status Tracking* and *Motion Pattern Recognition.* After these three go through load data processing, the results are utilized to make user-defined applications.

3.4 Load Data Processing

3.4.1 Force Line Retrieval

A *force line* is the key element of the architecture for the load data processing, which is mathematically represented as a set of *force* f and *intersection* a as shown in Fig.3-1. Here we describe how to retrieve a force line from raw load data. We assume the sets of measured force f_i, sensed at i-th load module ($i = 1, 2, ..., N$). For simplicity, we could assume that all the sensors are placed at p_i on the same $z = 0$ plane. The touch force f and its torque τ is derived as $f = \sum_i f_i$ and $\tau = \sum_i p_i \times f_i$ by definition.

Here, the line of action for manual touch is expressed as $x = a + pd$, parameterized by scalar p. The normalized direction vector d is $d = f/|f|$ and the anchor point is $a_0 = f \times \tau / |f|^2$. Since we can take an arbitrary point along the line as the anchor, we obtained the *intersection* a as the anchor point intersecting with the modular plane, where $a = a_0 - \frac{d_z}{a_{0z}} d$. On this formulation, the scalar p becomes regularized by being zero at all times when the point is on the $z = 0$ plane.

3.4.2 Touch Interaction

We provide the algorithm to detect the touch point on a 2D surface or 3D object on load sensitive modules in Fig.3-3. Here especially, we describe a unique algorithm of *3D Touch Detection*, which exploits the unsteadiness of a hand during touch interaction. We assume enough rigidity in the object, but it does not have to be composed

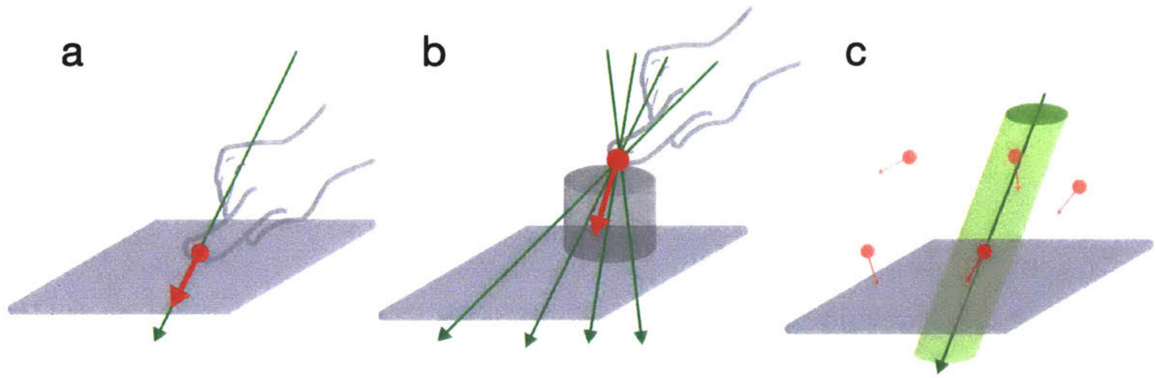

Figure 3-3: Process for *Touch Detection*: (a) 2D Touch Detection (b) 3D Touch Detection (c) Touch Classification

of a single uniform material. Our framework accepts multi-material objects (e.g. a wooden desk with metal legs), as long as they convey force from a touch point to the sensors without internal dispersion.

2D Touch Detection

As illustrated in the previous section for *Force Line Retrieval*, we used the intersecting point a between the force line and $z = 0$ plane as the touch point, as shown in Fig.3-3(a). By constraining the existence area onto $z = 0$ plane geometrically, we can solve the mathematical ambiguity along the force line.

Another type of geometric constraint is investigated in a prior project called INTACT [24]. Instead of the $z = 0$ plane stated above, as the geometric constraint on the force line, they introduced the 2D surface envelope of an object. This approach was clever enough to expand the interaction area from a 2D surface to a 2D envelope in 3D volume, however, it is still less scalable since this approach requires prior shape information and its orientation of the object on the surface in advance. That means it is difficult to expand the application range of the method to an object with an unknown shape.

3D Touch Overview

To address the problem stated above, we propose an algorithm to localize the touch point in 3D space without any geometric constraints, with focus on the unsteadiness of a human hand. Even though our approach is still constrained on the 2D surface envelope of an object as well, this approach outstands since it does not require any prior shape information and can be applicable to any rigid object.

The key insight of our solution lies in the fact that when we touch an object with our hand, the touch is never stable. As illustrated in Fig.3-3 (b), when we aggregate the several recent lines they should have slight differences in direction. By looking at these lines, we can find the touch point as the most possible intersecting point of all the lines. Instead of assuming a geometric constraint, our approach equivalently introduces the temporal continuity of human touch. This assumption is thought to be valid when human touch is much slower than the frequency of load sensing, such as 80 Hz sensing with the sped-up ADC, which we introduced in the implementation section.

3D Touch Algorithm

Here we describe the detail of the algorithm to localize the 3D touch. Firstly for simplicity we transformed the equation for a force line $x = a + pd$ into the form of a matrix equation, where I_3 is a 3x3 unit matrix.

$$\begin{bmatrix} I_3 & -d \end{bmatrix} \begin{bmatrix} x \\ p \end{bmatrix} = \begin{bmatrix} a \end{bmatrix} \tag{3.1}$$

This equation is apparently under-determined, so we must make the equation over-determined in order to calculate the touch point $x = [x \quad y \quad z]$ by the pseudo-matrix method. The touch point x can be assumed to be constant during a touch, and the system obtains different force lines $x = a_t + p_t d_t$, where the discrete time stamp is denoted as t. When we collect the most recent T data during the touch, the matrix equation mentioned above naturally expands in the manner below:

$$\begin{bmatrix} I_3 & -d_1 & 0 & \cdots & 0 \\ I_3 & 0 & -d_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ I_3 & 0 & 0 & \cdots & -d_T \end{bmatrix} \begin{bmatrix} x \\ p_1 \\ \vdots \\ p_T \end{bmatrix} = \begin{bmatrix} a_1 \\ \vdots \\ \vdots \\ a_T \end{bmatrix} \tag{3.2}$$

Here we abbreviate the equation as $DX = A$ for simplicity, where $D \in R^{3T \times T+3}$, $X \in R^{T+3}$ and $A \in R^{3T}$. Even though this equation has the worse condition number in terms of the inverse problem framework since the force lines are thought to be quasi-parallel, we can solve the equation by the support of appropriate regularization terms. Finally, we reach the least-squares solution X by using the Moore-Penrose pseudo-inverse matrix method. On this framework, the solution x tends to be constrained around the origin of the space, and slightly gets closer to the surface under the influence of the regularization on p_i as well.

Note that here we introduced *generalized Tikonov regularization*, rather than the standard Tikonov method with a uniform regularization parameter λ, to obtain a stable solution X by reducing the effect of sensing errors, which has introduced in a multi-modal sensing method [45]. This is because the regularization parameters, λ_x for x and λ_p for p_i, have different physical dimensions, such as x as a spatial position in mm and p_i as a dimensionless scalar. We experimentally adopted 20 for T, 0.1 for λ_x and 0.01 for λ_p. Here we finally reach the touch point $x = [X1 \quad X2 \quad X3]$ as picking the first three components in X:

$$X = (D^T D + \mathrm{diag}(\lambda_x^2, \lambda_x^2, \lambda_x^2, \lambda_p^2, \cdots, \lambda_p^2))^{-1} D^T A \tag{3.3}$$

Touch Classification

The touch classification algorithm is shown to classify an immediate touch to a corresponding registered touch point. It takes T samples, typically 0.25 sec or more, to register a touch point as shown above. However, with the classification algorithm the system can detect the touch to registered points immediately with only a single

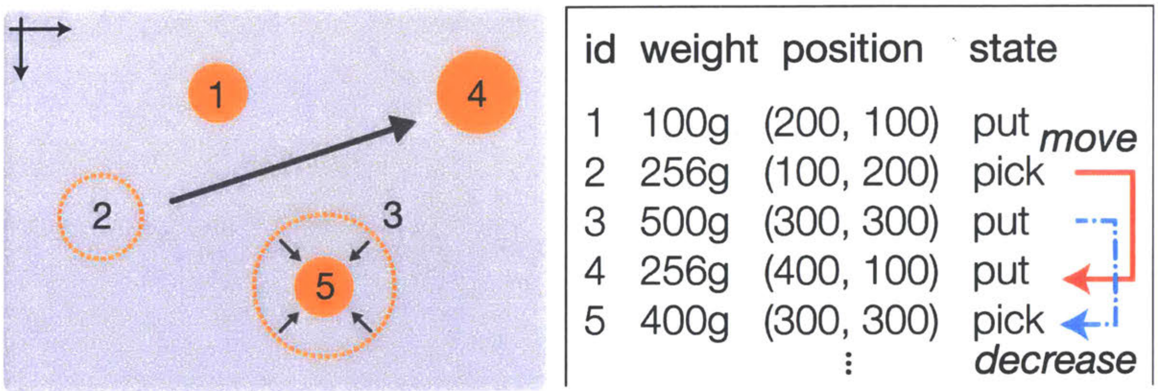

Figure 3-4: Process for *Object Status Tracking*: (left) Status Change for Objects (right) Database Manipulation

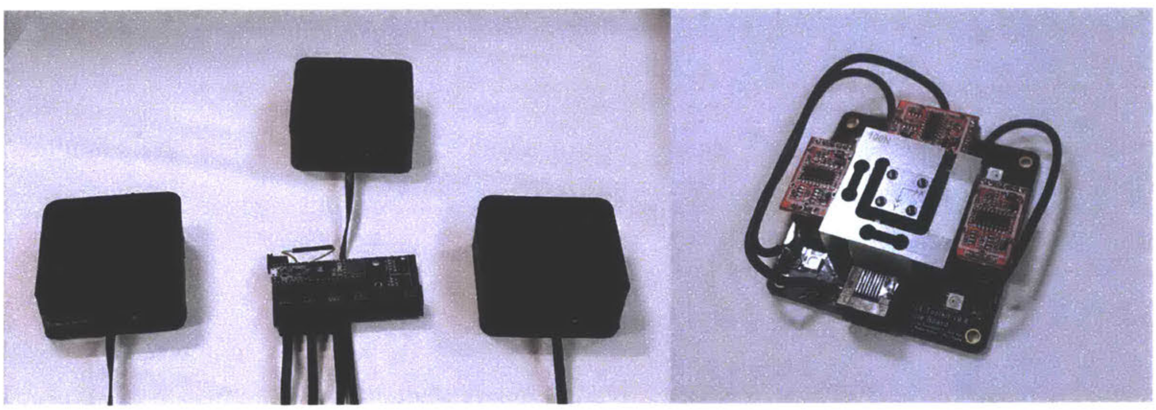

Figure 3-5: Load sensitive modules: (left) Outside (right) Inside

sample of force line.

We will classify the green force line as the most possible registered point shown in Fig.3-3 (c). There are two steps of selection: *Cylindrical Search* and *Direction Similarity*. For the first step of Cylindrical Search, we will ignore all of the distant registered points from the force line with the threshold radius r. For the second step of Direction Similarity, we will calculate the inner product of normalized directions between input force line and that of the registered point. The appropriate parameter r heavily depends on the application, yet we generally adopt 30 mm for the threshold radius.

3.4.3 Object Status Tracking

To detect an object with weight and position and identify its status from load signals, there are two steps of load processing. The first step is called *Stability Check*, where the system determines the weight and position of a new object or the removal of an existing object. The second step called *Database Manipulation* is where the system accesses the internal database to identify the type of action. The core concept of the second part is *Weight as ID* insight, which claims weight information is useful to distinguish two or more different objects on a scale with required precision [10].

Stability Check

In the first part of Object Status Tracking, we focus on weight data $w_i = f_{iz}$, which is the equivalent z-component of load from each module. Here we have $w_{total} = \Sigma w_i$. To check the emergence or disappearance of objects, the system needs to distinguish *Stable* status, where every raw load data is almost static, from *Unstable* status. This stability check is conducted through simple thresholding by subtracting the slow LPF-ed (low pass filter) from the fast LPF-ed data.

$$stability = slow\text{-}LPF(w_{total}) - fast\text{-}LPF(w_{total})$$

If *stability* is small enough, it means the objects on the surface are *Stable*. This stability has a trade-off with response of the system. We experimentally adopted 2 grams as the threshold value for *stability*. Also, LPF is implemented as the exponentially weighed moving average, with the filter strength α at 0.04 for slow-LPF, and 0.25 for fast-LPF. Once the status is classified to *Stable*, the center of total weight x_{total} can be calculated as $x_{total} = \Sigma w_i x_i / \Sigma w_i$, where x_i is the position of i-th module.

Database Manipulation

In the second part of Object Status Tracking, the system handles the internal database and reflects the result to SCALE GUI, as shown in Fig.3-4. If the detected total weight w_{total} is above zero, the object is to be labelled as *put*. If the weight is not above zero, the object is labelled as *pick*. In either case, the objects are then added to the database. In Fig.3-4, the newly detected object #4 has the same weight as that of object #2, which is picked. Here these two objects are identified as the same, and merged into object #4. This operation is called *move*. If the new object #5 appears on the same position as the existing object #3, the system subtracts the weight from that of the existing object #5. This is *decrease* of the weight. The same procedure will apply for *increase*.

In our practical implementation, since the system faced errors in weight and position we need to set a tolerance to identify the values. We experimentally applied thresholds to identify two slightly different values to one value. As a result, we adopted 5 grams for the weight threshold and 3 cm for the position threshold.

3.4.4 Motion Pattern Recognition

Here we describe the pipeline to distinguish two or more activities from each other based on load signals. It is out of our scope to construct a pipeline to build *general* Motion Pattern Recognition framework, so we drew from activities that follow the same raw signal with periodic patterns.

In our pipeline, the incoming raw signals are converted into a feature vector, which expresses a specific type of motion by feature extraction. The user can choose any feature extraction method, including fast fourier transform, average, standard deviation and etc. The feature is fed to be classified by a support vector machine (SVM) algorithm.

Specifically for our applications, we record the force and torque vectors for 1 sec with 30 Hz sampling rate, and then we derived the standard deviation in each component as a 6 dimensional feature vector. Also we adopted the fine Gaussian kernel for the detailed algorithm for classification.

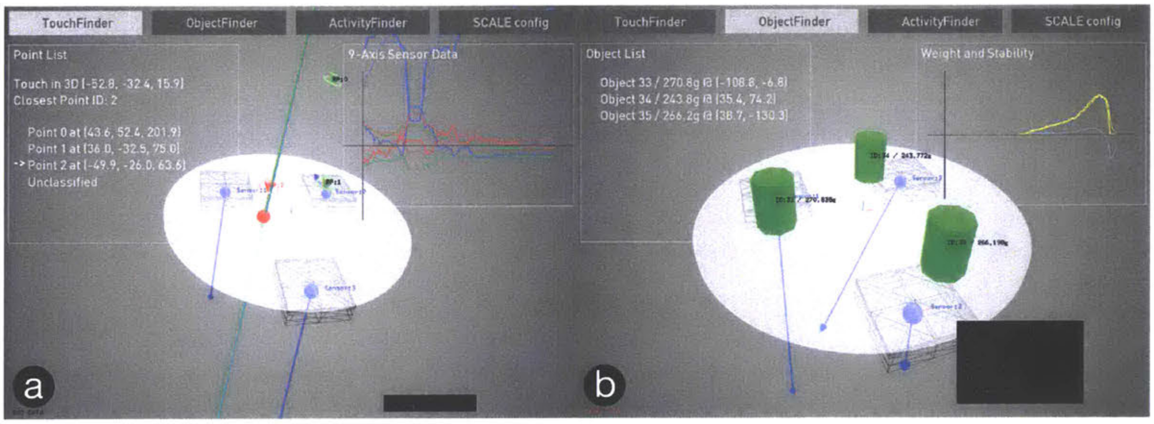

Figure 3-6: Interactive GUI for SCALE: (Left) It shows a current touch point and registered points for *Touch Detection* (Right) It shows the objects with its position and weight for *Object Status Tracking*

Figure 3-7: Application ideas from workshop presentations: (a) tangible music composer (b) interactive narrative with voiced characters (c) fish pointer for aquarium exhibition (d) activity sensing on the floor

Application Space	Ideas	# of Ideas per Group				FBI types		
		A	B	C	D	Object	Touch	Motion
Healthcare	Daily Health Check	5		3	4		●	●
	Help Medical Examination	3	1				●	●
Surveillance	Personal Identification	2	1				●	●
	Tracking Living things / Objects	1	2	3	5	●	●	●
Cooking	Help Cooking	2	3			●	●	●
	Record Cooking	1	1	1	1	●	●	
Enter-tainment	Game UI		1	1		●	●	
	Help Creative Works		3			●	●	●
	Compose & Play Music	5	1	2	1	●	●	
Home	Control Air / Sound / Light / TV	2	6	5		●	●	●
	Help Non-Verval Communication		1				●	●
Learning	Storytelling with Figures				1	●	●	
	How to Use Device				1	●	●	
	Observe & Point to Living Things		1				●	●
	Dance / Instruments / Yoga	1		1			●	●

Figure 3-8: Classification and analysis of application scenarios from corporate designers and engineers

3.5 SCALE Prototype

3.5.1 Modular Hardware

The overall SCALE architecture is illustrated on Fig.3-2. We designed two types of hardware to maximize usability of the entire system: modules and a controller.

Each module contains a three-axis load cell (FNZ100N, Forsentek.inc) with a load capacity of 10 kg, and three amplifiers (HX711) with analog-digital converters in the fastest mode at 80 Hz. The module is cuboid with an 90 x 90 x 35 mm form factor. To make the entire module compact enough to fit on the palm of your hand, we designed an original PCB and put all of these elements inside of a 3d-printed cabinet, as shown in Fig. 3-5. To maximize the grip between the module and floor or object, we put layered rubber onto both sides of the module surface.

The load sensitive modules are to be connected to a single controller with ethernet cables, which has a detachable and regularized connector so that a user can

Figure 3-9: Demonstration for *Touch Detection*: (a) *Virtual Interface on Physical Objects* (b) *Embedded Usage Tracker* (c) *General Shape Capturing* with a tripod as a target (d) a close-up picture of captured shape of the tripods

easily reconfigure the number and the placement of modules. A single controller is capable of being connected with 8 modules at maximum, which leads to *scalability* in weight tolerance and *variability* in spatial configurations. A controller contains a micro processor (Teensy 3.6) to aggregate and pre-process all of the raw data from the modules, and transmit them to the host computer.

3.5.2 Software GUI

All of the software composed of real time signal processing and Graphical User Interfaces (GUI) is implemented on the open-source library (openFramework) by C++, as shown in Fig.3-6, except the *Motion Pattern Recognition* feature, which is implemented on Matlab environment.

The GUI provides three different primitive modes, including *Touch Detection*, *Object Status Tracking*, and *Motion Pattern Recognition* (only for capturing signals), and the user can develop an integrated system on top of these three basic functions. For all primitive modes, the user is capable of interactively registering a current touch point or object to the database and selectively serialize them for further analysis for other applications.

3.6 Workshop for Exploring Application

We conducted a SCALE hands-on workshop to evaluate the utility and to explore potential applications which we had never expected. The workshop procedure was

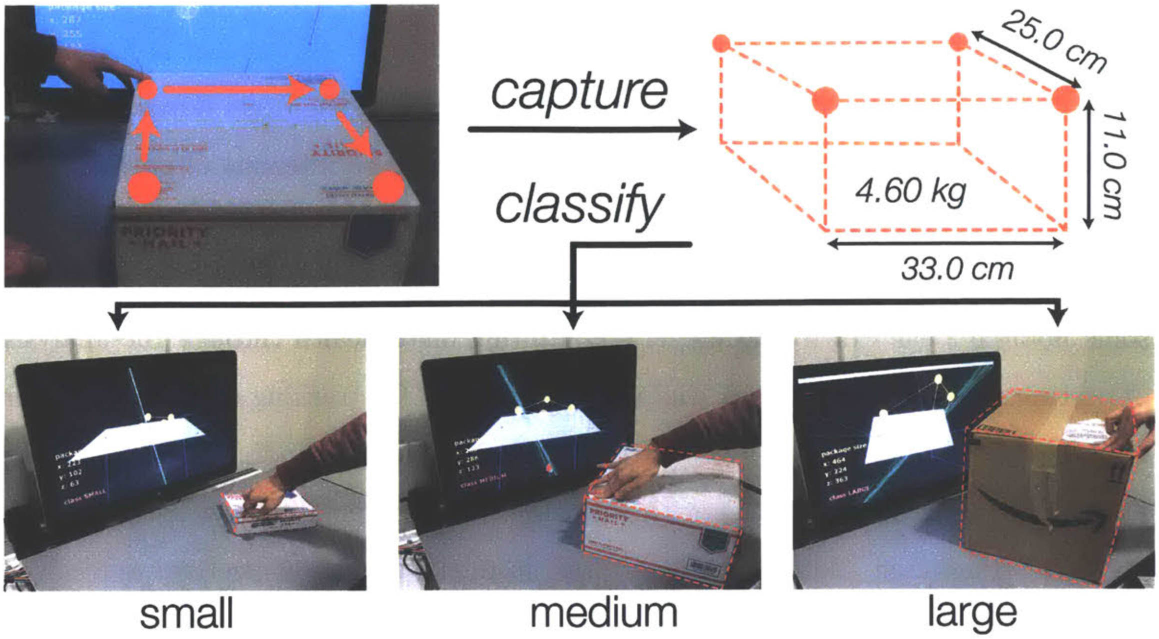

Figure 3-10: Demonstration for *Shape Capturing*: The system classifies the type of a package based on weight and its size.

designed in a way participants can accomplish prototyping their ideas and present their narratives with the developed demonstrations.

3.6.1 Designing the Workshop

With the support of a product corporation, 12 designers and 8 engineers attended the workshop and were divided into 4 teams to evenly distribute expertise in each group. There were three sessions in the workshop. The first 2-hour slot was designed to brainstorm new application scenarios. The participants were asked to come up with as many small use cases possible, to then merge them into a larger concept. The second session was 6 hours of hands-on participation to develop functional applications with the SCALE development kit. After providing detailed instructions to use the kit, each team that is composed of 5-6 people started to collaborate with colleagues to prototype their own ideas. We concluded with a one hour session to present the developed ideas and prototypes and to receive feedback from peers.

3.6.2 Exploring Application Space

We have compiled the ideas that corporate designers and engineers developed from the brainstorming session into Fig. 3-8. To catch the core interests of participants, we classified the ideas into six categories: Health-care, Surveillance, Cooking, Entertainment, Home and Learning. Among a range of promising scenarios, we picked some notable ideas worth sharing: (1) monitoring one's health through the analysis of posture changes while sitting, walking and sleeping; (2) tracking activity of pets or growth of babies and plants; (3) controlling home devices, including speakers, lights and air, through direct contact with furniture, walls or floor, rather than through digital interfaces. In addition to these ideas, from the user's perspective we received comments that mention a guideline on how to develop or implement each idea on top of our software pipeline, and the best use cases for *Touch Interaction* where the system becomes *the best from a practical point of view* among all sensing technology.

From the hands-on session, we had four different functional prototypes, as shown in Fig. 3-7. We briefly describe them in the following list:

- *Group A*: The **tangible music composer** is implemented on *Object Status Tracking*, and allows the user to play and mix up music based on the placement of different types of objects on specified disk locations, as shown in Fig. 3-7(a).
- *Group B*: The **interactive story-telling** with voiced characters is designed for children to breathe life into their favorite toys through a pre-recorded voice-over triggered by touch interactions, which is implemented on *Touch Detection* and *Object Status Tracking*, as shown in Fig. 3-7(b).
- *Group C*: A **fish pointing system for future aquarium** utilizes the *3D Touch Detection* technique to select a specific fish swimming in the middle of a large tank with the assumption that the 3D position of all fish are tracked by computer vision, as shown in Fig. 3-7(c). This application provides detailed knowledge of the selected fish, such as the name, species, habitat and food, by touching on the load sensitive glass window.
- *Group D*: The last application is the **posture-aware floor for Yoga practi-**

tioners, designed to identify individuals and analyze their posture and to allow the system to advise the individual on how to modify a post for safe practice, as shown in Fig. 3-7(d).

3.6.3 Evaluating Utility

To analyze the utility of the toolkit from a viewpoint of practicality, we conducted the subjective evaluation by distributing a questionnaire after the workshop. The questions were along the lines of, "How did you feel about SCALE as a ubiquitous sensitive system?" by using Likert's five points scale from "Very Good" to "Very Bad" and, "What are the pros and cons of the toolkit?" through open response. We received answers from 10 participants. The resulting scores from the first question are 4.6 / 5.0(*Average*), 5.0(*Median*) and 0.66(*SD*).

Regarding the comments from the second question, the positive comments are as follows: "It's really useful to be able to sense a variety of different things about the physical state of objects or people using a surface and invisible sensor" *(Female, Industrial Designer)*, "The interface is intuitive" *(Male, Chemical Engineer)* and "Detecting not only the single touchpoint but a series of touchpoints that translate into an activity" *(Female, Experience Designer)*. Among the negative comments were: "The threshold of SCALE should be adjusted so that people can act by elbow, body and so on" *(Male, Cognitive Psychologist)*, "The necessity of detection range and UI for ease to control" *(Male, Software Engineer)* and "Accuracy across large surfaces, sensitivity across multiple touch points at different densities" *(Female, Experience Designer)*.

The results of the questionnaire and brainstorming session as shown in Fig.3-8, which allow us to consider the following points. Firstly, we can see that an advantage of SCALE is the capability to recognize a wide variety of *Touch Interaction* with invisible forces. Secondly, SCALE is expected to use its *Motion Pattern Recognition* for grasping multiple interaction touch points. Finally, improvements on the versatility and application of SCALE are needed.

On the other hand, we found a issue regarding the constrain of the number of

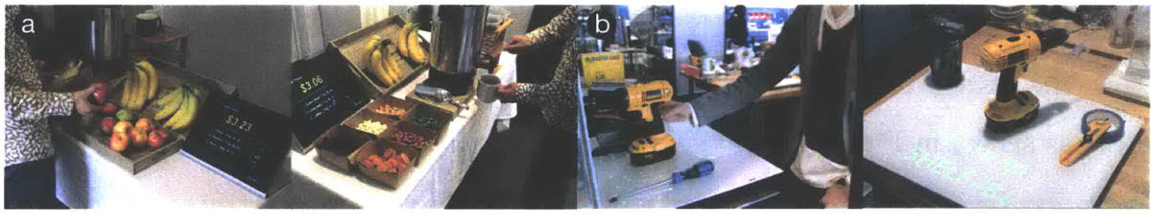

Figure 3-11: Demonstration for *Object Status Tracking*: (a) *Retail Automation* enables to capture object movement and liquid consumption (b) *Smart Workspace* is monitoring the location and usage of the tools

sensor module. While the reconfigurability of sensor modules made it easy for participants to quickly customize the layout of the modules, our prototype was constrained to use three module. This limitation made it unstable for some of the large scale interaction prototypes (e.g. body gesture detection). We plan to improve our User Interface software and force vector calculation algorithms to accommodate multiple (more than three) sensor modules placements.

3.7 SCALE Applications

Reflecting on the concluding remarks from the workshop, we identified 4 application areas where we felt that SCALE could have a potential impact - either as a useful enhancement to an established application or a novel application, uniquely enabled by our approach:

- making everyday objects and surfaces force sensitive
- capturing the general shape of an object by touching it
- locating objects, including liquids, through weight identification
- making home fixtures an activity tracking platform (eg. floors)

In the rest of this section we propose a few exemplary applications for each category, shedding light on the utility and scope of our sensing approach.

Volume Slider on PC Monitor

If everyday objects can be sensitive to touch, including touch position, direction, and intensity, they can configure functions in productive ways. The canonical example would be a PC monitor with a user-defined touch point, as shown in Fig. 3-9(a). When a user touches the top-right corner, the audio volume changes from low to high according to pressing force. A user can also assign a power button just next to the mute button, since the system can differentiate two overlapped registered points with the classification algorithm.

Shelf Usage Tracker

In addition to enhancing a PC monitor, *making everyday objects force sensitive* can be useful for objects with no feedback system inside. A user can easily augment a tool shelf containing different types of screws into a trackable activity tool by putting only three modules beneath the shelf or table surface, as shown in Fig. 3-9(b). When a user opens the third drawer and grabs some screws, the quantity and the type of screws are distinguished immediately.

Shape Capturing By Touch

Our 3D touch algorithm allows a user to capture the general shape of an object, like a notebook PC, by touching its outer points. After a user repeatedly touches multiple points around the object, the detected points are connected, and a contour of the object is captured, as shown in Fig.3-9(c).

Since our system is capable of capturing the general shape of an object from only load data, the system classifies an object into the user-defined categories based on its weight and estimated size, as shown in Fig.3-10. This could be useful for the application requiring simultaneous acquisition of weight and rough shape, including the measurement of packages at postal offices, or the airport counter to check-in the bags for flight, to estimate its cost and rough volume.

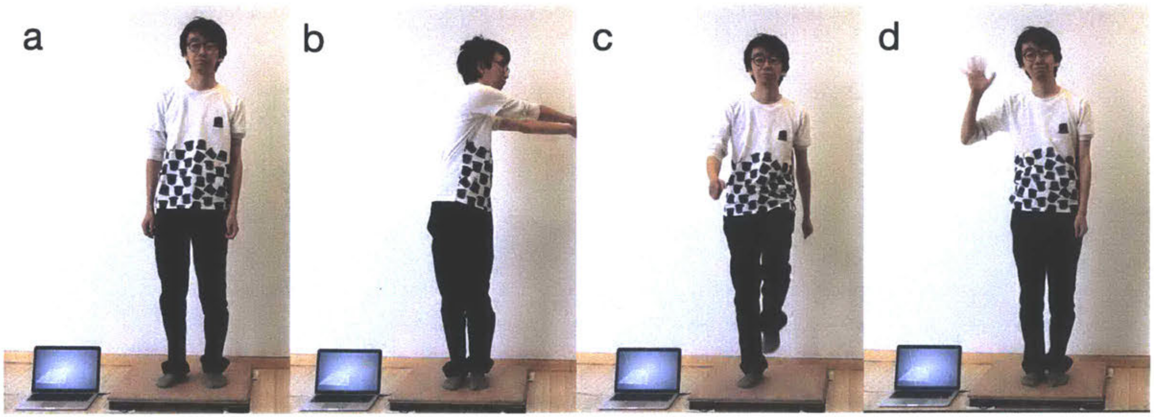

Figure 3-12: Demonstration for *Motion Pattern Recognition*: The system classifies four different activities: (a)*stand* (b)*stretch* (c)*walk* (d)*wave*

Retail Automation

On top of the *object status detection* mode, combined with an external database of product information, it is possible to prototype an automated checkout system on a load sensitive table as shown in Fig. 3-11(a). Recently, this type of application has been well-investigated using machine-vision systems, yet our load sensitive approach is adding an essential value of weight-based interaction, including selling-by-weight. In addition to discrete objects, liquids or granular products are under coverage of the SCALE system. A customer can take as much coffee as they want, and be charged according to the exact amount of consumption, since the change in weight is captured with its position.

Smart Workspace

The workbenches or tables enhanced by load sensitive modules are becoming smart enough to track the usage and positions of tools, like a handy drill, as shown in Fig. 3-11(b). The system remembers the previous position of a handy drill, so that the user can indicate the current location of the tool through other display techniques. Additionally, if the user forgets the place where the drill should be returned, the system will notify you of the location by searching in its database.

Posture Estimation on a load sensitive floor

Once load sensitive modules are embedded beneath the room floors, the surface immediately becomes capable of *motion pattern recognition.* From the different wave shape of load signals, the system classifies the type of movement (running) and displays a caution to stop running inside the room, as shown in Fig. 3-12. Further analysis including affection inference or user recognition could be implemented on top of the load processing framework we have proposed in this paper.

3.8 Technical Evaluation

Here we provide the performance of our prototype we experimentally evaluated to support the viability of the system. We setup the measurement on accuracy and precision concerning spatial position, and conducted two different experiments for the *horizontal* plane and the *vertical* axis, as shown in Fig. 3-13.

The *horizontal* accuracy is measured on xy plane, especially related to 2D touch detection or object localization. As shown in Fig. 3-13(a), we achieved less than 1 cm accuracy in the prototype, tested with the three different weights (300, 600, 900 gram) to check the weight consistency of the algorithm.

The *vertical* accuracy is measured along the z axis to evaluate 3D touch point detection, shown in Fig. 3-13(b). We put a fixed size shelf on the SCALE platform, and keep touching a point on each surfaces for 1 sec. We repeatedly obtain the estimated height for 10 times. In the figure, we illustrated the tested height as a small red dot and the standard error as a bigger red circle. At most we have 7cm accuracy at the height of 50 cm. While the error seemingly expands according to the height, it would be useful to distinguish two different surfaces in a shelf.

Also, as shown in Fig.3-13(c), we classified four motion patterns according to the proposed pipeline, and evaluated the accuracy of the prediction by making a confusion matrix. For the specific four different body motion, the prototype successfully classified them with more than 90 % accuracy.

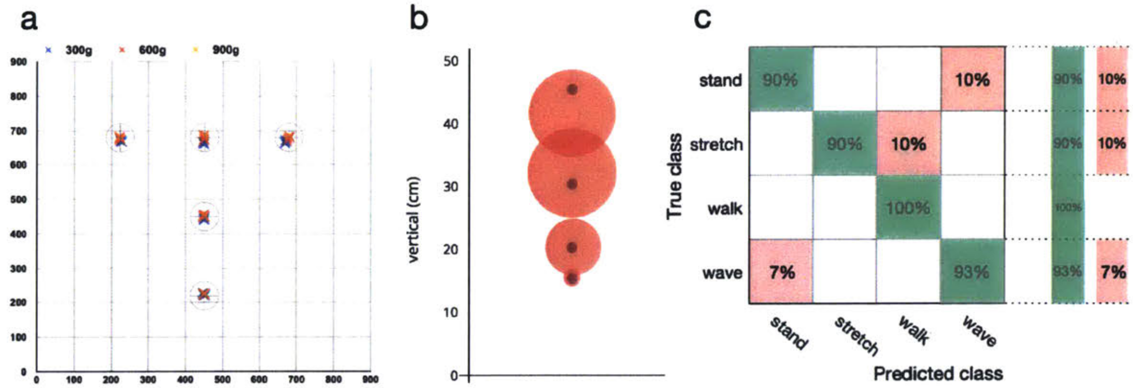

Figure 3-13: Results from Technical Evaluations: (a) Horizontal Accuracy from *Object Status Tracking* (b) Vertical Accuracy from *3D Touch Detection* (c) Confusion Matrix for *Motion Pattern Recognition*

3.9 Discussion and Limitation

Multi-touch Inability

Our system is not designed to accept multi-touch input in parallel to other load sensitive systems. If two different people are interacting on the surface or handling the objects simultaneously, the data processing framework would fail. This is because as shown in Fig.3-1 we combine all of the signals from load cells into one force line at the very beginning of the processing pipeline. Thanks to the modularity of our system, we can apply different modules beneath two areas where a user would like to separately detect multi-touches.

Database

An additional database about *product*, which is composed of product name, sale price, or materials would be useful to build wider applications, especially for *Object Status Tracking*. For example, our basic system stores the set of (*weight*, *position*) as shown in Fig.3-4. Once we assign the initial relation between weight and product, or position and product, the system is capable of tracking all changes during its execution.

Speed

This system has 80 Hz throughput of touch point detection, yet we are facing an unavoidable latency of at least 0.25 sec, since the system requires this for the acquisition of a bundle of quasi-parallel action lines, and it usually takes more than 20 samples. Although our system could apply to 3D input, there are limitations in expanding to temporal critical applications, such as making instant musical instruments with pieces of cardboard.

Scalability

We can deploy a much larger system, such as a load sensitive floor on an architecture scale, with the advantage of area and weight scalability. Thanks to the modularity of our system, we can put as many modules as a user requires to meet the maximum load requirement. If the user exceeds the load tolerance of the system, they have another option of using higher-capacity load cells, such as ones with 100kg tolerance, in turn sacrificing the minimum distinguishable weight on the platform.

3.10 Conclusion

We proposed a load processing framework with load sensitive modules for enhancing force-based interaction, and explored its design space with scalable and variable architecture. The workshop with corporate designers shows a range of applications and the utility of a modular prototyping kit with the algorithm including 3D touch detection. We envision the SCALE framework provide ubiquitous interactive surfaces with scalable load sensitive architecture to capture scalable Force-based Interactions of everyday activities for further analysis of human object interaction.

Chapter 4

KIOSK: SCALE Application for Customer UX in Farmer's Market

4.1 Introduction

The increasing availability of information communication technologies in retail stores has created retail technology solutions in recent years[16]. For example, recognition technologies for customer buying behavior[71, 25, 55, 63] has taken an important part for emerging automated goods control and checkout[3, 43]. On the other side, regarding the trend of retail stores, small temporary stores such like directly selling by producers has been increasingly attracting attentions to meet diversified demands of customers with differentiated products[50].

In a condition like this, toward the future, retail technology solutions should be enhanced enough to expand to every stores from supermarket to farmers market. The reason for that is the fact that it is increasingly needed to improve efficient supply management in stores as well as comfortable customer's experiences, even if in small temporary stores. However, the limitation that makes seller's efforts in installation of system tends to be an obstacle to deploy to every stores. For example, regarding computer vision(CV) or radio frequency(RF) technologies based recognition system, the installation needs seller's efforts for setting sensors up at a proper place for sensing. Especially, for farmers market's sellers, the problem of installation becomes significant due to its low-cost operation strategy. Thus, we started exploring a minimum viable goods control system so as to realize easy installation for small temporary stores.

In this study, we assume that the experimental goods control system should be designed on the basis of two concepts that we defined as follows. The first is to be system embedded configuration. In order to enable easier installation and the versatility of use, the electronic components such as sensor devices should be embedded into selling equipments itself like shelves or tables for goods display. Thus, the sensor device component must be designed for minimal simplicity, and also the component should be comprised of conventional devices. The second is to recognize consumer buying behaviors in real-time. As we can see in large store's tend, recognition of consumer's behaviors is naturally considered as a primary function for automated solutions and in-store marketing. But then, we have only to focus on the necessary behaviors to keep simplifying the system.

This study aims to create the first prototype which has the characteristic of easier installation and recognition of customer behaviors with a simplified configuration for smaller temporary stores. Though this study suggests an early-stage prototype without a field verification, we believe that this study will be the first step to helping an expansion of emerging retail solutions to the long tail of retail stores.

4.2 Related Works

4.2.1 Customer Buying Behavior Recognition

Regarding customer buying behavior recognition and analysis, the studies have been increasingly conducted thanks to progression and generalization of behavior sensing technologies including CV or RF technologies.

In this study, the challenge is to enable unobtrusive for the interaction behaviors between customers and goods in farmers market's small stores. In the light of the condition, several studies have explored how accurately the behaviors are recognized by using conventional image sensors including Kinect. Popa et al suggest a system for analyzing customer behavior patterns related to products interaction, and they obtained an accuracy of 80 percent for six basic handling actions they defined[54, 55]. Regarding utilizing RF technology, several researchers have investigated to easily grasp the customer behaviors[63, 62, 73]. They suggest motion recognition methods by detecting RFID tags which are attached on goods. In addition to them, Avrahami et al present an unobtrusive recognition system which utilizes an RF-rader sensor mounted under a table surface[5]. The system works with high recognition accuracy in a lab environment. Moreover, Sharma and Lee suggest a unique method of sensing sound that customer's smart watch emits to grasp customer behaviors[64]. Although these systems perform high recognition accuracy by using conventional devices, the systems have the limitation of target goods and a range of goods handling behaviors, and need to user's installation efforts.

4.2.2 Surface-Embedded Object Sensing

Here, we aim for the unobtrusive sensing on a surface, which means that the sensing system requires no attachment, wrapping or coverage on the object itself. Instead, the focus is on the interactions with the object placed on a surface. One popular technique is object recognition based on computer vision systems [33]. To address the issue of occlusion, or line of sight, here we focus on table-embedded vision systems. There has also been research based on the token-based touch sensing technique, yet this approach requires pre-defined tokens or markers [31, 32, 7].

We decided to utilize load-based approach as the touch sensing platform, since it has an advantage that is capable to capture both of the total load and the center position of the load based on simple triangulation. This simultaneous acquisition of weight and position is widely used in the applications from robotics to human computer interactions [10, 59, 60, 47], however, the spatial resolution is limited in horizontal 2D plane and no way to detect the height of the touch point from the sensory plane.

Here as a conclusion, we are focusing on load-based approach which allow a user to measure the both of position and force directly and provide a thin and single form factor for easy installation.

4.3 Design

A first step in our exploration of prototyping of the goods control system, we named KI/OSK is the definition of the design specifications. Before that, we need to consider the requirements of smaller temporary stores. To figure out the specification of our goods control system, here we provide *Preliminary Investigation*, *Design Requirements* and *System Specifications.*

4.3.1 Preliminary Investigations

The requirements are compiled in keeping with preliminary investigations by observations of farmer markets and interviews with the persons who concern about organizations of farmers markets in Tokyo, Japan as follows. Regarding the limitations of space regulations, each shop is basically assigned to a space(typical size, 3.0m square) and one or two tables for goods display shelves(typical size, W:1.5m x D:0.6m x H:1.0m). And, as characteristic of farmers markets, in most of cases, each store in a farmer's market is managed by one seller who almost is producer of the goods.

Accordingly, the stores provide unique shopping experiences like talking about goods with sellers. Moreover, farmer's market has usually a lot of fresh foods stores due to customer's high-demand for purchasing fresh foods. Most of the customers tend to be interested in higher quality and lower price foods according to their lifestyle. On the other side, seller tends to be bothered about unaccustomed checkout, packing operations and compiling sales data. And also, they are interested in the customer's interaction with goods to know the interests of customers.

4.3.2 Design Requirements

The requirements for KI/OSK are as follows, (1) KI/OSK focuses on recognitions of two customer's buying behaviors phases, selection of goods and check-out goods.

In the selection phase, since customers tend to select goods while handling them, the recognition naturally record physical interactions between customers and goods on shelves. In the chef-out phase, we aim to realize an automated registering and calculating transactions at cashier to support the seller's check-out operations. (2) KI/OSK focuses on fresh foods and drinks which are major commodities in farmer's markets. We aim to enable the recognition system as mentioned above to apply for a wide range of foods. (3) KI/OSK employs an approach to embed electronic components into a table equipment in order to realize easy installation for regular farmers market stores. We aim that seller have only to set a table up and connect with PC, and they are able to use KI/OSK.

4.3.3 System Specifications

We decided for KI/OSK to employ a load-based sensing approach among other available technologies mentioned in related works, since this approach is capable to satisfy the requirements above. The reasons are (1) load is occurred by object's weight that is an intrinsic property of object and force acting on an object. Thus, object's load variation is recognized via on everywhere the object contacts, (2) load sensing suitable for selling fresh foods and drinks by weight, (3) load sensor is a conventional which is low-cost and small-size. On the other hand, CV based recognition approach has the limitation of configuration design owing to considering the position of cameras or tags and the light condition, and recognizing an amount of foods.

In the prototyping, the challenging are how the load sensing system should be designed to accurately recognize load sequences, further customer's buying behaviors and simplify embed into a table top board. We hypothetically design KI/OSK configuration as shown in Figure.1. The first prototype system is basically comprised of the minimum number of load cell sensors with the cell amplifiers and a microprocessor to grasp object position on the top board and embed between table top boards.

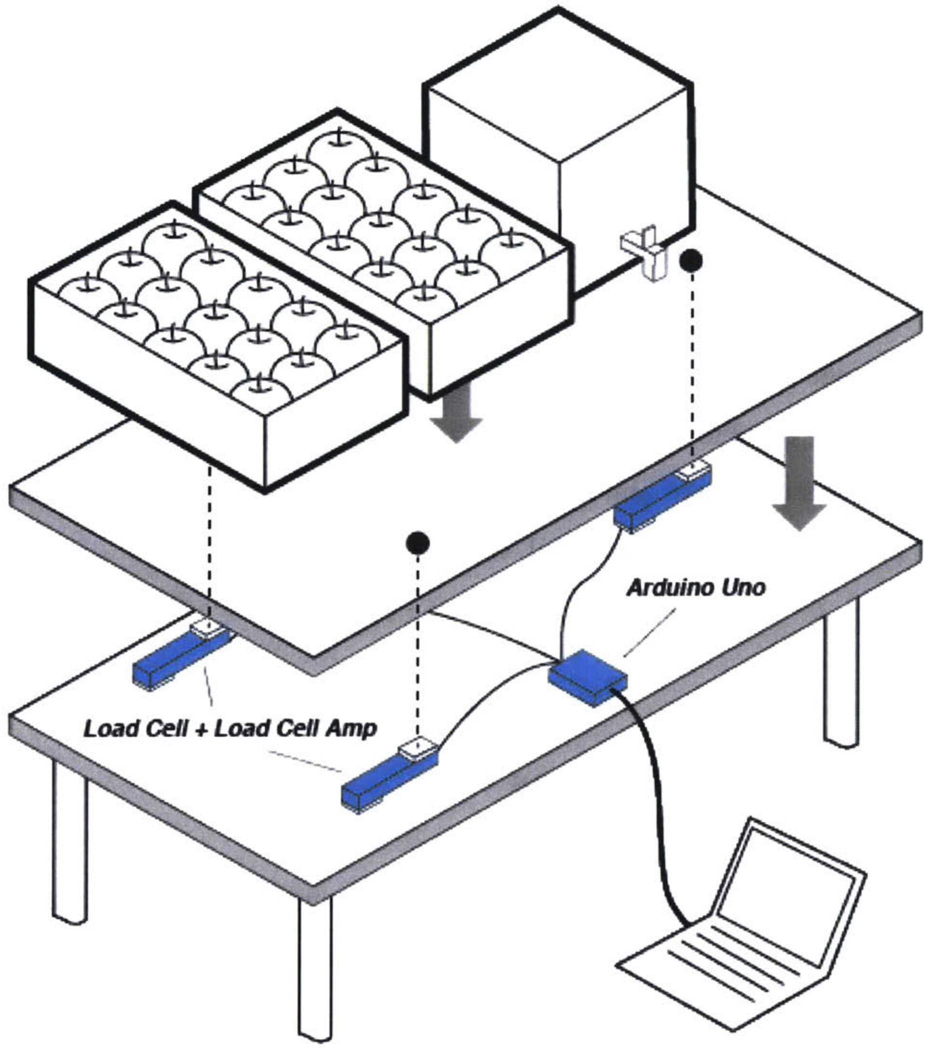

Figure 4-1: KI/OSK system configuration. the load cells, load cell amplifiers and microprocessor are located between top boards

4.4 Implementation

Here we explain how the hardware and software in KI/OSK system is implemented, followed by the design guideline we mentioned in advance.

4.4.1 Hardware

KI/OSK platform contains both of the electronics to achieve the position sensing with load cells and the wooden board interface to provide the natural affordance as well as a common table in a farmers market. In the electronics, three sets of a load cell (TAL220) and its amplifier circuit (HX711) are connected to one microprocessor (Arduino Uno), which sends the measured data to a laptop (Macbook Pro).

Each load cell is fixed at a vertex of a triangle beneath the wooden board with screws to make tight and rigid connection for better sensitivity. Every single load cell has maximum capacity of acceptable load, which is 10 kg with the low cost sensor in the prototype. This is durable enough for a table in a farmers market, but a designer who has much severe requirement could choose a load cell with more than 100 kg capacity. Also, we prepared the wooden board with 0.6m x 1.2m, which has the same dimension as the top panel of an off-the-shelf table, to conduct actual-scale experiments.

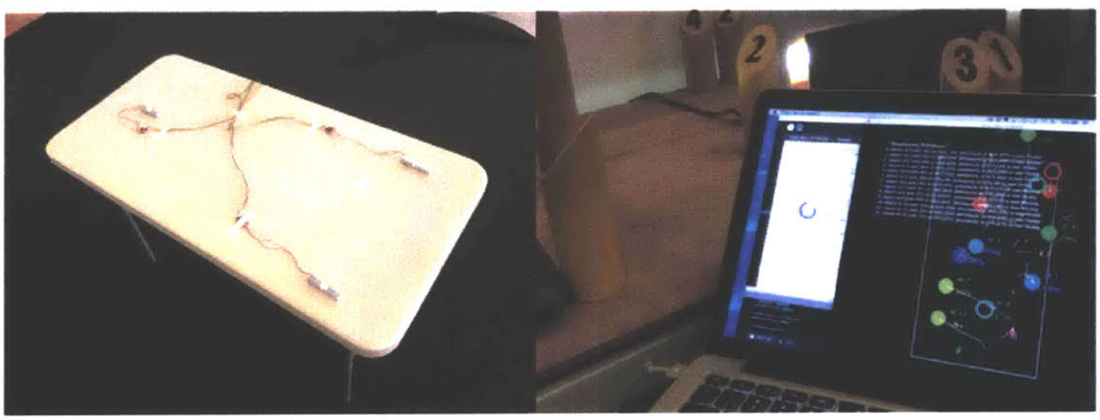

Figure 4-2: (Left) a hardware prototype of KI/OSK platform, with the removal of top board (Right) the software capturing all the object transfer and user interactions on the platform

4.4.2 Software

The design goal of the software is to process the acquired data and recognized the basic interactions for inferring customer buying behavior. The data sequence containing the triplet of measured weight is continually sent from the Arduino to KI/OSK software in a laptop, developed in the Processing environment. Here we explain the software pipeline, including *Load Localization*, *Object Detection* and *Object Tracking*.

Load Localization

We implemented the localization system based on a simple triangulation with the positions of load cells and the incoming triplets of measured weight, denoted as $\mathbf{p_i}$ and w_i respectively, where $i = 1, 2, 3$ represents the id of each load cell. Here the system could estimate the current total weight w and its position $\mathbf{p}$ by following formula.

$$w = \sum_i w_i \quad \text{and} \quad \mathbf{p} = \sum_i w_i \mathbf{p_i} / \sum_i w_i$$

Object Detection

The goal of *Object Detection* is to distinguish the signal pattern when a new object appears from the other patterns caused by user interactions, including touching or dragging. These interactions by a human hand make fast fluctuations in the sequence of measured load and hence distinguishable difference from an object emergence. Since we treat $w > 0$ as object emergence and $w < 0$ as object missing, the system is able to capture object missing in the same manner.

To capture the fluctuation arising from user interactions, we applied two different low pass filters (LPF) on the calculated total weight w, called *Fast LPF* and *Slow LPF*. Since these two filters are implemented based on exponentially weighted moving average, where the smoothing weight for the newest data sample is 0.25 and 0.04 respectively, Slow LPF has to have more stable and steady value. Therefore the system recognize the emergence when the absolute difference between Slow and Fast LPF becomes lower than a certain threshold. For detecting multiple object on a single surface, once an object is detected and recorded to the database the sensor values are calibrated to be zero again for next sampling.

Object Tracking

We implemented *Object Tracking* function on the software, lying on the fact that each object has unique mass inside and hence this could be utilized as unique identification when it appears again after a object with the same weight was missing. Here we assume that two objects with the same weight are not picked up simultaneously, in other words, the system expects that missing objects with a certain weight is unique. This functionality of tracking object provides a user to return a product after their pick-up since the system identifies it as the same product by the stored weight.

4.5 Technical Evaluation

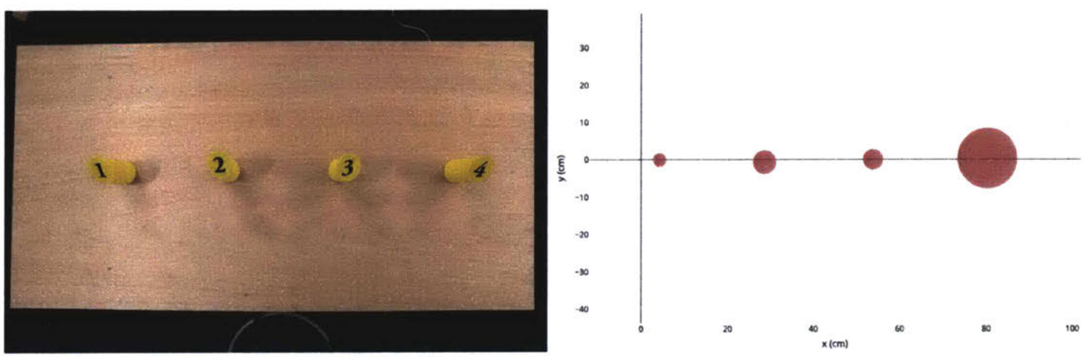

Figure 4-3: The evaluation of the accuracies in position and weight. (left) the setup of technical evaluation (right) the plot of measured points and standard deviation in repetitive measurement

We conducted technical evaluations on following basic components: weight, position and recognition time. For each target value we prepare 4 reference point and a test cylindrical object with 167.5 gram weight as shown in Figure.4-3 and put the object on the reference points repeatedly. The measurement of weight, time and position is conducted 10 times respectively and then we calculate its average and standard deviation. As a result, averaged weight error is 6.1 gram, averaged position error is 7.3 cm at maximum and averaged recognition time is 1.48 seconds.

Also, we checked if three basic human interactions, including touching, dragging and tracking are surely captured by the system. You can see the colored circle appears when touched, dotted lines are emerged when dragged and rings are connected when tracked, as shown in Figure.4-3.

4.6 User Study

We conducted user studies for test the applicability of the system into a real scenario of farmers market. We assigned the following task for 3 participants as behaved as customers in the store. The task statement is: "Choose three better apples and two better oranges, then pick them up into your basket within 30 seconds". We compared the count of detected interaction, including pick-ups and returns, with the ground truth, and then calculated the detection rate.

With the 63 accumulated trials, we have obtained the detection rate is 76 percentages. There is less confusion in the interaction classification level, such as confusion pick-ups with returns, since there is a significant difference in positive sign and negative sign of measured weight. However, we found that most of the confusion is caused in area classification level, which means the precision of localization process in our system. In this practical setting-up, it would be better to work to remove the external noise and incomplete isolation from vibration sources.

Figure 4-4: The photographs from user studies. (left) A test subjects choosing and picking up a set of fruits. (right) The system can recognized the amount of a cup of coffee as well.

4.7 Discussion

In this section, based on the evaluation results above, we compile the effectiveness, the problems and the future works of KI/OSK according to the design concepts which we defined to realize.

4.7.1 Easy installation

We suppose that KIOSK realized easy installation due to its embedded system configuration. We could prepare the experiments just by set the table and connect with a PC. Besides this, KI/OSK basically makes it easier to register goods before selling. KI/OSK's *Object Tracking* function allows to easily register the location where each goods is put on. For example, a seller can intuitively input the location just by tracing around the goods on the top board by his/her pointer finger.

On the other hand, we found that the load sensor module should be designed for easy mounting and removing. Because sellers actually use various shapes and sizes tables according to their layout design or might change the layout during selling, the design specification of module structure should be reconsidered.

4.7.2 Recognition customer buying behavior

We obtained the result that KI/OSK allows to recognize the customer buying behaviors that we expected. In selection phase, recognition of touching, picking up and return goods worked while distinguishing the signal patterns as we explained above. In check-out phase, KI/OSK determines pricing as much as participants take as we expected. Similarly, the system could recognize goods which have both characteristics of solid and fluid, and determine pricing by weight.

However, the current recognition system has the limitations that should be addressed for the future field study. At first, KI/OSK should be improved to make a distinction between goods load variations and others such as load variations from the environment or goods handling by sellers. As one possibility, the system should understand a lot of customer behavior patterns to distinguish from the noisy load information. To do this, we have to define more goods handling patterns based on field investigations as Popa et al attempted to define customer goods handling patterns for accurate recognition[54]. Secondary, KI/OSK should be improved to recognize multiple customers' behaviors at the same time. As an approach, we assume that load sensor is embedded into shopping baskets which customers use to recognize the person who takes the goods.

4.8 Conclusion

In this chapter, toward efficient supply chains and comfortable shopping experiences in physical stores, we proposed KI/OSK platform, which allows unobtrusive sensing of customer buying behaviors, including touches, pickups and returns. We clarified the design specifications based on preliminary investigation, implemented the prototype and conducted both technical evaluation and a user study. We envision the future ICT solutions weaving into not just only large retailers but also small owners in farmers market.

Chapter 5

DepthTouch: Augmenting Surface Interaction for Volumetric Displays by Force Sensors

5.1 Introduction

The rising demand for 3d imagery in the past few years has been pushing the envelope of 3d display research. Volumetric displays, which put the light-sources in a 3d volume, have the advantage over other methods in terms of no accommodation vergence mismatch etc., including swept-volume, static-volume and free-space methods [65]. Researchers also have been investigating how to interact with the 3d contents displayed in the display volume, including selection, manipulation and data input [44].

Among these basic interactions, depth selection is an essential technique so that a variety of principles has been proposed so far, including mechanical or optical methods etc. [20, 1]. However, these methods require the use of additional devices, providing a less than ideal sense of interaction. Previous studies have reported the usefulness of interacting with a built-in transparent enclosure surface, which is inherently part of the volumetric display [6].

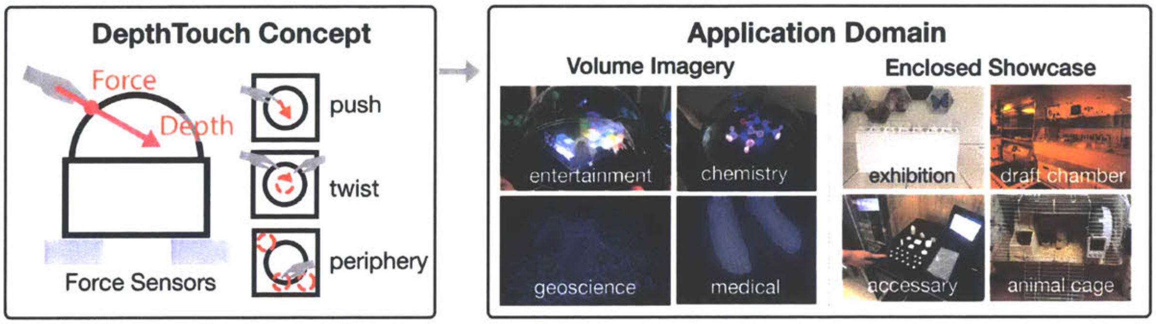

Figure 5-1: Design Space for DepthTouch: (left) The basic concept of force-depth conversion. Atomic functions of the system includes pushing, twisting and assigning peripheral inputs. (right) Application Domain of the system, including interaction with Volumetric Imagery and interaction with Enclosed Showcases. Photos for Volume Imagery are from Voxon website [53]

In this chapter, we have explored the design space of an interactive display enclosure combined with the 6 DoF force vector capturing method [70]. We propose DepthTouch system, that determines the cursor position along loaded force direction, enables us to interact with contents / imagery in a physically separated volume by an enclosure boundary, through detecting the exact location of the touch point on the dome and its force intensity. This method provides volumetric displays depth selection capability with spatial coordinate consistency between CGI and our body, without having the argument of controller-mapping for the conventional input methods. Moreover, our framework is capable of augmenting a variety of enclosures, including glass showcases in a museum, animal cages at a zoo, and chemical chambers in a lab. We have clarified the design space and architecture for DepthTouch system, and have implemented prototypes. Also, we have evaluated the user performance in a pointing task, comparing to the conventional 3d mouse method. and demonstrated multiple application scenarios.

Our contributions described in this paper include:

- An architecture and design space for DepthTouch to allow a range of force-based interactions, including quick depth selection, directional pointing, and other gestures with exact spatial coordination.
- Technical implementation of hardware and example demonstrations showing a

wide range of applications from volumetric displays to exhibition glass showcases.

- Outcome and findings from our thorough user study, revealing the performance of *depth selection* conducted with 12 human subjects.

5.2 Related Work

5.2.1 Basic Interaction with Volumetric Displays

To display 3d imagery to a user without wearable devices, researchers have been investigating different principles, including ray-based lightfield methods [49, 30], point-based volumetric methods[14, 36] and wave-based holographic methods [65]. Among these methods, a volumetric display is capable of focusing a light source at an arbitrary point in 3d in a display area, so that makes multiple users simultaneously to watch 3d imagery with both of correct motion parallax and correct binocular disparities, without accommodation-vergence mismatch [14].

In addition to the volumetric method, researchers have proposed and explored a variety of interaction methods with displayed content. According to the survey paper [44], basic interactions are generally categorized into four different types: selection, navigation, manipulation and data input. To select displayed content in 3d volume, the user can use methods such as 3d mouse and peripheral controllers, optical tracking, and touch-sensitive enclosures. 3d mouse and peripheral controllers have been widely accepted in commercial products [1, 53], and optical tracking using hand-held devices are also well-investigated by researchers [21, 20]. For touch-sensitive enclosures on a non-planar surface, researchers have attempted this method, however it is difficult to fabricate a capacitive panel onto an arbitrary 3d curved shape [34].

In this chapter, we explore depth selection and other basic interactions with touch-sensitive enclosures based on the force-depth conversion technique, which is inspired by the pencil-type product that converts its touch pressure onto the surface into positional depth 'under' the display surface [37].

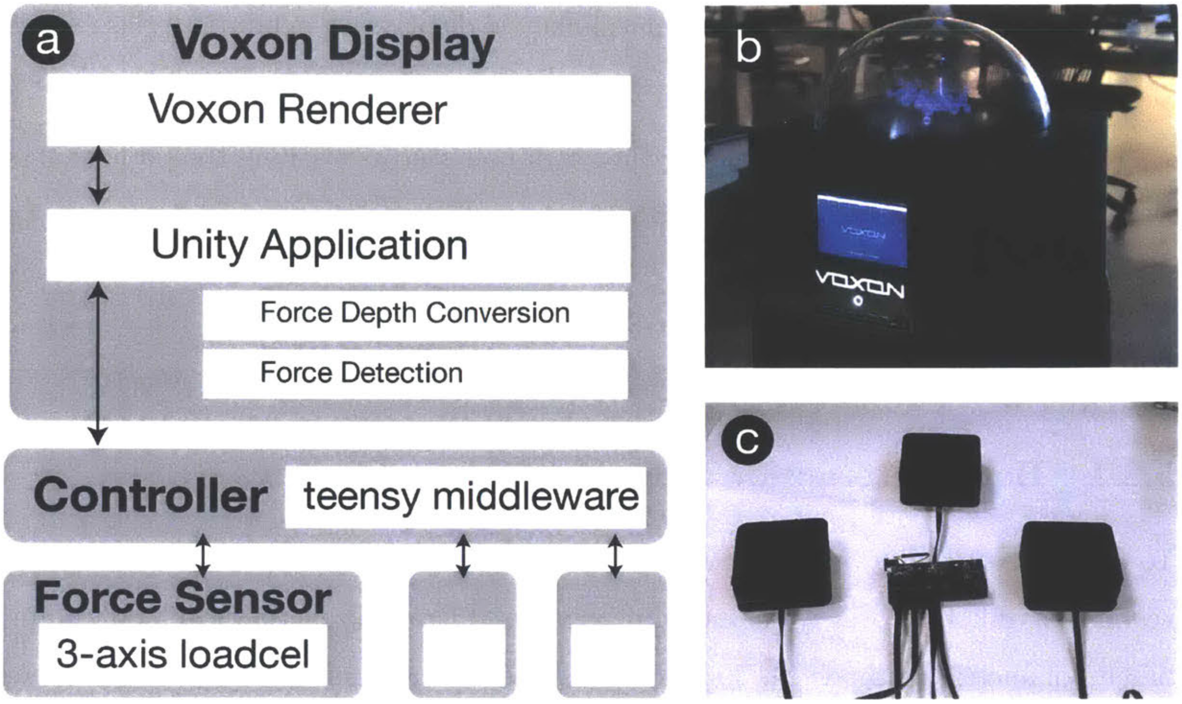

Figure 5-2: DepthTouch System: (a) Workflow (b) Voxon VX1, 3d Image Output Device (c) 3d-axis Force Sensors and Controller

5.2.2 Force Vector Sensing

To capture force-based interaction between the user and the display enclosure, we have several design choices including a capacitive touch screen [56, 61], swept frequency capacitive sensing [23, 58], acoustic sensing [51] and force sensing [68, 12].

Among these methods, it is critical to capture the force vector with spatial and 3-axis resolution, meaning we have two design choices: a single centralized 6 DoF (force and torque) sensor [24] and a set of distributed 3 DoF (force) sensors [70]. From a mechanical engineering point of view, it is better to have multiple supports for the structure above, so that we utilized distributed force sensors. Note that even though 6 DoF force sensors are generally much more expensive than a set of 3 DoF sensors, if the target enclosure size is relatively small, eg. less than 10 cm, it is better to use the 6 DoF sensor due to the limitation of size.

5.3 DepthTouch system

DepthTouch is an input technique that enables direct selection and manipulation of 3d contents in a volumetric display, with additional sets of force sensors placed beneath / beside the volumetric display. Force sensors calculate the force intensity, direction and the contact point simultaneously, so that the system is capable of interpreting the information into accurate input results for the contents in the display, as shown in the left of Fig. 5-1.

The immediate processing of force information results in a wide range of real-time application, as shown in the right of Fig. 5-1. The application domain covers two different digital / physical contents: Volume Imagery and Enclosed Showcase, which we will describe in detail in the following section.

5.3.1 Volume Imagery

A user could deploy DepthTouch system with a volumetric display, followed by the display immediately obtaining interactivity, with the user input of the enclosure dimensions.

The user can manipulate a 3d point as a cursor by pushing on the enclosure, and selecting the content inside by hovering over or applying more force on the desired area. In addition to using the system as a point cursor, manipulating a 'plane' could exploit more advantages for the DepthTouch method. For instance, snapping layers on a geological landscape model or creating an incision during surgery by changing the direction and position of the manipulated plane according to pushing force, can be a beneficial scenario on the proposed system [11, 48].

5.3.2 Enclosed Showcases

The DepthTouch system can be applied not just to digital content in 3d volume, but also for physical objects in an enclosed space, such as displayed items in jewelry shops, museum exhibitions and animal enclosures. With the simple input of the enclosure dimensions, eg. width $\times$ height $\times$ length, and the relative position of force sensors to the enclosure's center, the system is capable of detecting the force line of touch and its contact points, which is intersecting with the enclosure surface.

In this application domain, the enclosure starts to understand the user intention, on top of the specific application context. For example in a jewelry shop, the user can point to and select a specific wedding ring in a glass showcase by pointing at the ring with their forefinger on the surface, as a natural extension of pointing at objects. In an aquarium, a child could point at a fish by touching the glass surface, and through voice input, ask, "what's that?" to the system. With the additional input of the relative position of the fish in the tank, the system is capable of answering the child's question and identifying the fish.

5.4 Implementation

5.4.1 Hardware

Force Input

For the force sensor, we utilized load cells (FNZ100N, Forsentek.inc) with a load capacity of 10 kg, and packaged each sensor into a small cuboid with a 90 x 90 x 35 mm form factor, as shown in Fig. 5-2. We have placed three force sensors on the support structure with 24-inch square, at the front corners and the middle of the backside edge. More detailed fabrication protocol of the force sensors are available in the related project [70]. For the communication between force sensors and the central PC, we connected the sensors to teensy 3.6, the Arduino compatible processor.

Imagery Output

Here we describe the hardware that displays volumetric imagery. As a display, we have used Voxon VX1 (Voxon Photonics.inc), which has the volume for imagery area with a cuboid of 180 x 180 x 80 mm [53]. The device has a vertically-moving translucent screen inside the acrylic dome, and a high-speed projector with 4,000 fps projecting a patterned image corresponding to the swept height of the actuated screen.

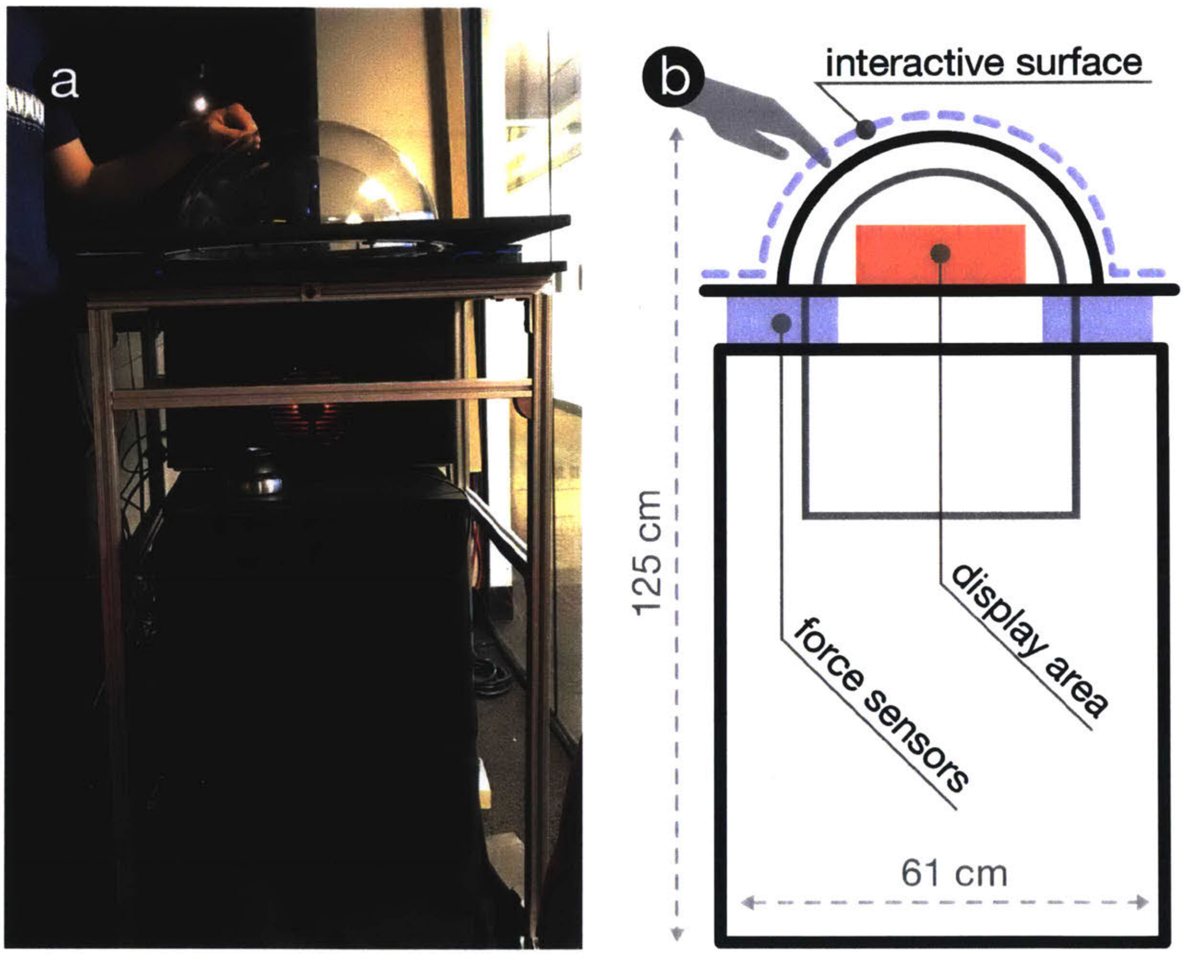

Figure 5-3: DepthTouch Hardware: (a) the photograph of system outlook (b) the structure and dimensions: Voxon display in gray lines, mechanical support in black lines and force sensors in blue squares

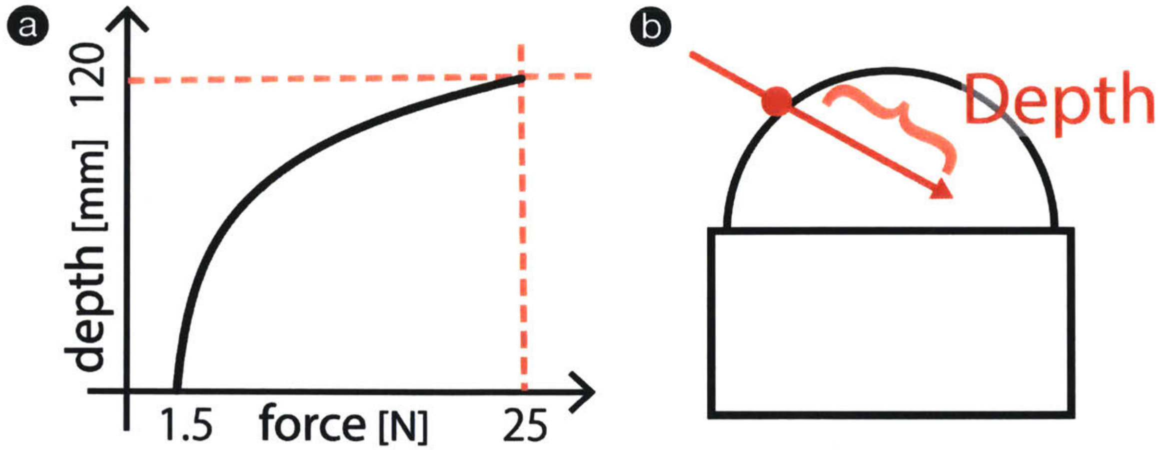

Figure 5-4: (a) Logarithmic Conversion from Force to Depth based on Weber Fechner Law (b) The depth is measured along the force direction from the contact point

Coverage for Mechanical Isolation

To construct a touch-sensitive acrylic enclosure over the 3d display, one can immediately imagine the architecture with the force sensors just beneath the legs of the display. However, force sensors are so sensitive to external vibration that they need to be mechanically isolated from any vibration sources around the force sensors. Therefore, we designed an acrylic cover over a dome of the 3d display to mechanically isolate both the sensors and the display, as shown in Fig. 5-3. We utilized aluminum frames with a 50 mm square intersection to build a support of 610 x 610 x 950 mm, and install an acrylic hemisphere with 400 mm in diameter to cover the entire display area. At the top, the outer dome height is 1250 mm.

5.4.2 Software

3d Application Environment

We implemented an interactive 3d application in a Unity 2017 environment with Voxon VX1 SDK[26]. We exported the application into an execution file for Windows, and sent it to a Windows 10 environment, which has a built-in OS for Voxon VX1 hardware, as shown in Fig.5-2.

Force Line & Contact Point Detection

A *force line* is the key element for detecting the contact point, which is mathematically represented as a set of *force* f and *contact point* x. Here we describe how to retrieve a single force line from force sensors. We assume the sets of measured force f_i, sensed at i-th load module ($i = 1, 2, 3$). For simplicity, we can assume that all sensors are placed at p_i on the same plane. We placed three sensors at $p_1 = (250, 250, -30)$, $p_2 = (-250, 250, -30)$ and $p_3 = (0, -270, -30)$ in millimeters.

The touch force f and its torque τ are derived as $f = \sum_i f_i$ and $\tau = \sum_i p_i \times f_i$ by definition. The force line is expressed as $x = a + pd$, parameterized by scalar p. The normalized direction vector d is $d = f/|f|$ and the anchor point is $a = f \times \tau/|f|^2$. With the geometry of the enclosure, eg. hemisphere with 400 mm diameter, the system is capable of numerically detecting the point of contact, at the intersection of the force line and the enclosure geometry.

Force-Depth Conversion

In converting force to depth, we experimentally designed a logarithmic transformation formula, based on a discussion of Weber Fechner law [15].

$$depth = d_0 \log(\frac{force}{f_0})$$

We experimentally determined the constants, 1.5 N as a minimum threshold of touch force and 25 N as a maximum. As shown in Fig. 5-4 we have selected the constants as follows: $d_0 = 100$ mm and $f_0 = 1.5$ N. This parameter can also be determined and updated in the future through a user study.

Peripheral Functions

In addition to direct touch input onto the enclosure dome, this system is capable of assigning a variety of functions triggered by the users touch onto the surface outside of the transparent enclosure. Here we describe two example functions enabled by force input: *button* and *slider*. Since the dimension of the enclosure dome and its

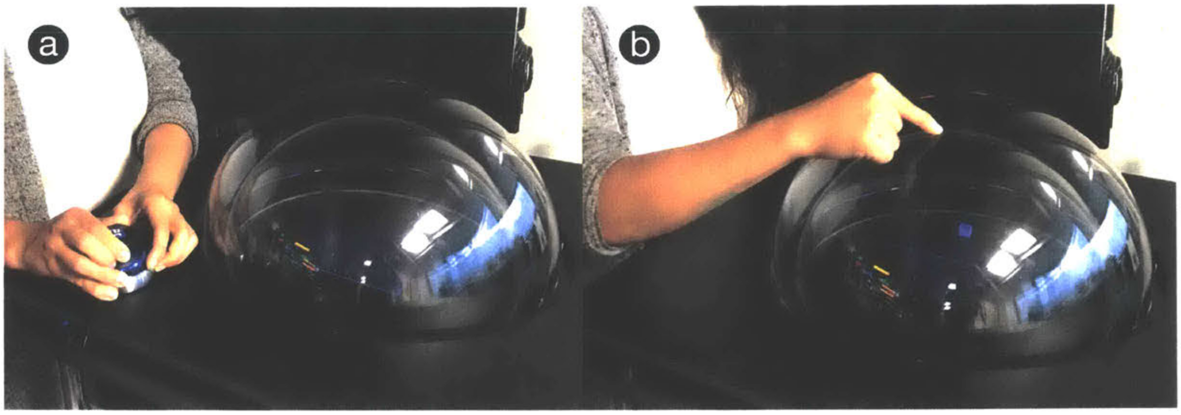

Figure 5-5: Input Methods for User Study: (a) With 3d mouse, a user controls the cursor position by wriggling the device neck (b) With DepthTouch technique, a user points the target through touching the surface and forcing differently in intensity and direction.

mechanical extension is known in design, the system is able to judge if the force line is intersected with a certain point. The user can assign *a virtual button* at arbitrary points onto the structure's surface. If a user put several buttons in a line, one could create *a virtual slider* as well.

5.5 User Study

We have designed and conducted user study to experimentally validate the concept of DepthTouch. To reveal performance in a pointing task, we decided to conduct the same experiment for 3d mouse in parallel. As shown in Fig. 5-5 these input methods have different design principles, still the result from 3d mouse experiment is thought to be useful as a baseline performance for the result from DepthTouch experiment.

For experimental design, we followed the protocol reported in a related work which proposed the pointing method with an optically-tracked hand-held device [20].

5.5.1 Procedure

We designed a 'static target acquisition' task to measure the performance of the pointing method. Targets are rendered as yellow wire-frame cuboids with 18 mm

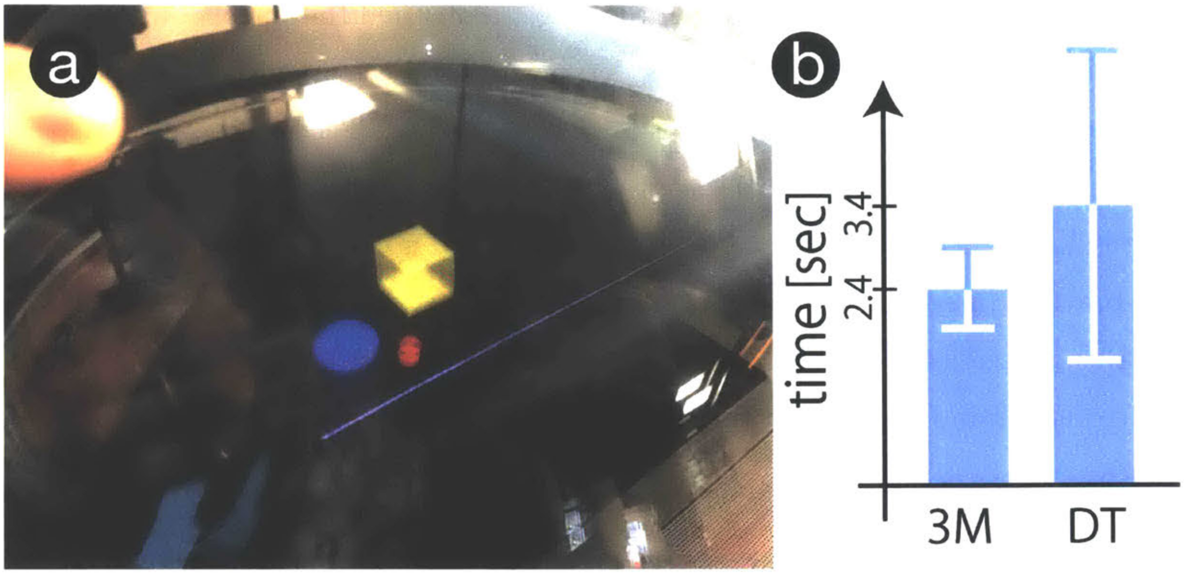

Figure 5-6: Results of User Study: (a) a static target acquisition task with a yellow target cuboid and a red cursor sphere (b) the elapsed time from one target to another for DepthTouch method. 3d mouse provides the baseline performance of the task.

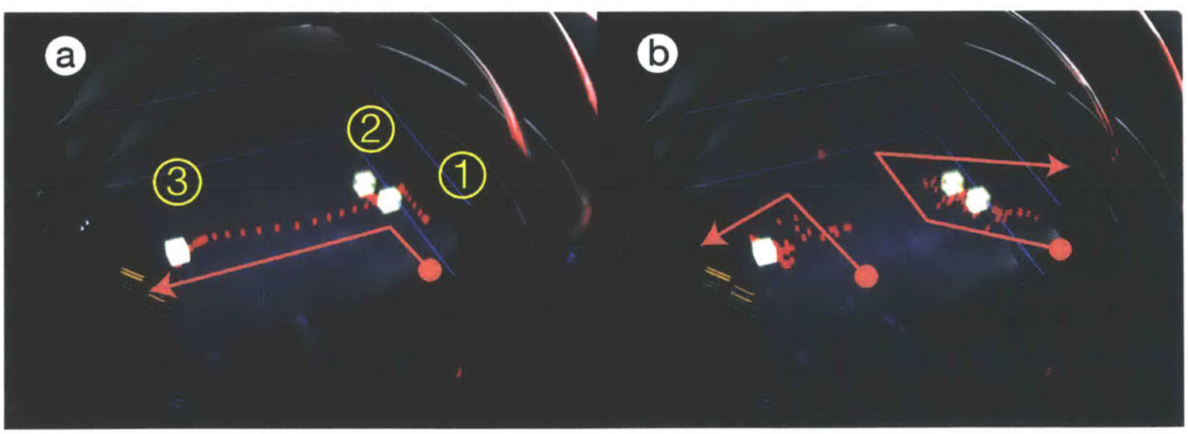

Figure 5-7: Comparison of the Cursor Trajectory. A user is asked to pass by yellow check points sequentially. The red dots are the trajectory of the cursor. The pink lines are the illustrated path. (a) With 3d mouse, the trajectory tends to be continuous and sequential. and (b) With DepthTouch method, the trajectory tends to be jumped and disconnected.

edges, as shown in Fig.5-6. The cursor is rendered as a red sphere with 10 mm diameter. Once the distance between the target center and the cursor center is close enough less than 14 mm, the target is determined as selected. A start target would randomly appear in the displayed volume of the display. The trial began once this target was selected, and a trial ended when 120 seconds elapsed. Once a target is selected, a next target would appear at a random position, with the distance from the previous target of longer than 90 mm. The mean elapsed time from one target to another is used as the task performance indicator. For control purposes, participants were centered in front of the display and were told not to move their feet during the trials.

As a baseline performance of the task, we compared the pointing performance by DepthTouch method with that by a commercial 3d mouse (SpaceNavigator, 3dconnexion Inc.) [1]. Before each session there was a 5 minute demonstration and had practice periods to warm up with 5 min each for DepthTouch and 3d mouse. After the practice, a user had 2 min fist trial, 1 min break and 2 min second trial. Therefore we had 4 min accumulative log data per method per person. The participant is asked to try the tasks in random sequence, such that 50 % of the users are asked to try DepthTouch first, and other half to try 3d mouse first. As the unpaid volunteer participant for user study, we had 5 males and 7 females with the age ranging from 19 to 35. One of the 12 participants was left handed and the rest were right handed. Participants touched the enclosure or manipulated the 3d mouse with their dominant hand.

5.5.2 Results

The result of the static target acquisition is shown in Fig.5-6. The vertical axis shows the mean elapsed time, which is the average of the travel time from one box to next box. This value is calculated per person at first, and then summarized into single value over all the participants.

The mean elapsed time of DepthTouch is 3.4 seconds (SD = 1.9 sec), and that of 3d mouse is 2.4 seconds (SD = 0.5 sec), as shown in Fig. 5-6. The result shows with the proposed method a user take 1.0 seconds longer than with 3d mouse. Also, the standard deviation for the proposed method is larger by 1.4 seconds.

Notably, we found user # 8 showed the quickest performance, with 1.24 seconds (SD = 0.85 sec.), in DepthTouch among all the task including 3d mouse. We also found the a certain set of users could not get familiar with the proposed method. They are almost double-scored participants, compared to the DepthTouch average, including 8.0 sec of user # 6 and 6.3 sec of user #7. Without having them as outliers, we got the 2.7 seconds (SD = 0.8 sec). The individual difference is seen larger than 3d mouse, in the limited practice time.

Note that we analyze data of 12 participants for DepthTouch and of 11 participants for 3d mouse, since the 3d mouse experiment for user # 3 couldn't be conducted due to machine trouble. Also we treated data from 2nd trial, omitting 1st trial data as part of practice time.

5.5.3 User Comments

We also have received a variety of feedback including the positive and the negative, through the questionnaire which the participants have answered right after the user study. Toward qualitative understanding of user experience, here we have categorized the comments to similar groups, and show the dominant comment, which is mentioned from at least three participants.

For the positive feedback, the dominant comments mentioned *Control*, *Response* and *Enjoyment*: "I feel like I could control every aspect of the touch and interaction. The amount of force and angling of my finger were a couple of the things that made the DepthTouch experience feel more authentic and personalized." *(Female, Software Engineer)*, "faster than 3d mouse, more degrees of freedom because both position and direction vector were present." *(Male, Mechanical Engineer)"* and "New experience, fun to play!" *(Male, Chemical Engineer)*. For the negative feedback, the dominant comments mentioned *Stability* and *Accuracy*: "At the shallow layer, the DepthTouch is easily influenced by vibration." *(Male, Audio Engineer)* and "It would be nice if the control of depth/pressure is more sensitive." *(Female, Mechanical Engineer)*.

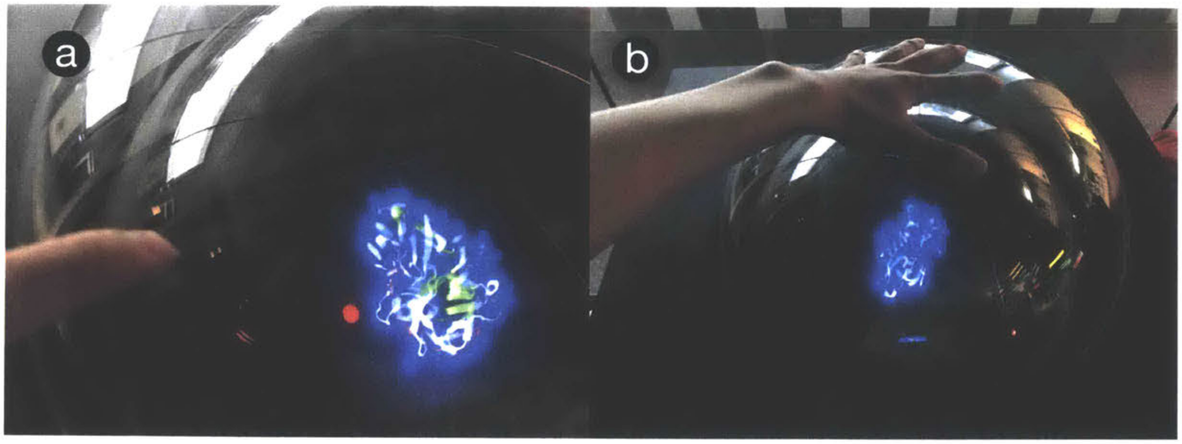

Figure 5-8: Application - *Protein Viewer* (a) selecting a molecule by touch (b) rotating a model by twisting

5.5.4 Discussion

We found a significant difference between two methods in terms of way of interaction, as shown in Fig.5-7. Here, a user is passing by yellow boxes one by one from the box 1 to the box 3. With 3d mouse, the trajectory is continuous and on a straight line, since the mouse input reflects to the differential change from the current position. It is easy to stop at a certain point, or 'hover', since a user can just stop to touch the device. With DepthTouch system, the trajectory tends to be jumped and fluctuated. This results in the difficulty of hovering a single point with the method. A user collects the box 1 and the box 2 in one motion, pushing at a same contact position but into different direction. This technique, pointing two or more targets without changing the contact position, contributes to quick access to multiple objects. Also in order to point the next box 3 in a distance from the box 2, users tend to change the contact position to the closer place, typically in the middle of the box 3 and user's body position.

According to the user's comment, the design guideline of our method starts to be formed in comparison with the established 3d mouse method. First of all, the Depth-Touch method could not be thought as the replacement of conventional 3d mouse input. Similar to other gestural midair input methods, DepthTouch users tend to be feeling tired during the interaction since they need to pull their arm up. However,

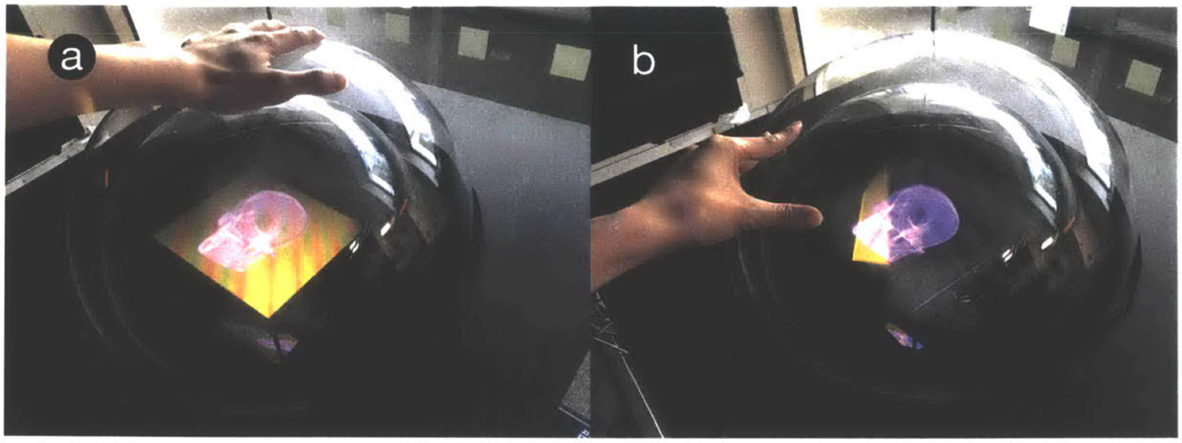

Figure 5-9: Application - *Medical Section Cut*: Slicing a 3d model with an intersecting plane at a different depth by pushing

specifically for instant pointing inside the object enclosure, some participants agree with usefulness and seamlesness of our method.

From technical point of view, some participants complained on the unintended fluctuation of the cursor position. This is thought to be partially from unsteadiness of a hovering hand, partially from incomplete mechanical isolation of the force sensors from the vibration source. The better mechanical design and comprehension of touch characteristics could be important future works.

5.6 Application Examples

5.6.1 Interacting with Volumetric Imagery

Protein Viewer

Seeing a digital scientific model in a 3d image viewer, such as a protein structure, it is required for the software to capture the user input, including selecting, scaling, rotating and so on, towards smooth interaction with the contents.

To demonstrate the capability of basic interactions, we displayed a protein model of pespin, and implemented the functions of selection by touching and rotation by twisting, as shown in Fig.5-8.

Medical Section Cut

In some 3d contents in medical or geoscience applications, it is critically important to see the planar intersection of the volumetric model.

To demonstrate this function, we implemented the 3d human model with manipulable 2D plane. A user can manipulate position and surface normal of the plane by force input to see the model from better point of view, as shown in Fig. 5-9. A user could push harder to see the contents at deeper intersection, and push less to see it at shallower position. If a user would like to watch the intersection from different point of view, one can achieve that only by touching from different point on the enclosed dome.

5.6.2 Interacting with Enclosed Showcases

Accessory shop

In an accessory shop with a bunch of different kinds of jewerly rings beyond a glass showcase, it is better to provide a user the quick pointing method to candidate rings. Augmenting the enclosed showcases with our force sensors, the system could help pointing remote objects at enclosed place.

As shown in Fig. 5-10, we demonstrate this scenario with small accessories in an acrylic transparent case, on the illuminating surface to indicate which object is currently under pointing. This enclosure understands a user's selection, and could interprets deep push to selecting, not just pointing, to convey more input information.

5.7 Limitations and Discussions

5.7.1 Scalability

We demonstrate the augmentation for display enclosure or showcases around 400 mm scale in this paper. However our system is inherently scalable to size of target enclosure, because we could choose different force sensor specification, specifically

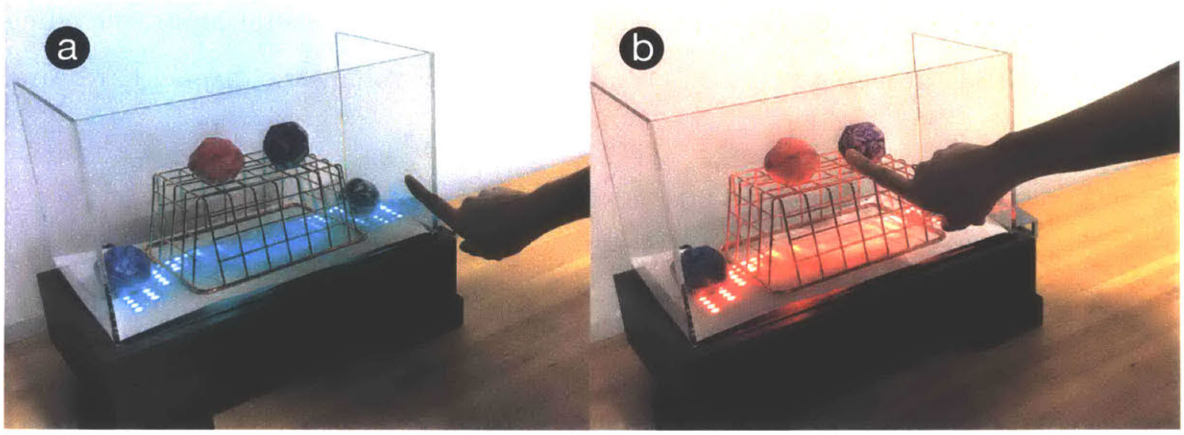

Figure 5-10: Application - *Jewery Shop* Selecting a exhibited jewery over the glass window case by pointing

the maximum weight torelance and load resolution of loadcels inside, based on the applications.

5.7.2 Multi-touch Inability

Our current prototype is not designed to accept multi-touch input, in parallel to other force-based approaches. If two different people are interacting on the surface or handling the objects simultaneously, the data processing framework would fail. This is because we combine all of the signals from load cells into one force line at the very beginning of the processing pipeline. Combining with additional spatially resolved layer, such as non-planar capacitive panel, could be a future possible direction towards multi-touch force-based sensing on non-planar surface [34].

5.7.3 Other Possible Interactions

Since our projects is implemented on the similar architecture suggested in the other loadcel-based sensing architecture, a user also could implement other possible interactions, including object detection or activity recognition, in our DepthTouch system as well [70]. For example, weight-based object detection technique could be used as a physical token to turn on the corresponded applications. Force signal-based activity recognition could be useful to classify different types of gestural input, including swiping, writing, dragging and etc., onto the surface of volumetric display hardware.

5.8 Conclusion

In this paper we proposed DepthTouch system that captures force vector of human touch, and interprets its intensity and direction to the 3d point or planar cursor in the isolated enclosed volume of 3d display. This method could also augment inert physical enclosures, including glass showcases and animal cages, to enhance pointing and selecting applications. We envision the living environments with the distributed and embedded force sensors are capable of capturing human object interaction unobtrusively to comprehend human behavior and intentions.

Chapter 6

Conclusion

6.1 Summary of Force-based Interaction

In this thesis, I have explored the force-based interaction, the most fundamental conveyor of our intention to physical objects through our body, specifically a hand / a finger. Throughout the two years research of Human-Object Interaction, I have revealed the exciting capabilities of force vector information.

- Scalability - Force sensing can be applicable from a minute small object sensing with 0.1 gram to a room-scale large body motion with 100 kilogram weight change.

- Expressibility - Force vector conveys the spatial-temporal information of human body part movement. Through the detailed analysis of the force signals, I revealed a variety of the human activity can be reconstructed from the data.

- Reconfigurability - Force preservation is the one of the fundamental law of universe according to the basic physics. Force doesn't disappear anywhere. This characteristics allow us to place the sensors onto / into the target object with a broad range of attachment position.

6.2 Future of Force-based Interaction

I'm writing this concluding section to note the broaden perspectives with the foreseeable future of my overarching concept, Force-based Interaction, with a strong confidence underpinned on top of the intensive two-years investigation.

We human being with bright mind and open sight to see and understand the world mostly through the channel of eyes, however, at the same time, we are the being with physical bodies that feel and experience the world through the body, which can be behaved as both of an actuator and a sensor for force signals.

Photons, which is emitted from the "Pixel Empire" or just bounced from a surface of objects, are literally 'light', actually assumed to have no mass in theory, so that it is thought to be the quickest way to convey the summarized information to the eyes. However our body and way of life also generate Forces, which is the another broad channel of information, as revealed in every chapter of this thesis.

On top of the expectation, we have schemed Force-based Interaction. I hope that this fundamental insight could be contributing to establish the next grand vision towards enhancing Human Object Interaction. This perspective of Force-based Interaction could be the great complement of GUI, VR and AR and any other technologies from the Pixel Empire. I know that this is not purely my invention, from the experience of extensive survey for past researches, however, I believe this thesis work could assist the circulation, distribution and communication of the idea, insight and perspective of Force-based Interaction to the rest of the world.

In conclusion, I envision the world, covering everyday life in our residence to an architectural space, augmented with unobtrusive force sensors to capture the human intention through the subtle features hidden and embedded in force signals.

Bibliography

[1] 3Dconnexion. Spacenavigator official website. https://www.3dconnexion.com/spacemouse$_{c}ompact/en/$. $Accessed$: $2019 - 08 - 10$.

[2] T. Achituv. Hx711 simultaneous read library. https://github.com/compugician/HX711-multi, 2016.

[3] Amazon.com. Introducing amazon go and the worldâĂŹs most advanced shopping technology. (2016), 2016.

[4] Mihai Andries. *Object and human tracking, and robot control through a load sensing floor.* PhD thesis, Université de Lorraine, 2015.

[5] Daniel Avrahami, Mitesh Patel, Yusuke Yamaura, and Sven Kratz. Below the surface: Unobtrusive activity recognition for work surfaces using rf-radar sensing. In *23rd International Conference on Intelligent User Interfaces*, pages 439–451. ACM, 2018.

[6] Ravin Balakrishnan, George W Fitzmaurice, and Gordon Kurtenbach. User interfaces for volumetric displays. *Computer*, 34(3):37–45, 2001.

[7] Patrick Baudisch, Torsten Becker, and Frederik Rudeck. Lumino: tangible blocks for tabletop computers based on glass fiber bundles. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1165–1174. ACM, 2010.

[8] Jeffrey Traer Bernstein, Avi Cieplinski, Brett W Degner, Duncan Kerr, Patrick Kessler, Paul Puskarich, Marcelo H Coelho, and Aleksandar Pance. Touch pad with force sensors and actuator feedback, March 1 2016. US Patent 9,274,660.

[9] Dunn Brian. *Load Cells Introduction and Applications.* BD Tech Concepts LLC, 2016.

[10] Andrew Carvey, Jim Gouldstone, Pallavi Vedurumudi, Adam Whiton, and Hiroshi Ishii. Rubber shark as user interface. In *CHI '06 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '06, pages 634–639, New York, NY, USA, 2006. ACM.

[11] Alvaro Cassinelli and Masatoshi Ishikawa. Volume slicing display. *SIGGRAPH ASIA Art Gallery & Emerging Technologies*, 88, 2009.

[12] D. Antony Chacon, Eduardo Velloso, Thuong Hoang, and Katrin Wolf. Spinallog: Visuo-haptic feedback in musculoskeletal manipulation training. In *Proceedings of the Thirteenth International Conference on Tangible, Embedded, and Embodied Interaction*, TEI '19, pages 5–14, New York, NY, USA, 2019. ACM.

[13] Liwei Chan, Stefanie Müller, Anne Roudaut, and Patrick Baudisch. Capstones and zebrawidgets: sensing stacks of building blocks, dials and sliders on capacitive touch screens. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2189–2192. ACM, 2012.

[14] Gregg E Favalora. Volumetric 3d displays and application infrastructure. *Computer*, 38(8):37–44, 2005.

[15] Gustav Theodor Fechner, Davis H Howes, and Edwin Garrigues Boring. *Elements of psychophysics*, volume 1. Holt, Rinehart and Winston New York, 1966.

[16] World Economic Forum. Shaping the future of retail for consumer industries, January 2017.

[17] Gustav Gautschi. Piezoelectric sensors. In *Piezoelectric Sensorics*, pages 73–91. Springer, 2002.

[18] Daniel J. Goble, Brian L. Cone, and Brett W. Fling. Using the wii fit as a tool for balance assessment and neurorehabilitation: the first half decade of "wii-search". *Journal of NeuroEngineering and Rehabilitation*, 11(1):12, Feb 2014.

[19] Daniel Hayim Goodman. *Aware surfaces: large-scale, surface-based sensing for new modes of data collection, analysis, and human interaction.* PhD thesis, Massachusetts Institute of Technology, 2015.

[20] Tovi Grossman and Ravin Balakrishnan. The design and evaluation of selection techniques for 3d volumetric displays. In *Proceedings of the 19th annual ACM symposium on User interface software and technology*, pages 3–12. ACM, 2006.

[21] Tovi Grossman, Daniel Wigdor, and Ravin Balakrishnan. Multi-finger gestural interaction with 3d volumetric displays. In *Proceedings of the 17th annual ACM symposium on User interface software and technology*, pages 61–70. ACM, 2004.

[22] Jefferson Y. Han. Low-cost multi-touch sensing through frustrated total internal reflection. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology*, UIST '05, pages 115–118, New York, NY, USA, 2005. ACM.

[23] Colin Honigman, Jordan Hochenbaum, and Ajay Kapur. Techniques in swept frequency capacitive sensing: An open source approach. In *NIME*, pages 74–77, 2014.

[24] Charles Hudin, Sabrina Panëels, and Steven Strachan. Intact: Instant interaction with 3d printed objects. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pages 2719–2725. ACM, 2016.

[25] Sam K Hui, Yanliu Huang, Jacob Suher, and J Jeffrey Inman. Deconstructing the âĂIJfirst moment of truthâĂİ: Understanding unplanned consideration and purchase conversion using in-store video tracking. *Journal of Marketing Research*, 50(4):445–462, 2013.

[26] Unity inc. Unity official website. https://unity.com. Accessed: 2019-08-10.

[27] Hiroshi Ishii, HR Fletcher, J Lee, S Choo, J Berzowska, C Wisneski, C Cano, A Hernandez, and C Bulthaup. Musicbottles. In *ACM SIGGRAPH 99 Conference abstracts and applications*, page 174. ACM, 1999.

[28] Hiroshi Ishii and Brygg Ullmer. Tangible bits: towards seamless interfaces between people, bits and atoms. In *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems*, pages 234–241. ACM, 1997.

[29] Hiroshi Ishii, Craig Wisneski, Julian Orbanes, Ben Chun, and Joe Paradiso. Pingpongplus: design of an athletic-tangible interface for computer-supported cooperative play. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 394–401. ACM, 1999.

[30] Andrew Jones, Ian McDowall, Hideshi Yamada, Mark Bolas, and Paul Debevec. Rendering for an interactive 360 light field display. *ACM Transactions on Graphics (TOG)*, 26(3):40, 2007.

[31] Sergi Jordà, Günter Geiger, Marcos Alonso, and Martin Kaltenbrunner. The reactable: exploring the synergy between live music performance and tabletop tangible interfaces. In *Proceedings of the 1st international conference on Tangible and embedded interaction*, pages 139–146. ACM, 2007.

[32] Yasuaki Kakehi, Kensei Jo, Katsunori Sato, Kouta Minamizawa, Hideaki Nii, Naoki Kawakami, Takeshi Naemura, and Susumu Tachi. Forcetile: tabletop tangible interface with vision-based force distribution sensing. In *ACM SIGGRAPH 2008 new tech demos*, page 17. ACM, 2008.

[33] Hirokazu Kato and Mark Billinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *Augmented Reality, 1999.(IWAR'99) Proceedings. 2nd IEEE and ACM International Workshop on*, pages 85–94. IEEE, 1999.

[34] Hong-Ki Kim, Seunggun Lee, and Kwang-Seok Yun. Capacitive tactile sensor array for touch screen application. *Sensors and Actuators A: Physical*, 165(1):2–7, 2011.

[35] Sven Kratz, Tilo Westermann, Michael Rohs, and Georg Essl. Capwidgets: tangile widgets versus multi-touch controls on mobile devices. In *CHI'11 Extended Abstracts on Human Factors in Computing Systems*, pages 1351–1356. ACM, 2011.

[36] Knut Langhans, Detlef Bahr, Daniel Bezecny, Dennis Homann, Klaas Oltmann, Krischan Oltmann, Christian Guill, Elisabeth Rieper, and Götz Ardey. Felix 3d display: an interactive tool for volumetric imaging. In *Stereoscopic Displays and Virtual Reality Systems IX*, volume 4660, pages 176–190. International Society for Optics and Photonics, 2002.

[37] Jinha Lee and Hiroshi Ishii. Beyond: collapsible tools and gestures for computational design. In *CHI'10 Extended Abstracts on Human Factors in Computing Systems*, pages 3931–3936. ACM, 2010.

[38] M.H Lee and H.R Nicholls. Review article tactile sensing for mechatronics âĂŤ a state of the art survey. *Mechatronics*, 9(1):1 – 31, 1999.

[39] Rong-Hao Liang, Liwei Chan, Hung-Yu Tseng, Han-Chih Kuo, Da-Yuan Huang, De-Nian Yang, and Bing-Yu Chen. Gaussbricks: magnetic building blocks for constructive tangible interactions on portable displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 3153–3162. ACM, 2014.

[40] Rong-Hao Liang, Kai-Yin Cheng, Liwei Chan, Chuan-Xhyuan Peng, Mike Y Chen, Rung-Huei Liang, De-Nian Yang, and Bing-Yu Chen. Gaussbits: magnetic tangible bits for portable and occlusion-free near-surface interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1391–1400. ACM, 2013.

[41] Rong-Hao Liang, Kai-Yin Cheng, Chao-Huai Su, Chien-Ting Weng, Bing-Yu Chen, and De-Nian Yang. Gausssense: attachable stylus sensing using magnetic sensor grid. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*, pages 319–326. ACM, 2012.

[42] Rong-Hao Liang, Han-Chih Kuo, Liwei Chan, De-Nian Yang, and Bing-Yu Chen. Gaussstones: shielded magnetic tangibles for multi-token interactions on portable displays. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*, pages 365–372. ACM, 2014.

[43] Jie Liu. Autonomous retailing: A frontier for cyber-physical-human systems. In *Principles of Modeling*, pages 336–350. Springer, 2018.

[44] Mark Mine. Isaac: A virtual environment tool for the interactive construction of virtual worlds. *UNC Chapel Hill Computer Science Technical Report TR95-020*, 1995.

[45] Leo Miyashita, Ryota Yonezawa, Yoshihiro Watanabe, and Masatoshi Ishikawa. 3d motion sensing of any object without prior knowledge. *ACM Trans. Graph.*, 34(6):218:1–218:11, October 2015.

[46] I. Muller, R. M. de Brito, C. E. Pereira, and V. Brusamarello. Load cells in force sensing analysis – theory and a novel application. *IEEE Instrumentation Measurement Magazine*, 13(1):15–19, February 2010.

[47] Kazuya Murao, Junna Imai, Tsutomu Terada, and Masahiko Tsukamoto. Recognizing activities and identifying users based on tabletop activities with load cells. In *Proceedings of the 17th International Conference on Information Integration and Web-based Applications & Services*, page 39. ACM, 2015.

[48] Ken Nakagaki, Daniel Fitzgerald, Zhiyao John Ma, Luke Vink, Daniel Levine, and Hiroshi Ishii. inforce: Bi-directionalforce'shape display for haptic interaction. In *Proceedings of the Thirteenth International Conference on Tangible, Embedded, and Embodied Interaction*, pages 615–623. ACM, 2019.

[49] Tomoharu Nakamura, Tomoya Yano, Kohki Watanabe, Yui Ishii, Hideki Ono, Ippei Tambata, Nobuki Furue, and Yuji Nakahata. 360degreee transparent holographic screen display. In *ACM SIGGRAPH 2019 Emerging Technologies*, SIGGRAPH '19, pages 1:1–1:2, New York, NY, USA, 2019. ACM.

[50] United State Department of Agriculture. Farmers markets and direct-to-consumer marketing, 2017.

[51] Makoto Ono, Buntarou Shizuki, and Jiro Tanaka. Touch & activate: adding interactivity to existing objects using active acoustic sensing. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*, pages 31–40. ACM, 2013.

[52] Leonel Paredes-Madrid, Arnaldo Matute, Jorge Bareño, Carlos Parra Vargas, and Elkin Gutierrez Velásquez. Underlying physics of conductive polymer composites and force sensing resistors (fsrs). a study on creep response and dynamic loading. *Materials*, 10(11):1334, 2017.

[53] Voxon Photonics. Voxon vx1 official website. https://voxon.co. Accessed: 2019-08-10.

[54] Mirela Popa, Alper Kemal Koc, Leon JM Rothkrantz, Caifeng Shan, and Pascal Wiggers. Kinect sensing of shopping related actions. In *International Joint Conference on Ambient Intelligence*, pages 91–100. Springer, 2011.

[55] Mirela C Popa, Léon JM Rothkrantz, Caifeng Shan, Tommaso Gritti, and Pascal Wiggers. Semantic assessment of shopping behavior using trajectories, shopping related actions, and context information. *Pattern Recognition Letters*, 34(7):809–819, 2013.

[56] Jun Rekimoto. Smartskin: an infrastructure for freehand manipulation on interactive surfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 113–120. ACM, 2002.

[57] Roy E Ritzmann and Sasha N Zill. Control of locomotion in hexapods. In *The Oxford Handbook of Invertebrate Neurobiology*. 2017.

[58] Munehiko Sato, Ivan Poupyrev, and Chris Harrison. Touché: Enhancing touch interaction on humans, screens, liquids, and everyday objects. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 483–492, New York, NY, USA, 2012. ACM.

[59] Albrecht Schmidt, Martin Strohbach, Kristof Van Laerhoven, Adrian Friday, and Hans-Werner Gellersen. Context acquisition based on load sensing. In *International Conference on Ubiquitous Computing*, pages 333–350. Springer, 2002.

[60] Albrecht Schmidt, Martin Strohbach, Kristof Van Laerhoven, and Gellersen Hans-W. Ubiquitous interaction using surfaces in everyday environments as pointing devices. In *ERCIM Workshop on User Interfaces for All*, pages 263–279. Springer, 2002.

[61] Martin Schmitz, Jürgen Steimle, Jochen Huber, Niloofar Dezfuli, and Max Mühlhäuser. Flexibles: Deformation-aware 3d-printed tangibles for capacitive touchscreens. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 1001–1014, New York, NY, USA, 2017. ACM.

[62] Longfei Shangguan, Zimu Zhou, and Kyle Jamieson. Enabling gesture-based interactions with objects. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '17, pages 239–251, New York, NY, USA, 2017. ACM.

[63] Longfei Shangguan, Zimu Zhou, Xiaolong Zheng, Lei Yang, Yunhao Liu, and Jinsong Han. Shopminer: Mining customer shopping behavior in physical clothing stores with cots rfid devices. In *Proceedings of the 13th ACM conference on embedded networked sensor systems*, pages 113–125. ACM, 2015.

[64] Amit Sharma and Youngki Lee. Audiosense: sound-based shopper behavior analysis system. In *Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers*, pages 488–493. ACM, 2017.

[65] Daniel Smalley, Ting-Chung Poon, Hongyue Gao, Joshua Kvavle, and Kamran Qaderi. Volumetric displays: Turning 3-d inside-out. *Optics and Photonics News*, 29(6):26–33, 2018.

[66] Julien Tripette, Haruka Murakami, Katie Rose Ryan, Yuji Ohta, and Motohiko Miyachi. The contribution of nintendo *Wii Fit* series in the field of health: a systematic review and meta-analysis. *PeerJ*, 5:e3600, September 2017.

[67] Geoff Walker. A review of technologies for sensing contact location on the surface of a display. *Journal of the Society for Information Display*, 20(8):413–440.

[68] Yuzhang Wei and Qingsong Xu. An overview of micro-force sensing techniques. *Sensors and Actuators A: Physical*, 234:359 – 374, 2015.

[69] Takatoshi Yoshida, Xiaoyan Shen, Tal Achituv, and Hiroshi Ishii. 3d touch point detection on load sensitive surface based on continuous fluctuation of a user hand. In *SIGGRAPH Asia 2018 Posters*, SA '18, pages 39:1–39:2, New York, NY, USA, 2018. ACM.

[70] Takatoshi Yoshida, Shen Xiaoyan, Yoshino Koichi, Nakagaki Ken, and Ishii Hiroshi. Scale: Enhancing force-based interaction by processing load data from load sensitive modules. In *Proceedings of the 32th Annual Symposium on User Interface Software and Technology*, pages XX–XX. ACM, 2019.

[71] Xiaoling Zhang, Shibo Li, Raymond R Burke, and Alex Leykin. An examination of social influence on shopper behavior using video tracking data. *Journal of Marketing*, 78(5):24–41, 2014.

[72] Yang Zhang, Chouchang Yang, Scott E. Hudson, Chris Harrison, and Alanson Sample. Wall++: Room-scale interactive and context-aware sensing. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*, page 273, 2018. Exported from https://app.dimensions.ai on 2019/04/04.

[73] Zimu Zhou, Longfei Shangguan, Xiaolong Zheng, Lei Yang, and Yunhao Liu. Design and implementation of an rfid-based customer shopping behavior mining system. *IEEE/ACM Transactions on Networking*, 25(4):2405–2418, 2017.

[74] Sasha Zill, Josef Schmitz, and Ansgar BÃijschges. Load sensing and control of posture and locomotion. *Arthropod Structure Development*, 33(3):273 – 286, 2004. Arthropod Locomotion Systems: from Biological Materials and Systems to Robotics.