

國立臺灣大學電機資訊學院資訊工程學系
博士論文

Department of Computer Science and Information Engineering
College of Electrical Engineering and Computer Science
National Taiwan University
Doctoral Dissertation

具彈性及可擴充性之穿戴式平台
A Flexible and Extensible Wearable Platform

李爭原

Cheng-Yuan Li

指導教授：朱浩華博士
Advisor: Hao-Hua Chu, Ph.D.

中華民國 105 年 12 月
December, 2016

Abstract

The wearable devices for healthcare applications are a rapid growth market in recent years and are expected to further grow in the near future. The researchers also present many emerging healthcare projects with wearable device from a variety of disciplines. However, development of a wearable device needs much iteration to go through the steps between ideate, prototype and test. Different healthcare applications may need to sense diverse types of physiological data and/or activities and use different form factors. The iterative design process may take a lot of time, cost and effort to advance the prototype from low-fidelity to high-fidelity in order to achieve a right design. Prototyping also requires a lot of engineering skills to implement a device to validate the concept and hypothesis of the design. To address these challenges, this thesis proposes a wearable sensing platform that allows developers to rapidly prototype their health applications. We first consider the users of our wearable platform are researchers and designers who may not have sufficient engineering skills to build a prototype to test their design in developing phase. Toward an easy-to-use wearable platform for those users, we first conduct focus group and participatory design workshops to explore the needs from the users with medical background. We therefore design a wearable platform based on the key findings summarized from the design exploration studies, called the BioScope, which is a wearable sensing platform with flexibility and extensibility. Each sensor is one single patch in the

platform. The flexibility enables users to choose the needed sensor patches and assemble them together like building LEGO bricks. The platform is extensible which can simplify the process to extend new component without re-design entire system. The platform provides several basic sensor patches that can easily collect physiological data from human body. An interaction patch has a compact display to quickly feedback sensory data and a gesture input sensor to simply control the wearable device. Bandage-like is the basic form factor to simply affix the device on arbitrary body location like a bandage. The platform has a library that contains the basic sensing and feedback functions of each patches that user can build their application that just simply select required sensing patches and corresponding feedback function. We also provide APIs on both wearable device and mobile device to serve advanced developer who attempts to construct customized function on existing patches.

Contents

Abstract	ii
1 Introduction	1
1.1 Motivation	1
1.2 Contribution	1
1.3 Organization	1
2 Background and Related Work	2
2.1 Lesson Learned from Case Studies of Developing Health Applications	2
2.2 Wearable Devices with Healthcare Applications	3
2.3 Rapid Prototyping for Wearable Applications	4
2.3.1 Integrated Design Devices	4
2.3.2 Prototyping platforms	5
2.3.3 Customization in HCI	7
2.3.4 Modular Design: Flexible and Extensible	8
2.4 Design Exploration	9
2.4.1 Focus Groups	9
2.4.2 Participatory Design Workshops	10
2.4.3 Findings form Explorative Studies	12
3 BioScope: a Flexible and Extensible Wearable Platform	15
3.1 Design Exploration	15
3.1.1 Focus Groups	16

3.1.2	Participatory Design Workshops	17
3.1.3	Findings from Explorative Studies	19
3.2	BioScope System Design and Implementation	21
3.2.1	Flexible and Extensible Sensing Bandage	22
3.2.2	Stacking Mechanism	24
3.2.3	Patches	27
3.2.4	Explorative Experiment	30
3.3	Visualization of Sensory Data	33
3.3.1	Configuration	33
3.3.2	Data Processing and Representation	35
3.3.3	User Interface on Mobile Device	36
3.4	Platform APIs	37
3.4.1	Device APIs	37
3.4.2	Phone APIs	40
3.5	Summary	42
4	Tangram: Simplify PCB Design Process with Modularization	43
4.1	Design considerations	44
4.1.1	Prototyping with off-the-shelf developing toolkit	45
4.1.2	Fabricate customized PCB	46
4.2	System design	47
4.2.1	TangramScope	48
4.2.2	TangramFab	49
4.3	Explorative study	51
4.4	Discussion	52
4.4.1	Flexibility and Difficulty in PCB Design	52
4.4.2	Developing a Well-provisioned Library	53
4.4.3	Limitation and Future Work	53
4.5	Summary	53

5 Portfolio	55
5.1 Sensor-Embedded Teeth for Oral Activity Recognition	55
5.1.1 System Overview	56
5.1.2 Experimental Evaluation	59
5.1.3 Discussion	60
5.1.4 Summary	62
5.2 Sensing Fork: Persuasive Technology to Improve Eating Behavior using a Sensor-Embedded Fork	63
5.2.1 System Overview	64
5.2.2 Prototype Design: Sensing Fork	65
5.2.3 Persuasive Game Application: Hungry Panda	70
5.2.4 Experimental Evaluation	72
5.2.5 Limitations	75
5.2.6 Summary	76
5.3 Enabling Daily Self-monitoring to Assist Recovery from Drug Addiction through a Phone-based Support System	77
5.3.1 System Overview	79
5.3.2 Saliva Screening Design	81
5.3.3 User Interface	89
5.3.4 Summary	91
6 Conclusion	92
Bibliography	93

List of Figures

3.1	BioScope: a flexible and extensible wearable sensing platform.	16
3.2	Design of the extensible sensing bandage. (a) Four patches with distinctive embossed icons are stacked on the inner and contact layers according to the direction indicated by the red dotted arrows. (b) The sound-collecting structure (box with red dashed border), a thermocouple wire (box with blue solid border), and two electrodes coated with a conductive gel (two green circles) directly contact the skin.	22
3.3	Stacking mechanism to connect sensor patches and mainboard.	24
3.4	Four steps for applying bandages.	25
3.5	An example of using BioScope to prototype a breathalyzer.	26
3.6	Interaction patch includes OLED display and on-the-air gesture sensor .	29
3.7	Recharging patch.	30
3.8	Four steps for applying bandages.	31
3.9	The steps of configuration for data visualization. (a) System off. (b) Assembly: stacking patches and casing. (c) display on interaction patch. (d) display on mobile phone.	33
3.10	Representations of data visualization on wearable display: digit, pattern, icon.	35
3.11	Extract the initialization and loop sections from conventional structure of microcontroller codes	37
3.12	An example of BioScope device APIs for reading temperature sensor and transmitting sensory data through Bluetooth radio	38

4.1	Flowchart of the Tangram.	47
4.2	(a) Tans of TangramScope. (b) Stacking mechanism.	48
4.3	Snapshot of TangramFab.	50
4.4	Tans imported into TangramFab from TangramScope.	50
4.5	Situate Tans to desired positions and draw the dimension.	50
4.6	(a) The toy doll embedded electronic device. (b) The customized PCB. . .	51
5.1	The breakout board with (b) tri-axial accelerometer and (a)(c) sensor embedded denture.	56
5.2	System configuration. Left: The Sensing Fork prototype. Right: The software on the smartphone.	65
5.3	Electrodes and resistance sensing schematic. Marker 1 and 2 represent two fork's tines. Marker 3 is the fork's grip.	66
5.4	State transition diagram of eating actions.	67
5.5	The story and flow of Hungry Panda game application.	69
5.6	The examples of food items in three cuisines.	74
5.7	Proposed system for daily self-monitoring of drug use, comprising saliva screening device and mobile smartphone.	78
5.8	The system architecture.	79
5.9	The reaction zone of a strip and its potential line patterns. "C" ("T") means the control (test) line of the pattern.	80
5.10	Electrical components on the circuit boards and connections between boards. .	81
5.11	Image of reaction zone illuminated using monochromatic LEDs: (a) green, (b) blue, (c) red, and (d) white. The RGB readings across the middle row of pixels in the image are plotted as a line chart below each image. . . .	83
5.12	Auxiliary board with test strip and three equivalent circuits. When the strip is dry, the equivalent circuit (b). Saliva samples are absorbed by the front cotton wick resulting in equivalent circuit (c). When the back cotton wick is also soaked, the equivalent circuit is in (d).	84
5.13	Voltage samples of determining saliva test progress.	86

5.14 The row-by-row intensity analysis while validating the control line and classifying the test line.	87
5.15 User interface of smartphone application: (a) saliva screening, (b) screening results, (c) event logging and (4) ranking of risky scenario.	90
5.16 F-measure accuracy of three learning classification algorithms (SVM, RF, and LR).	91

List of Tables

5.1	Adopted features for oral activity recognition.	59
5.2	F-measure accuracy of oral activity recognition with a person-dependent classifier.	60
5.3	F-measure accuracy of oral activity recognition with a person-independent classifier.	61
5.4	The accuracy of detecting each eating action. The numbers in parentheses denote the average time (sec.) required to detect each eating action. . . .	73
5.5	The confusion matrix of classifying food types. (1)Shiitake Mushroom. (2)Eryngii Mushroom.	73

Chapter 1

Introduction

1.1 Motivation

...

1.2 Contribution

This thesis makes the following contributions:

- ...

1.3 Organization

The remainder of this thesis is organized as follows. Chapter 2 presents ...

Chapter 2

Background and Related Work

This chapter starts by first surveying related wearable devices for healthcare applications that are available on the market today. We describe technical challenges that may be faced by developers in developing and prototyping wearable devices. Second, we describe our own case study of building a wearable device from scratch without using any platform/tool support. We detail the amount of development effort and skill needed to motivate the need for wearable platform & tool support. Third, we present related application development platforms and tools in which designers and developers can use to build their devices & applications. We describe how BioScope addresses the issues of flexibility and extensibility lacking from these related platforms & tools.

2.1 Lesson Learned from Case Studies of Developing Health Applications

This section discusses our lesson learned from prior case studies that we experienced the design and implementation of a wearable oral sensory system that recognizes human oral activities, such as chewing, drinking, speaking, and coughing from scratch without using any platform/tool support.

In order to provide a easy to use system, it requires a lot of effort and time to go through many iterations in refining modality, trouble-shooting and usability test. In addition to

the cost in building such system, developers also require significant engineering skills. However, some designers and/or developers may not be able to deal with those processes as well. Therefore, we consider that designers and/or developers should need a way to mitigate the effort and cost in the design process for building their system.

2.2 Wearable Devices with Healthcare Applications

Recently, there have been many commercial products leveraging mobile sensing and wearable technologies to promote health in people's daily life. As smartphones become ubiquitous in people's daily life, people have explored the use of smartphones together with their built-in sensors to monitor human health condition, such as basic vital-signs. For example, Instant Heart Rate [14] is a phone application that leverages the smartphone's built-in camera and flashlight to sense a person's heart rate. Strava [28] is a smartphone application that uses the smartphone's GPS sensor to quantify human performance of running and cycling exercises. When built-in sensors are insufficient, smartphones can also connect with external sensors, such as wearable heart rate sensors and/or cycling sensors, to collect more detailed information for further analysis of the exercise events.

Wearable devices are not only an extension of a smartphone. They can also work as a standalone device. Being wearable, it reduces people's burden of carrying additional devices. In recent years, a variety of wearable products have successfully immersed in people's daily life. Wristband is the most common form factor in wearable devices. For example, the Fitbit [11] is a popular wristband device that enables people to track and monitor their physical activities. The Apple watch [4] and the Android wear [3] provide wearable computing platforms to build wearable healthcare applications in the wristband form factor. Other non-wristband wearable form factors are also emerging. For examples, the Under Armour [29] releases a sensor-embedded running shoe that can track and collect running metrics. The Athos [7] is a smart apparel that monitors muscle activities and heart rate to assist training and exercise. X2 xGuard [35, 44] is a product that embeds sensors inside a mouth guard for tracking athletes' accumulative head impacts in contact sports, e.g.: football. The forecast for wearable devices worldwide from Gartner [58] shows that the

market of wearable healthcare devices will experience rapid growth in the near future. Since wearable devices are deployed and worn on humans constantly, usability plays a particularly important factor to successful people's acceptance and adoption of wearable technologies. Reaching usability goal is costly, requiring designers and/or developers to perform many iterations of prototyping, testing, analyzing, and refining to uncover and fix problems. Such development and prototyping effort can be reduced with platform support.

2.3 Rapid Prototyping for Wearable Applications

The iterative design [80, 85, 87] between ideate, prototype and test is the most costly part in the entire design process. The key challenge here is how to rapid prototyping in iterative design. In addition to product developers, academic researchers also have a great demand of seeking an efficient way to speed up the prototyping process. For achieving the design goal, several options can be adopted in different iteration in the design process. At the beginning of the iterative design, a low-fidelity prototype [89] is the ideal tool to rapidly examine the feasibility, check the usability and refine the ideal with minimum cost of building prototype. However, low-fidelity prototype may not be able to test the functionality on certain aspects. For example, we attempted to develop a wearable oral sensory system that recognizes human oral activities related to health [69]. For examining its feasibility, it requires building a wearable device integrated with the sensor. Therefore, people always demand a useful tool that contains electronic and/or mechanical components in the design process. In the subsequent subsections, we describe the related works of rapid prototyping for wearable applications.

2.3.1 Integrated Design Devices

Developers, designer and researchers would like to develop a wearable application with uncomplicated process and without significant engineering skill to work with it, such as wiring, soldering, coding and so on. For non-engineering background users, an off-the-shelf device would be a better choice. For examples, activPAL [1] equips motion sensor

and storage for collecting human's physical activity. Shimmer [26] is a device that integrates various types of sensors, storage, wireless connectivity and software for accessing the device. The Mercury [71] is an example that uses Shimmer in their study for high-fidelity motion analysis of patients being treated for neuromotor disorders. The Health-Patch [33] is a bandage-like wearable device that embedded ECG sensor, accelerometer and temperature sensor to track human's health. Smart phone is also a versatile device that integrated many sensors. Eric C. Larson et.al. [67] leverage the microphone on the smart phone carried in shirt pocket or using a neck strap for preserving the cough activity. But a fixed design can only offer limited functionalities and meet the needs of limited applications. Once the user need an extra function or different form factor, it can not satisfy the need of developing new application and service. Thus, the people may be willing to pay more cost and effort for better prototyping their design.

2.3.2 Prototyping platforms

Several tools can help people to easily prototype a device for developing and testing a design. For example, the Arduino [5] is an open-source platform that provides easy-to-use hardware and software for making electronic device. It also has the version, e.g.: Arduino Gemma, that targets the user for developing wearable application. Arduino is a successful platform that you can easily get many resources to prototype your idea including plenty of compatible sensors, actuators, feedback components and software libraries. The Intel Edison [15] presents a development kit that offers high computational performance and wireless connectivities including Wi-Fi and Bluetooth. Through a breakout board, it can connect other components as well as the Arduino. The Xadow [25] is another platform which is similar to Arduino. It adopts a well-defined connector and cable to simplify the process of connecting the components. Thus, the user is not necessary to worry about the wiring for connecting components in pin-to-pin fashion and the robustness of those wires. The Arduino provides not only the hardware boards in its platform but also the IDE and many software libraries to support writing codes and uploading binary codes to the board. The Processing [21] is an IDE to program the codes running on the PC side

for communicating with a device, processing data and building 2D/3D visualization. The Arduino IDE and Processing are both open-source. Therefore, there have many third-party libraries which users can leverage in their development. The microcontroller adopted by the Arduino has only limited computational capability and types of IO interface. The Arm mbed [6] is an web-based IDE that can support for programming different devices from different manufacturers which adopt the microconotrlers with the Arm Cortex-M architecture.

Several works focus on simplifying software development process of building application. Many applications rely on machine learning algorithms to construct their functions. The Weka [34] from University of Waikato integrates several popular machine learning algorithms that can be used in prototyping phase. But such type of tool still requires a lot of basic knowledge of data processing and machine learning. Several researchers present the mechanism to reduce the threshold for first-time users. For example, the Jigsaw [72] proposes a system for simplifying the effort of building the applications that require continuous sensing on a mobile phone. The users can use this platform to build an application that need to recognize every day activities using a mobile phone with a proper balance between performance and resource demand. Choudhury et. al. [49] present a wearable sensing platform designed for constructing activity-aware system that can support ubiquitous computing applications. The wearable sensing platform implements a wearable device that equips Bluetooth radio, flash storage, microphone, light sensor, accelerometer, digital compass, barometer and humidity/temperature sensor. This study also is involved in developing a software framework running on their device for data logging, feature processing and activity recognition. The d.tool [61] demonstrates a platform that facilitates the iterative process between design, test and analysis for prototyping information appliances with low threshold. A plug-and-play hardware board allows prototyping device by choosing needed components including button input, accelerometer, digital compass, LED, LCD screen, speaker and so on. An IDE on the PC supports the iterative process for programming prototype, testing builded functions and reviewing the data from testing. Since many applications on wearable devices need to work with a mobile phone

together. Chi et. al. presents the Weave [48] that provides a web-base IDE with library to enable programming and testing for constructing behavior interactions and UI layouts across wearable and mobile devices.

However, those platforms may not be friendly to the users who are not engineering background. They still need to pay a great effort to learn how to coding and prototyping using those platforms. After a prototype device is constructed and programed, its wiring mechanism is not robust in connecting component or its form factor is not suitable in wearable while use in testing. Then troubleshooting would be a considerable cost in iterative design process. When they leap over the prototyping phase and advance to a finished device. The effort in prototyping phase is wasted due to the constructed device and the codes can not be reuse in their finished device that should have further considerations, e.g.: compact and power saving.

2.3.3 Customization in HCI

After several iterations of prototyping a system of the wearable application in low-fidelity and medium-fidelity prototypes, the next step is to build a high-fidelity prototype for conducting a field trial and further refine the design. In product development, the implementation of the wearable system will involve with a great cost and effort in customization before launching the application and service to the market. In some cases, low-fidelity prototype may not be able to satisfy the requirement of design goal and validate the design hypothesis. Therefore, many HCI studies customize their wearable system in order to have a workable and robust device used in field trial or to better demonstrate their design and concept. For examples, the mPuff [37] introduces a chest-worn device for detecting smoking behavior using motion, heart rate, respiration and galvanic skin response sensors. The MARS [75] customizes wearable devices with inertial sensors to recognize whole body movement. However, a number of studies for wearable application [37, 66, 75, 84, 74, 71, 90] use common components in their customized device. The effort, cost and time to revise the prototype in customization will further increase with more iterations they go through.

2.3.4 Modular Design: Flexible and Extensible

Modular design is a design concept that separates each functional partition of the system as a module and standardizes the communication interface between modules to enable flexibility and extensibility. Each module can be reuse to reduce the cost in customization. A modularized system can easily add new functionality without re-design entire system to shorten developing process. Google Ara [12] and LG G5 [16] are flexible and extensible smart phones that allow users to make the choice of modules for building a smart phone as their need. The modular design also enables easily to extend new module, update existing module and replace broken module. The Nascent [19] allows users to build a ready-to-use electronic device using modular components such as speaker, inertial sensor and microphone, which has a similar quality to a commercial product.

Recently, modular design also emerged in wearable area. Similar to modular smart phones, the BLOCKS [8] is a modular smartwatch that lets users to tailor the features as they needed. It contains several modules that can function on the wrist, such as temperature sensor, ECG, GPS, camera, NFC, gesture control and so on. The Vigekwear [32] is a developing kit that offers micro and modularized components including BLE microcontroller, inertial sensors, temperature sensor, barometer, heart rate sensor and OLED display. This developing kit aims for allowing users to develop an application on the wrist.

The concept has been widely explored in product manufacturing to discuss the tradeoff between modular design integrated design[82, 86]. In addition to making a modular design in manufacturing phase, i.e., each component is modularized physically, modularization also can be done in design phase. For example, the Altium Designer [2] is a PCB design software that can encapsulate a part of circuit as a module. Thus, the designer can rearrange the module to create different configurations and variants. However, there still is a great effort to wire each module in pin-to-pin fashion.

Those examples show that the modular design can be a useful concept to benefit rapid prototyping for developing wearable applications. In our BioScope platform, we leverage the concept of the modular design to enable flexibility and extensibility that can assist researcher, designer and developer for rapid prototyping their health applications. How-

ever, the design of the wearable platform has a lot of challenges that the platform can meet the needs of our users who may not have engineering background and/or related training. In the next section, we will introduce the design exploration that explore the design criteria of prototyping wearable for health applications through two focus groups and two participatory design workshops.

2.4 Design Exploration

The first step of iterative design for a wearable sensing platform is to understand the needs of potential users. We conducted two focus groups and two participatory design workshops to explore the needs from healthcare professionals.

2.4.1 Focus Groups

We conducted two focus groups to better understand the practical needs of wearable technology in clinical setting. In the first focus group, we recruited four nursing professionals and two physical therapists from National Taiwan University Hospital. In the second focus group, we recruited four nursing professionals from Taipei City Psychiatric Center (TCPC).

Participants

In the first focus group, we recruited four healthcare workers (four experienced nursing experts and two physiotherapists), aged from 27 to 46 years. The four experienced nursing experts had worked as registered nurses for more than three years and the two physiotherapists have worked in the Rehabilitation Department of National Taiwan University Hospital at least six years. These participants have prior experienced in oncology, cardiology, and orthopedics. Four experts in the second focus group, aged from 35 to 45 years, have worked as registered nurses for more than ten years at the Taipei City Psychiatric Center (TCPC) and specialized in providing inpatient psychiatric care for children, adolescents, geriatrics, and patients with substance abuse.

Procedures

The moderator of the focus group was a researcher with background in engineering. After explaining the objective of the focus group, we provide five mock-up wearable devices with different form factors to the participants. Then, the moderator asked their perceptions regarding the usability, safety, and potential applications of those form factors. The second topic of the focus group is to encourage stories with clinical scenarios that can reveal how wearable devices may benefit different healthcare applications.

2.4.2 Participatory Design Workshops

We conducted participatory design workshops to look for opportunities and constraints concerning the use of wearable sensing technologies in healthcare applications. Following the participatory design process [59, 79], two workshops were held to co-design devices to assist healthcare workers in designing a wearable sensing platform.

Participants

Eight nursing professionals (four experienced nursing experts and four senior nursing school students), aged from 22 to 46 years, with experience in clinical nursing, were recruited. Four experienced nursing experts had worked as registered nurses for at least seven years and the four senior nursing school students had been nursing interns in hospitals for at least half a year. All participants were experienced in nursing patients in multiple medical divisions, with the main specialty spanning in divisions of oncology, pediatrics, cardiology, and orthopedics.

Prior to a Workshop

The nursing professionals were briefly introduced to the goal of this study and the structure of the workshop several days (about four to six days) prior to the workshop. Before attending the workshop, participants were asked to finish the pre-workshop homework similar to “homework” assignment to participants prior to each workshop in the PICTIVE project [78]. This pre-workshop homework involved a design sheet to help participants describe

and/or sketch five potential ideas for designs of devices that used any related wearable sensing technology to assist them in resolving difficulties that they faced at work or in their daily life. Since the purpose of this pre-workshop homework was to encourage the participants to generate a wide range of designs in the workshop, the participants were not asked to evaluate the feasibility of their prototype devices for solving the described problems or to limit them in their focus on post-operative caring scenarios. This pre-workshop homework helped the participants to brainstorm a wide range of ideas and collect observations from their working or everyday life as a warm-up for the upcoming workshop.

Procedures

This workshop comprised of three sessions, which were (1) an engagement session, (2) a guiding session, and (3) a development session. Two participatory design workshops were held on two separate days. Each workshop lasted approximately four hours, including a half-an-hour break for lunch.

Engagement session:

In the engagement session (which lasted for half of an hour), all participants introduced themselves and presented the five design ideas that they had prepared for their pre-workshop homework. While presenting a design idea, each participant stuck a design note that also described the idea on a white board. Researchers generated appropriate categories and grouped similar ideas as the participants were placing their notes. After all of the notes on the white board, researchers and participants discussed and moved the notes among categories or to a new category using the affinity diagramming method. All of the categories indicate potential directions or building blocks in the development of the solution in the final design of the workshop.

Guiding session:

In the guiding session (which lasted for half of an hour), a moderator, who was a researcher with an engineering background, introduced the sensing technologies or projects related to healthcare applications. The purpose of the workshop was explained to participants, which was to use wearable and/or environmental sensing technologies in the design of

devices that can help healthcare workers in caring for post-operative patients in hospital.

Development session:

In the final developing session (which lasted two and a half hours), four participants were separated into two pairs in which each participant could share design opinions. To stir cross-disciplinary thinking within each pair, one researcher with an engineering background helped participants to identify alternative sensing technologies for use in their design work, and another researcher with a background in industrial and commercial design expressed their design scenarios using concrete storyboards. Consistent with the work-oriented participatory design process, the goal of each design was set, which was to help healthcare workers care for patients in hospitals. To assist participants in performing the design task effectively, an A3 sheet of paper with the following fields was provided; name of their solution, target user, needs of the target user, targeted sensing events and required feedback, form factor of the designed device. The first stage of the development session involved the participants discussing the name and the target user of the designed solution using categories that were obtained from the pre-workshop homework. Following the identification of the target users, the second stage involved determining the potential needs of the target user that are associated with the solution. In the third stage, the engineering researcher discussed with participants the selection of sensors and actuators for detecting targeted events or generating the required feedback in each application scenario. When the group completed its final design, the designer cooperated with the participants to finalize the form factor of the designed device by sketching an appropriate storyboard. In the final stage, each group presented their final design, about which all participants discussed and commented.

2.4.3 Findings from Explorative Studies

The design goal of BioScope towards a flexible and extensible platform that can fit in with different types of users. We describe the design considerations for both non-engineering background and engineering background users.

For data analysis, the audio recordings of focus groups and interviews, and the video

recordings of the participatory design workshops were transcribed and coded. With lessons learned from our study and observations, we summarize the findings of design exploration for developing a wearable sensing platform as follows.

Findings

Form factor:

Several wearable form factors are discussed to identify what are the nice characteristics as a wearable device, including wristband, neckband, sticker, bandage, badge, garment, flip flop, strap and so on. The explored characteristics include easy wearing, taking off, compact and comfortable. The device should be comfort to wear and quick to deploy. The device also can be placed on proper body location to sense physiological data.

Flexibility:

Because there are various types of physiological sensors that can be used to develop different healthcare applications, a fixed design can only offer limited functions on a device. It is difficult to have one single design that can fit all requirements to develop healthcare application. Therefore, if a wearable sensing platform has flexibility that developers can build a wearable device with required sensing abilities. Then, developers can use this device as a prototype to test their tailored healthcare application. In different iterations of design process, a platform with flexibility can easily deal with the situation that developers may need to add or remove functions on the wearable device.

Extensibility:

A wearable sensing platform must be easily extend and upgrade its abilities. In terms of the hardware part, the platform must be able to extend new component without re-design entire system. In terms of the software part, the platform must provide comprehensive APIs to allow users to construct new method, function and algorithm on top of existing components.

Sensing and data:

To develop a healthcare application, several types of common information are collected

using wearable device, such as vital sign, physical activity and so on. Since a wearable device collects sensory data, the next step is to process the data. Then, the system can feedback the processed data to corresponding users depending on their requirement. The platform should provide a meaningful representation of data that can be understood by the user, e.g.: a pedometer shows step count rather than raw data of motion sensor.

Configuration:

The complexity of configuration is a key whether users will adopt a platform for developing their application or not. For non-engineering background users, they desire there is no complex steps to configure the device and the components are plug and play. For advanced developers, they would desire further customization on top of the platform to realize some specific functions and applications.

Battery life:

The work period of the wearable device is a critical concern for a healthcare application. The device must be able to survive until its mission is complete. However, enabling power saving relies on correct configuration on both hardware and firmware. A proper power saving mechanism can prolong the work period of the wearable device as long as possible and can work transparently to the user without complex configuration.

Chapter 3

BioScope: a Flexible and Extensible Wearable Platform

This chapter presents the design and implementation of BioScope, a flexible and extensible wearable platform for rapidly prototyping healthcare applications (Figure 3.1). BioScope is an extensible bandage system with components that can be stacked like Lego blocks. Using this system, users can simultaneously collect the four most commonly monitored biosignals (i.e., heart rate, body temperature, acoustic signals emitted from the body, and inertial readings of human movement) from multiple bandages to assess and diagnose physical conditions. BioScope extracts the processing and communication functions into a core building block, and hosts the required sensors. Each sensor is affixed as a patch that collects one biosignal. By stacking the required sensors onto a bandage-like platform, users can easily create a customized bandage that can be affixed to the skin of the patient. The data collected by the sensors are sent through a Bluetooth interface to the device screen used by the healthcare worker.

3.1 Design Exploration

The first step of iterative design for a wearable sensing platform is to understand the needs of potential users. We conducted two focus groups and two participatory design workshops to explore the needs from healthcare professionals.



Figure 3.1: BioScope: a flexible and extensible wearable sensing platform.

3.1.1 Focus Groups

We conducted two focus groups to better understand the practical needs of wearable technology in clinical setting. In the first focus group, we recruited four nursing professionals and two physical therapists from National Taiwan University Hospital. In the second focus group, we recruited four nursing professionals from Taipei City Psychiatric Center (TCPC).

Participants

In the first focus group, we recruited four healthcare workers (four experienced nursing experts and two physiotherapists), aged from 27 to 46 years. The four experienced nursing experts had worked as registered nurses for more than three years and the two physiotherapists have worked in the Rehabilitation Department of National Taiwan University Hospital at least six years. These participants have prior experience in oncology, cardiology, and orthopedics. Four experts in the second focus group, aged from 35 to 45 years, have worked as registered nurses for more than ten years at the Taipei City Psychiatric Center (TCPC) and specialized in providing inpatient psychiatric care for children, adolescents,

geriatrics, and patients with substance abuse.

Procedures

The moderator of the focus group was a researcher with background in engineering. After explaining the objective of the focus group, we provide five mock-up wearable devices with different form factors to the participants. Then, the moderator asked their perceptions regarding the usability, safety, and potential applications of those form factors. The second topic of the focus group is to encourage stories with clinical scenarios that can reveal how wearable devices may benefit different healthcare applications.

3.1.2 Participatory Design Workshops

We conducted participatory design workshops to look for opportunities and constraints concerning the use of wearable sensing technologies in healthcare applications. Following the participatory design process [59, 79], two workshops were held to co-design devices to assist healthcare workers in designing a wearable sensing platform.

Participants

Eight nursing professionals (four experienced nursing experts and four senior nursing school students), aged from 22 to 46 years, with experience in clinical nursing, were recruited. Four experienced nursing experts had worked as registered nurses for at least seven years and the four senior nursing school students had been nursing interns in hospitals for at least half a year. All participants were experienced in nursing patients in multiple medical divisions, with the main specialty spanning in divisions of oncology, pediatrics, cardiology, and orthopedics.

Prior to a Workshop

The nursing professionals were briefly introduced to the goal of this study and the structure of the workshop several days (about four to six days) prior to the workshop. Before attending the workshop, participants were asked to finish the pre-workshop homework similar

to “homework” assignment to participants prior to each workshop in the PICTIVE project [78]. This pre-workshop homework involved a design sheet to help participants describe and/or sketch five potential ideas for designs of devices that used any related wearable sensing technology to assist them in resolving difficulties that they faced at work or in their daily life. Since the purpose of this pre-workshop homework was to encourage the participants to generate a wide range of designs in the workshop, the participants were not asked to evaluate the feasibility of their prototype devices for solving the described problems or to limit them in their focus on post-operative caring scenarios. This pre-workshop homework helped the participants to brainstorm a wide range of ideas and collect observations from their working or everyday life as a warm-up for the upcoming workshop.

Procedures

This workshop comprised of three sessions, which were (1) an engagement session, (2) a guiding session, and (3) a development session. Two participatory design workshops were held on two separate days. Each workshop lasted approximately four hours, including a half-an-hour break for lunch.

Engagement session:

In the engagement session (which lasted for half of an hour), all participants introduced themselves and presented the five design ideas that they had prepared for their pre-workshop homework. While presenting a design idea, each participant stuck a design note that also described the idea on a white board. Researchers generated appropriate categories and grouped similar ideas as the participants were placing their notes. After all of the notes on the white board, researchers and participants discussed and moved the notes among categories or to a new category using the affinity diagramming method. All of the categories indicate potential directions or building blocks in the development of the solution in the final design of the workshop.

Guiding session:

In the guiding session (which lasted for half of an hour), a moderator, who was a researcher with an engineering background, introduced the sensing technologies or projects related

to healthcare applications. The purpose of the workshop was explained to participants, which was to use wearable and/or environmental sensing technologies in the design of devices that can help healthcare workers in caring for post-operative patients in hospital.

Development session:

In the final developing session (which lasted two and a half hours), four participants were separated into two pairs in which each participant could share design opinions. To stir cross-disciplinary thinking within each pair, one researcher with an engineering background helped participants to identify alternative sensing technologies for use in their design work, and another researcher with a background in industrial and commercial design expressed their design scenarios using concrete storyboards. Consistent with the work-oriented participatory design process, the goal of each design was set, which was to help healthcare workers care for patients in hospitals. To assist participants in performing the design task effectively, an A3 sheet of paper with the following fields was provided; name of their solution, target user, needs of the target user, targeted sensing events and required feedback, form factor of the designed device. The first stage of the development session involved the participants discussing the name and the target user of the designed solution using categories that were obtained from the pre-workshop homework. Following the identification of the target users, the second stage involved determining the potential needs of the target user that are associated with the solution. In the third stage, the engineering researcher discussed with participants the selection of sensors and actuators for detecting targeted events or generating the required feedback in each application scenario. When the group completed its final design, the designer cooperated with the participants to finalize the form factor of the designed device by sketching an appropriate storyboard. In the final stage, each group presented their final design, about which all participants discussed and commented.

3.1.3 Findings from Explorative Studies

The design goal of BioScope towards a flexible and extensible platform that can fit in with different types of users. We describe the design considerations for both non-engineering

background and engineering background users.

For data analysis, the audio recordings of focus groups and interviews, and the video recordings of the participatory design workshops were transcribed and coded. With lessons learned from our study and observations, we summarize the findings of design exploration for developing a wearable sensing platform as follows.

Findings

Form factor:

Several wearable form factors are discussed to identify what are the nice characteristics as a wearable device, including wristband, neckband, sticker, bandage, badge, garment, flip flop, strap and so on. The explored characteristics include easy wearing, taking off, compact and comfortable. The device should be comfort to wear and quick to deploy. The device also can be placed on proper body location to sense physiological data.

Flexibility:

Because there are various types of physiological sensors that can be used to develop different healthcare applications, a fixed design can only offer limited functions on a device. It is difficult to have one single design that can fit all requirements to develop healthcare application. Therefore, if a wearable sensing platform has flexibility that developers can build a wearable device with required sensing abilities. Then, developers can use this device as a prototype to test their tailored healthcare application. In different iterations of design process, a platform with flexibility can easily deal with the situation that developers may need to add or remove functions on the wearable device.

Extensibility:

A wearable sensing platform must be easily extend and upgrade its abilities. In terms of the hardware part, the platform must be able extend new component without re-design entire system. In terms of the software part, the platform must provide comprehensive APIs to allow users to construct new method, function and algorithm on top of existing components.

Sensing and data:

To develop a healthcare application, several types of common information are collected using wearable device, such as vital sign, physical activity and so on. Since a wearable device collects sensory data, the next step is to process the data. Then, the system can feedback the processed data to corresponding users depending on their requirement. The platform should provide a meaningful representation of data that can be understood by the user, e.g.: a pedometer shows step count rather than raw data of motion sensor.

Configuration:

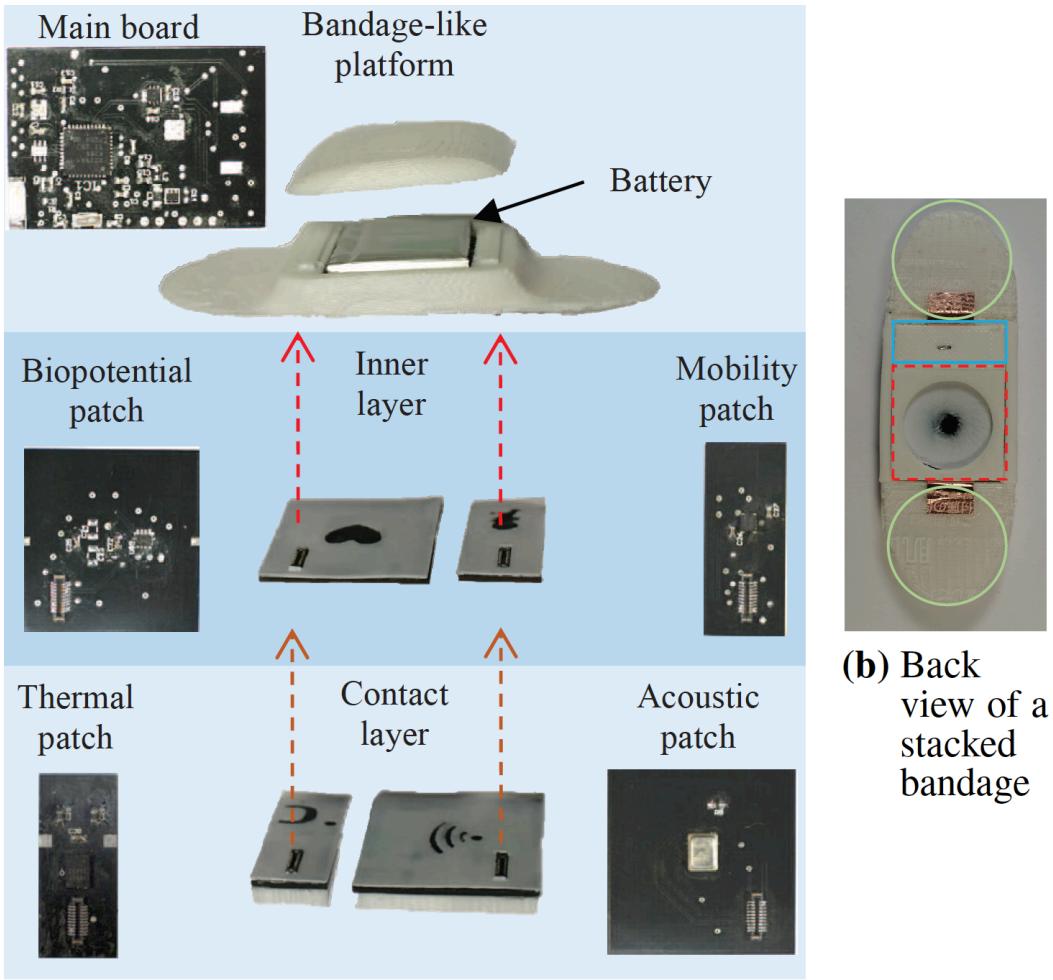
The complexity of configuration is a key whether users will adopt a platform for developing their application or not. For non-engineering background users, they desire there is no complex steps to configure the device and the components are plug and play. For advanced developers, they would desire further customization on top of the platform to realize some specific functions and applications.

Battery life:

The work period of the wearable device is a critical concern for a healthcare application. The device must be able to survive until its mission is complete. However, enabling power saving relies on correct configuration on both hardware and firmware. A proper power saving mechanism can prolong the work period of the wearable device as long as possible and can work transparently to the user without complex configuration.

3.2 BioScope System Design and Implementation

Based on the design considerations, we designed and implemented the BioScope system. The following subsections include (1) flexible and extensible sensing bandage, (2) stacking mechanism, (3) sensor patches, (4) explorative experiment.



(a) Assembled circuit boards and bandage stacking

Figure 3.2: Design of the extensible sensing bandage. (a) Four patches with distinctive embossed icons are stacked on the inner and contact layers according to the direction indicated by the red dotted arrows. (b) The sound-collecting structure (box with red dashed border), a thermocouple wire (box with blue solid border), and two electrodes coated with a conductive gel (two green circles) directly contact the skin.

3.2.1 Flexible and Extensible Sensing Bandage

Based on the design considerations from our explorative studies for designing a flexible and extensible platform that can facilitate data collection, we collaborated with an expert from the Department of Nursing at National Taiwan University, Taipei, Taiwan. Through biweekly meetings with the expert, we first identified the necessity for an extensible sensing device that can assist healthcare workers in efficiently collecting commonly monitored data used in nursing assessments (e.g., monitoring the functional recovery of postoperative patients). Based on input and feedback provided by the expert and two experienced

healthcare workers (three women, aged 30~42 years), we explored alternative designs for the flexible and extensible sensing device. During these brainstorming sessions, we recorded the ideas of all participants and then organized these ideas by using affinity diagrams to gain a deeper understanding of the primary design considerations.

The system should enable healthcare workers to tailor the sensors to specific assessments. Depending on the diagnostic results, healthcare workers may wish to reassess the patient by collecting additional data. The system should therefore enable preliminary screenings in which the healthcare workers can add or remove sensors to the device. For developer, the system should be able to extend new component without re-design entire system.

The system should be efficient and simple to use, even for healthcare workers with no technical background. After determining the types of data required for patient assessment, healthcare workers should be able to easily identify which sensors to add to the device. To enable healthcare workers to attach the required sensors to appropriate locations on the patient, the device should have a compact adhesive design, enabling it to be easily affixed to the skin without undue inconvenience or skin irritation [39].

The system should be able to analyze trends in data that are collected continually over durations ranging from several hours to several days. Healthcare workers can use this long-term data to review and reassess the physical state of patients and identify potential health complications. To continually collect data within a given period, (e.g., half of a day), a fully charged battery should contain sufficient energy to power the sensing device and the system should be able to enable power saving mechanism to prolong the battery life.

To create a flexible and extensible system, we designed a device with two distinct types of module (Figure 3.2): (1) the basic bandage platform and (2) sensor patches. The bandage-like platform resembles an adhesive bandage. We drew a 3D model of the platform and then printed it using a 3D printer and elastic filaments. Figure 3.2(a) depicts the platform, in which a hollow space is reserved to encase the customized sensing patches. To provide processing and communication capabilities, we designed a customized circuit board, called the main board, that could be mounted in the hollow space. The main board and

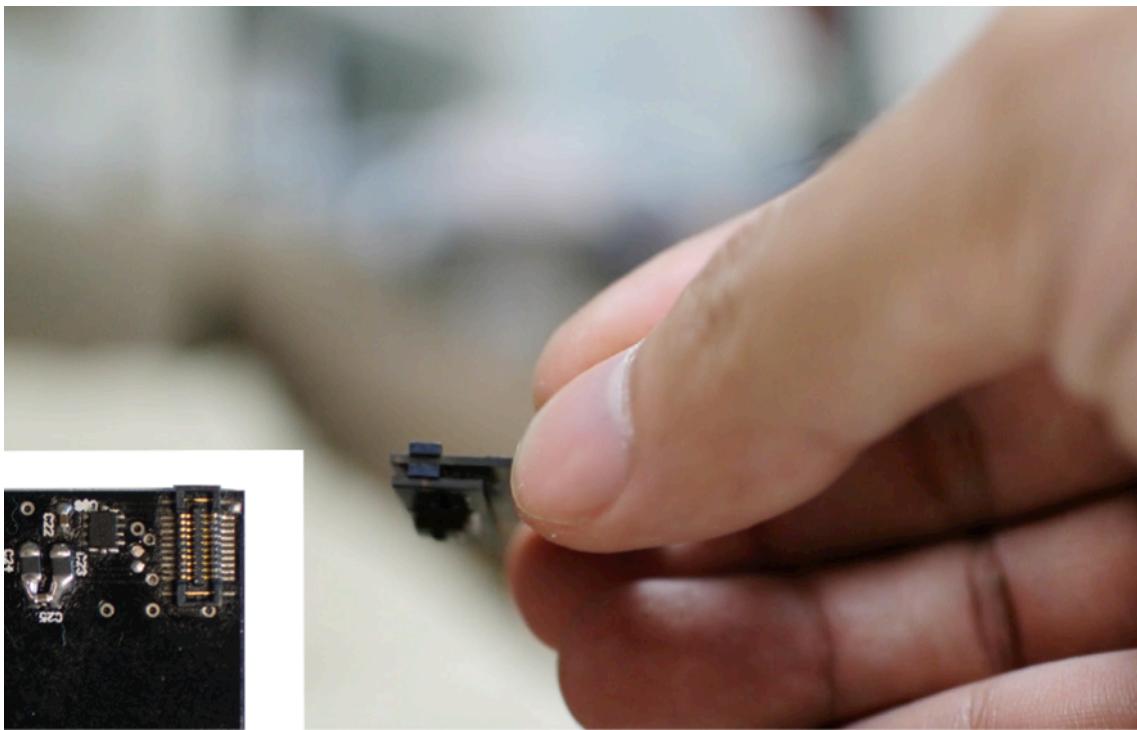


Figure 3.3: Stacking mechanism to connect sensor patches and mainboard.

the stacked sensor patches are powered by a 130-mAh Li-ion battery situated in the upper layer of the platform. On the main board, a Microchip PIC32MX150 microcontroller receives data from the sensor patches through board-to-board connectors, and then relays the processed data to the monitoring screen through a Texas Instruments CC2451 microcontroller with Bluetooth radio.

3.2.2 Stacking Mechanism

Users can choose different combinations of LEGO-like sensing and interaction blocks and assemble them inside a bandage form factor. The key mechanism that enables BioScope to be flexible and extensible is stacking (Figure 3.3). In this subsection we We define a connecting protocol to unify the power supply and data communication between the patches and the mainboard.

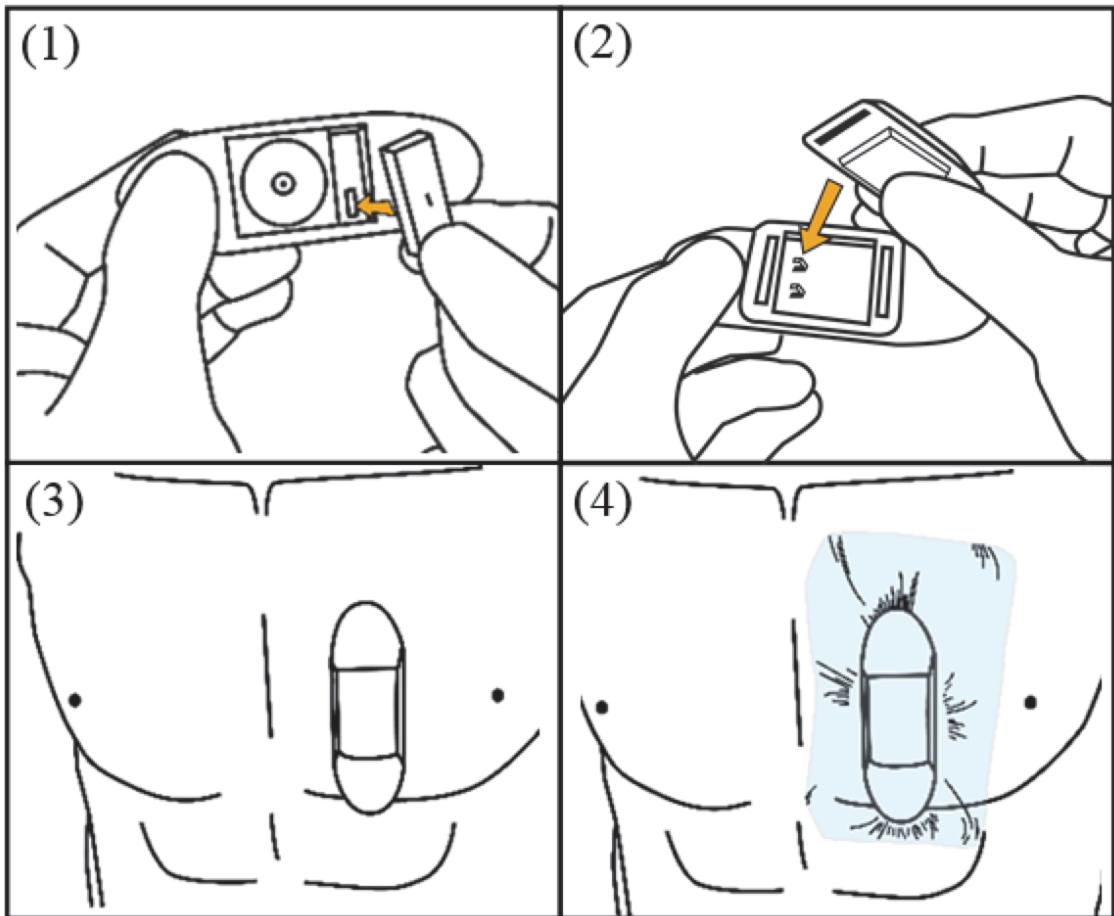


Figure 3.4: Four steps for applying bandages.

Assembling

Figure 3.4(a) illustrates these patches stacked in two layers in the hollow space of the platform; temperature and microphone sensors directly contact the skin to collect high-quality signals. To create accessible patches for the users, each patch was punched with a representative icon on both sides of the covering material. Figure 3.4 illustrates the BioScope application process: (1) A healthcare worker selects the appropriate patches (or dummy patches) by using the embossed icon as a reference, stacks the patches on (or filling in empty spaces that are originally occupied by unused patches on) the platform, (2) inserts a battery and closes the protection cap, (3) affixes the bandage to chest, and (4) covers the entire bandage with transparent film dressings if water-proof is needed.

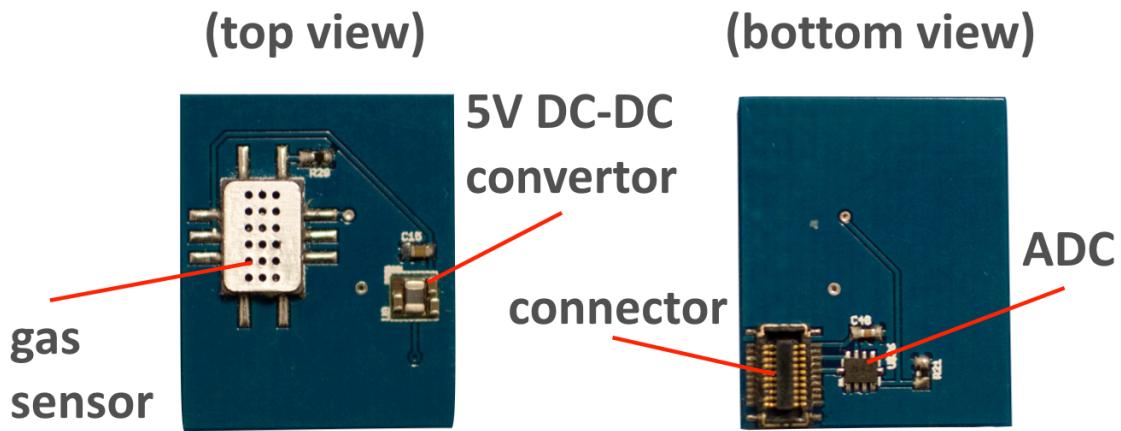


Figure 3.5: An example of using BioScope to prototype a breathalyzer.

Power source

Each component in the system may require various power supply range. Since a component may require a supply voltage higher or lower than the voltage from li-ion battery (4.2V), the voltage of power source should be converted to proper voltage. A linear regulator can be used to lower the power source to the working supply voltage. The mainboard equips a Microchip MIC5392 regulator with two 3V outputs, one output supplies the microcontroller and another output supplies peripheral patches that can be turned on/off by microcontroller. If the supply voltage of a component is higher than power source, a DC-DC converter can converts the voltage to a higher voltage. In figure 3.5, the supply voltage of the gas sensor is 5V which is higher than the voltage of the battery. Therefore, we place a compact DC-DC convertor, a Texas Instruments TPS81256, to boost the voltage to 5V and provide sufficient current to enable the sensor. A compact switch button is mounted on the mainboard used to wake-up and reset the system. For power saving purpose, the system must be able to enter low frequency mode which only can be waked up by external interrupt. All peripheral components and radio frequency function can not work in low power mode. Also, the system will scan plugged patches when reset is triggered and wait the command from terminal device, i.e., mobile phone.

Data communication

For a sensor that communicate to the microcontroller through digital bus, I2C and SPI are two most common protocols. Those protocols is bus topology that allows microcontroller can access multiple slave components with only few pins. SPI requires slave select pin to enable particular slave component for communication, our first prototype supports up to three slave components.

However, some sensors are designed to output analog signals to deliver sensory data rather than digital signals. In Figure 3.5, a Texas Instruments ADS1114 analog-to-digital converter translates the analog signals from gas sensor to digital signals in I2C. Thus, the system doesn't have to provide multiple pin-to-pin analog channels to serve individual analog sensor.

Horizontal expansion

In addition to vertical stacking (Figure ?), the system provide horizontal expansion supported by a tiny H-bridge board (Figure ?), which has two sets of plug and receptacle connectors. The H-bridge board can be connected on plug connector or receptacle connector on the top or bottom of the board (Figure ?). For advanced developers, they may need to connect a component the system doesn't support. A breakout board (Figure ?) can be used to test new component for developing new patch for BioScope platform. The developer can evaluate customized component as well as off-the-shelf development toolkit.

3.2.3 Patches

To collect biosignals, such as electrocardiogram (ECG) signals, two pre-allocated electrodes (i.e., two conductive copper areas situated 6.4 cm apart at opposite ends of the bandage) are coated with a thin layer of electrical gel (Figure 3.2(b)). The sensor patches, consisting of small sensor boards sandwiched between two thin layers of 3D-printed elastic filaments, are mounted on the bandage-like platform using connectors. To demonstrate the concept of this system, we designed six types of patch —biopotential, thermal, acous-

tic, mobility, interaction and recharging patches —to facilitate the collection in the most commonly monitored data in healthcare applications. The detail of those six types of patch are described in follows.

Biopotential patch:

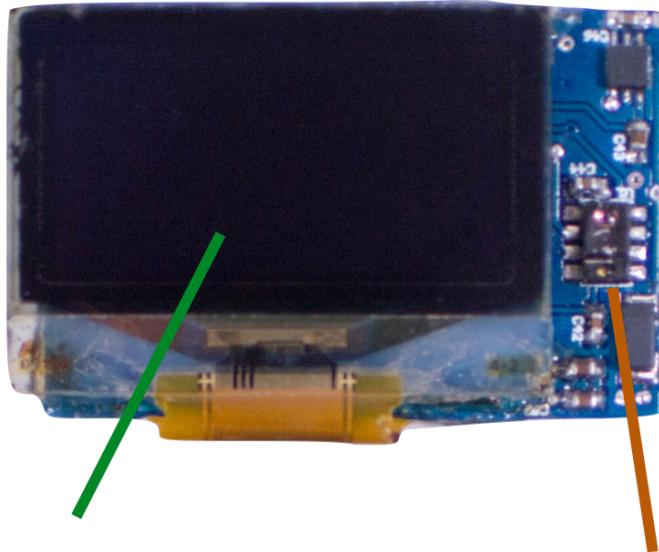
This $23\text{ mm} \times 24\text{ mm}$ patch, stacked in the inner layer, amplifies and filters ECG signals to enable continual cardiovascular monitoring. Cardiac activity, which can be characterized by ECG signals, is a crucial biosignal for assessing the cardiac functions of people. By amplifying the electrical potential difference measured between the two electrodes by using a Texas Instruments ADS1115 analog-to-digital converter on the patch, ECG signals can be monitored by allowing the passing of low-frequency signals from 0 to 100 Hz [36] by using a low-pass filter. A pulse can be identified by detecting spikes in the signal, thus enabling users to assess heart and respiratory rates.

Acoustic patch:

This $24\text{ mm} \times 24\text{ mm}$ patch, stacked in the contact layer, records acoustic signals emitted by a person's body or while the person is phonating. By identifying the unique sound patterns that the body's organs generate, users can assess person's conditions. Furthermore, person's phonation can indicate social interaction, according to which users can assess whether a person is depressed or impaired cognitively. To clearly record the internal sounds of the body, a mediating instrument (e.g., a stethoscope) is required. Inspired by the design of electronic stethoscopes, we designed and attached a small sound-collecting structure (Figure 3.2(b)) on the patch that effectively amplified acoustic signals from the body and occluded environmental noise. Above the sound-collecting structure, an opening is aligned with the receiving hole of an InvenSense INMP441 microphone with I₂S interface on the board to guide sound waves towards the hole. In this study, we detected a person phonation, which reflected social activity, by analyzing the frequency components of the collected sound.

Thermal patch:

This $10\text{ mm} \times 24\text{ mm}$ patch is stacked in the contact layer and measures the skin tem-



OLED display gesture sensor

Figure 3.6: Interaction patch includes OLED display and on-the-air gesture sensor

perature, which can indicate a person's health. users can evaluate a person by identifying abnormal or varying temperatures [57]. A Maxim MAX31850 K-type thermocouple-to-digital converter detects body temperature through a thermocouple wire that protrudes from the covering elastic material to contact the skin of the person (Figure 3.2(b)).

Mobility patch:

This 11 mm × 24 mm patch, stacked in the inner layer, monitors the mobility level of a person. For example, to prevent complications caused by reduced mobility levels and assess functional recovery, users must track the mobility level of patients. On this patch, an InvenSense MPU6050 6-axis motion sensor is used to collect motion readings, which indicate whether a person is moving or stationary. The mobility level can be derived by calculating the percentage of time a person is moving.

Interaction patch:

To enrich data feedback and user interaction for exercise applications, we also expanded an interaction patch (Figure 3.6) that consists of a tiny display (0.96" OLED display), and a touchless gesture sensor (Avago APDS-9960) that is used to recognize six directions (near, far, up, down, left and right) of on-the-air swipe gesture. The gesture can be used

to switch between different sensor's data or different data representations. The interaction patch is connected to main board through stacking mechanism as well. When a mobile device attempts to establish connection with particular one of multiple wearable devices. To simplify Bluetooth pairing procedure for quickly retrieving data or configuring device, a NXP NTAG203 NFC tag is embedded in wearable device.

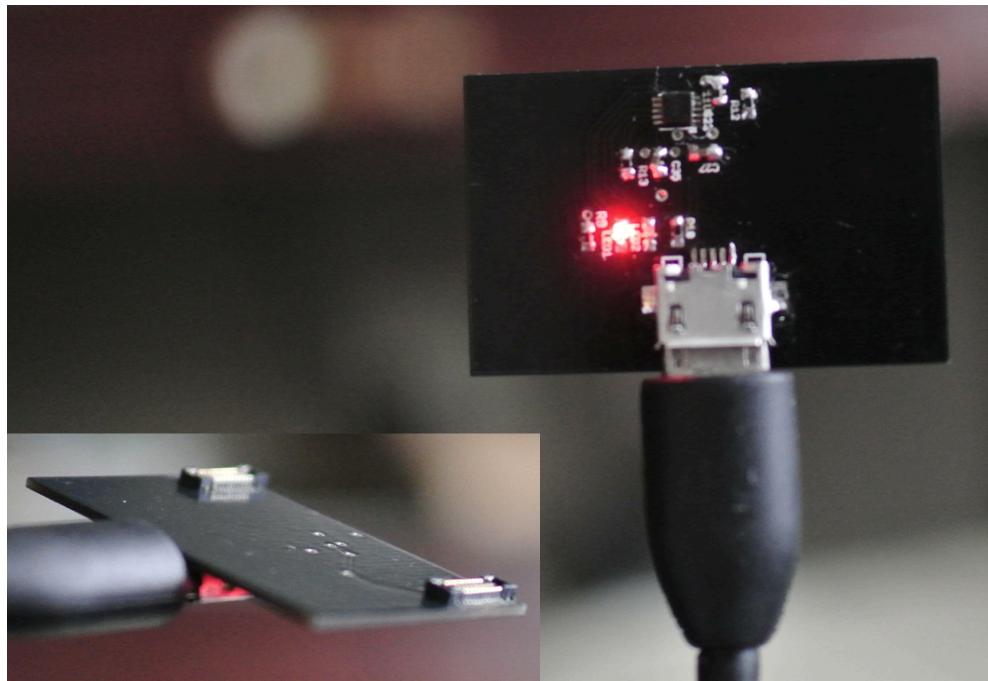


Figure 3.7: Recharging patch.

Recharging patch:

The entire system is powered by a Li-ion battery with 130-mAh capacity. This patch has a USB connector to connect a 5V power source for charging and a Texas Instruments BQ24040 to regulate the voltage in charging process (Figure 3.7).

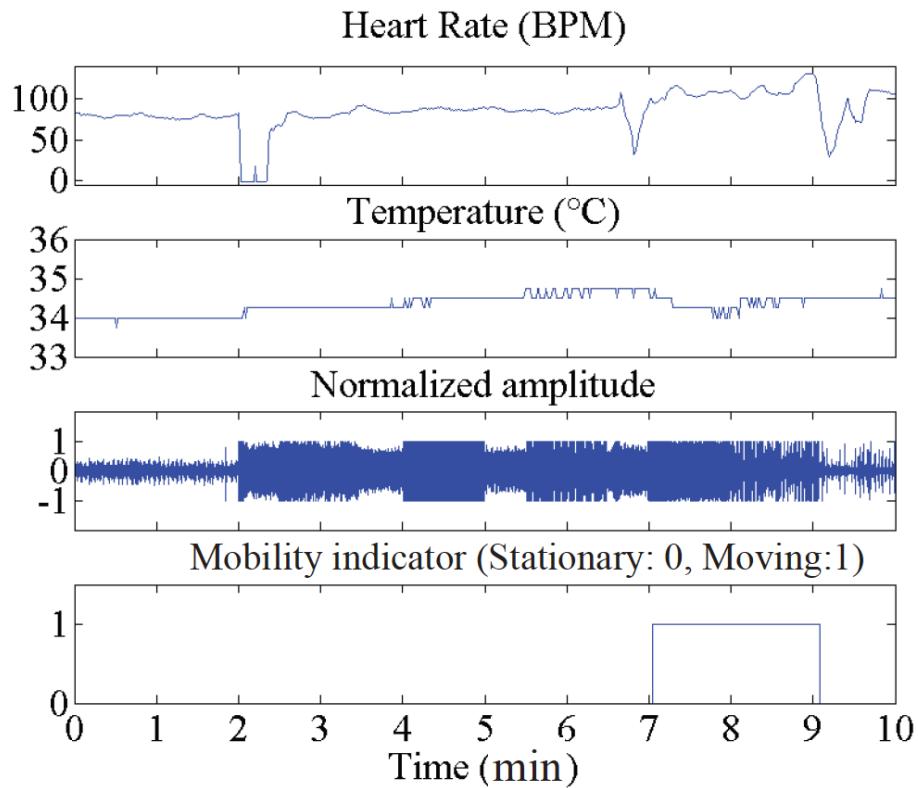
3.2.4 Explorative Experiment

To validate system functionality, we scripted a sequence of activities to simulate conditions arising when a patient with basic functional mobility is hospitalized. Two volunteers performed the specific activities while wearing bandages equipped with all four patches on their chests, enabling us to collect data (Figure 3.8(a)). The simulations were conducted for 10 and 30 minutes in the cases of the first and second participants (P1 and P2),

respectively. Activities comprised (1) lying down on a bed, (2) having a phone conversation, (3) watching TV, (4) having a face-to-face conversation, and (5) performing walking. In the experiments, the data captured were heart rate, skin temperature, received acoustic signals, and mobility indicators.



(a) Stacked bandage affixed to patient's chest



(b) Results from data collected by BioScope

Figure 3.8: Four steps for applying bandages.

Figure 3.8(b) shows the results obtained by analyzing the data collected from P1. The readings obtained from the mobility patch indicated that P1 moved between the seventh

and ninth minutes; this was an accurate assessment of the patient's behavior during that time. While walking, P1's heart rate increased relative to that while stationary between the start and the seventh minute. When the posture of the patient drastically changed, such as when P1 stood up near the second, seventh, and ninth minutes, the ECG signals were distorted [45], producing a dip in the calculated heart rate. The sounds generated by clothes rubbing against the bandage when P1 moved adversely affected the quality of detected internal sounds, causing the amplitudes to increase between the seventh and ninth minutes. After filtering out sounds generated by movement, however, we could still detect when P1 phonated between the second and seventh minute. Based on the vocal resonance of the body [50], we detected phonation by identifying the frequency components of sounds higher than the 0- to 3-kHz frequency range of the human voice [7]. Finally, the body temperature varied minimally ($34^{\circ}\text{C} \sim 35^{\circ}\text{C}$) and was near the normal skin temperature of the human chest [57]. Overall, the results accurately reflected the activities performed by the participants.

To examine whether the system can detect reasonable values for the average heart rate, total moving duration, average skin temperature, and total phonating time, we analyzed the data collected from P2 over 30 minutes. The total moving duration was determined to be 7.2 minutes (actual value: 6.9 minutes), with an error of 4.0%. The average heart rate was 81.5 and 100.3 beats respectively. Because P2 did not perform intensive exercise, the average temperature did not vary significantly, remaining near 33.9°C . By identifying the high-frequency components embedded in the high-pitched sounds collected when P2 was stationary, P2 was determined to have phonated for 635.5 seconds (actual value: 564.0 seconds), with an error of 12.7% per min (BPM) when P2 was stationary and moving, respectively. Because P2 did not perform intensive exercise, the average temperature did not vary significantly, remaining near 33.9°C . By identifying the high-frequency components embedded in the high-pitched sounds collected when P2 was stationary, P2 was determined to have phonated for 635.5 seconds (actual value: 564.0 seconds), with an error of 12.7%.

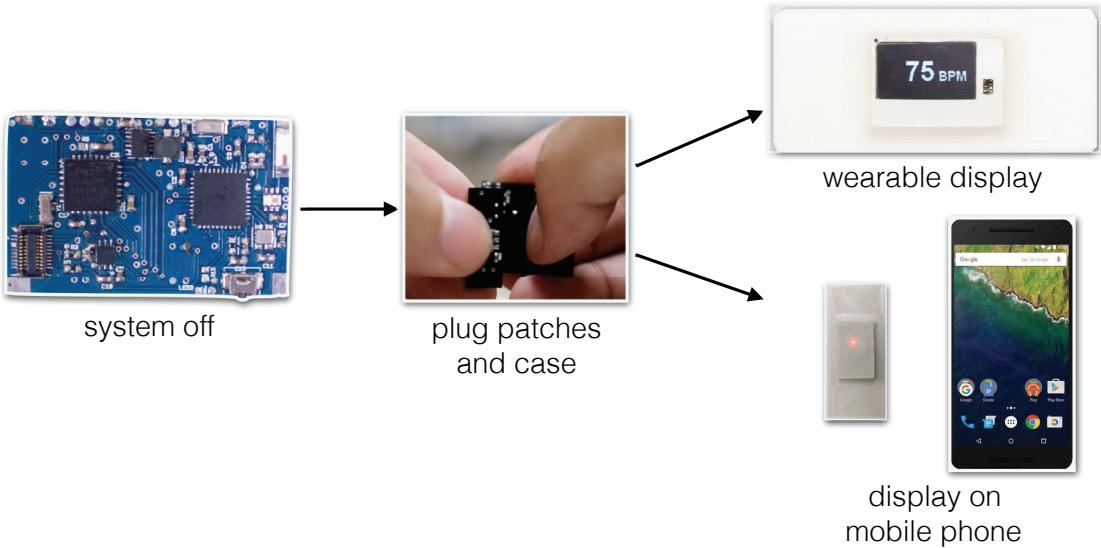


Figure 3.9: The steps of configuration for data visualization. (a) System off. (b) Assembly: stacking patches and casing. (c) display on interaction patch. (d) display on mobile phone.

3.3 Visualization of Sensory Data

The hardware design of BioScope sensing platform enables flexibility and extensibility, its firmware and software have corresponding design consideration to promote the prototyping phase of design process. This section describes the visualization of sensory data that allows users can simply receive raw and/or primitive processed data from sensors. Visualization is an efficient way that can assist users to easily understand the characteristics of sensors. In order to simplify the usage of the system for non-engineering background users, BioScope platform provides the basic mechanisms provided by BioScope platform to visualize sensory data: (1) display on interaction patch and (2) display on mobile device. The users can simply stacks needed patches on the mainboard of BioScope. The default firmware can detect attached patches and run corresponding process to collect sensory data and provide feedback.

3.3.1 Configuration

BioScope platform allows users to easily configure the system for collecting sensory data and receiving feedback from chosen patches with simple steps. We provide a mobile app

running on top of Android for data configuration and data visualization. The main steps of configuration for data visualization is shown in Figure 3.9.

System off

This is the initial state of the system, when the device is not applied a power source, i.e., install/recharge battery, or user configures the device to turn off the device. As well as many off-the-shelf electronic products, the device doesn't have an on/off switch to shutdown the device. Instead, the system will enter to low power mode, because the consumed current in low power mode is negligible ($\sim 16 \mu\text{A}$). The device will be turned to system off mode caused by two conditions: (1) pressing reset button on the device then idle for one minute, and (2) receiving the off command from mobile phone. In low power mode, the microcontroller switches to secondary oscillator with low system frequency (32.768kHz) and turns off all internal components of the microcontroller for power saving.

The device can only be waked up by external interrupt in system off mode.

Assembly: Stack patches and casing

User can choose needed sensor patches and components to construct a customized device by simply stacking patches on the mainboard. Then the device is placed into a 3D printing case with the elastic filaments, e.g., bandage case. When the device is booted, the system will blink the on-board LED and scan stacked patches on the mainboard through I2C and SPI bus. For the patches with I2C bus, the system will query the specific addresses defined in the BioScope system to check whether the patch is attached. For the patches with SPI bus, the system will query the connected patch by setting each slave select pin to high voltage individually. Thus, the system can automatically determine attached patches without users' effort to manually input the chosen patches.

Display data

The mobile phone is used as a terminal for configuring device and visualizing sensory data via Bluetooth radio. The mobile phone is the central and the BioScope device is peripheral



Figure 3.10: Representations of data visualization on wearable display: digit, pattern, icon.

in the wireless topology. After stacking the chosen patches and casing, user can use the mobile phone to establish the connection with device. Then user can configure the mobile phone to display sensory data on the screen.

The device can also operate without communication with the mobile phone. When the interaction patch is detected on the I2C bus, the system will run the corresponding process for basic sensing and feedback functions with wearable display. Users can switch and control the displayed sensory data through the touchless gesture sensor on the interaction patch. The system also allows the device to be further configured and accessed through the user interface on mobile phone as well.

3.3.2 Data Processing and Representation

Considering application need, wireless bandwidth and power efficiency, the sensory data should be processed in different ways. For example, the device transmits raw data of ECG to mobile phone for further computation that requires a sampling rate of 200 Hz (400 bytes per second). Such high data rate will consume a lot of power for data transmission. If the application needs heart rate data rather than the raw data of ECG, the device can determine every single peak of the raw data for computing average heart rate and then transmit data with sampling rate of 1 Hz (1 byte). Thus, the overall power consumption can be significant reduced by processing data using the microcontroller. However, the microcontroller may not be able to compute high complexity work, e.g., activity recognition with motion data. For this case, developers have to adopt higher performance processor or transit raw data to mobile phone for processing.

The data can be illustrated in different representations for visualization depending on users' needs. For example, displaying heart rate data with interaction patch can be different representations: (1) digit: beats per minute, (2) pattern: raw data and (3) icon: blink with beat (Figure 3.10).

3.3.3 User Interface on Mobile Device

The mobile device is versatile and pervasive in our daily life. Therefore, we built a mobile app running on top of Android platform to facilitate data visualization and device configuration.

Device scan:

The first page (Figure ? (??)) of the mobile app shows the BioScope devices nearby. The device without specific prefix in device ID is filtered out. By pressing the particular device in the list, the system will start to establish the connection with the device.

Attached patches:

When the connection between phone and device is established, the mobile app will list all attached patches on the screen (Figure ?(??)).

Select function:

A patch may have multiple functions, user can configure and visualize the data of each function separately (Figure ?(??)). If the check button of each function in the list is checked, the sensory data will be logged in the storage of mobile device. User can export these log data to PC for further analysis.

Config visualization:

By choosing a function in the list, the data is visualized on the screen of the mobile phone. User can further config the data and its representations (Figure ?(??)).

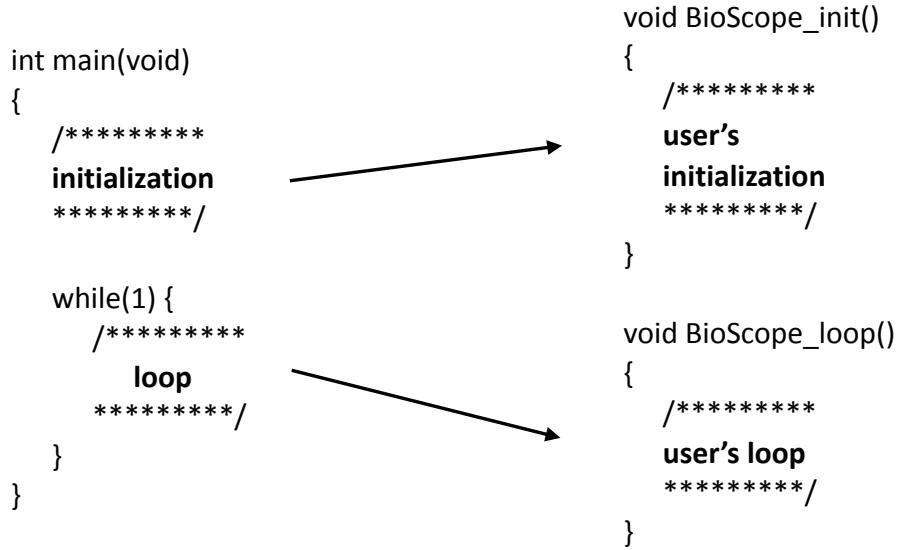


Figure 3.11: Extract the initialization and loop sections from conventional structure of microcontroller codes

3.4 Platform APIs

In previous section, we introduced the basic data visualization and collection on BioScope platform that enables plug-and-play stacking mechanism and easy-to-use interface. In addition to visualizing and collecting data, advanced developers may need to further program the system for building their own application. We provides device APIs and phone APIs to facilitate building application on BioScope platform. The device APIs allow developers to simply access the sensors and Bluetooth radio on BioScope device and phone APIs to retrieve sensory data and configure device on the smart phone.

3.4.1 Device APIs

The BioScope system adopts a Texas Instruments CC2451 microcontroller as the core of the wearable device. However, the software developing environment of microcontroller may be too complicated for the users. To facilitate the access of components on BioScope platform, we encapsulates the detail codes to the APIs based on the software stack provided by Texas Instruments. As well as the prototyping platforms in the market, we extract the initialization and loop sections from conventional structure of microcontroller codes (Figure 3.11). Thus, developers only need to handle initialization and periodical routine

and callback events.

```
void BioScope_init()
{
    BluetoothLE ble;
    TemperatureSensor temp;
    setLoopInterval(1000);
}

void BioScope_loop()
{
    uint16_t tempReading = temp.readTemp();
    ble.send(temp.type, 2, &tempReading);
}
```

Figure 3.12: An example of BioScope device APIs for reading temperature sensor and transmitting sensory data through Bluetooth radio

Figure 3.12 shows an example for reading data from temperature sensor and transmitting sensory data through Bluetooth radio. In *BioScope_init* function, developers only need to declare the struct variables of Bluetooth component and temperature sensor. And the configuration of the loop interval must be set up in the initialization section for power sensing requirement. Then the *BioScope_loop* function runs sensory reading and radio transmission periodically with the specific interval. The provided APIs in C language are summarized as follows.

System

uint16_t getPatchList()*: Get the list of attached patches on the device.

void setLoopInterval(uint16_t interval): Set the interval(ms) of the loop.

void setLoopIntervalSec(uint16_t interval): Set the interval(sec) of the loop.

void delayMS(uint16_t ms): Delay for the number of millisecond.

void delayUS(uint16_t us): Delay for the number of microsecond.

void enablePowerSaving(): Enable power saving. If idle, the system will turn to low-

power mode until next interrupt occurs.

void disablePowerSaving(): Disable power saving. The processor stays in fastest frequency and all peripheral components are enabled as well.

Sensor

bool isAttached(uint8_t patch): Determine whether a patch is attached on the device or not.

void detached(uint8_t patch): The callback function is invoked when a patch is detached.

uint16_t readTemp(): Read temperature data from thermal patch.

uint16_t readHeartRate(): Read heart rate from Biopotential sensor.

uint16_t readAccelerometer():* Read 3-axis accelerometer data from mobility sensor.

uint16_t readGyro():* Read 3-axis gyro data from mobility sensor.

uint16_t readMagnetic():* Read 3-axis magnetic data from mobility sensor.

Interaction patch

*void writeChars(uint8_t *chars, uint8_t size):* Write characters on the OLED display with specific size.

*void drawIcon(uint8_t *icon):* Draw an icon on the OLED display represented by bitmap format.

void isGesture(uint8_t gesture): Check whether a specific gesture is detected.

void gesture(uint8_t gesture): The callback function is invoked when a gesture is detected.

BluetoothLE radio

*void bleConnected(uint8_t *centralAddress):* The callback function is invoked when a central device has connected with this device.

*void bleDisconnected(uint8_t *centralAddress):* The callback function is invoked when a central device has disconnected with this device.

*void send(uint8_t type, uint8_t length, uint8_t *data):* Send data to central device through Bluetooth radio.

*void received(uint8_t type, uint8_t length, uint8_t *data):* The callback function is invoked when received a command from central device through Bluetooth radio.

3.4.2 Phone APIs

The BioScope system provides APIs on Android platform for collecting sensory data and accessing the BioScope device. The phone APIs consists of two classes: *BioScopeBle* and *BioScopeBleListener*. The *BioScopeBle* is the class that provides the functions to communicate with BioScope device through Bluetooth radio. The *BioScopeBleListener* is the interface class that allows developers to implement the callback functions of communication with BioScope device in their codes. The provided APIs in Java language are summarized as follows.

BioScopeBle

BioScopeBle(String device): The constructor of the class is used to new an object for communicating with specific BioScope device.

void connect(): Start to establish connection with the BioScope device.

void disconnect(): Disconnect an existed connection with the BioScope device.

void writeCharacteristic(int type, byte[] command): Write the command to a characteristic on the BioScope device.

int[] getPatchList(): Get the list of attached patches on the BioScope device.

BioScopeBleListener

void bleNotSupported(): The callback function is invoked if BLE is not supported in this device.

void bleConnectionTimeout(): The callback function is invoked when the connection establishment is timeout (10 sec).

void bleConnected(): The callback function is invoked when the connection establishment is success.

void bleDisconnected(): The callback function is invoked when the established connection is disconnect.

void bleWriteSuccess(): The callback function is invoked when the execution of the *writeCharacteristic* method is success.

void bleWriteStateFail(): The callback function is invoked when the execution of the *writeCharacteristic* method is fail.

void bleDataReceived(int type, byte[] data): The callback function is invoked when the data sent from BioScope device are received.

3.5 Summary

In this chapter, we introduced BioScope, a flexible and extensible wearable platform designed to facilitate rapidly prototyping of the design process. Users can choose the needed components to build a customized device for their application. The system provides visualization of sensory data that users can simply access various primitive representations of sensory data. The platform APIs allows advanced users to program the system to meet their requirement. Users can therefore easily create a high-fidelity prototype for evaluating and testing required functions in their design.

Chapter 4

Tangram: Simplify PCB Design Process with Modularization

In previous chapter, we started from medical-background users' viewpoints to design a wearable platform for developing health applications. However, a team to develop an application may involve cross-disciplinary members. The users with different background knowledge and skill levels have various requirements for prototyping. Thus, we extend the concept of BioScope and further consider the needs of engineering-background users who develop and implement the device for wearable, mobile and IoT applications. This process often requires numerous design iterations before reaching its design goals. In the past, makers have to spent significant efforts on implementing outer cases, electronic components, firmwares and corresponding softwares, while designing and exploring various novel applications. With the evolutions of technology on 3D printing, laser cutting and CNC engraving, it becomes much easier and affordable for HCI researchers and the emerging makers [54] to do rapid prototyping on the appearance of devices.

For developers to prototype on digital functionalities, Arduino, by providing easy-to-use hardware and software, is the most popular open-source development toolkit. However, users may meet some problems with using such development tools to prototype their devices. Robustness would be an important issue when users will have tangible interactions with the device. Troubleshooting of a bunch of wires bridging the main controller and

external components will bring a lot of bother to users. Also, a device built by development tools may not be able to meet users' needs in dimensions and power saving. To address these issues, many HCI researchers then choose to build their device with customized PCB, which has high quality with low cost [53, 81]. Nevertheless, designing PCB is not only a hard work for those who has no background knowledge, but it also costs experienced users a lot of time and effort.

In HCI community, several attempts have been made on reducing the effort spends on prototyping and fabrication processes. D.tools [62] proposed a toolkit that supports design thinking processes for device prototyping with iteration cycle between analysis, designs and testings. Knibbe et al. [64] built the Smart Makerspace to guide makers to build devices with immensely instructed environment. FaBrickation [77] presents an approach on speeding up 3D printing process by substituting sub-volumes of devices with standard building blocks. Previous researches [43, 63] demonstrate how to fabricate PCB for rapid prototyping using a standard inject printer.

This work introduces a system designed for makers to build a customized PCB effectively avoiding tedious tasks. The system consists of two parts: (1) TangramScope, a flexible and extensible physical developing toolkit; and (2) TangramFab, a modular PCB layout software. TangramScope allows developers to select required components for prototyping and evaluating the functions of a dedicated device for their application. After the functionalities of devices built by TangramScope reach expectation, developers can easily import the configurations into TangramFab. With a few more customization settings, TangramFab can generates all necessary design files for PCB fabrication.

4.1 Design considerations

Developers usually go through several design iterations before their products reaches their design goals using off-the-shelf developing toolkit (e.g., Arduino or Raspberry Pi) and peripheral components (e.g., sensor modules, actuators, displays, etc.) in early stages. To better understand the needs in making electronic devices, we conduct semi-structure interviews on 6 participants, who have experience on HCI projects (4 male and 2 female

age from 22 to 33 with computer science or electrical engineering background). We also demonstrated our early prototype with a simple example, a device equips Bluetooth radio and 9-axis motion sensor, to collect their feedbacks and comments. In this section, with lessons learned from our study and observations, we describe the challenges to develop a system that assist rapid prototyping with high quality customized devices.

4.1.1 Prototyping with off-the-shelf developing toolkit

We summarize several issues below, reported by our participants based as their experience to prototype with off-the-shelf developing toolkit.

Robustness

Robustness is the most-frequently reported reason that drives developers to fabricate customized PCB board. Developers complained that off-the-shelf developing toolkit often have issues on poor connections and wires being pulled out while testing, especially when they use jumper wires or cable connectors for devices in mobiles, wearables, and interactive applications. Developers also mentioned that it costs a lot of time on connecting the wires correctly.

Form factor

Most of the off-the-shelf developing toolkits are designed to support general purpose and extendibility, equipped with more components and interfaces to communicate with different external components. The result is that they usually have larger size and heavier weights. In many cases, developers then have to fabricate customized PCB boards with ideal shape and compact size for their applications.

Power consumption

One of our interviewee reported that a device's battery life is a key effect on a long-term user study. Short battery life causes bad user experience, enabling power saving mechanism is important to prolong the battery life. However, off-the-shelf developing toolkits

may not be able to support power saving due to the hardware configurations.

4.1.2 Fabricate customized PCB

To address the above issues caused by using off-the-shelf developing toolkit, developers therefore have to customize PCB to build high-quality electronic devices. However, developers still confront several challenges when designing PCB.

Difficulties in implementations

All of our participants agree that PCB design requires relatively high technical skill and background knowledge, comparing to off-the-shelf developing toolkit. Challenges accounted in different stages of design process are: (1) Besides simply select components that matches their requirements, developers have to spend a lot of time on checking the compatibilities between each components. (2) Different setting on components produce difference functionalities on the same component, e.g., power saving mechanism relies on secondary oscillator and correct power regulation in hardware. Developers may misconfigure their components if they do not have enough information. (3) Correctness of the circuit board is another difficulty, since developers may only notice the errors after the board is fabricated. The entire PCB will not work even there is only a mistake in a single pin or wire. Although there are many PCB design softwares that check design and electrical rules automatically, logical errors are still inevitable. (4) A beginner may get stuck to getting started in PCB design, if he/her doesn't have relevant background knowledge.

Cost

Nowadays, although the price of PCB fabrication service has decreased significantly, the overall cost on time, money and effort is still considerably high for developers. To prevent revisions caused by unexpected behaviors and misconfigurations, it is important to provide physical testing on their PCB designs and generate correct PCB designs afterwards.

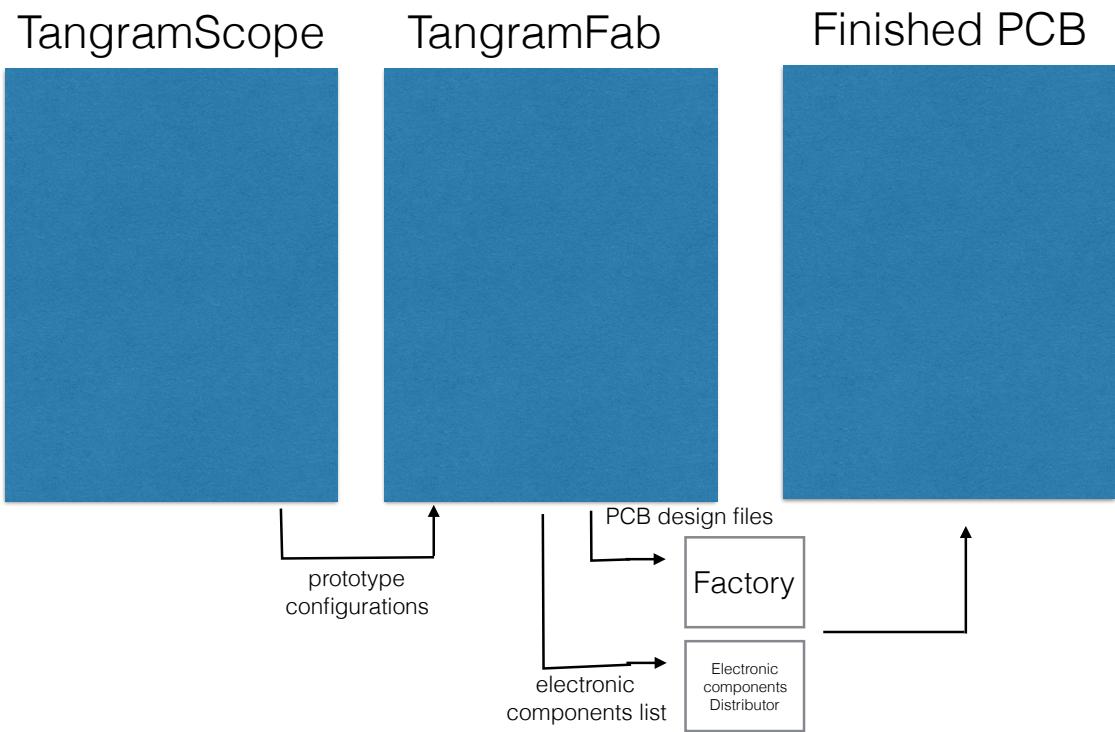


Figure 4.1: Flowchart of the Tangram.

4.2 System design

To simplify the procedure of creating a customized PCB, we propose a service with two parts: (1) TangramScope, a flexible and extensible physical development toolkit; and (2) TangramFab, a web-based PCB layout software with modular components. A Tan is a module that wrap the main electronics components, (e.g., chips), capacitors, oscillators, etc.

Figure 4.1 shows the entire flow of our service. Developers can first easily prototype their devices using TangramScope to evaluate and test their prototypes. After their prototypes conform to their applications, the configurations of their prototypes (e.g., the components used, parameters of the components, connections between components) will be automatically imported to TangramFab. Developers use TangramFab to design the shapes of their boards and arrange the locations of the components according to their needs. TangramFab then exports the standard design files required for PCB fabrication. Finally, by sending the standard design files and electronic components to PCB Manufacturer, the final PCB

is made.

With our system, developers avoid issues when using off-the-shelf development toolkit and difficulties when customizing PCB board in the ordinary way. First, with customized PCB, the device is more robust and can be made into desired shapes. Second, all Tans in our system have no compatibilities problem between Tans and are designed with power saving mechanism. Third, each Tan has only one function. Electronic components that have different function according to their configurations are separated into different Tans with corresponding configurations. Fourth, since the Tans in TangramScope and the Tans in TangramFab differ only from the position of electronics components within and connectors between Tans, the final PCB works exactly like the prototype built with TangramScope. In addition, programs that runs on TangramScope can also run on the final PCB without modification.



Figure 4.2: (a) Tans of TangramScope. (b) Stacking mechanism.

4.2.1 TangramScope

Since there is no chance to turn around after standard format files are sent to factory, most developers prototype their devices with an off-the-shelf development toolkit before designing a dedicated PCB. In this work, we present TangramScope, a flexible and exten-

sible development toolkit that allows developers to easily prototype their electronic device and evaluate their design. We leverage the same design concept as well as BioScope, and extend the application area to mobile and IoT. In addition, the previous code running on TangramScope can also run on the PCB produced by TangramFab without modifications. By programming the prototyped device with the peripheral detection firmware , the configuration of selected Tans can be captured and then output to TangramFab for further PCB layout process.

To demonstrate the design of TangramScope, we implement several sample modules (Figure 4.2(a)) including microcontroller board, Bluetooth radio board, motion sensor, temperature sensor, ECG heart rate sensor, camera, OLED display, microphone, audio DAC, flash storage, RFID reader and Lithium-battery recharger. TangramScope provides the hands-on platform for prototyping that can assist developer to design a customized PCB with correctness in TangramFab. TangramScope adopts the stacking mechanism that developers can easily assemble modular components through compact 30-pin connectors like LEGO bricks. The connectors involve SPI and I2C bus that the modules can communicate to each others, and provides battery power and regulated 3V supply voltage sourced from microcontroller board (Figure 4.2(b)).

4.2.2 TangramFab

TangramFab is a web-based UI that let users arrange the desired Tans to form shapes that matches their applications. Developers can either use information imported by TangramScope or start from scratch with TangramFab. TangramFab is divided into three main parts: the Workspace, the Tan library and the TodoList, as shown in Figure 4.3. The Workspace is where developers situate Tans and draw the dimension of the final PCB board. Tans that are not on the workspace can be added by dragging items from the Tan library below. Key information (e.g., number of pins, bus type, etc.) of the Tans are displayed for developers. The TodoList on the right helps developers get track of missing Tans and warnings from system. After editing, TangramFab will output a BOM (bill of material) file of used components and standard design files for fabrication including Ger-

ber, stencil and pick & place files. The BOM file can be imported to the shopping cart of online electronic components distributor (e.g., Mouser, Digikey, etc) for purchasing required electronic components..

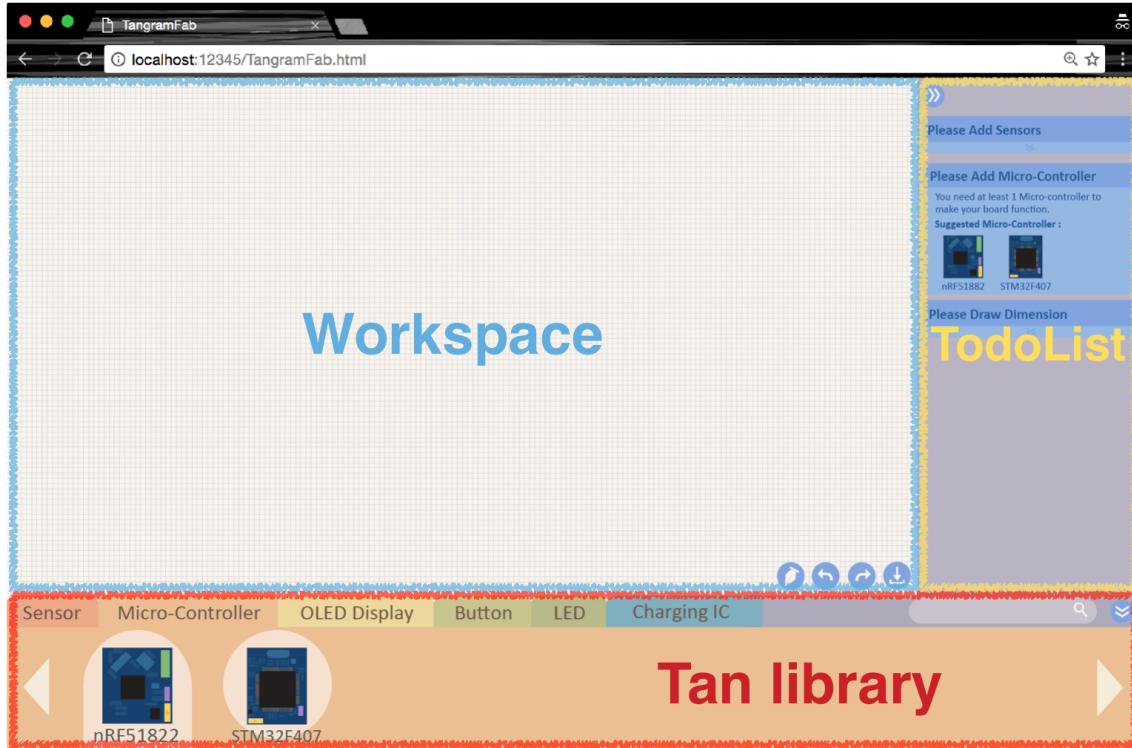


Figure 4.3: Snapshot of TangramFab.



Figure 4.4: Tans imported into TangramFab from TangramScope.



Figure 4.5: Situate Tans to desired positions and draw the dimension.

Sample scenario

Here we have an example to show how developers use TangramFab to produce PCB designs. To build a device with a bluetooth radio (Nordic nRF51822) and a 9-axis motion sensor (InvenSense MPU9250), developers first use TangramScope to build their proto-



Figure 4.6: (a) The toy doll embedded electronic device. (b) The customized PCB.

type and test it with their programs. With TangramScope the settings are then imported into TangramFab as Figure 4.4. Developers can alter the default configurations of a Tan (e.g., type of connector, battery, etc) if needed. Developers then rotate and move the Tans, which is the motion sensor here, according to their needs. After developers draw the dimension of the PCB board as Figure 4.5 and click the export button, the components list and standard files for fabrications are produced.

4.3 Explorative study

We recruited one participant from previous interviews to explore and evaluate the capability of our design. The participant seeks PCB customization for his project, but doesn't have experience in PCB design. The goal is to build a toy doll [73] embedded electronic device (Figure 4.6(a)) that can interact with children to guild children's daily routine without procrastination, e.g., get up and breakfast. He first prototyped his device using Raspberry

Pi, sensor modules, microphone and speaker. In previous interview, he reflected that he is discontented with and robustness, size, weight, battery life and cost of the prototype built by these components. He therefore want to customized the device for addressing these issues.

Based on the interview with the participant, we prepared the TangramScope modules with sample codes according to the requirement of his device, including the microcontroller board, camera module, microphone, audio DAC and RFID reader. After the hands-on prototyping with TangramScope, the configuration is exported to TangramFab for PCB design. With several steps to layout on TangramFab, the system generated PCB design files and sent to factory for fabrication (Figure 4.6(b)).

When the customized PCB has arrived, we conducted a interview the participant to collect feedback. The participant commented: *“As a beginner in PCB design, I feel it is hard to getting started to know how to do part selection. That would be nice to have some examples to see what components they typically used in their applications. . . When I make my circuit board with TangramFab, I had some doubts that can really produce what I want with those simples steps. Although I have yet program my codes on this board, all modules on the board function correctly as I expected with demo codes. I think it will work well in my application. And the finished PCB is much compact and light weight than my previous prototype!”* As the result, our system can assist participant to produce a customized PCB that can really meet the requirements for his application.

4.4 Discussion

4.4.1 Flexibility and Difficulty in PCB Design

Conventional PCB design software provide the flexibility to design any details on the board. Considering the threshold is high for the beginning developers who do not have required skills and background knowledge in PCB design. We design our system with modularization that can simplify the PCB design process by finding a balance between flexibility and difficulty. With our system, developers can build a customized PCB without

undergoing low-level design processes.

4.4.2 Developing a Well-provisioned Library

As a proof-of-concept system, we provide only limited Tans in current TangramScope and TangramFab. A well-provisioned library is a necessary condition to provide good usability of this system for users. The crowd's power would be a possibility to achieve this goal. For example, Arduino has a great community that have contributed a comprehensive library and tutorials that people can easy to access.

4.4.3 Limitation and Future Work

In this work, we studied a small group of participants our system still has room to improve the usability and the limited library. Many participants in our study reflected the importance of supporting users in software programing with our system. Many developers may already familiar with a development tool. TangramFab should be compatible with other development toolkits by designing a system that can transfer the configurations of prototype device for PCB design. For advanced developers who may desire more flexibility to customize their PCB, they can also get benefits in our system. We will provide the function that can export the layout in TangramFab to external PCB design software (e.g., Eagle format in XML [52]) that developers can further edit the details of the layout. Thus, our system can assist they to rapidly generate a reference design that contain all required electronic component to save the time.

4.5 Summary

In this work, we intend to refine the conventional process of building an electronic device by considering and exploring the balance between flexibility and difficulty. TangramScope, a flexible and extensible development toolkit, allows users to build an electronic device by choosing modular components for prototyping their application. Developers can evaluate and test the device built by TangramScope whether the functions are work-

ing properly as they expected. TangramFab, a modular PCB layout software, provides a UI for simplifying PCB layout process. Our explorative study demonstrates how this system can assist users to rapidly prototype their design and easily build the customized PCB.

Chapter 5

Portfolio

5.1 Sensor-Embedded Teeth for Oral Activity Recognition

The human mouth is one part of the human body that is almost always in constant use. We use our mouth to perform some of the most important daily functions, such as eating, drinking, speaking, coughing, breathing, and smoking. Because our mouth is an opening into assessing the health of the human body, it presents the opportunity for the placement of a strategic sensor for detecting human oral activities. For this study, we developed such an oral sensory system, where we explored the use of a small motion sensor embedded inside artificial teeth for the recognition of human oral activities. The detection of human oral activities can enable numerous health care applications, such as food and fluid intake monitoring. Previous research has explored wearable sensory devices installed in various locations of the upper body for detecting human oral activities. For example, Amft et al. [1] used an earphone-attached microphone sensor to record human chewing sounds and detect food types based on their acoustic profiles. Amft et al. [2] proposed another approach that combined surface Electromyography (EMG) and a microphone worn around the neck area to recognize low- or high-volume swallowing actions. BodyScope [3] placed an acoustic sensor around the neck area to recognize different oral activities (e.g., eating, drinking, speaking, laughing, and coughing) by analyzing sounds generated

from the throat area. In comparison, our oral sensory system explores a unique sensor placement not on the human body, but inside the human body, specifically the mouth. Because a sensor placement inside the mouth has the advantage of being in proximity to where oral activities actually occur, this enables our oral sensory system to accurately capture the motion of oral activities. This study presents the design and evaluation of this in-mouth oral sensory system, which uses a small accelerometer sensor embedded inside artificial teeth. Our motivation was based on our observation that most oral activities, such as chewing, drinking, speaking, and coughing, each produce a unique teeth motion. By recording and identifying teeth motion profiles for each oral activity, the proposed oral sensory system builds classifiers that distinguish different human oral activities.

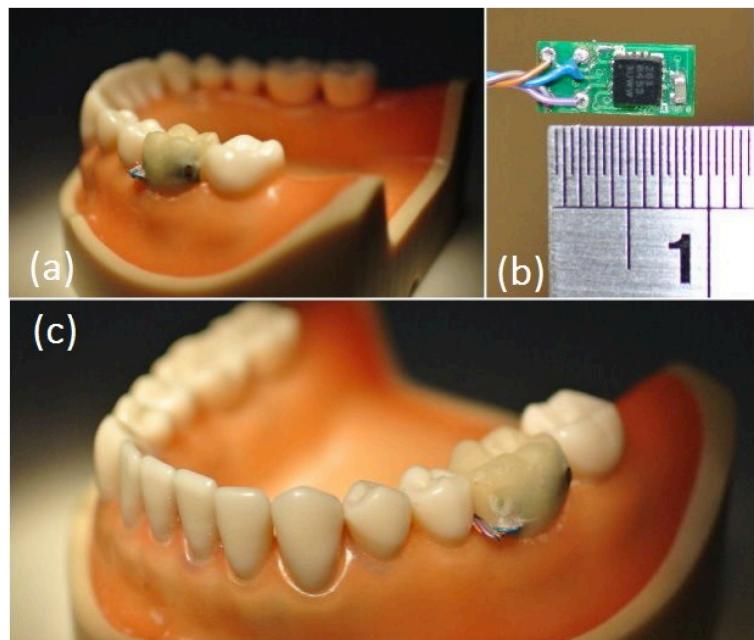


Figure 5.1: The breakout board with (b) tri-axial accelerometer and (a)(c) sensor embedded denture.

5.1.1 System Overview

The system consists of two main components: (a) an oral sensory unit; and (b) oral activity classifiers.

Oral Sensory Unit

Figure 1(b) shows a small breakout board with a tri-axial accelerometer sized 4.5 mm x 10 mm. This small breakout board is sufficiently small to be embedded inside a removable artificial tooth, as shown in Figs. 1(a) and 1(c). To ensure that this sensor board is safe and saliva-proof for human mouth placement, we carefully coated the sensor board with dental resin. In actual system deployment, this sensor board would include a small Bluetooth radio capable of wirelessly transmitting sensor data to a nearby mobile phone for data analysis and oral activity recognition. In the current proof-of-concept system, we have yet to place a Bluetooth radio on this oral sensory unit; therefore, thin wires are used to connect the sensor board to an external data-logging device for data retrieval and power. These thin wires also protect users from accidentally swallowing the sensor units. Oral Activity Analysis Oral activity recognition comprises the following three steps: (1) data preprocessing, (2) feature extraction, and (3) data classification. These steps are described as follows:

Data preprocessing

The sampling rate of the accelerometer sensor is set to 100 Hz. The system first divides the accelerometer data into windows of 256 samples with a 50% overlap between consecutive windows [65]. In each data window, the system extracts the time-domain and frequency-domain features shown in Table 5.1. Because people have different mouth and teeth specifications, sensor orientation can change for different users. Thus, accelerometer readings must be adjusted and calibrated using a rotation matrix. During the calibration phase, the proposed system asks users to hold their head straight and still for a few seconds during the application of Rodrigues' rotation formula to compute this rotation matrix. Each oral sensor unit has its own rotation matrix, which is used to transform its sensor readings from the device's coordinates to real-world coordinates. This normalization procedure reduces the negative effect of errors caused by varying device orientation. Because this normalization procedure reduces, but does not completely remove this error, the system also extracts orientation-independent features based on the magnitudes of x-, y-, and z-axis acceleration. For each sample at time t, its magnitude data is calculated as

$$\sqrt{x_t^2 + y_t^2 + z_t^2}.$$

The normalized (orientation-dependent) feature set and the orientation-independent feature set have different characteristics. The normalized feature set retains separate information on tri-axial acceleration values, which are required for distinguishing activities involving both vertical and horizontal movements. In contrast, the orientation-independent feature set is based on the magnitude value, in which its precision is less affected by changes in device orientation; thus, it is suitable for distinguishing activities that depend on the movement scale.

Feature extraction

Table 5.1 lists the time-domain and frequency-domain features extracted from each data window. Frequency-domain features are computed using the FFT algorithm. Both real and imaginary components of FFT coefficients in the 256-sample window are used to generate 256 features. Overall, the system trains activity classifiers by computing and extracting the following two feature sets: 807 features (269 features per axis acceleration) from the normalized feature set, and 269 features from the device-independent feature set.

Training Classifiers

Our system implements three classifiers: the C4.5 Decision Tree (DT), the Multivariate Logistic Regression (MLR), and the Support Vector Machine (SVM). The SVM classifier uses the radial basis function kernel and one-against-one multiclass classification, and it is further optimized by an additional parameter selection and data scaling. To filter out redundant and irrelevant features, we performed feature selection based on the correlations between features. Low-relevance features with low correlation are filtered out. We adopted principal component analysis [5] as a feature selector, in which the number of relevant features was reduced to 137. For each classification algorithm of the DT, the MLR, and the SVM, we trained two classifiers: person-dependent and person-independent. A person-dependent classifier uses data from all users (i.e., 8 participants in our study) to train a generalized activity model for recognizing the oral activities of different users. Conversely, a person-independent classifier uses 7 users' data to train a specific activity model for recognizing remaining person's oral activities.

	Features
Time-domain	Mean, absolute value mean, maxima, minima, max-min, zero crossing rate, root mean square, standard deviation, median, 75% percentile, inter-quartile range, inter-axis correlation
Frequency-domain	Entropy, energy, FFT coefficients

Table 5.1: Adopted features for oral activity recognition.

5.1.2 Experimental Evaluation

We conducted a laboratory experiment to evaluate the accuracies of different classification algorithms.

Experimental Procedure

Eight users (5 males and 3 females) participated in this experiment. They were asked to install the oral sensor unit inside their mouth while performing each of these four oral activities: chewing, drinking, talking, and coughing. Because it was not possible to customize a removable tooth for each participant, we used dental cement to fix the sensor units to each participant's tooth. For each activity, we collected 15 samples from each participant. Each sample consisted of 2.56 seconds of a specific activity performance. For the coughing data, participants were asked to cough continuously. For drinking data, participants were asked to drink a bottle of water. For chewing data, participants were asked to chew gum or to imitate the action. For speaking data, participants were asked to read a section of an article. We collected 480 activity samples from the 8 participants performing these four oral activities. Person-dependent and person-independent classifications used the same data set collected from the experiment.

Results

We conducted 10-fold cross-validation and leave-one-person-out cross-validation to measure the accuracies of the person-dependent and person-independent classifiers. For the person-dependent classifiers, each round of cross-validation involved using all of each

participant's data for both training and testing. Table 5.2 shows the mean F-measure accuracy results. The SVM (93.8%) classifier outperforms both the DT (52.2%) and MLR (60.5%) classifiers.

For the person-independent classifiers, each round of cross-validation involved using 7 participants' data for training, and the remaining participant's data for testing. Table 5.2 shows the mean F-measure accuracy results. Again, the SVM (59.8%) classifier outperformed both the DT (40.8%) and MLR (55.9%) classifiers. Reasons for the low-accuracy result in the person-independent classifier are as follows. First, because people's teeth and mouth structure are different, their sensor placements are also (slightly) different, thus creating variations in the motion data. Second, people perform oral activities differently; for instance, some people chew or talk faster, slower, harder, or softer. It is possible to improve the accuracy of person-independent classification by extending the training set to include different sensor placements and oral activity types.

Oral Activities	Person-dependent Classification Accuracy		
	C4.5 DT	MLR	SVM
Coughing	62.5%	62.9%	94.3%
Drinking	74.5%	70%	94.9%
Chewing	34.8%	57.5%	92.9%
Speaking	37.9%	56.8%	93.1%
Overall	52.2%	60.5%	93.8%

Table 5.2: F-measure accuracy of oral activity recognition with a person-dependent classifier.

5.1.3 Discussion

This is a feasibility study of an oral sensory system that detects human oral activities. We identified the following challenges for the proposed oral sensory system.

Data logging and transmission

If an application does not require real-time monitoring, sensory data can be temporarily stored on the sensor device. When users remove their artificial teeth, for example, for

Oral Activities	Person-independent Classification Accuracy		
	C4.5 DT	MLR	SVM
Coughing	34.6%	50.7%	49.7%
Drinking	72.1%	86.5%	78.9%
Chewing	36.9%	61.9%	50.3%
Talking	19.6%	24.6%	60.5%
Overall	40.8%	55.9%	59.8%

Table 5.3: F-measure accuracy of oral activity recognition with a person-independent classifier.

disinfection and storage, small electrodes on the surface of the artificial teeth are used to connect to the sensor board and retrieve stored sensor data. For a real-time monitoring application, the sensor board must have a low-power wireless radio (e.g., Bluetooth) to transmit sensor data to a nearby smartphone for data analysis and activity recognition. Another possible data transmission medium is intra-body communication [60], which has lower power consumption compared to wireless radio communication.

Energy

Because users must remove artificial teeth for daily disinfection and storage, we surmise that a recharging and storage station will be required, similar to that of an electric toothbrush, in which users would place the cleaned artificial teeth on this station for battery recharging and data retrieval.

Safety

Because of the sensor placement inside the mouth, the safety concern is paramount. All electronic components must be sealed securely and tightly. In the event that the sensor units are mistakenly swallowed, they will pass the human body without causing any harm. Its safety requirements are similar to those of capsule endoscopy, in which patients swallow a camera pill. Because our current prototype (Figure 5.1) was not considered safe, we attached a safety string to the sensor unit so that participants would not be able to swallow it.

5.1.4 Summary

For this study, we designed and developed an oral sensory system that can recognize human oral activities. Our results from a laboratory experiment with 8 participants demonstrate the feasibility of this oral sensory system in recognizing the following four human oral activities: speaking, chewing, drinking, and coughing. We found that a person-dependent SVM classifier achieved a high F-measure accuracy of 93.8%, whereas a person-independent SVM classifier achieved only an F-measure accuracy of 59.8%. Because the mouth is an opening into human health, this oral sensory system has the potential to enhance exiting oral-related healthcare monitoring applications such as dietary tracking.

5.2 Sensing Fork: Persuasive Technology to Improve Eating Behavior using a Sensor-Embedded Fork

Parents often experience situations where verbal persuasion has only limited effectiveness with their children. A particular case is at mealtime when parents are attempting to instill healthy eating habits in their children. For example, parents often say, “*Eat your green food*” or “*Finish up your plate.*” Children, however, often have little or no incentive to comply with these directives. Is it possible to apply computing technology to make the eating behavior training experience more enjoyable for children? Several HCI (Human Computer Interaction) projects have explored persuasive technologies [55] for motivating users in diverse daily activities such as brushing teeth, and avoiding a sedentary lifestyle [47], [70]. We believe that the playful aspects of persuasive technology and interaction are especially applicable to children.

A well-balanced diet is an important factor in maintaining good health. From a broader perspective, dietary education today includes how to eat, how to select ingredients, and dietary culture. Because diet and health are closely related, dietary education is and will continue to be a universally important health topic. In many developed countries, governments have been promoting dietary education by providing dietary information on websites [20, 31] and running dietary classes at schools.

In Japan, owing to concern over increasing unhealthy dietary habits, particularly in unbalanced nutrient consumption, the Dietary Education Basic Law was enacted in 2005. The goal of this legislation was to promote healthy dietary habits such that people could enjoy active and healthy lives [9]. Consequently, most kindergarten schools are required to have dietary education classes in which teachers encourage children to eat diverse types of food. However, it is difficult for many parents and children to continue learning and adopting such dietary activities at home because of the lack of time and knowledge. We believe that a new method for dietary education suited to the mealtime period is needed.

We therefore aim to provide an alternative method for motivating children to improve their eating behavior by enhancing a fork, a basic and simple eating utensil. This sensor-

embedded fork, called a Sensing Fork, in conjunction with a mobile game, called a Hungry Panda, can facilitate dietary education activities for parents and children at home. To encourage eating a variety of food items, this system makes use of food *color*, something children can easily identify and understand. By encouraging children to eat foods of different colors, the system teaches children about eating a balanced meal. This simple color-based food selection strategy is an alternative to more complex nutritional lessons that teach different food groups, their nutrition types, functions and roles. Given its simplicity, some nutritionists and physicians recommend a color-based dietary education method [41], that we also adopted in our system. Japanese nutritionists have also proposed the “Five-colored Balanced Food Hygiene,” which is expected to be adopted to support children’s dietary education [83]. The basic idea is to eat five food colors (RED, WHITE, YELLOW, GREEN, and BLACK) each mealtime. Our color-sensing approach is designed for children following this simple color-based dietary education strategy.

To detect the food color in contact with the fork, we embed a small color sensor at the tip of the Sensing Fork device (Figure 5.2). The Sensing Fork has other sensory and communication components including three electrodes, a motion sensor, a microcomputer, a Bluetooth radio, and a battery. By processing the sensory data, our system recognizes a child’s eating behavior (eating actions and food colors). The sensory data is transmitted to the Hungry Panda game executed on a smartphone. The game is interactive and provides feedback to encourage children to improve their eating behavior.

5.2.1 System Overview

Embedding miniature sensors in utensils enables the detection of natural eating behaviors without introducing new tracking devices and without changing the normal operation and function of the eating utensil. To facilitate interaction with children, the Sensing Fork is designed to work with a mobile device. The Sensing Fork forwards the sensory data to the mobile device. The mobile device then processes the data to recognize the child’s eating behaviors associated with the fork. The Hungry Panda game running on the mobile device interacts with the child according to the recognized eating behaviors.

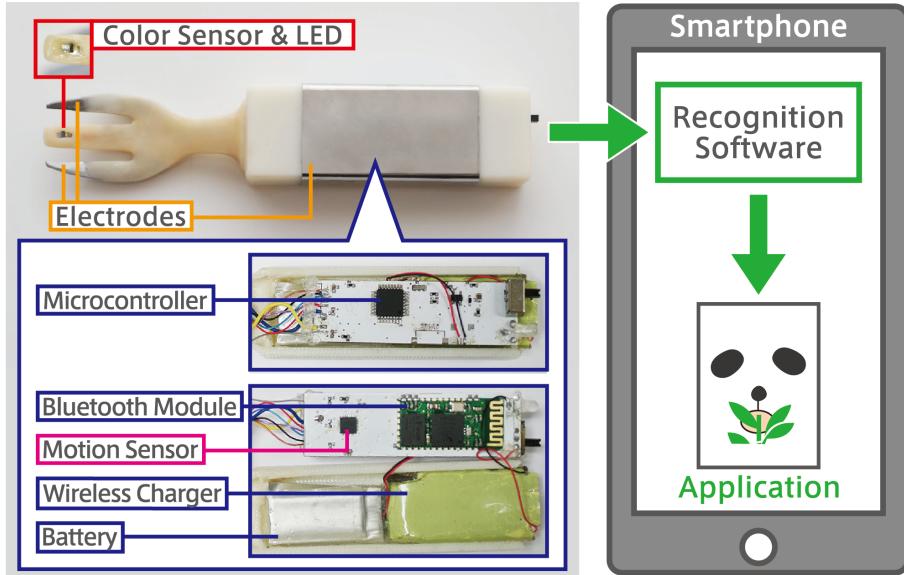


Figure 5.2: System configuration. Left: The Sensing Fork prototype. Right: The software on the smartphone.

5.2.2 Prototype Design: Sensing Fork

Components of the Sensing Fork

Most of the electronic components are packaged and hidden inside the fork grip (Figure 5.2). The controller is an Atmel ATMEGA328P microcontroller. Sensor readings are collected at a rate of ten samples per second and then transmitted to the mobile device through a Bluetooth module. A rechargeable 120 mA lithium battery powers the Sensing Fork. There is sufficient battery power for the fork to run for two hours. Because it must be washed, the fork is waterproof and includes a wireless charging system. The fork grip is $L80 \times W25 \times H15\text{mm}$ in size, making it easily held by small children.

The fork's middle tine contains a tiny breakout board that houses a color sensor and a tiny LED light to illuminate the food contacting the fork. The breakout board is carefully placed between two metal tines and coated with white and transparent resin. The resin is dental grade, ensuring that the fork is washable and safe for eating.

Three-electrode Conductive Probe

Because both food and the human body are conductive, we leveraged two naive voltage dividers with three-electrodes to recognize two eating actions: **Poking** and **Biting**. As shown in Figure 5.2 and Figure 5.3, two metal tines and one metal surface grip are wired

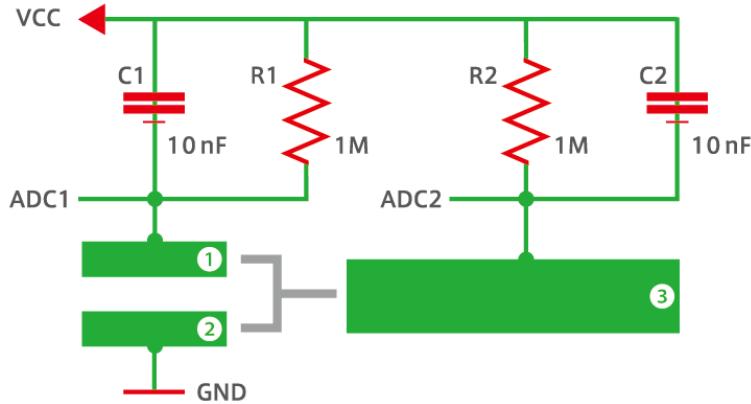


Figure 5.3: Electrodes and resistance sensing schematic. Marker 1 and 2 represent two fork's tines. Marker 3 is the fork's grip.

to the circuit as electrodes. In the case of **Poking**, the measured voltage equals the supply voltage when the two fork's tines (electrodes) do not make contact with food. When the two fork tines do contact food, a circuit is created in which the measured voltage is altered because of the food's impedance. The function for recognizing **Biting** is similar. When a child's mouth contacts food while holding the fork's grip, i.e., the third electrode, it forms a circuit between the fork's grip and tine, the child's mouth, and the hand. The electrical current that passes through the food and child's body is low and safe.

Six-axis Motion Sensor

To capture human motion with the fork, the Sensing Fork includes an InvenSense MPU-6050 that integrates a three-axis accelerometer and a three-axis gyroscope into a single chip. The motion data is used to infer changes to the fork's state.

Single-pixel RGB Color Sensor

The single-pixel RGB color sensor is a TAOS TCS3414 that outputs four-channel color readings (RGB and intensity). It is very small, $L2.1 \times W1.9 \times H0.9\text{mm}$. The color sensor samples the food color when a child pokes the food. A tiny white LED next to the color sensor illuminates the food in case the tine penetrates the food too deeply for the sensor to determine the color.

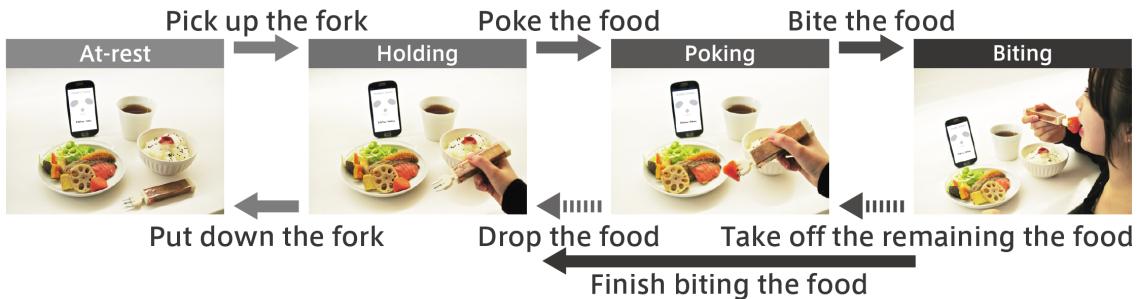


Figure 5.4: State transition diagram of eating actions.

Detecting Eating Actions

The eating action detector analyzes the sensor readings from the fork and determines the state of the fork during a child's eating action. Figure 5.4 presents the fork's four states and describes the method used by the proposed system to distinguish between them.

At-Rest

The fork is **At-Rest** state when it is not being held by a hand. If the fork is the only eating utensil, this state suggests that the user is not engaging in any eating action. The system is able to infer the **At-Rest** state given that the accelerometer and gyroscope sensors report no motion.

Holding

The fork is in the **Holding** state when it is being held by the hand without any food on the fork. This state suggests that the user has lifted the fork and may be deciding what food item to grab with the fork. The system infers this **Holding** state based on the presence of motion from the accelerometer and gyroscope sensors.

Poking

The fork is in the **Poking** state when it is being used to poke a food item. The fork's tines contact the food. This state suggests that the user is in the process of selecting food, but has not inserted it into his/her mouth. The system infers the **Poking** state from two sensor inputs: (1) two-electrode tines measure a non-zero resistance value indicating the presence of food; and (2) the photocell sensor detects food colors. Touching food with the fork triggers the food color recognizer.

Biting

The fork is in the **Biting** state when the fork or food on the fork contacts the mouth. This state suggests that the user is biting or tasting the food on the fork. The system infers the **Biting** state by measuring an altered reading from the fork circuit connecting the child's hand, fork, and mouth.

Classifying Food Types

Our system uses the food's color as the feature to classify the type of food that contacts the fork. Food type classification includes the following four functions: (1) sample selection, (2) extracting feature, (3) training classifier, and (4) online testing.

Sample Selection

Because the fork's sampling rate (10 Hz) is higher than in human's eating actions, our system observes a window of sensor data for each eating action. Consider the case where a child is deciding what to eat next. The child might start by touching one food item before moving on to another, and then another. This can continue until the child sticks the fork tine inside a target food item. Thus, the sensor readings will vary initially and then stabilize when the fork penetrates a food item. To determine this stable window, our system uses a sliding window with a length of five samples and then computes a variance value for this window. If the variance value is lower than the threshold value, our system uses the sensor data in this window for inferring the food type. Note that the fork must also be in the **Poking** or **Biting** state.

Extracting Feature

Four color readings and one resistance value in the sample are used for classifying the food types. The raw color readings can vary between each poking of the same food item. The RGB readings increase linearly with light intensity. Since the ambient lighting may affect the RGB sensory readings, we calculate two-dimensional chromaticity values [40] using the RGB readings as features to mitigate the effect from ambient luminance. Note that chromaticity represents the quality of a color regardless of its luminance. Our system also computes the difference between the RGB values and clear readings as features to train a food detection classifier. When a child is poking a food item, the attached LED

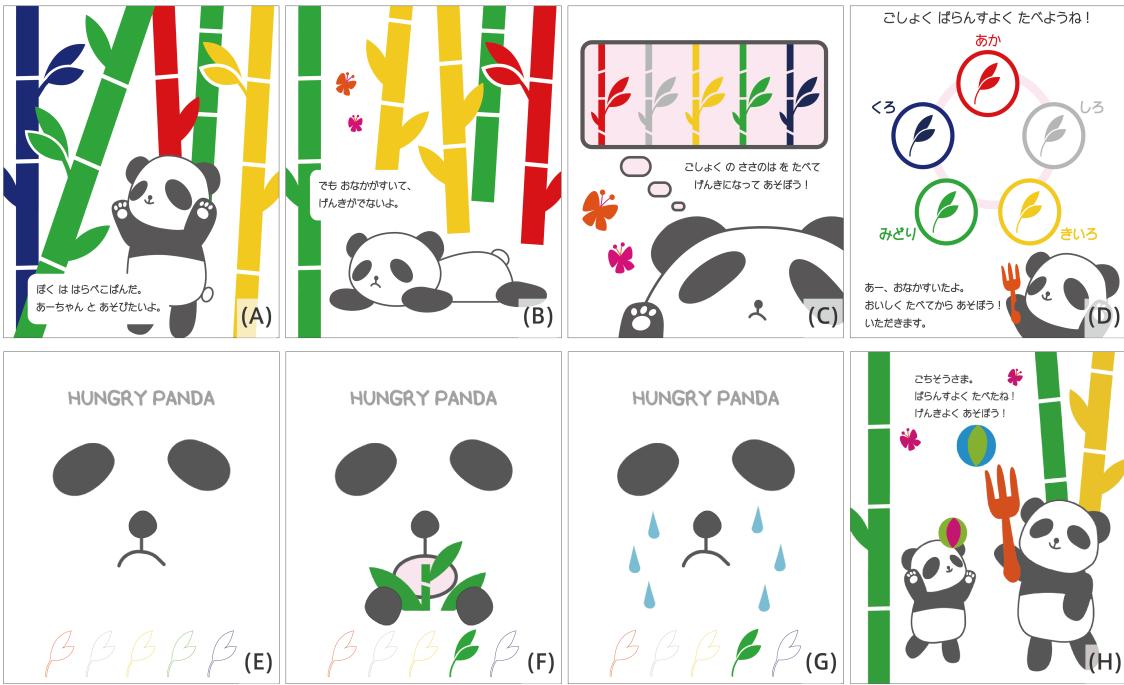


Figure 5.5: The story and flow of Hungry Panda game application.

on the fork tine becomes the primary light source to illuminate the food’s interior color. Based on our experiment, the fork’s LED is primary in detecting food color, whereas the ambient light has only a limited effect.

Training Classifier

A Support Vector Machine (SVM) algorithm is used to train the classifier. The classifier is implemented with LibSVM [46]. The SVM classifier uses the polynomial kernel and one-against-one multiclass classification.

The method used to train the classifier results in differing food classification performance. As the number of food types increases, the performance of the classifier decreases. This is because foods may have similar colors that are ambiguous to the classifier. The classifier can be trained either right before the meal or in pre-training. Training right before the meal (per-meal training) requires the user to collect samples before the meal is about to start. This determines the significant differences between ingredients that are of the same type though not identical. Identical ingredients may still have a slight difference in color. Cooking-style might also change the food’s color. A pre-trained classifier is trained prior to any system deployment with food samples that are commonly eaten by children in the

community. By comparison, a per-meal classifier is trained with food samples collected from a meal immediately prior to consumption. The advantage to using a pre-trained classifier is that it saves the user from needing to train the classifier. The advantage to using a per-meal classifier is accuracy, because the training samples are identical to the testing samples and because every family serves different food or at least prepares the same food differently, i.e., the same food but with a slightly different color. In other words, the per-meal and pre-trained classifier form a tradeoff between accuracy and convenience in real-life use.

Online Testing

In our application, the system tests five consecutive samples determined by sample selection. The classifier tests each sample separately. The system reports the results according to the most occurrences of the tested results from the five selected samples. Real-time classification on the Android platform was implemented using the LibSVM library for JAVA. Our application is able to perform online testing smoothly using an Android phone. Our application classified the food items into five color categories (RED, WHITE, YELLOW, GREEN, and BLACK) according to the “Five-colored Balanced Food Hygiene” [83].

5.2.3 Persuasive Game Application: Hungry Panda

In this study, we focus on improving a child’s dietary behavior, particularly by eating diverse foods with different colors. We developed a persuasive game application, called a Hungry Panda, on a smartphone to be used together with the Sensing Fork. The basic design for the Hungry Panda game has a digital panda imitating the child’s eating behavior, including the eating actions and selected food colors. The rationale for designing it this way is to build rapport and foster sympathy for the panda, so that the child wants to help the panda. Because the panda imitates the child’s eating behavior, the child is presented with an opportunity and a choice to help the panda by improving his/her own eating behavior. The use of a smartphone is a convenient computing and display device for many people in Japan and elsewhere. We use the panda as a virtual pet, because the panda is familiar

to children in most Asian countries. The Hungry Panda game consists of two parts: (1) an interactive picture book for dietary education prior to eating, and (2) an interactive and persuasive game application for collecting points during a meal.

Picture Book for Dietary Education

It is common for parents to read picture books aloud to their children. Such activities are not only important for educating children but also serve to foster good parent-child relationships. Therefore, we designed a short and simple picture book to teach color-based dietary education. Reading this picture book before a meal helps both parents and children to become familiar with the color-based food game.

The game's first screen is shown in Figure 5.5 (A), with a panda greeting the child: "*Hello, (the child's name). I am Hungry Panda. I want to play with you.*" This panda greeting is an attempt to initiate friendship with the child and encourage the child's sympathy for the panda. Figure 5.5 (B) is the second screen that shows a tumbled Hungry Panda: "*However, I do not feel perky because I am very hungry.*" This screen gives a subconscious hint to the child that mealtime is about to begin. Figure 5.5 (C) presents the third screen: "*Let's eat five different colored bamboos to become cheerful.*" This sets the goal for a properly balanced diet of different food color. Figure 5.5 (D) shows the next screen "*Let's play after the meal. Itadakimasu (Japanese expression of gratitude before meals).*" This screen also provides a detailed explanation about the five-colored food items. This short and simple picture book encourages both children and parents to engage in a process that is both fun and entertaining. Because we encourage parents to read the picture book aloud to their children, we removed any vocal narration from the picture book and retain only the background sound effects.

Interactive and Persuasive Game Application

After the picture book introduction, the unhappy panda appears on the smartphone screen and says, "*I am very hungry. I want to eat*" (Figure 5.5 (E)). The panda imitates the child's eating actions, for instance by **Holding** or **Bitting**, and eats the bamboo grass whose

color matches the child's consumed food item (Figure 5.5 (F)). All consumed food items correspond to the five colors.

This game application specifically targets two common eating problems¹. **Picky Eating**: The score (expressed as bamboo grass points and displayed at the bottom of the screen) improves when the user eats food with a different colors. This encourages the child to continue eating to collect more points. At the same time, the child is helping to feed the Hungry Panda. **Distraction**: The system generates various sounds as an alert to the child. For example, when the child stops eating or puts down the Sensing Fork for a certain time period, the panda cries and exclams “*want to eat more*” (Figure 5.5 (G)). When the child picks up the Sensing Fork again, the panda laughs. In the game design, holding the Sensing Fork is not only a way for the child to show an interest in helping, it also encourages the child to concentrate on eating, thereby avoiding distractions. Pressing the finish button produces an animated reward for the child’s good behavior —the higher the points obtained, the more impressive the animation (Figure 5.5 (H)).

5.2.4 Experimental Evaluation

We conducted two types of experimental evaluation on the Sensing Fork using a smartphone. This section describes the results for detecting eating actions and classifying food colors.

Results for Detecting Eating Actions

To confirm the ability to detect eating actions, we recruited six participants (three men and women, aged 21-28) and asked them to eat a certain food item, i.e., a bite-size cube of cheese, five times. We instructed each participant to put the Sensing Fork back on the table to examine the recognition accuracy for the completion of a single sequence in eating actions. Each eating sequence included **Holding-1**, **Poking**, **Biting**, **Holding-2**, and **At-Rest**, in that order. **Holding-2** was the time when the fork left the participant’s

¹ Statistical Surveys conducted by Ministry of Health, Labour and Welfare of Japan in 2005.
<http://www.mhlw.go.jp/toukei/list/83-1.html>

Action	Participant						Avg.
	A	B	C	D	E	F	
Holding-1	100% (0.3)	100% (0.5)	100% (0.3)	100% (0.5)	100% (0.6)	100% (0.4)	100% (0.4)
Poking	100% (0.6)	100% (0.3)	100% (0.9)	100% (0.6)	100% (0.6)	100% (0.5)	100% (0.6)
Biting	100% (0.3)	80% (0.9)	60% (0.9)	80% (1.2)	80% (1.1)	60% (0.7)	77% (0.9)
Holding-2	100% (1.1)	100% (1.8)	100% (1.5)	100% (1.1)	100% (0.5)	100% (1.2)	100% (1.2)
At-Rest	100% (1.5)	100% (1.4)	100% (1.5)	100% (1.5)	100% (1.1)	100% (1.4)	100% (1.4)

Table 5.4: The accuracy of detecting each eating action. The numbers in parentheses denote the average time (sec.) required to detect each eating action.

Prediction \ Actual	Tuna	Tomato	Pumpkin	Spinach	Seaweed	Pork	Carrot	Red pepper	Yellow pepper	Green pepper	Egg	Mushroom(1)	Hamburger	Sweet potato	Broccoli	Mushroom(2)	Rice	Recall (%)
Tuna	45	1	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	90.0
Tomato	2	42	0	0	0	0	1	5	0	0	0	0	0	0	0	0	0	84.0
Pumpkin	0	0	48	0	0	0	0	0	1	0	1	0	0	0	0	0	0	96.0
Spinach	0	0	0	43	3	0	0	0	0	3	0	1	0	0	0	0	0	86.0
Seaweed	0	0	0	5	37	1	0	0	2	5	0	0	0	0	0	0	0	74.0
Pork	0	0	0	0	1	39	0	0	0	0	0	0	10	0	0	0	0	78.0
Carrot	0	2	0	0	0	2	45	0	1	0	0	0	0	0	0	0	0	90.0
Red pepper	1	6	0	0	1	0	0	42	0	0	0	0	0	0	0	0	0	84.0
Yellow pepper	0	0	3	0	1	0	0	0	43	0	0	0	3	0	0	0	0	86.0
Green pepper	0	0	0	4	3	0	0	0	0	41	0	0	0	0	1	1	0	82.0
Egg	0	0	2	0	0	0	0	0	0	48	0	0	0	0	0	0	0	96.0
Mushroom(1)	0	0	0	3	0	0	0	0	0	0	0	46	0	0	1	0	0	92.0
Hamburger	0	0	0	0	1	11	0	0	2	0	0	0	36	0	0	0	0	72.0
Sweet potato	0	0	0	0	0	0	0	0	0	0	0	0	50	0	0	0	0	100
Broccoli	0	0	0	2	1	0	0	0	0	1	0	1	0	5	40	0	0	80.0
Mushroom(2)	0	0	0	0	0	0	0	0	0	0	0	0	1	0	49	0	0	98.0
Rice	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	50	0	100
Precision (%)	93.8	82.4	90.6	75.4	77.0	73.6	97.8	82.4	87.8	82.0	98.0	95.8	73.5	89.3	95.2	98.0	100	

Table 5.5: The confusion matrix of classifying food types. (1)Shiitake Mushroom. (2) Eryngii Mushroom.

mouth. Each participant repeated the above eating sequence five times. All participants were using the Sensing Fork for the first time. We did not describe the system to them in advance.

Table 5.4 shows the accuracy of eating action detection and the average time required for detection. The overall recognition rate was 95%. However, the **Biting** recognition rate was only 77%. **Biting** was falsely detected as **Holding-2**. On observing the falsely recognized **Biting** action on the video, we noticed that the average time taken was only 0.1s, whereas a correctly recognized **Biting** lasted around 0.9s. Thus, when the fork touched the participant’s mouth for only a brief moment, it was incorrectly recognized. This prob-



Figure 5.6: The examples of food items in three cuisines.

lem stems from the limitation in the implementation that set the recognition interval as 0.1s. This interval was set low in order to conserve battery power. We believe that the recognition rate will be much improved by modifying the interval.

As shown in Table 5.4, the average time for detecting the eating state was 0.9s. Notably, however, detecting **At-Rest** required 1.4s. This is because the fork infers it is **At-Rest** when it remains motionless for more than one second. However, in general, actions such as putting down and lifting the fork cannot be repeated at high speeds and therefore this is not considered a problem. Further, **Holding-2** required an average of 1.2s. This resulted from a problem with the experimental program. Specifically, a sample animation of around 2.0s was displayed after the **Biting** action and the state did not update until the end of the animation. The actual detection time is thought to be equivalent to **Holding-1**.

Results of Classifying Food Types

To examine the accuracy in classifying the food types, we collected food items from three different cuisines: Japanese, Chinese, and Western cuisines typical in Japanese households (Figure 5.6). Each cuisine included multiple food types providing a balanced diet.

We collected food samples using the Sensing Fork to perform a poking action with the food item. Based on the three different cuisines, we collected 17 types of food items and 50 samples of each, because the color of the same ingredient could change with the cooking method. In each experiment, we poked the food item ten times. The collected dataset contained 850 samples from the 17 food types. We conducted a 10-fold cross-validation to evaluate the F-measure accuracy of the food-color classifier. Each round of cross-

validation randomly divided all the samples into a training set and testing set. The overall F-measure accuracy was 87.5% in this experiment. Table 5.5 shows the confusion matrix for the result that evaluated classifying 17 food types under the cross-validation. Improper poking caused a few incorrect classifications. Most incorrect classifications were caused by foods with similar colors from the sensor’s viewpoint. For example, pork and ham-burger steak, yellow pepper and pumpkin, tuna and red pepper, and carrot and tomato were similar groupings. We further classified the 17 food types into three categories — Japanese, Chinese, and Western— and evaluated their F-measure accuracy individually. Because individual cuisine has less ambiguous food types, the accuracies of the Japanese, Chinese, and Western cuisines were 94.7%, 96.2%, and 93.8%, respectively. Samples from the three cuisines were also used in our user study with the children. The food samples were not only used to evaluate the performance of the Sensing Fork but also to train the classifiers used in the Hungry Panda game during the user study.

5.2.5 Limitations

Food Classification: Pre-trained or Per-meal Classifier: A pre-trained classifier provides low food-classification accuracy but requires no user effort to train the classifier. A per-meal classifier provides high food-classification accuracy but requires some user effort to train the classifier. In consideration for the parent, we adopted a pre-trained classifier in the real-life user study —hence the negative feedback from the mother respecting classification error. To reduce the effort involved in training a per-meal classifier, we have created a step-by-step tool in the application with which a parent can poke each food sample in his/her child’s meal multiple times using the Sensing Fork. Upon completion, the application will automatically send the food-sample data for training and build a meal-specific classifier.

Fork, Spoon and Other Utensils:

To eat food such as soups or stews, a spoon is a more appropriate utensil than a fork. It is possible to reuse the design for the Sensing Fork and apply it for building other Sensing Utensils. For example, motion sensors embedded in a Sensing Spoon can recognize spoon

actions. A pair of electrodes can be used to distinguish liquid from solid food, because liquids have a lower resistance value than solids.

Game Application Design for Persuasion and Education:

With repeated use, children may become weary of an unchanging picture book and game. However, because the picture book and game are mobile applications, it is relatively easy for users to download new educational content in the picture book and/or update the Hungry Panda game with a new game that a child has not yet played.

5.2.6 Summary

In this study, we implemented a sensor-embedded eating utensil called the Sensing Fork for detecting a user's eating behavior (eating actions and food colors). As a sample application compatible with the Sensing Fork, we designed the Hungry Panda game targeting children's eating behavior. The aim of the Hungry Panda game was to address children's eating issues, including picky and distracted eating. Embedding the sensing technology into a conventionally-used item is more acceptable to the children if persuasive education tools are included in daily life.

5.3 Enabling Daily Self-monitoring to Assist Recovery from Drug Addiction through a Phone-based Support System

Drug addiction has long been recognized as one of the most challenging social problems facing society. Drug use poses health risks, such as drug over-dose and exposure to HIV infection and other blood-borne diseases, and also leads to crime and poverty. Seriously drug-dependent patients commonly are forced to enter rehab centers for treatment; however, individuals returning to daily life face renewed temptations that often lead to relapse. Previous studies [17] have reported that 40% to 60% of the patients leaving rehabilitation programs relapse. An on-going support system is often necessary to help patients exert control over their behavior, increase their self-efficacy, and encourage them in maintaining abstinence. Support systems, such as cognitive behavior therapy (CBT), involve the self-monitoring of one's own drug-use behavior. This self-monitoring approach enables users to track and assess their progress, enhance their awareness regarding the need for behavior changes, and give them the confidence required to overcome addiction.

Several monitoring systems and screening kits have been developed for the detection of drug usage; however, they have a number of shortcomings, including: (1) the need for costly specialized equipment, (2) limited to specific environments, (3) vulnerabilities to cheating (4) lack of personalized data collection and feedback support. No existing system is able to overcome all four of these shortcomings. The Alere DDS MobileTest System [42] uses a specialized handheld device with a test cartridge to screen for drug usage; however, this system costs several thousand dollars and is tailored to law enforcement for the roadside drug screening of drivers. The intelligent fingerprinting system [18] is a portable device capable of analyzing a tiny quantity of sweat from fingerprints in detecting traces of drug metabolites. It essentially minimizes the likelihood of cheating; however, the costly device is several thousand US dollars [27]. Numerous cost-effective FDA-approved drug test kits are now available in grocery stores. Most of them use membrane strips [24, 23] coated with chemicals that react to the presence of drug metabolites in

urine or saliva. Unfortunately, the collection of urine samples requires access to a private bathroom and saliva-based systems are susceptible to cheating, when performed anywhere except under supervision in a drug test center.

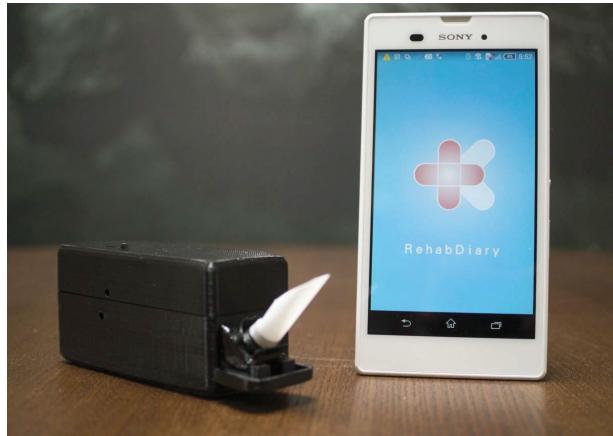


Figure 5.7: Proposed system for daily self-monitoring of drug use, comprising saliva screening device and mobile smartphone.

The RehabDiary system (Figure 5.7) proposed in this study leverages the capabilities of smartphone to reduce the system costs, thereby making the devices available to even those from a low socio-economic background. Drug testing can be also conducted anywhere and the built-in smartphone application provides a cheating-prevention mechanism as well as personalized data collection, progress-based feedback, and motivational support. Furthermore, the proposed system provides the same degree of accuracy as any test strips used in existing FDA-approved drug test kits.

In this study, we targeted a schedule III controlled substance [10], ketamine, which poses a moderate to low potential for physical and psychological dependence and is an emerging drug of choice among students, particularly in East and South-East Asia [30]. In 2010, ketamine users accounted for 1.7% of the general population of the UK, making it the fourth most popular of the so-called club drugs [76]. Note that our system design is generalizable to self-monitoring other types of drugs, such as amphetamines, cocaine, ecstasy, marijuana, opiate, etc., which have inexpensive test strips.

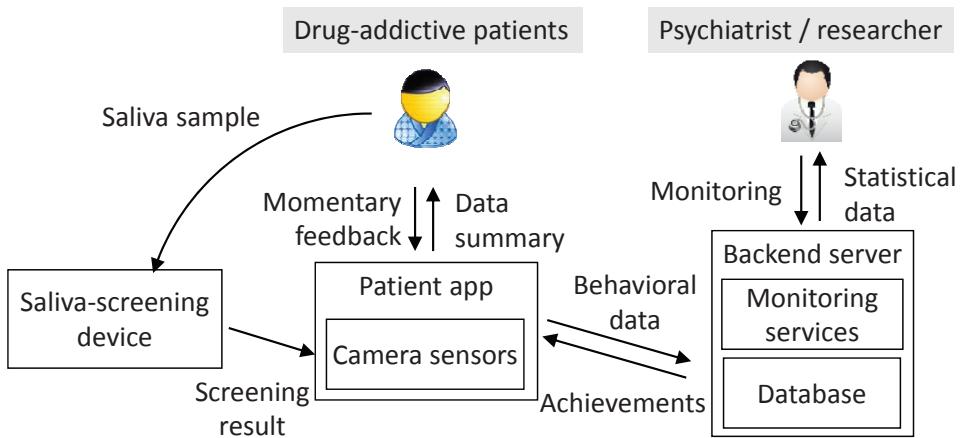


Figure 5.8: The system architecture.

5.3.1 System Overview

Figure 5.8 shows the RehabDiary system, which consists of a portable Bluetooth saliva-screening device, a smartphone app, and a backend server. Phones that sense participants' drug-use behavior. Patients start a saliva screening test by inserting a replaceable cassette into a customized screening device. Once the patient initiates a saliva test, the phone carried by patients installed with the RehabDiary phone application automatically connects with the screening device via Bluetooth and examines the validity of the cassette. To make sure that it is the patient and not someone else taking the test, the patient is required to show his/her face using the screening device in front of the phone camera. After the strip is soaked by patient's saliva, the app waits ten minutes for the test strip to absorb the saliva, during which time the patient can browse through the app's psychiatrist-provided coping skills. At the end of 5 minutes wait, the phone app identifies line patterns appearing in the reaction zone of the strip and notifying patients of the results, which are also uploaded to a backend server for future analysis. These components are described as follows.

Saliva-screening Device

Figure 5.7 shows the saliva-screening device, which facilitates patients to collect their saliva and screens patient's ketamine use. Before each test, patients simply plug in a replaceable and single-use test cassette, which contains a saliva test strip [22]. After the

strip interacts with absorbed saliva, the camera sensor inside will photograph the image of the strip. All images will send back to the phone to further analyze the screening results appearing on the strip.

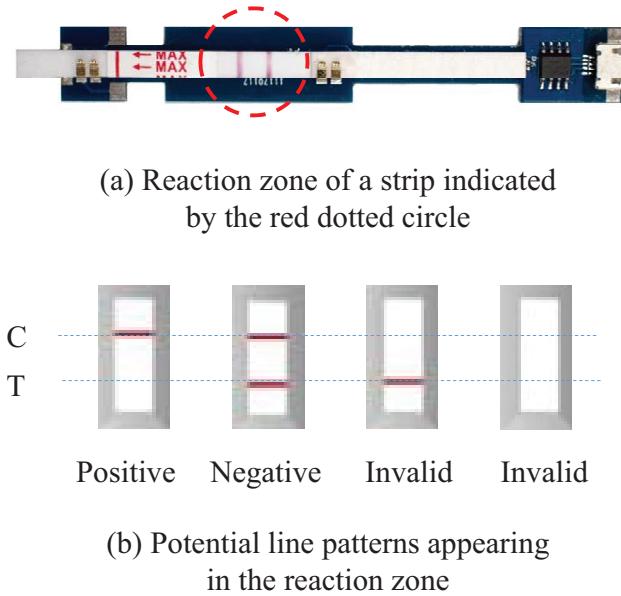


Figure 5.9: The reaction zone of a strip and its potential line patterns. “C” (“T”) means the control (test) line of the pattern.

Phone Application

The RehabDiary phone application (1) enables ketamine-dependent patients to self-monitor ketamine use with saliva tests, (2) summarizes their progress in maintaining abstinence, and (3) incorporates face recognition techniques to prevent cheating. To increase patient awareness of ketamine use behavior through self-monitoring, this application processes the photos of the test strip sent from the screening device to detect if the test outcome is negative (two lines), positive (one control line), or invalid (no control line) in Figure 5.9(b). To visualize patients’ progress of ketamine abstinence as an immediate feedback of their daily usage, the application shows the phone app notifying patients of the results, which are also uploaded to a backend server. Since this patient population tends to cover up their drug use with lies, we ask patients to take multiple photos of their frontal faces. The application will compare the features of the frontal face with those collected when setting up the application.

Backend Server

The backend server records test results and patients' feedback sent from the phone application. These records can be further summarized for psychiatrists to review or sent back to patients' phones for their progress assessment and encouragement.

5.3.2 Saliva Screening Design

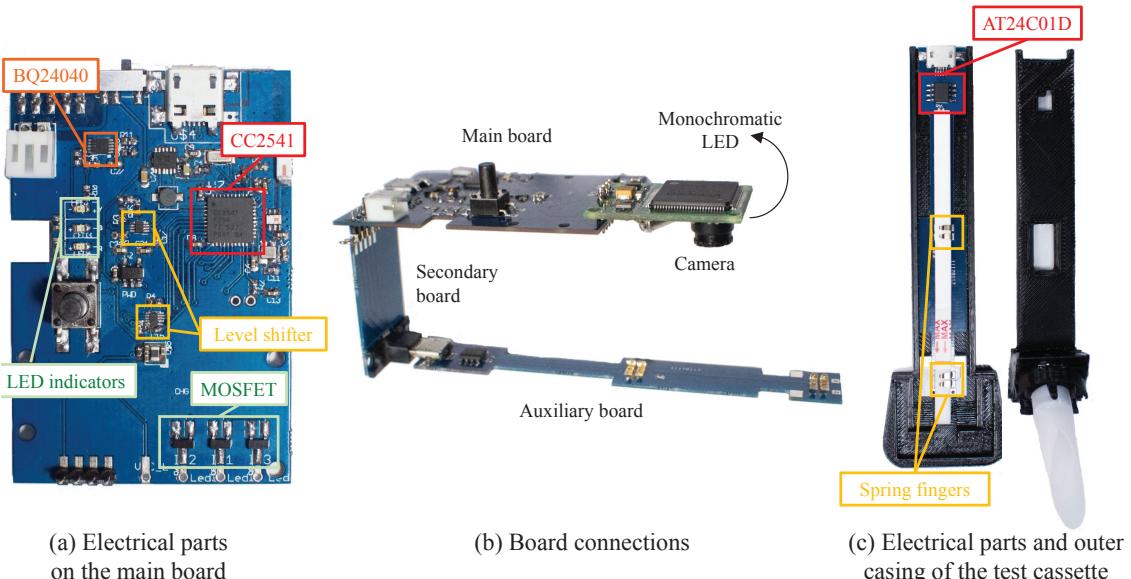


Figure 5.10: Electrical components on the circuit boards and connections between boards.

The proposed saliva screening system comprises two components: a saliva-screening device and a smartphone application. Excretion of ketamine into the saliva enables detection even one or two days after ingestion [22]. The app reminds the patients to perform at least one saliva test per day to enable the screening of ketamine before it is metabolized by the liver. In the following, we describe the hardware used in the collection of saliva samples for the screening of drug metabolites using off-the-shelf saliva strips [22] loaded into a replaceable test cassette. The cassette sends data to the smartphone via Bluetooth for processing by the app. In the following, we outline the design of the hardware and software components, including hardware, firmware, and the smartphone application.

Portable Saliva-screening Device

Figure 5.10 illustrates the proposed saliva-screening device, comprising a *portable* screening module and a *replaceable* cassette that holds test strips. After spitting saliva into a straw for drug-use screening, the test cassette can simply be disposed of, without the need to clean the device.

Portable screening module:

Figure 5.7 presents a side-view of our portable screening module into which the user inserts a cassette holding the test strips. The screening module tracks the operational state of the inserted cassette and captures a photograph of the line patterns that emerge in the reaction zone.

The main board (Figure 5.10(a)) is connected vertically to the secondary board (Figure 5.10(b)) through a 5-pin header to enable tracking of the operational state of the cassette, such as whether the strip in a cassette is ready for detection or whether it has been previously used. When a cassette is inserted into the screening module, the device interfaces with a small memory device in the cassette in order to retrieve the unique identification number associated with each cassette. An off-the shelf camera module is mounted in a fixed position within the device to enable the capture of a photographic of the the line patterns emerging in the reaction zone of the test strip. Monochromatic light emitting diodes (LEDs) surround the lens of the camera to function as a light source to illuminate the reaction zone.

Figure 5.11 presents images of the reaction zone captured by the camera when illuminated using monochromatic LEDs of (a) green, (b) blue, (c) red, and (d) white, showing luminescence values as measured in the R, G, and B channels. In the RGB color model [56], subtractive colors are seen when pigments in an object absorb particular wavelengths of white light while reflecting the rest. The placement of a Red, Green, or Blue filter in front of the image sensor does not eliminate all wavelengths outside the monochromatic spectral region [68]. This results in the generation of cone-shaped spectral sensitivity curves, colored conjugated antibodies (the reagent) [51] interact with then chemical coating in the control line or test line. An improper light source would reduce the likelihood of correctly

classifying line patterns as being positive or negative when few amount of reagent accumulates in the test line. Thus, in this study, a green LED provided the clearest rendering of the test strips.

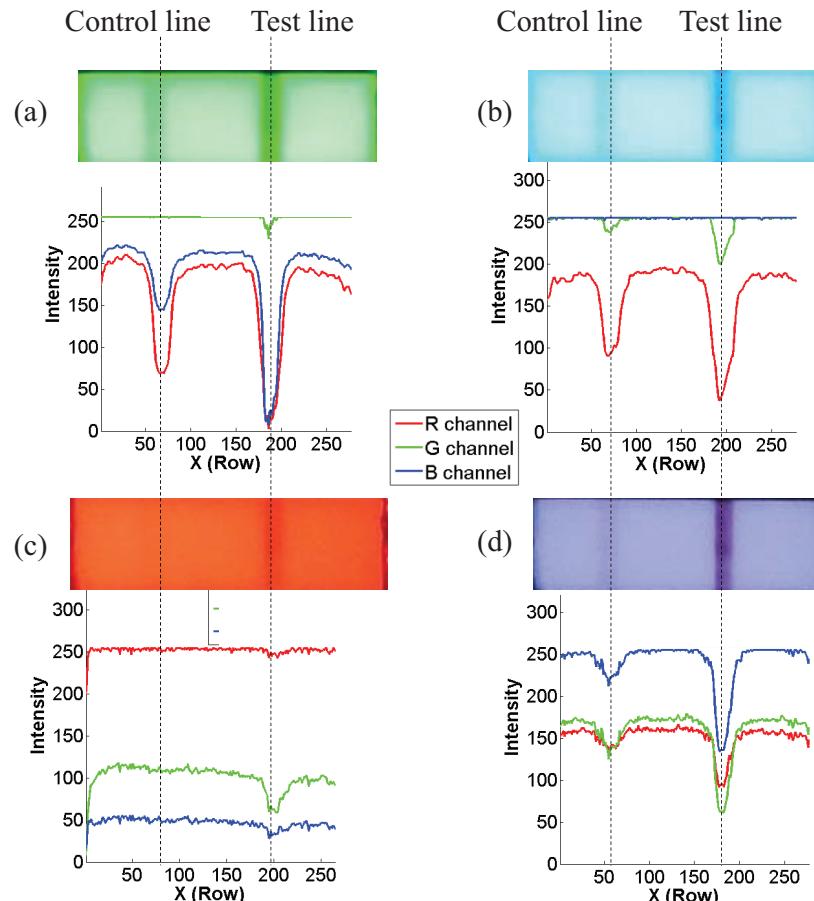
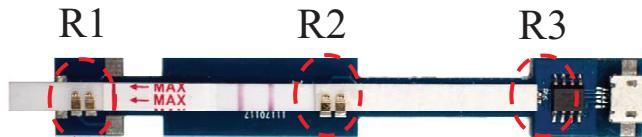


Figure 5.11: Image of reaction zone illuminated using monochromatic LEDs: (a) green, (b) blue, (c) red, and (d) white. The RGB readings across the middle row of pixels in the image are plotted as a line chart below each image.

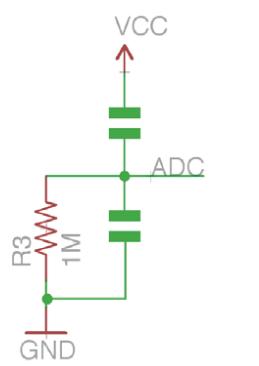
Replaceable test cassette:

Figure 5.10(c) presents a replaceable test cassette with an embedded auxiliary board enclosed within upper and lower cases. During each drug test, the patient spits saliva into the cassette, after which it is simply thrown away. Because the units are meant to be disposable, we used inexpensive components for the nonvolatile memory, micro-USB connector, spring fingers [13], and testing strip and adopted surface mount manufacturing techniques for assembly. Non-volatile EEPROM memory is also mounted on the board to store the unique ID for each cassette. As shown in Figure 5.10, two pairs of spring fingers are

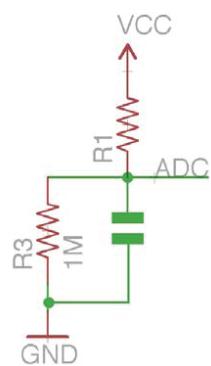
placed at the front and back end of the strip to hold the absorbent cotton wick of the test strip, thereby forming an electrical component with infinite resistance (Figure 5.12(b)). These two equivalent resistances (R_1 and R_2) and a high load resistance ($R_3 = 1\text{M}$ Ohms) form a voltage dividing circuit. By interfacing with the main board through the micro USB connector, the microcontroller makes it possible to obtain data from the cassette.



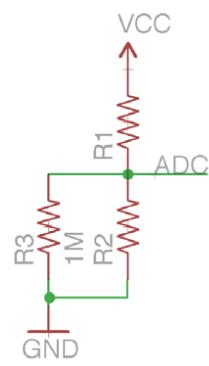
(a) A strip on the auxiliary board



(b) Dry strip



(c) Spit saliva (R_1 decreases)



(d) The reaction zone soaked (R_2 decreases)

Figure 5.12: Auxiliary board with test strip and three equivalent circuits. When the strip is dry, the equivalent circuit (b). Saliva samples are absorbed by the front cotton wick resulting in equivalent circuit (c). When the back cotton wick is also soaked, the equivalent circuit is in (d).

Cassette state sampling:

The status of each test cassette is monitored by relaying data related to the saliva absorption and screening to ensure their integrity. These actions are performed after receiving state transition commands from the app. Briefly, the device retrieves the ID number of the cassette, samples the voltage of the dividing circuit, and captures an image of the reaction zone.

ID retrieval: Unique ID identifiers prevent the falsification of saliva samples through the insertion of previously used cassettes. Throughout the test, the device periodically

retrieves the ID number from EEPROM to ensure that the cassette is not removed.

Voltage sampling: After reading the ID number, the screening device engages in continuous sampling of the voltage across R_3 (Figure 5.13) via the USB connector. Figure 5.12 presents three equivalent circuits associated with the cassette. In the case where the strip is dry, the resistance values of the cotton material overlaid on the strip (R_1 and R_2) are nearly infinite, such that the current does not pass R_3 (Fig. 5.12(a)). Figure 9(c) shows the equivalent circuit in which the front end of the strip soaked with saliva. Figure 5.12(d) shows the equivalent circuit in which the front and back ends of the strip are soaked with saliva. Establishing profiles pertaining to the sequences of voltage samples makes it possible for the system to determine whether the saliva sample has been provided correctly

Photo capturing: The microcontroller communicates with the camera using a serial protocol to record the line patterns appearing in the reaction zone of the test strip. Following the collection of saliva and confirmation that both ends of the strip are soaked, the app waits for 5 minutes to enable a reaction with chemicals embedded within the reaction zone of the test strip. The microcontroller then sends a command to the camera to acquire a photo of the reaction zone after receiving a shooting command from the app on the smartphone. Finally, the ID, voltage samples, and captured images are sent to the app for processing.

Line pattern classification pipeline:

After receiving images from the screening device, the smartphone app performs analysis in three steps: (1) window localization, (2) control pattern validation, and (3) test pattern classification.

Window localization: In this step, pixels within the pattern window are identified in order to locate the reaction zone of the strip. Manufacturing the upper casing of each test cassette using a 3D printer leads to differences in the expansion and contraction of the PLA material, which can alter the location of the pattern window by $5 \sim 10$ pixels. Thus, we included a window localization step to frame the exact position of the pattern window. The color contrast between the pattern window (green) and the outer casing (black) is significant; therefore, the pattern window can be identified by selecting light-colored pixels in the estimated locality of the pattern window. In this manner, we can

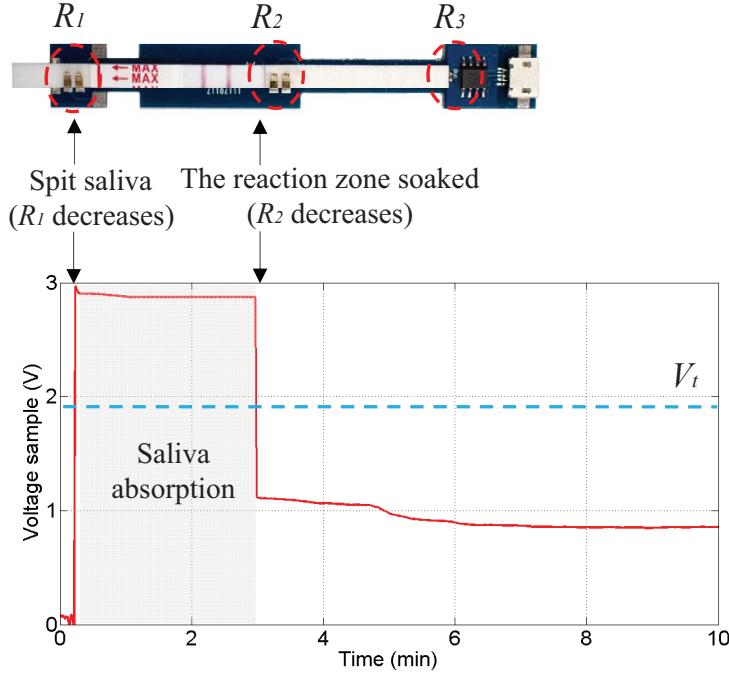


Figure 5.13: Voltage samples of determining saliva test progress.

determine the location of the pattern window by finding the upper-left and lower-right corners, i.e., $\{x_{ul}, y_{ul}\}$ and $\{x_{lr}, y_{lr}\}$, of a box bounding the over-exposed pixels.

Control pattern validation: The proposed algorithm scans one half of the reaction zone row-by-row in order to validate the screening results. No control line emerges when the screening result is invalid. Based on the location of the pattern window, we can extract each row within the pattern window by searching for a dip in the intensity of the R channel in order to identify the control line. After reviewing all of the patterns collected, we determined that the intensity of the control line was always lower than that of the test line. Figure 5.14 plots intensity in the R channel across one row of pixels. Based on this sharp control line, we designed a heuristic detection algorithm for the detection of the control line in each row. We first search for a significant decrease in intensity in the control line by identifying the position of the lowest dip in each row. The location of the control line can then be pinpointed by calculating the average positions of all dips, with the average column of all dips. Once the location of the control line has been determined, the region of the test line (test region), can be located by assuming that the distance between the control and test lines is constant. Identifying the lowest dip in the test region makes it possible to

pinpoint any position in the test line. Most of the reagent accumulates along the control or test line; therefore, the curve presents two dips; i.e., a dip in the intensity values of the control and test lines with a flat area between the two dips. To characterize the drop in intensity associated with the two dips, we respectively calculate the average intensity in the control region, test region, and flat area, i.e., I^c , I^t , and I^f , respectively. The *control (test) drop* of the control (test) dip can be defined as the difference between average values of the control (test) line and the flat area, i.e., D^c (D^t). One row is used as a part of the control row only when the dip in the control region is large enough to be differentiated from smaller noisy disturbances; i.e., $D_i^c > S_i^f$. Otherwise, the row is not recognized as a part of the control line. Therefore the score of a i th row can be defined as the follows.

$$V_i = \begin{cases} 1, & \text{if } (I_i^w - I_i^t) = D_i^c > S_i^f \\ 0, & \text{otherwise} \end{cases} \quad (5.1)$$

where V_i is the contributing score of row i as the control line. By summarizing the scores of all the rows, the screening results can be considered valid when the total score is larger than half the total number of rows.

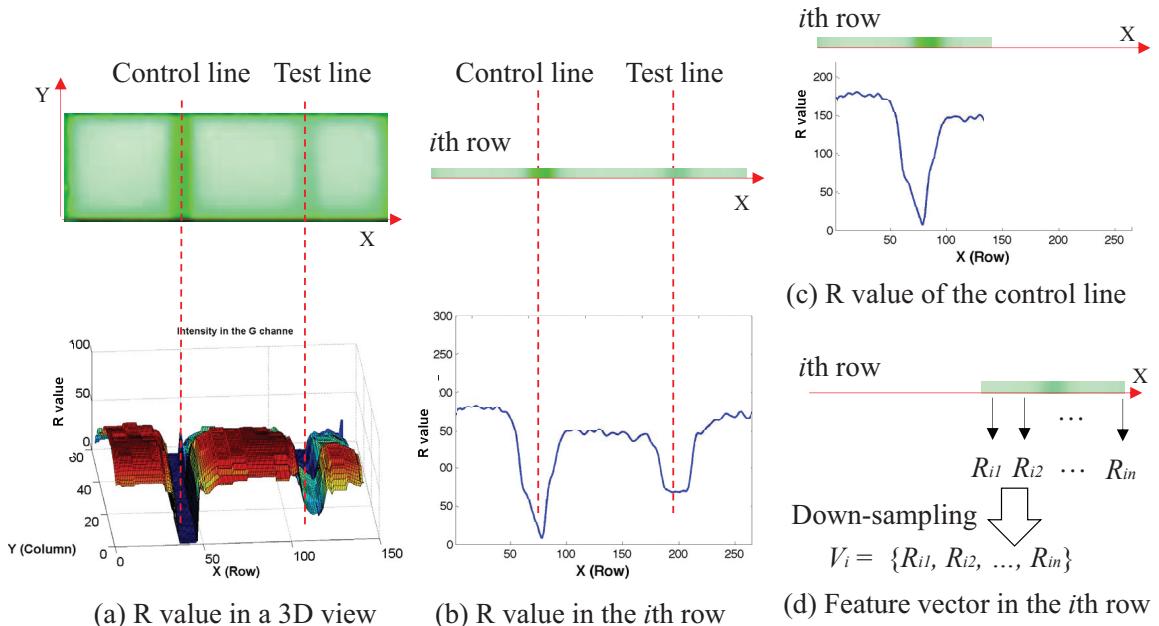


Figure 5.14: The row-by-row intensity analysis while validating the control line and classifying the test line.

Test pattern classification: Following validation of the screening results, the patterns in the test line are classified as positive or negative. The quantity of magenta reagent that attaches to the test line varies according to the amount of the target drug ingested by the patient. The accumulation of the magenta marker influences the intensity of green light reflected back to the camera. The heuristic algorithm used to detect the test line may fail to identify lines with a slight magenta color; therefore, we implemented a learning algorithm to facilitate the accurate classification of all test lines as either positive or negative line patterns.

The system crops out half of the image to prepare the input feature vector associated with the test line pattern in the pattern window (Figure 5.14(d)). The system then down-samples the intensity values in the R, G, and B channels of each row and concatenates all three sequences of R, G , and B values together to form an input vector of the pattern. We also normalize each component using the svm-scale tool [46]. Feature selection is performed on the training set to enable the selection of a feature subset of arbitrary size, which best characterizes the statistical properties of the given target classes based on the ground truth labels. The final step of classification involves the use of a support vector machine (SVM) [46] for the classification of all of the pattern lines as either positive or negative. Using the selected feature subset, the SVM is trained to predict the class label of each line pattern in the test set.

Cassette validation processing pipeline:

Validating the testing operation involves the processing of ID data and voltage samples from the screening device to ensure that the current cassette is unused that the patient has performed saliva collection as prescribed.

ID data is compared with an ID number downloaded from the server to ensure that a valid cassette has been inserted. Verifying that saliva collection is performed correctly involves checking the sequence of voltage samples to ensure that the changes in voltage proceed as expected. Figure 5.13(a) presents the voltage readings recorded during the saliva collection phase. The high resistance of the absorbent cotton wick on the strip means that the resistance between adjacent spring fingers (R_1 and R_2) is infinite, as shown in

Figure 5.12(b). Thus, the voltage across the load resistance (R_3) is pulled down to ground (0V).

Saliva that soaks into the cotton wick slightly decreases resistance at R_1 , which pulls up the supply voltage of the regulator along the conductive path formed by R_1 and R_3 to a value close to 3 volts, as shown in Figure 5.12(c). With the further absorption of saliva into the cotton wick on the other end of the strip results in resistance values at R_2 comparable to that at R_1 , as shown in the 3rd minute in Figure 5.13. Figure 5.12(d) shows the resulting equivalent circuit when the strip is soaked at both ends, in which the decrease in R_2 lowers the equivalent resistance between the sampling point and the ground, which results a voltage drop to below the given threshold (V_t) of 2 volts in this study.

5.3.3 User Interface

The user interface is used to summarize the patients' drug use and summarize their progress in maintaining abstinence. The system is based on various behavioral theories, self-efficacy theory [38] and cognitive behavior therapy (CBT) [88], which are commonly used to help drug dependent patients remain abstinent. We also introduced a mechanism to prevent cheating in order to reinforce the validity of testing. Through self-monitoring, participants show their determination to exert control over their behavior and thereby increase their self-efficacy. The review of drug use logs helps them to recognize drug cravings for which they should develop strategies to cope. The smartphone application can be altered to enable the screening of different drugs as well as the mandatory screening frequency in accordance with the rate at which patients metabolize a given drug. In this study on ketamine, we set the screening frequency at one test each day in order to avoid undue inconvenience. Self-efficacy theory to increase patient awareness of ketamine use behavior through self-monitoring [21], cognitive behavior therapy (CBT) to emphasize triggers and coping strategies to prevent relapses [19], and self-determination theory to internalize the motivation for recovery [9]. The smartphone app (Figure 5.15(b)) provides four functional modules: (1) daily saliva screening, (2) screening results (3) event logging and (4) ranking of risky scenario.



Figure 5.15: User interface of smartphone application: (a) saliva screening, (b) screening results, (c) event logging and (4) ranking of risky scenario.

Classification Accuracy in the Recognition of Line Patterns

To evaluate the accuracy of the proposed system in recognizing line patterns, we recruited patients from the Taipei City Psychiatric Center (TCPC) who were believed to be the most likely to test positive for drug use. This sample included outpatients who visited the hospital to seek treatment for ketamine use as well as inpatients who had recently started receiving treatment. These users commonly attempt to hide their ketamine use; therefore, they were asked to take urine as well as saliva tests in screening for positive samples. Ketamine differs in its effects on urine and saliva as well as in the rate of metabolism; therefore, the results of urine testing were adopted as the ground-truth and the results of saliva testing were accepted only in cases where they agreed with the urine tests. Images of the reaction zone were obtained only when both the saliva and urine tests were positive. Researchers and graduate students who passed urine tests were recruited to provide negative samples for the saliva test. A total of 128 negative samples and 80 positive samples were obtained.

Following the collection and manual labeling of samples, we followed a leave-one-set-out scheme for evaluation. The data was divided into ten randomly distributed groups, with one group serving as testing data and the other nine groups serving as training data. For the classification line patterns, we compared the accuracy of support vector machine (SVM),

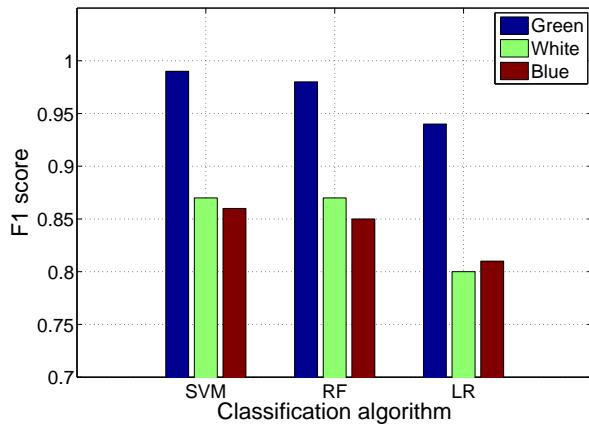


Figure 5.16: F-measure accuracy of three learning classification algorithms (SVM, RF, and LR).

random forest (RF) and logistic regression (LR) ported them to run on the android smartphone. Figure 5.16 presents the mean F-measure accuracy results obtained under green illumination, which indicated that SVM provides the highest accuracy in the classification of line patterns (0.99) followed by random forest (0.87) and logistic regression (.80). Changing the light source to a blue (white) LED reduced the overall accuracy to 0.86 when using SVM, due to a decrease in the *avg_delta*. Our experiment results demonstrate the efficacy of the proposed system in providing robust, accurate results in the screening of drug use.

5.3.4 Summary

This study reports the design and implementation of a cost-effective and cheating-prevention daily drug self-monitoring system. A saliva-screening device transmits data wirelessly to a mobile smartphone for analysis of drug use. The system also uses ID validation and image capture to prevent cheating. In this study, ketamine was selected as a target drug; however, the saliva-screening device could be applied to other drugs simply changing the saliva test strip [24]. The screening device could also be extended as multi-panel screening to deal with multiple drugs in a single saliva test. Our results demonstrate the accuracy and efficacy of the RehabDiary system as a low cost alternative to drug testing.

Chapter 6

Conclusion

Bibliography

- [1] activPAL. <http://www.paltechnologies.com/>.
- [2] Altium Designer. <http://www.altium.com/altium-designer/>.
- [3] Android wear. <http://www.android.com/wear/>.
- [4] Apple watch. <http://www.apple.com/watch/>.
- [5] Arduino. <http://www.arduino.cc/>.
- [6] Arm mbed. <http://www.mbed.com/>.
- [7] Athos. <http://www.liveathos.com/>.
- [8] BLOCKS. <http://www.chooseblocks.com/>.
- [9] Cabinet Office, Government of Japan. <http://www8.cao.go.jp/syokuiku/index.html>.
- [10] Commonly Abused Drugs Charts | National Institute on Drug Abuse (NIDA). <http://www.drugabuse.gov/drugs-abuse/commonly-abused-drugs-charts>.
- [11] Fitbit. <http://www.fitbit.com/>.
- [12] Google Ara. <http://www.projectara.com/>.
- [13] Harwin S7141-45R. <https://www.harwin.com/products/S7141-45R/>.
- [14] Instant Heart Rate. <http://www.azumio.com/apps/heart-rate/>.

- [15] Intel Edison. <http://www.intel.com/content/www/us/en/do-it-yourself/edison.html>.
- [16] LG G5. <http://www.lg.com/us/mobile-phones/g5/>.
- [17] “Mindfulness” Meditation Can Help Reduce Addiction Relapse Rates. <https://ncadd.org/in-the-news/1062-mindfulness-meditation-can-help-reduce-addiction-relapse-rates>.
- [18] Mobile drug testing. <http://www.intelligentfingerprinting.com/product/>.
- [19] Nascent. <http://www.nascentobjects.com/>.
- [20] National Health Service (England). <http://www.nhs.uk/livewell/5ADAY/Pages/5ADAYhome.aspx>.
- [21] Processing. <http://processing.org/>.
- [22] Rapid response signal drug saliva test cassette. <http://www.btnx.com/Product.aspx?id=1997>.
- [23] Saliva Drug Test DrugWipe® S. <http://www.securetec.net/en/products/drug-test/drug-test-saliva/>.
- [24] Saliva drug test with salivascan oral fluid drug test including sponge saturation indicator. http://www.drugcheck.com/dc_salivascan.html.
- [25] Seeed studio. <http://www.seeedstudio.com/>.
- [26] Shimmer. <http://www.shimmersensing.com/>.
- [27] Start Up Hopes to Put Drug Test at Companies’ Fingertips. <http://www.cnbc.com/id/47195398>.
- [28] Strava. <http://www.strava.com/>.
- [29] Under armour. <http://www.underarmour.com/>.

- [30] United Nations Office on Drug and Crime (UNODC). World Drug Report 2010. https://www.unodc.org/documents/wdr/WDR_2010/World_Drug_Report_2010_lo-res.pdf.
- [31] U.S. Department of Agriculture. <http://www.choosemyplate.gov>.
- [32] Vigekwear. <http://www.giayee.com/bluetooth-bracelet/>.
- [33] vitalconnect. <http://www.vitalconnect.com/>.
- [34] Weka. <http://www.cs.waikato.ac.nz/ml/weka/>.
- [35] X2 xGuard. <http://www.x2biosystems.com/>.
- [36] B. Al-Shaikh and S. Stacey. *Essentials of Anaesthetic Equipment*. Churchill Livingstone, 1995.
- [37] A. A. Ali, S. M. Hossain, K. Hovsepian, M. M. Rahman, K. Plarre, and S. Kumar. mpuff: Automated detection of cigarette smoking puffs from respiration measurements. In *Proceedings of the 11th International Conference on Information Processing in Sensor Networks*, IPSN '12, pages 269–280, New York, NY, USA, 2012. ACM.
- [38] A. Bandura. Self-efficacy. *ncyclopedia of human behavior*, 4:71–81, 1984.
- [39] P. M. Barrett, R. Komatireddy, S. Haaser, S. Topol, J. Sheard, J. Encinas, A. J. Fought, and E. J. Topol. Comparison of 24 Hour Holter Monitoring Versus 14 Day Novel Adhesive Patch Electrocardiographic Monitoring. *The American Journal of Medicine*, Oct. 2013.
- [40] A. D. Broadbent. A critical review of the development of the cie1931 rgb color-matching functions. *Color Research and Application*, 29(4).
- [41] G. L. Broadbent, Arthur D. Blackburn and B. A. Waltman. Physician's guide to the new 2005 dietary guidelines: how best to counsel patients. *Cleveland Clinic Journal of Medicine*, 72(7):609–618, 2005.

- [42] M. C., K.-B. T., and L. J. Field testing of the alere dds2 mobile test system for drugs in oral fluid. *J Anal Toxicol.*, 37(5):305–307, 2013.
- [43] V. P. C and D. Wigdor. Printem: Instant printed circuit boards with standard office printers and inks. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology*, UIST ’15, pages 243–251, New York, NY, USA, 2015. ACM.
- [44] D. Camarillo, J. Mattson, M. Flynn, S. Yang, P. Shull, R. Shultz, G. Matheson, and D. Garza. Head contacts in collegiate football measured with an instrumented mouth-guard. *British Journal of Sports Medicine*, 47(5):e1–e1, 2013.
- [45] A. Chan, N. Ferdosi, and R. Narasimhan. Ambulatory Respiratory Rate Detection using ECG and a Triaxial Accelerometer. In *Proc. IEEE EMBS 2013*, pages 4058–4061, July 2013.
- [46] C.-C. Chang and C.-J. Lin. Libsvm: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):27:1–27:27, May 2011.
- [47] Y.-C. Chang, J.-L. Lo, C.-J. Huang, N.-Y. Hsu, H.-H. Chu, H.-Y. Wang, P.-Y. Chi, and Y.-L. Hsieh. Playful toothbrush: ubicomp technology for teaching tooth brushing to kindergarten children. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI ’08, pages 363–372. ACM, 2008.
- [48] P.-Y. P. Chi and Y. Li. Weave: Scripting cross-device wearable interaction. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI ’15, pages 3923–3932, New York, NY, USA, 2015. ACM.
- [49] T. Choudhury, G. Borriello, S. Consolvo, D. Haehnel, B. Harrison, B. Hemingway, J. Hightower, P. . Klasnja, K. Koscher, A. LaMarca, J. A. Landay, L. LeGrand, J. Lester, A. Rahimi, A. Rea, and D. Wyatt. The mobile sensing platform: An embedded activity recognition system. *IEEE Pervasive Computing*, 7(2):32–41, April 2008.

- [50] J. Dacre and P. Kopelman. *A Handbook of Clinical Skills*. Manson, 2002.
- [51] A. Dasgupta. *A Health Educator's Guide to Understanding Drugs of Abuse Testing*. Jones & Bartlett Learning, 2011.
- [52] C. Eagle. Retrieved September 21, 2016 from <https://cadsoft.io/>, 2016.
- [53] S. F. electronic service. Retrieved September 21, 2016 from <https://www.seedstudio.com/fusion.html>, 2016.
- [54] M. Fair. Retrieved September 21, 2016 from <http://makerfaire.com/>, 2016.
- [55] B. J. Fogg. Persuasive technology: using computers to change what we think and do. *Ubiquity*, 2002(December), 2002.
- [56] D. A. Forsyth and J. Ponce. *Computer Vision: A Modern Approach*. Prentice Hall Professional Technical Reference, 2002.
- [57] R. A. Freitas. *Nanomedicine, Vol. I: Basic Capabilities*. Landes Bioscience, 1999.
- [58] Gartner. Forecast: Wearable electronic devices, worldwide. 2016.
- [59] J. Greenbaum and M. Kyng, editors. *Design at Work: Cooperative Design of Computer Systems*. L. Erlbaum Associates Inc., Hillsdale, NJ, USA, 1992.
- [60] K. Hachisuka, A. Nakata, T. Takeda, Y. Terauchi, K. Shiba, K. Sasaki, H. Hosaka, and K. Itao. Development and performance analysis of an intra-body communication device. In *TRANSDUCERS, Solid-State Sensors, Actuators and Microsystems, 12th International Conference on*, 2003, volume 2, pages 1722–1725 vol.2, June 2003.
- [61] B. Hartmann, S. R. Klemmer, M. Bernstein, L. Abdulla, B. Burr, A. Robinson-Mosher, and J. Gee. Reflective physical prototyping through integrated design, test, and analysis. In *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology*, UIST '06, pages 299–308, New York, NY, USA, 2006. ACM.

- [62] B. Hartmann, S. R. Klemmer, M. Bernstein, L. Abdulla, B. Burr, A. Robinson-Mosher, and J. Gee. Reflective physical prototyping through integrated design, test, and analysis. In *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology*, UIST '06, pages 299–308, New York, NY, USA, 2006. ACM.
- [63] Y. Kawahara, S. Hodges, B. S. Cook, C. Zhang, and G. D. Abowd. Instant inkjet circuits: Lab-based inkjet printing to support rapid prototyping of ubicomp devices. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '13, pages 363–372, New York, NY, USA, 2013. ACM.
- [64] J. Knibbe, T. Grossman, and G. Fitzmaurice. Smart makerspace: An immersive instructional space for physical tasks. In *Proceedings of the 2015 International Conference on Interactive Tabletops & Surfaces*, ITS '15, pages 83–92, New York, NY, USA, 2015. ACM.
- [65] K. V. Laerhoven and O. Cakmakci. What shall we teach our pants? In *Proceedings of the 4th IEEE International Symposium on Wearable Computers*, ISWC '00, pages 77–, Washington, DC, USA, 2000. IEEE Computer Society.
- [66] N. D. Lane, P. Georgiev, C. Mascolo, and Y. Gao. Zoe: A cloud-less dialog-enabled continuous sensing wearable exploiting heterogeneous computation. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '15, pages 273–286, New York, NY, USA, 2015. ACM.
- [67] E. C. Larson, T. Lee, S. Liu, M. Rosenfeld, and S. N. Patel. Accurate and privacy preserving cough sensing using a low-cost microphone. In *Proceedings of the 13th International Conference on Ubiquitous Computing*, UbiComp '11, pages 375–384, New York, NY, USA, 2011. ACM.

- [68] V. Lebourgeois, A. Bégué, S. Labbé, B. Mallavan, L. Prévot, and B. Roux. Can commercial digital cameras be used as multispectral sensors? a crop monitoring test. *Sensors*, 8(11):7300, 2008.
- [69] C.-Y. Li, Y.-C. Chen, W.-J. Chen, P. Huang, and H.-h. Chu. Sensor-embedded teeth for oral activity recognition. In *Proceedings of the 2013 International Symposium on Wearable Computers*, ISWC ’13, pages 41–44, New York, NY, USA, 2013. ACM.
- [70] J. J. Lin, L. Mamykina, S. Lindtner, G. Delajoux, and H. B. Strub. Fish’n’steps: Encouraging physical activity with an interactive computer game. In *Proceedings of the 8th International Conference on Ubiquitous Computing*, UbiComp’06, pages 261–278. Springer-Verlag, 2006.
- [71] K. Lorincz, B.-r. Chen, G. W. Challen, A. R. Chowdhury, S. Patel, P. Bonato, and M. Welsh. Mercury: A wearable sensor network platform for high-fidelity motion analysis. In *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*, SenSys ’09, pages 183–196, New York, NY, USA, 2009. ACM.
- [72] H. Lu, J. Yang, Z. Liu, N. D. Lane, T. Choudhury, and A. T. Campbell. The jigsaw continuous sensing engine for mobile phone applications. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, SenSys ’10, pages 71–84, New York, NY, USA, 2010. ACM.
- [73] L.-F. L. T.-W. L. W.-C. H. C.-y. C. R. L. K.-Y. C. M.-h. L. Meng-Ying Chan, Yi-Hsuan Lin and Y.-J. J. Hsu. Wakey: Assisting parent-child communication for better morning routines. In *Proceedings of the 20th ACM Conference on Computer-Supported Cooperative Work and Social Computing*, CSCW ’17. ACM, 2017.
- [74] F. Mokaya, R. Lucas, H. Y. Noh, and P. Zhang. Myovibe: Vibration based wearable muscle activation detection in high mobility exercises. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp ’15, pages 27–38, New York, NY, USA, 2015. ACM.

- [75] F. O. Mokaya, B. Nguyen, C. Kuo, Q. Jacobson, A. Rowe, and P. Zhang. Mars: A muscle activity recognition system enabling self-configuring musculoskeletal sensor networks. In *Proceedings of the 12th International Conference on Information Processing in Sensor Networks*, IPSN '13, pages 191–202, New York, NY, USA, 2013. ACM.
- [76] C. Morgan and H. Curran. Ketamine use: a review. *Addiction*, 107(1):27–38, 2012.
- [77] S. Mueller, T. Mohr, K. Guenther, J. Froehlfen, and P. Baudisch. fabrickation: Fast 3d printing of functional objects by integrating construction kit building blocks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pages 3827–3834, New York, NY, USA, 2014. ACM.
- [78] M. J. Muller. -an exploration in participatory design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '91, pages 225–231, 1991.
- [79] M. J. Muller. The human-computer interaction handbook. chapter Participatory Design: The Third Space in HCI, pages 1051–1068. L. Erlbaum Associates Inc., Hillsdale, NJ, USA, 2003.
- [80] J. Nielsen. Iterative user-interface design. *Computer*, 26(11):32–41, Nov. 1993.
- [81] S. Prototyping. Retrieved September 21, 2016 from <http://smart-prototyping.com/PCB-Prototyping.html>, 2016.
- [82] M. A. Schilling. Toward a general modular systems theory and its application to interfirm product modularity. *Academy of management review*, 25(2):312–334, 2000.
- [83] K. Sugimoto. *Fun and Delicious Five-colored Balanced Food Hygiene: For Preventing and Overcoming of Lifestyle Diseases (in Japanese)*. Footwork Publishers, 1998.

- [84] R. Thompson, I. Kyriazakis, A. Holden, P. Olivier, and T. Plötz. Dancing with horses: Automated quality feedback for dressage riders. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '15, pages 325–336, New York, NY, USA, 2015. ACM.
- [85] S. D. Tripp and B. Bichelmeyer. Rapid prototyping: An alternative instructional design strategy. *Educational Technology Research and Development*, 38(1):31–44, 1990.
- [86] K. Ulrich. The role of product architecture in the manufacturing firm. *Research policy*, 24(3):419–440, 1995.
- [87] D. Van Duyne, J. Landay, and J. Hong. *The Design of Sites: Patterns for Creating Winning Web Sites*. Prentice Hall, 2007.
- [88] N. Volkow. *Principles of Drug Addiction Treatment: A Research-Based Guide (2nd Ed.)*. DIANE Publishing Company, 2011.
- [89] M. Walker, L. Takayama, and J. A. Landay. High-fidelity or low-fidelity, paper or computer? choosing attributes when testing web prototypes. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 46, pages 661–665. SAGE Publications, 2002.
- [90] K. Yatani and K. N. Truong. Bodyscope: A wearable acoustic sensor for activity recognition. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, UbiComp '12, pages 341–350, New York, NY, USA, 2012. ACM.