

CapBand: Battery-free Successive Capacitance Sensing Wristband for Hand Gesture Recognition

Hoang Truong

University of Colorado, Boulder
hoang.truong@colorado.edu

Phuc Nguyen

University of Colorado, Boulder
vp.nguyen@colorado.edu

Qin Lv

University of Colorado, Boulder
qin.lv@colorado.edu

Shuo Zhang

University of Colorado, Boulder
jasonzhang@colorado.edu

Nam Bui

University of Colorado, Boulder
nam.bui@colorado.edu

Kaushik Chowdhury

Northeastern University
krc@ece.neu.edu

Tam Vu

University of Colorado, Boulder
tam.vu@colorado.edu

Ufuk Muncuk

Northeastern University
umuncuk@coe.neu.edu

Anh Nguyen

University of Colorado, Boulder
anh.tl.nguyen@colorado.edu

Thang Dinh

Virginia Commonwealth University
tndinh@vcu.edu

ABSTRACT

We present CapBand, a battery-free hand gesture recognition wearable in the form of a wristband. The key challenges in creating such a system are (1) to sense useful hand gestures at ultra-low power so that the device can be powered by the limited energy harvestable from the surrounding environment and (2) to make the system work reliably without requiring training every time a user puts on the wristband. We present successive capacitance sensing, an ultra-low power sensing technique, to capture small skin deformations due to muscle and tendon movements on the user's wrist, which corresponds to specific groups of wrist muscles representing the gestures being performed. We build a wrist muscles-to-gesture model, based on which we develop a hand gesture classification method using both motion and static features. To eliminate the need for per-usage training, we propose a kernel-based on-wrist localization technique to detect the CapBand's position on the user's wrist. We prototype CapBand with a custom-designed capacitance sensor array on two flexible circuits driven by a custom-built electronic board, a heterogeneous material-made, deformable silicone band, and a custom-built energy harvesting and management module. Evaluations on 20 subjects show 95.0% accuracy of gesture recognition when recognizing 15 different hand gestures and 95.3% accuracy of on-wrist localization.

CCS CONCEPTS

- Human-centered computing → Interaction devices; • Hardware → Power and energy; • Computer systems organization → Sensors and actuators;

KEYWORDS

capacitance sensing, wristband, battery-free

ACM Reference Format:

Hoang Truong, Shuo Zhang, Ufuk Muncuk, Phuc Nguyen, Nam Bui, Anh Nguyen, Qin Lv, Kaushik Chowdhury, Thang Dinh, and Tam Vu. 2018. CapBand: Battery-free Successive Capacitance Sensing Wristband for Hand Gesture Recognition. In *Sensys '18: Conference on Embedded Networked Sensor Systems*, November 4–7, 2018, Shenzhen, China. ACM, New York, NY, USA, 14 pages. <https://doi.org/10.1145/3274783.3274854>

1 INTRODUCTION

Hand gesture has long been used as a form of human-computer interface thanks to its intuitiveness [67]. Existing hand gesture recognition techniques are based on sensors that belong to one of the two broad categories: off-body sensors (e.g., camera [5, 61, 63, 64, 84], RF signals [59], LED or IR light [20, 21, 41, 43, 51, 53]), and on-body sensors (e.g., using pressure [12, 32], EMG [14, 36, 65, 66], impedance tomography [11, 85, 86], accelerometer, and gyroscope [10, 24, 40, 57, 83]). While off-body sensing approaches often yield high accuracy and free users from wearing additional devices, they require infrastructure support that limits users' mobility and versatility. In comparison, on-body sensing approaches provide much better mobility support at the cost of usability. In particular, these wearable sensors often require frequent battery recharging, complex setup, re-calibration, and sometimes, per-use training. For example, Myo [76], a commercially-available, arm-worn hand gesture recognition device, requires users to recalibrate the device every time the user puts it on, and the device needs to be recharged at least once a day [46]. These requirements are the key barriers that limit on-body hand gesture recognition from becoming a mainstream interface.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Sensys '18, November 4–7, 2018, Shenzhen, China

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-5952-8/18/11...\$15.00

<https://doi.org/10.1145/3274783.3274854>

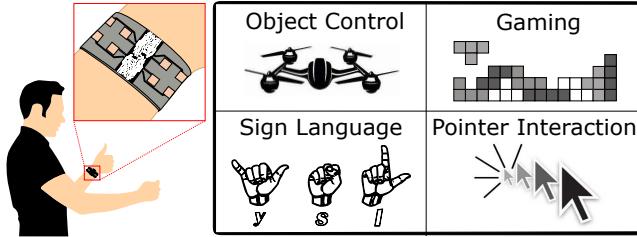


Figure 1: CapBand and application examples.

These limitations are common across hand-gesture recognition wearable devices for two reasons. First, those wearable devices must be able to continuously sense multiple streams of signals in order to capture complex gestures performed by users. As a result, power consumption is typically high. At the same time, these devices' batteries are usually small for aesthetic appearance and convenience of use. Therefore, frequent recharging is needed. Second, per-use recalibration or training is often required because every time a user puts the device on, it may be at a position that is slightly different from the position that the device was trained for.

In this work, we propose CapBand, a battery-free hand gesture recognition wristband to address these two fundamental problems. Such a wristband can benefit diverse applications ranging from remote object control and gaming, to sign language translation, as illustrated in Figure 1. The key challenges in creating such a system are: (1) minimizing the required energy to sense useful hand gestures so that the device can be powered by the limited energy harvestable from the surrounding environment, and (2) ensuring reliable hand gesture recognition without per-use training. Note that users may vary in their wrist size and shape, and the wristband may be positioned differently every time it is put on, making it challenging to design robust hardware and software to perform consistently well across users.

Aiming for limited harvestable energy consumption, we introduce *successive capacitance sensing*, an ultra-low power sensing technique to capture small skin movements on the user's wrist, which corresponds to specific groups of wrist muscles as gestures are being performed. To increase the sensitivity of the system, we design a flexible electronic circuit that contains a capacitance array of 15 pairs of electrodes wrapping around the user's wrist. The device is further molded inside a soft band to improve durability and prevent electronic component exposure. By analyzing human wrist and hand gesture anatomy, we build a *wrist-muscles-to-gesture model*. From this, we develop a hand gesture classification method using both motion and static features. To eliminate the need for per-use training, we propose a *kernel-based on-wrist localization technique* to detect CapBand's position on a user's wrist and instruct the user to adjust the wristband position accordingly. To demonstrate the feasibility of our battery-free scheme, we also fabricate the energy management module to scavenge energy from the surrounding environment. We also provide the open-source, open-hardware release of current CapBand prototype¹.

More importantly, the techniques in this paper, i.e. capacitance sensing under battery-free scheme and customized sensor fabrication process, have high potential in various applications for mobile

and wearable devices. The proposed sensors are not limited to a standalone wristband prototype as presented in this paper. We envision the integration of these sensors into commercial-off-the-shelf smartbands or smartwatches to capture both the dynamic movement of wrists and the delicate movement of fingers. Moreover, the proposed capacitance sensors have great potential to be used in various skin deformation sensing scheme, such as chest/abdominal movement (respiration rate measurement), facial skin jaw, and neck movement (facial gesture detection and silent speech application).

In summary, we make the following key contributions:

- Developing a wrist muscles-to-gesture model that informs the design of the hardware and algorithm to detect hand gestures;
- Introducing a successive capacitance sensing technique for ultra-low-power and highly sensitive detection of small skin movements on the users' wrist;
- Developing a kernel-based on-wrist localization technique that instructs users to adjust the wristband position and eventually eliminate the need for per-use training;
- Prototyping CapBand to confirm system feasibility with a custom-built flexible electronic circuit, a custom PCB hardware, and a custom-built energy harvesting and management module; and
- Evaluating CapBand on 20 participants with 95.3% accuracy of on-wrist localization and 95.0% accuracy of gesture recognition for 15 different hand gestures, and conducting a user study on system usability in terms of responsiveness, robustness, and form factor.

The remainder of the paper is organized as follows. In Sec. 2, we first present the background of the human wrist and hand gesture anatomy, which guides our muscles-to-gesture modeling and leads to our design implications of using capacitance array to sense hand gesture. In Sec. 3, we present an overview of our system, goals, challenges, and high-level solutions for each system module. Following that, Section 4–7 elaborates on each module, including custom sensor and hardware design, the gesture classification model, the on-wrist localization algorithm, and the energy harvesting module. We show the prototype implementation details in Section 8 and provide performance evaluations in Section 9, including on-wrist localization, gesture recognition, power consumption, and user feedback. Section 10 discusses in detail existing approaches and their advantages and limitations. We conclude the paper with a discussion of CapBand's limitations, possible applications, and future improvements.

2 WRIST MUSCLE AND HAND GESTURE ANATOMY BACKGROUND

In this section, we analyze the anatomy of the human hand to identify the relationship between muscle movement and skin surface deformation. We derive a model to link hand gestures (i.e., the firing of different parts of the wrist and hand) with specific muscle group movements, which lead to skin surface deformation. The high-level intuition in distinguishing gestures is that when a user performs a hand gesture, the whole wrist and hand structure (i.e., bones, joints,

¹Open-source, open-hardware release: <https://github.com/tonys2berry/capband>.

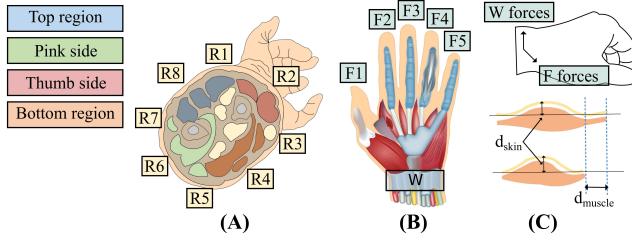


Figure 2: Muscle groups and corresponding muscle movement areas (A and B); contraction forces and muscle contraction to surface deformation (C).

muscles, and skin) change both their patterns and shapes simultaneously. These shape transitions lead to skin surface deformation and affect the measurements of on-body sensors attached to it.

Given a gesture, which muscle groups cause the surface deformation of our interest? The human hand has an intricately complex structure in which approximately 35 muscle groups [19] in the hand and forearm collaborate to perform a hand gesture. Even though these muscles provide our hands with unsurpassed flexibility, extremely precise control, and gripping strength, wrist movements are made of four basic types of movement (without loss of generality, consider the left-hand structure): up, down, left, right. Furthermore, these four basic movements are caused by two fundamental muscle operations: contraction and relaxation.

The muscle groups of interest are those that control each finger and the wrist on the human hand. Specifically, (i) *carpi radialis* moves the thumb up and down; (ii) *digitorum* is the large muscle group that activates pointer, middle and ring fingers as it joins at the middle point of the wrist; (iii) *carpi ulnaris* controls the movement of the pinky; and finally, (iv) *retinaculum* is a special muscle group that wraps around the wrist [73]. Among these muscle groups, there are subgroups that further control the direction of the movements, such as: (i) *the flexor muscle*, a skeletal muscle that contracts to bring the palm toward the forearm; and (ii) *the extensor muscle*, which has the opposite effect of the flexor muscle, opens a joint and increases the angle between the palm and the forearm.

We cluster muscle groups near the wrist area and label the clusters from 1 to 8 as illustrated in Figure 2. As such, muscle groups around the wrist usually belong to one of the following four basic regions (R_1, R_3, R_5, R_7) of interest [35]: (i) *Top region*, top side of the forearm of retinaculum and extensor digitorum; (ii) *Bottom region*, bottom side of retinaculum and flexor digitorum group; (iii) *Thumb region*, abductor pollicis group; and (iv) *Pinky region*, carpi ulnaris group. Last but not least, movements towards regions R_2, R_4 (R_6, R_8) are the effect of *carpi radialis* (digit minimi) muscle group [9].

What are the contributions of different muscle groups to skin deformation? Hand gestures are the results of contracting the aforementioned muscle groups. Intuitively, the direction of a movement depends on the relative force magnitude of different muscle groups. For the purpose of modeling the wrist, we consider the original wrist position as the point in the middle of the intersection cavity between the retinaculum and the digitorum muscles. The total contraction force at that point is the composition of (1) the projection of forces from finger muscle paths denoted as $\vec{F}_{1..5}$, and (2) the projection of forces from the retinaculum muscle group denoted as

\vec{F}_{W_1} for the upper wrist muscle portion and \vec{F}_{W_2} for the lower wrist muscle portion. As an example, when the palm is at rest condition, the wrist (retinaculum) condition is $\vec{F}_{W_1} + \vec{F}_{W_2} = 0$ and the side (ulnar/radial) condition is $\vec{F}_1 + \vec{F}_5 = 0$.

When either of these conditions is violated, the hand will move to the corresponding direction. For example, if $\vec{F}_{W_1} < \vec{F}_{W_2}$, the hand will move according to the flexor activation direction. Thus each contraction force above will have a different weight coefficient to the total movement of the hand where $\vec{F}_{W_1}, \vec{F}_{W_2}, \vec{F}_1, \vec{F}_5$ contribute to the wrist direction and $\vec{F}_{1..5}$ contribute to the finger formation, which will form a muscle group movement decision function:

$$f(\text{gesture}) = \alpha |\vec{F}_{W_1} + \vec{F}_{W_2}| + \beta |\vec{F}_1 + \vec{F}_5| + \gamma \sum_{k=2}^4 |\vec{F}_k| \quad (1)$$

We use Hill's equation [8] to show the relationship between muscle contraction force and muscle distance change:

$$F(t) = f(v)f(l)a(t)F_0 \quad (2)$$

where $f(v)$ and $f(l)$ are normalized velocity and length, $a(t)$ is muscle activation and F_0 is the maximum isometric force.

In short, Hill's muscle model [2] can be transformed to find the muscle contraction as follows:

$$\Delta d_{\text{muscle}} = (b \frac{F_0 + a}{F + a} - b)\delta t_{\text{muscle}} \quad (3)$$

where δt_{muscle} is the duration of muscle contraction; a and b are coefficients that can be found via experimental isotonic records. Changes of muscle contraction distance lead to corresponding skin movement at that region: $\Delta d_{\text{skin}} \propto \Delta d_{\text{muscle}}$. Therefore, if we obtain movements at those 8 corresponding areas, the relationship between hand gesture and wrist shape deformation is:

$$f(\text{gesture}) = \sum_{i=1}^8 \alpha_i \Delta d_{\text{skin}_i} \quad (4)$$

where α_i is the weight coefficient of skin dislocation at position R_i and Δd_i is the distance change at R_i .

Since Equation 1 is a non-convex optimization problem [72], solving the Linear Regression problem of Equation 4 gives the weight coefficients of skin deformation at the corresponding 8 defined areas ($R_{1..8}$) and also gives the corresponding solutions. This observation provides the intuition for preferable positions for our sensor placement (more details in Section 4) and the basic foundation to construct the on-wrist localization algorithm which will be presented in Section 6.

3 CAPBAND SYSTEM OVERVIEW

In this section, we describe the overall design of CapBand, a battery-free successive capacitance sensing wristband system for hand gesture recognition. Our design goals include (1) ultra-low-power consumption; (2) high recognition accuracy for multiple gestures; (3) no need for per-use training; and (4) battery-free usage. To achieve those goals, we propose the CapBand system as illustrated in Figure 3. CapBand consists of four main components including ultra-low-power sensing, reliable gesture recognition, on-wrist localization, and an energy harvesting/managing module.

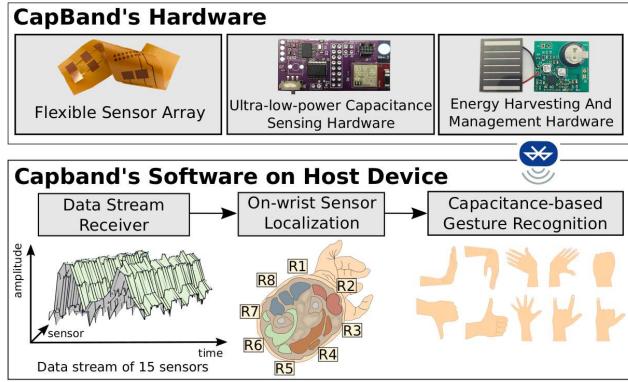


Figure 3: CapBand system overview.

Ultra-low-power sensing. Skin surface deformation due to wrist muscle movement can be detected by multiple sensing modalities including force resistor sensor [32], camera-based [37] or impedance tomography [86]. However, these sensing techniques require high energy consumption and have low sensitivity. In this work, we explore capacitance sensing since it can operate at very low power (at μW level) [22], with high stability, high resolution, and high sensitivity. However, these benefits come with challenges. Since capacitive sensors may accidentally couple with surrounding electrical objects, they are affected by the inherent non-deterministic floating ground and high ambient electrical noise. To overcome this problem, we design two capacitance sensor arrays consistently spaced with explicit connections to the system ground. We fabricate these two capacitance sensor arrays using flexible circuits and embed them in a heterogeneous silicon wristband to minimize the influence of heat and moisture from the surrounding environment (Section 4.1). Detailed implementation is discussed in Section 8.

Reliable gesture recognition. The gesture recognition module needs to be accurate, lightweight, and low latency while using a low sampling rate, in order to meet the system requirements of low-power and real-time operation. We investigate different machine learning techniques and propose a CNN-based gesture classification method using both motion and static data from capacitive sensors. The CNN model can be trained offline, and our evaluations show that the trained CNN model can be efficiently used for online gesture classification. (Section 5).

On-wrist localization. In daily use, the position of the wristband may change every time it is put on or due to user activities. Such changes often shift the sensors to untrained areas and lead to inaccurate gesture recognition. A naive approach is to require the user to perform the training again before every time he/she uses the device. Such *per-use training* is cumbersome and undesirable. To address this issue, we propose an on-wrist localization algorithm to automatically determine the current position of sensors on the user’s wrist. This helps users to adjust the wristband accordingly instead of retraining the model. Furthermore, the mismatched position information may be used for auto-calibration. In this work, we derive a universal coordinate system based on the human wrist and hand muscle structure and design a kernel-based localization algorithm that validates the on-wrist sensor positions (Section 6).

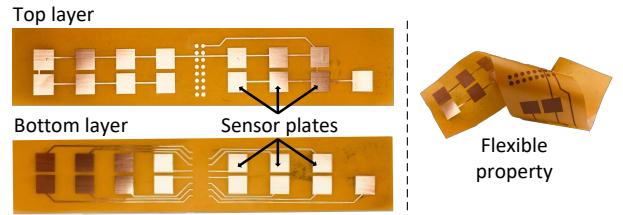


Figure 4: Flexible sensor arrays. Top connects to the system ground and Bottom connects to the measurement pins

Battery-free and transient powering. In order to support battery-free gesture recognition, we face difficulties in (1) harvesting enough energy from a suitable source in the surrounding environment and (2) stabilizing the harvestable energy that is prone to fluctuation due to environmental changes. Thus, we design a custom energy harvesting and a management module which can provide stable supply voltage to the sensing hardware and manage the charge/discharge cycle of the energy storage (Section 7). In addition, we employ solar cell energy harvesting which fits both harvest-and-use and harvest-and-use-later scenarios.

4 LOW-POWER CAPACITANCE SENSING WRISTBAND DESIGN

In this section, we first describe an ultra-low power capacitance sensing technique for recognizing the user’s hand gesture. We then present a design using a flexible sensing array and heterogeneous silicone wristband which accommodates the measurement technique and eliminates the environment noises (e.g., floating ground; heat and moisture condition of user’s wrist). In the following discussion, we present the basic principles and advantages of using a capacitance sensing method to detect wrist gestures. We then present a custom flexible sensor array and silicon wristband holder to increase the sensitivity and to reduce the impact of floating ground to reliably monitor capacitance variations. Last but not least, we design a method to reduce the energy consumption of the system and make it work at ultra-low-power mode.

4.1 Capacitance-based Skin Deformation Sensing

Capacitance sensing has been used to estimate physical properties such as touch, proximity, and deformation by measuring the capacitance variations between two or more conductors [22, 49]. These conductors can be made from metal, foils, plastics or human skin. A capacitance exists when two conductors are separated from each other by a distance d . They create a capacitance of $C = \frac{\epsilon_0 \epsilon_r A}{d}$, where A is the area in meters square, d is the distance between the two plates, and ϵ is dielectric constant (i.e., product of free space ϵ_0 and relative dielectric constant of the material, ϵ_r). In our case, when a user performs a gesture, the skin deformation will cause a change in distance between the plates. By capturing the capacitance variations measured from these sensors, CapBand recognizes the gesture being performed.

The capacitance sensing value is indirectly inferred by different methods that measure the variance in voltage, current, and frequency or pulses which are applied through the capacitor of

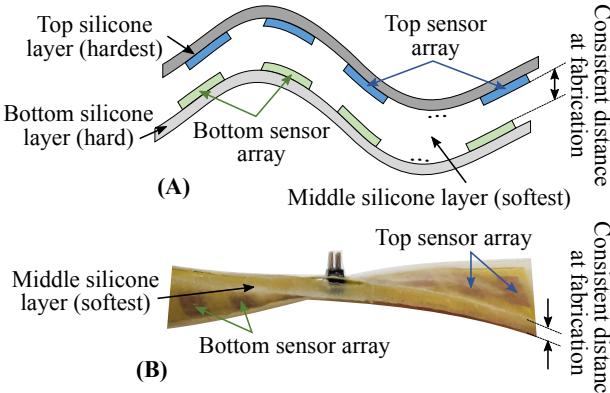


Figure 5: (A) Heterogeneous silicone layering design, (B) Actual fabricated wristband

interest [79, 80]. For example, relax oscillator, sigma-delta modulation, RC oscillator and pin oscillator are few popular methods which have power consumption in the range of pW to μW . The most intuitive measurement method is to charge the capacitor from a defined current source for a known time and measure the voltage across the capacitor. RC-oscillator method applies the same mechanism and uses a reference resistor to count the charge/discharge time period when the capacitor reaches a pre-defined high/low thresholds.

The circuit operating voltage and the pre-defined threshold level is denoted as V_{CC} , V_H , and V_L respectively. In this design, in order to achieve highly sensitive and low latency capacitance sensing, the value of the sensor needs to be relatively small (pF -order) and the reference resistor is chosen to be very large ($M\Omega$ -order). Then the charge/discharge duration can be expressed as:

$$t_{charge} = RCl \ln\left(\frac{V_{CC}}{V_{CC} - V_H}\right), t_{discharge} = RCl \ln\left(\frac{V_{CC}}{V_L}\right)$$

After counting charge/discharge time, the sensor capacitance value is inferred by averaging the calculation from the above equations. In each measuring cycle, the sensor pin of the main MCU will be switched between high and low output direction, which consumes power at the order of $20\text{-}50\text{pA}$. These selections for sensor size, reference resistor, and sensing technique yield the short measurement times at ms-order and ultra-low-power consumption.

4.2 Custom Flexible Sensor Array

We design capacitance sensor elements and combine them into a sensor array. Our design implication comes from the basic form of 2-plate-capacitor such that each sensor has the form of a capacitor with two parallel plates. One plate is connected to the voltage source and another one is connected to the ground. According to the wrist analysis, which clusters 8 areas for sensing (Sec. 2), we construct eight corresponding groups of sensors following those optimal sensing areas.

4.3 Ultra-low Power Capacitance Sensing

Then, we integrate these two sensor arrays into a wearable wristband by fabricating two separate layers of sensor plates as flexible PCB (Figure 4). Each layer consists of an array offi fteen plates. These plates are clustered to correspond to eight sensing

areas. Among those eight areas, seven contain sensor pairs and one has a standalone sensor). This custom sensor design and the standalone sensor pair placement give us (1) reliable control with the explicit capacitance sensor ground and (2) the convenience in guiding users the correct orientation of wearing the wristband later.

4.4 Heterogeneous Silicone Wristband

The next task is to integrate this structure of two flexible sensor arrays into a wristband form factor. To realize a sensitive yet reliable sensor measurement, the challenge here is to keep a consistent spacing between two layers; to maintain enough softness, and flexibility for the movement of the two layers, and to allow skin deformation to be propagated to the sensor whenever the user moves his/her hand. To that end, the wristband needs to be constructed from different layers of materials with different levels of softness.

In particular, we construct the wristband from three different silicone layers as shown in Figure 5 with the middle layer to be cured separately in order to have the desired thickness. This wristband design with a heterogeneity of silicone materials fulfills our requirements to have a consistent spacing between two flexible layers. Furthermore, the inner silicone layer is soft and tacky enough so that whenever the user's hand moves, skin deformation will squeeze the internal middle layer but not the outer one, changing the distance between the two corresponding plates of sensor arrays that will lead to the changes in capacitance.

Triggering measurements at all sensors at once requires much power and sophisticated MCU with large numbers of I/O pins. This leads to more complicated and expensive hardware which violates our design goals mentioned earlier. In CapBand, we use a multiplexer together with a simple MCU to alternatively trigger different capacitance sensors. There is only one sensor activated at a time. Due to this simplicity, the MCU could be very simple and require at most one ADC pin. Figure 6 shows the CapBand's overall architecture. The multiplexer successively rotates through all sensor inputs and allows only one signal path from a sensor to the main processor at any instance. This reduces the power consumption up to N times (N is the number of the sensor in the array). Note that this design decision makes a trade-off between power consumption and the resolution of sensing data. The data stream consists of packets which are the combination of all 15 sensing values. In addition, we use Bluetooth Low Energy (BLE) to streaming data from the device to a host computer for further processing. BLE is a popular, low-power technique for communication, and readily available in smart devices. One could argue that BLE still consumes too much power for a battery-free application. However, there are battery-free products on the market that utilize BLE such as LunaR smartwatch [68] and EnOcean PushButton [15].

Power Analysis. With the above design, assuming the capacitance value is at pF -order, charging time is at ms-order, and the device is operated at 3V, the sensing component requires μW -order. In particular, low-power MCU [28] and a MUX [29] requires μW of power. This power consumption is similar to that of force resistor sensor [32]. However, we select capacitance sensing due to its high sensitivity characteristics to small movements [22]. The power consumption is much lower than camera-based [37] (400mW) or

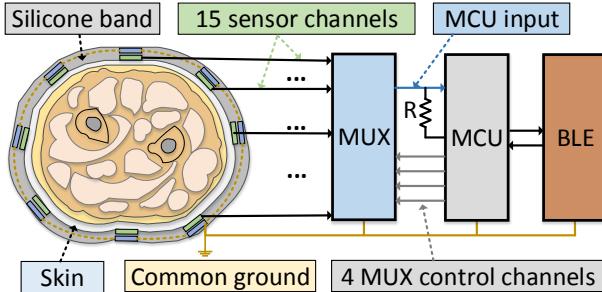


Figure 6: CapBand overall hardware structure

impedance tomography [86] (50mW). BLE components [26] usually consumes at a peak of 5mA for communication. In order to compensate for the consumption of BLE communication and to further investigate the ultra-low-power perspective of our system, we explore the possibility of rendering the battery-free prototype by designing an energy harvesting and management module from a solar cell (Section 7).

5 HAND GESTURE RECOGNITION

When using CapBand for hand gesture recognition, input data will be a series of data samples at a given frequency, each sample containing 15-dimensional vectors of the capacitance sensors on the wristband. One straightforward approach is to consider the static phase only (i.e., when the user holds a stationary gesture), and use traditional machine learning methods, such as support vector machines or random forest to classify every single sample in the static phase. However, this approach does not utilize sensor readings in the motion phase (i.e., when the user is in the process of making a gesture), which can be useful for more accurate and even faster gesture recognition. Intuitively, as the user makes a specific gesture, there are distinct gesture-specific characteristics to be captured in both the motion phase and the static phase. As such, a hybrid hand gesture recognition model that leverages both the motion and static phases would be desirable.

To accomplish this, we tried to implement traditional supervised learning algorithms such as SVM and Random Forest. However, these methods are not good at detecting the skin deformation variance within a given time series. The recent development of deep learning algorithms provides a better solution. We propose a temporal deep learning architecture using a series of data samples for hand gesture recognition. Different from previous hand gesture recognition approaches that consider only individual data samples, we combine the sensor readings in a given time frame as a feature matrix and develop a Convolutional Neural Network (CNN) based model for hand gesture recognition. CNN [39] is a popular deep learning technique for learning local structures of the feature matrix adaptively. It can take advantage of the strong proximity relation between features and is particularly suitable for capturing the various physical skin deformation relations due to muscle movement when performing hand gestures. Since skin deformation variance depends on the physical position of each individual sensor, a deep learning model to be beneficial in capturing these kinds of geometric variance. In addition, CNN is sensitive for detecting edge patterns; thus it can be more robust with the varying speeds and

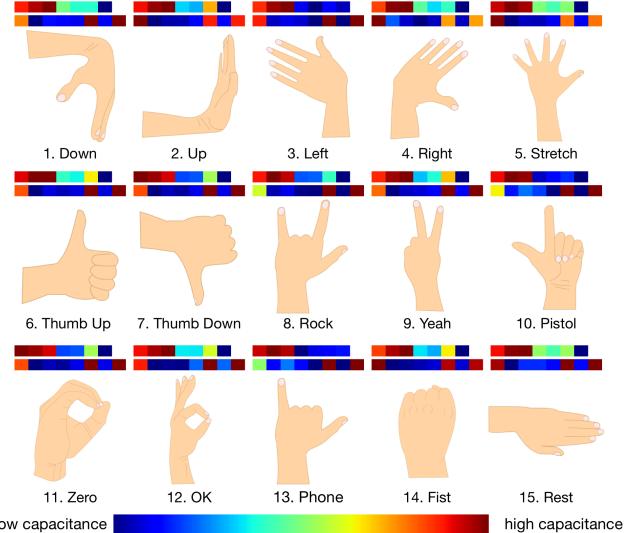


Figure 7: The 15 different hand gestures and corresponding capacitance measurements with dedicated sensor placement as shown in Figure 4.

magnitudes when people are making hand gestures. In comparison, traditional machine learning algorithms Support Vector Machine and Random Forest would link every single dimensional feature to a singleton equation, makes them less flexible in detecting edge patterns. Detailed performance evaluation will be explained in Section 9. Please note that in our design, the model will be pre-trained offline using training data, and then embedded into cloud for online recognition. Real-time recognition can be realized by doing this.

Specifically, we use a CNN to learn the temporal muscle pressure patterns for hand gesture recognition, which are encoded as a two-dimensional feature matrix and are used as input data to the network for CNN model training. We combine a sequence of 15-dimensional sensor readings in a gesture cycle. For instance, using 25 samples per second, a 4-second gesture cycle with both the motion and static phases would contain 100 samples, resulting in a 100×15 feature matrix. The proposed CNN architecture consists of two convolutional layers, each followed by a MAX pooling layer. The kernel size for each convolutional layer is 5×1 and the pooling size is 2×2 . For the convolutional layers, we use “same padding” to keep the feature information complete, whereas the max-pooling layers use the same stride as the filter size. We also use a Rectified-Linear (ReLU) activation function and local response normalization is applied after each layer. Dropouts of 0.6 is also applied to the fully connected layer. The initial training rate is set to 10^{-3} as commonly used in CNN-based solutions.

The reason to use a two-layer convolutional structure is to prevent overfitting with our training data. We specify the filter size to be 5×1 to tell the network that small clusters of locally-connected features will contain useful information for classification. After two convolutional layers and two max-pooling layers, a Multi-layer Perceptron is used to complete the classification task. The output layer in our architecture has one neuron for each gesture in the recognition process. We apply “softmax” as our activation function.

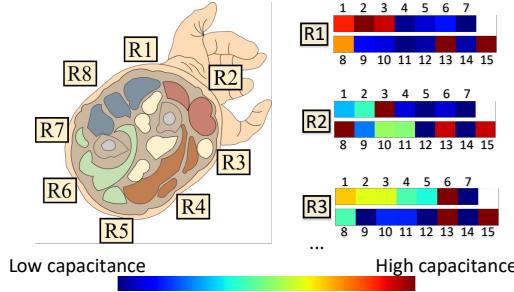


Figure 8: Sample of sensor measurement while the wristband is rotated at different position.

Our CNN architecture is implemented using the TensorFlow software library [1]. We first train the CNN model offline with sensor input from CapBand and labeling information using the TensorFlow framework written in Python. The reason for using TensorFlow is that it has a C++ API that can be used on smartphone platforms to run the inference (recognition)-only part of the CNN model. After training the offline CNN model, the trained weights are used as a recognition-only structure by removing the back-propagation, training layers and dropouts. Only the recognition module will be used for real-time gesture recognition on smartphone platforms.

6 CAPBAND ON-WRIST LOCALIZATION

To achieve higher gesture recognition accuracy and better user experience, proper positioning of CapBand on users' wrist is essential. While it is relatively easy to point out that the wristband is in the wrong position, it would be much more useful (and user-friendly) to determine the specific (incorrect) position so the user can adjust accordingly. Accurate on-wrist localization also makes it possible to automatically calibrate for any position without manual adjustment, which we plan to investigate as future work. The challenge of on-wrist localization is that there exists no universal coordinate system for on-body sensors, i.e. on the wrist. Moreover, if we define the axes and mark the coordinates based on fixed positions, it cannot generalize to diverse users with different wrist sizes.

Position definition. In order to preserve the generality, we approach to define the wristband coordination based on wrist muscle anatomy structure analysis that we did in Section 2. We define the correct position while the user wears the wristband by aligning region 1 (R_1) with the group of sensor input number 3 and 4 (sensor input configuration is shown later in Figure 10). Thus we denote this wristband position as R_1 . Each time the user wears the wristband, the alignment of the pair (4,5) will not be exactly at the R_1 position as the first time because the wristband might rotate a little bit, which leads to the alignment of the pair (4,5) will belong to other regions of interest ($R_{2..8}$). Thus we define 8 feasible localization positions of CapBand relatively corresponding to the alignment of pair (4,5) with given regions of interest $R_{1..8}$. Since the design of odd sensor number guides the user the correct direction of CapBand, we do not consider the case that user wears the wristband in flipping position.

Given data input from 15 sensors in a fixed period of time, our goal is to detect the current location of the wristband (Figure 8

shows the measurement samples while rotating wristband at different locations with gesture 'stretch'). Thus we can instruct the user to rotate the wristband back to the correct position if necessary.

Kernel-based localization. We propose a kernel-based localization method to automatically identify a user's wristband position based on the 15-dimensional sensor readings. The main advantage of using the kernel-based approach is that the relevance of a user's new wristband position can be calculated by aggregating the kernel similarity with previous training position data in our dataset. In addition, it provides a structured way to use a linear algorithm in a transformed feature space, for which the transformation in our problem is typically nonlinear. As such, the kernel method allows us to model the nonlinear patterns of wristband position effectively.

Specifically, given a set of training samples $(\mathbf{l}_i, y_i), i = 1, \dots, n$, where \mathbf{l}_i represents one sample of 15-dimensional sensor readings. Assuming that \mathbf{l}_i follows an unknown distribution g : $y_i = g(\mathbf{l}_i)$. Our goal is to model g with some function \hat{g} :

$$\hat{g} = \operatorname{argmin}_{g \in \mathcal{H}} \sum_{i=1}^n (y_i - g(\mathbf{l}_i))^2 + \lambda \|g\|_{\mathcal{H}_K}^2, \quad (5)$$

where $\|g\|^2$ is the regularization function.

We define kernel $K_H(\mathbf{l}_i)$ as a multivariate Gaussian Kernel:

$$K_H(\mathbf{x}) = \frac{1}{\sqrt{2\pi|\mathbf{H}|}} e^{-\frac{\mathbf{x}\mathbf{x}^T}{2\mathbf{H}}}, \quad (6)$$

where \mathbf{H} is the 15×15 symmetric and positive definite covariance matrix. For simplicity, in our training process, we set $\mathbf{H} = \mathbb{I}$.

We then define the function space as: $g(\mathbf{l}) = \sum_{i=1}^n \alpha_i K_H(\mathbf{l} - \mathbf{l}_i)$,

$$g(\mathbf{l}) = \sum_{i=1}^n \alpha_i K_H(\mathbf{l} - \mathbf{l}_i), \quad (7)$$

where $\boldsymbol{\alpha}$ will be learned during the training process by minimizing the least-squares function $\|\mathbf{y} - \mathbf{K}\boldsymbol{\alpha}\|_2^2$ using stochastic gradient descent. The main procedure of our kernel-based wristband localization algorithm is shown in Algorithm 1. The non-linear function g in Algorithm 1 outputs the estimated wristband position, a real value that is rounded up and mapped to one of the 8 on-wrist regions (Figure 8). This helps users to adjust the position accordingly and (as future work) possibly auto-calibrate the wristband without manual adjustments.

Algorithm 1: Kernel-based On-Wrist Localization

```

Input: Training data  $(\mathbf{l}_1, y_1), \dots, (\mathbf{l}_n, y_n)$ , the weighting vector  $\boldsymbol{\alpha}$ , and the regularization parameter  $\lambda$ .
Output: variables  $\boldsymbol{\alpha}$ , function  $g$ , step size  $\theta$ 
1 initialization:  $\boldsymbol{\alpha} = 0$ ;
2 repeat
3   for  $j = 1, 2, \dots, n$  do
4     Learning output:  $\hat{y}_j = \sum_{i=1}^n \alpha_i K_H(\mathbf{l}_j - \mathbf{l}_i)$ ;
5     Compute the error:  $e_j = \hat{y}_j - y_j$ ;
6     Update the solution:
7        $\alpha_j \leftarrow \alpha_j - \theta(2e_j \sum_{i=1}^n \alpha_i K_H(\mathbf{l}_j - \mathbf{l}_i) + 2\lambda\alpha_j)$ ;
8     Update the function:  $\hat{g}(\mathbf{l}_j) = \sum_{i=1}^n \hat{\alpha}_i K_H(\mathbf{l}_j - \mathbf{l}_i)$ ;
8   end
9 until converge;
10 return  $g(\mathbf{l}) = \sum_{j=1}^n \alpha_j y_j K_H(\mathbf{l}_j - \mathbf{l})$ 

```

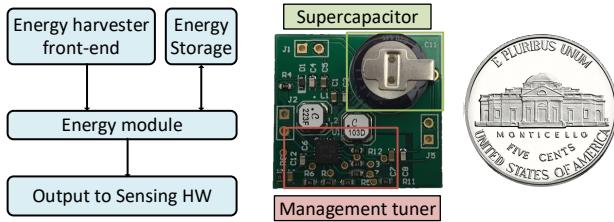


Figure 9: Energy management module schematic and PCB

7 ENERGY MANAGEMENT FOR TRANSIENT POWERING

To deal with the difficulties in (1) providing a stable supply for the sensing hardware with the fluctuation of harvesting energy source and (2) harvesting sufficient energy for the operation of sensing hardware, we first design an energy management module and then investigate suitable energy sources.

7.1 Energy management design

The energy management module contains (1) the energy harvesting front-end that scavenges the energy from the surrounding environment; (2) the power harvester management module that controls the charge/discharge cycle and provides stable DC output; and (3) the energy sink that stores the harvested energy. Ideally, this module should first provide enough energy for the sensing hardware (refer to the design in Sec. 4) and communication module boot-up, process the measurement, send the data to the host device and maintain the connection in the sleep mode, most of the time.

We denote the supply output to sensing hardware as V_{out} and energy sink to be a supercapacitor of size C_{sink} . Since the system boot-up happens only once, we investigate and design the duty cycle of the system to be under the coverage of the charge/discharge cycle of the power harvester management module. One cycle of the whole system includes the duration to measure all sensor values t_{sense} , the duration to transmit the measurement $t_{transmit}$, and the sleep duration t_{sleep} , which has the current consumption of i_{sense} , $i_{transmit}$, i_{sleep} , respectively. The charge/discharge cycle of the module is controlled by managing the appropriate V_{high} and V_{low} of the energy sink. At V_{high} , the supercapacitor starts to discharge and sensing hardware switches to active mode, measures sensor values, and transmits data. When the voltage across the supercapacitor reaches V_{low} , the sensing hardware switches to sleep mode and energy module starts to charge the wristband.

Minimum discharge time of energy sink is:

$$t_{supply} = t_{sense} + t_{transmit} = C_{sink} \left(\frac{\delta V_{sense}}{i_{sense}} + \frac{\delta V_{transmit}}{i_{transmit}} \right),$$

where $\delta V_{sense} + \delta V_{transmit} = V_{high} - V_{low}$.

Minimum charging time of energy sink:

$$t_{harvest} = \frac{E_{V_{high}} - E_{V_{low}}}{P_{harvest} - P_{sleep}} = \frac{1}{2} \frac{V_{high} - V_{low}}{P_{harvest} - P_{sleep}} C_{sink}.$$

Thus all the values of C_{sink} , V_{out} , V_{high} , and V_{low} must be tuned to fit the operation scheme of the sensing hardware.



Figure 10: Custom fabricated PCB of sensing hardware

7.2 Energy harvesting sources

There is a wide range of harvestable energy sources in recent research, which typically can harvest in range of μW to mW (e.g. TV signal [38, 81], Wi-Fi [52, 75], RF ambient [50, 58], solar cell or photodiode [4, 74], piezoelectric [54], and thermoelectric generator [33]). Including the communication module (i.e. BLE), CapBand requires 11.5mW consumption, on average. We note that BLE was selected for our design to provide reasonable communication range with the host device while not consuming too much energy. This selection was motivated by the existence of many battery-free products in the market that also utilize BLE, such as the LunaR smartwatch [68] and EnOcean PushButton [15].

Solar energy source. To demonstrate the operational capability of our system in the battery-free scheme for both indoor and outdoor environments, we employ solar cells as the energy harvesting front-end. This selection was made due to several advantages: (i) the availability of small size solar cells that can harvest energy with various light intensity [23] in both the indoors (500-800 lux) and outdoors (20k-120k lux), (ii) high power conversion efficiency, and (iii) low-cost and easy integration process. This energy source provides sufficient energy for the hardware to operate in both harvest-and-use schemes in normal light condition and harvest-and-use-later schemes in no light condition within 113s.

RF energy source. We also fabricate hardware for a multiband RF energy harvesting front-end that scavenges energy at LTE700, GSM850, and ISM900 frequency bands. This component includes a voltage rectifier that converts the incident RF power into functional DC power, an adaptive impedance matching network that allows the circuit to select the excited frequency band according to available RF power and provides the maximum power delivery from the antenna to the voltage rectifier for that band. Though we can show that RF can be used as an alternative energy source, this technique is not practical for two main reasons. First, it requires a strong incident RF source (i.e. more than -5dBm) nearby (i.e. within 1 meter) to provide enough power, which is subject to FCC regulations [16] that limit the maximum permissible output power. Second, this RF module requires an additional antenna that impacts its form factor.

8 PROTOTYPE IMPLEMENTATION

Flexible sensor array and wearable wristband. We fabricate two separated flexible sensor arrays from the flexible circuit material Pyralux [60], where each sensor plate has the size of 7×7mm. In order to meet the requirement for a silicone wristband (stated in Section 4), we construct our wristband carefully by combining three silicone layers with the different hardness levels of 10A, 00-10, and

30A Shore hardness silicone, respectively, using the Smooth-On Dragon Skin [69, 70] and Ecoflex silicone [71]. The pads between the two flexible circuits are cured separately to achieve the consistent spacing of 3mm. Given the size of the sensor, and the permittivity of silicone, the estimation of capacitance for each sensor in normal conditions is 25.34pF.

Custom low-power, low-cost hardware. We choose CC2650 module [26], TI MSP430G2553 [28], and MUX CD74HC4067 [29] to fabricate our custom 50.5×23.6mm PCB hardware (Figure 10). A bleeding resistor of $5.1M\Omega$ gives us the advantages of (1) a higher resolution in capacitance measurement and (2) small current drain (588nA with operating voltage of 3.0V). The BLE part in our hardware is implemented as a server so that it will just need to broadcast data during the connection interval after waking up from the sleep mode after pairing with the BLE device on the host device side.

Power management for transient powering. We prototype the energy management module (Figure 9) with the DC-DC buck controller TI BQ25570 [27] and a 47 mF supercapacitor as the energy sink. Hardware peripheral is tuned to stabilize the output voltage of 3V. The high and low voltage thresholds of the supercapacitor are also tuned at 3.9 and 3.2V, respectively. We choose the amorphous silicon solar cell AM-5610CAR [55], which has a size of 25×20mm and a maximum power of 16.8mW. The RF front-end uses a 4-stage Dickson-based rectifier, and we employed an HSMS285C Schottky diode [7], which has a turn-on voltage of 150mV measured at the forward current of $100\mu A$ as a non-linear component of the rectifier.

9 EVALUATIONS

In this section, we present in detail the set of experiments we conducted to evaluate the overall performance of CapBand and demonstrate the feasibility of using our battery-free successive capacitance sensing wristband for hand gesture recognition in the real-world under low-power scenarios.

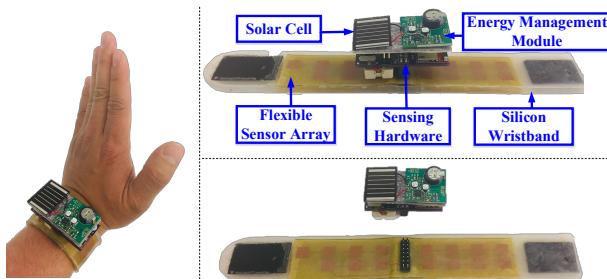


Figure 11: CapBand prototype

In particular, we aim to evaluate the following aspects of CapBand: (1) Effectiveness of hand gesture recognition algorithms using both motion and static phases; (2) accuracy of the kernel-based on-wrist sensor localization method; (3) power consumption of the capacitance sensing system; and (4) user experience survey.

9.1 Evaluation Methodology

Experimental Setup: As shown in Figure 11, the user wears CapBand to his/her left wrist (similar to wearing a watch) and follows

the instructions on the computer screen to make the gestures. During the data collection period (i.e. 30 minutes per subject), we use CapBand with a coin battery to avoid the charging cycles of the energy harvesting module.

Participant Demographics	
Age (years)	18 - 35 years old, average 26.4
Gender Ratio	Male: 16, Female: 7
Dominant Hand	Right: 19, Left: 4
Static set	Gestures: 15, Duration: 1 min
Motion set	Gestures: 15, 40 motions
Position data set	Positions: 8, Duration: 1 min
Participant rate	Position: 10, Static: 20, Motion: 20

Table 1: Data collection details.

Data Collection. To evaluate the effectiveness of the proposed algorithms for hand gesture recognition and on-wrist localization, we invited testers to collect three different types of data for evaluation: (i) static gesture data, (ii) motion gesture data, and (iii) wristband positions data.

Static gesture dataset is collected by asking each participant to perform every gesture statically for one minute. We also recorded the motion gesture dataset from the same 20 participants. The motion phase includes moving from the rest status to a specific gesture, and moving back to the rest status. We use three seconds of data for the motion cycle, which is sufficient to perform a complete gesture cycle in real life. Note that users may finish the gesture before three seconds, then he will wait for the next signal on the screen to start performing the next gesture. Recording this way would better reflect a real application scenario. Each motion gesture was repeated 40 times by each participant.

We instructed ten users to wear the wristband for one minute in each position in eight different positions as defined in Section 6. The sensing results for these gestures formulated our dataset.

9.2 Gesture Recognition Performance

For hand gesture recognition, we use three metrics: precision, recall, and F1 measure for multi-class classification, to evaluate whether the model provides comparatively robust performance.

	Method	Precision	Recall	F1
Static-only	LR	0.113	0.102	0.107
	DT	0.732	0.741	0.736
	SVM	0.896	0.904	0.900
	RFC	0.925	0.932	0.928
	CNN	0.802	0.796	0.799
Motion + Static	LR	0.121	0.115	0.118
	DT	0.744	0.741	0.742
	SVM	0.773	0.624	0.691
	RFC	0.883	0.889	0.886
	CNN	0.951	0.949	0.950

Table 2: Comparison of hand gesture classification performance of different methods using static-only features and motion+static features.

We compare our CNN-based method with different machine learning methods including Logistic Regression (LR), Decision Tree (DT), Support Vector Machines (SVM), and Random Forest Classifier (RFC). These methods have been widely used in gesture recognition and shown to be effective in specific settings. We evaluate

the methods using static-phase features as well as motion-phase features. For training and evaluation purposes, we use stratified 10-fold cross-validation and for each split, 10% data samples are held out as test data.

Table 2 shows the overall classification performance of different methods for “static-only” gestures and “static+motion” gestures. As can be seen from the table, our CNN-based method achieves the best performance when leveraging both motion and static features (3-second motion + 1-second static), with 0.951 precision, 0.949 recall, and 0.950 F1 measure.

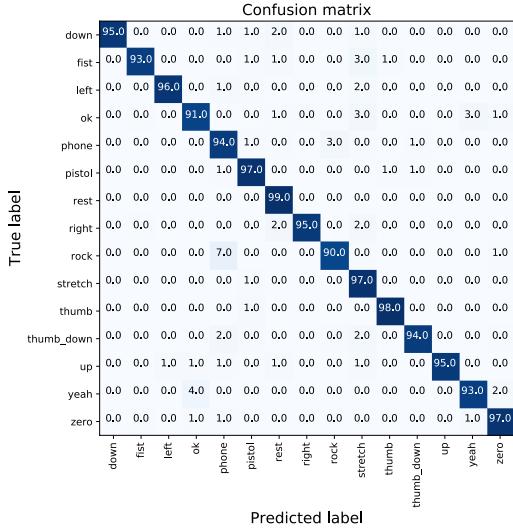


Figure 12: Confusion matrix of gesture classification using CNN with motion+static features.

CNN works better when utilizing motion features because it is effective in detecting feature matrix edges and grouping these nearby features and projecting them to a higher dimensional space to model the muscle pressure variances when users perform these gestures in motion. CNN does not perform well with only static features since they do not contain such variance patterns. Random Forest Classifier (RFC) performs the best for static gesture classification, achieving 0.925 precision, 0.932 recall, and 0.928 F1 measure. For real-world application scenarios, static classification usually does not apply. It is more common that a user would perform a gesture in motion. As such, improving the performance of the “static+motion” setting is our ultimate goal in this paper. Note that linear regression (LR) provides very low performance, with 0.107 F1 measure for the static case and 0.118 F1 measure for the motion+static case. Thus, a non-linear combination of the capacitance sensor readings can model the muscle group variances more effectively for hand gesture recognition.

To evaluate the run-time latency of gesture recognition, we used an iPhone X and an Android Pixel 2 to simulate the recognition process using the trained CNN model. The operating system is iOS 11.3 on iPhone X and Android 8.1 Oreo API 27 on Android Pixel. For each phone, we measured 10 rounds of testing, each round recognizing 100 motion+static sensor inputs. The average latency of each round is 0.15s (1.5ms/recognition) on iPhone X and 0.78s

(7.8ms/recognition) on Android Pixel. These results demonstrate that our CNN-based model is feasible for real-time hand gesture recognition in the wild.

Figure 12 shows the confusion matrix when classifying hand gestures using CNN with both motion + static dataset. Overall, our proposed method consistently performs well for different types of gestures, with slightly lower performance when identifying the “rock” and “phone” gesture. Rock gesture is misclassified the most with phone gesture since they have similar hand configuration and contain both the thumb and pinky side muscle group.

Figure 13 shows the impact of sample size on classification accuracy. Since the static features are less affected by the lower sampling rate, RFC performs stably when reducing the sample size. The motion features are more sensitive to the sample size. CNN still performs well with above 20 measurements per second, but the performance degrades quickly with a lower sampling rate.

We further investigate how using different combinations of motion and static features would impact the classification accuracy and latency. Specifically, we evaluate three different time-series combinations: (1) 1.5s motion, (2) 1.5s motion plus 1s static, and (3) 3s motion plus 1s static, which corresponds to a classification latency of 1.5s, 2.5s, and 4s, respectively. Figure 14 shows the results. While removing the static features and reducing the motion features have some impact on the classification accuracy, only using 1.5s motion features can still achieve 0.91 F1 measure, while reducing the classification latency from 4s to 1.5s. It proves that using motion features is beneficial in real-world scenarios.

9.3 On-Wrist Localization Performance

For on-wrist localization, we evaluate the accuracy of identifying the wristband position among 8 possible regions. Specifically, we evaluate the accuracy of each region, since the regions correspond to different muscle groups and deform differently when making hand gestures.

Figure 15 shows the localization accuracy and standard deviation for each of the eight regions. We can see that our method performs well for all eight regions, achieving between 92.4% and 99.5% accuracy with a small standard deviation. These results demonstrate the strength of our hardware design using successive capacitance sensing as well as the kernel-based on-wrist localization algorithm. This solution should be robust for real-world usage, which can notify users when the wristband is worn at the wrong position and instruct users to adjust the wristband position accordingly.

9.4 Energy consumption

We measure the consumption of CapBand using Monsoon power monitor. Our hardware can operate normally with the supply voltage ranging from 2.2V to 3V. Table 3 shows the highest energy consumption breakdown for the main processes of sensing hardware is at 3V. The cold start process consumes 27.06 μ W during 2.58s happens once to boot up hardware and establishes pairing with the BLE client in the host device.

After starting up, our hardware takes capacitance measurement of 15 sensors during 30ms with energy consumption of 69 μ W, where the MUX consumes a maximum of 0.33 μ W. Then capacitance measurements are sent by BLE transmission performing in 2.46ms

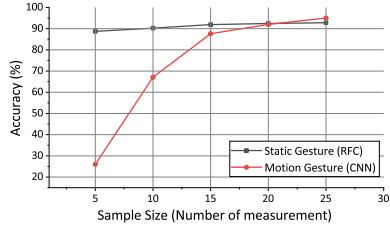


Figure 13: Classification accuracy vs. Sample size

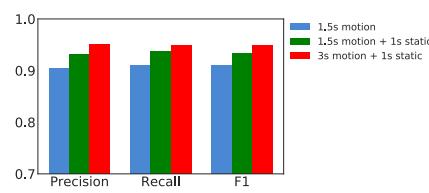


Figure 14: Classification accuracy vs. latency

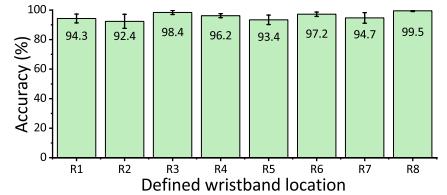


Figure 15: On-wrist localization accuracy and standard deviation for eight defined location.

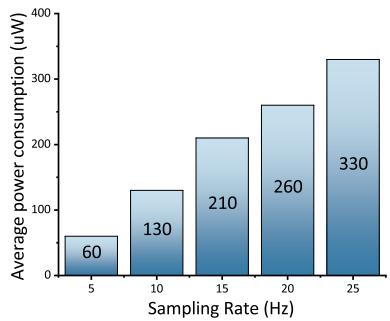


Figure 16: Average power of sensor measurement at different sample size of measurement numbers.

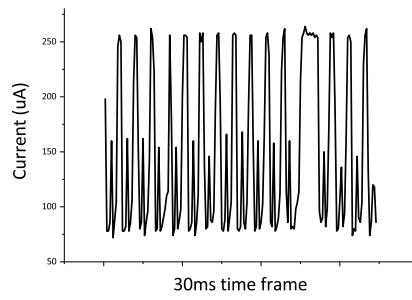


Figure 17: Current profile during measurement of 15 capacitance sensor.

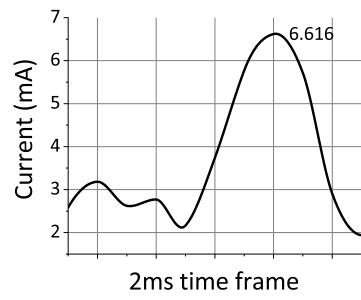


Figure 18: Current profile during BLE module connection interval.

Component	Consumption
Cold start	27.06 μ W
15 capacitance measurements	69 μ W
UART to BLE	375 μ W
BLE packet transmission	10.38 mW
Sleep mode	40 μ W

Table 3: Power consumption breakdown at 3V

with consumption of $10.38\mu\text{W}$ for each operation cycle. Figure 17 and 18 show the current profiling of capacitance measurement and BLE transmission with the average consumption of $148\ \mu\text{A}$ and 3.46mA respectively. Figure 16 shows the average power of sensor measurements with the different sample size of measurement numbers. After finishing measurement and BLE transmission, the sensing hardware goes to standby (defined as sleep mode - LPM3 in both MCUs) and consumes $1.5\ \mu\text{A}$ in total. In various light condition of 30k lux (outdoor clear sky), 2k lux (indoor hallway), 500 lux (office condition), the energy module with the solar cell can harvest 23.1mW , 7.7mW and 1.6mW respectively.

9.5 User Study

We also conduct a user questionnaire involving 12 users who participate in the data collection session to collect feedback of our CapBand. The survey includes questions to ask users' perspective on the usage convenience, interaction smoothness, research idea importance and future integration feasibility. Results show that 63.6% users like the idea of CapBand and think it is easy to use;

72.7% feel in control of the wristband when they are making gestures. Regarding smart device integration necessity, 50% provide positive responses while 41.7% selected "Maybe". In addition, 58.3% of the testers believe the battery-free idea is necessary for this kind of wristband. In summary, we receive positive feedback on the idea of our wristband in terms of gesture recognition and battery-free. At the same time, users also mention concerns about whether long-time wearing of our CapBand would make their wrist tiring. There are also participants who are not very familiar with similar devices express doubt about its stability and sensitivity for long-term use in real-life. All of these concerns should be carefully considered in our future improvement.

10 RELATED WORKS

Sensors and gesture recognition. Camera-based recognition for human hand gestures has been developed for a long time [61, 63, 84] including breakthrough techniques integrated into smart TVs [5, 64]. However, this approach is limited due to inherent constraints on the relative position between users and cameras, the environment, heavy computational workload, etc. [61]. To address these challenges, other approaches have been built to utilize the IMU embedded in mobile devices such as AcceleGlove [24], ASG [57], ViBand [40], GestGlove [3], and Finexus [10]. Furthermore, various devices such as Myo [76], EMPress [45], Force Myography [31, 82], EchoFlex [44] have been introduced to exploit special bio-signatures extracted from the electromyographic signal (EMG) or ultrasound images captured from human muscles as a new input type.

Wrist-worn devices. One of the ongoing research focus is to recognize gestures [17, 25, 30] by analyzing the tension of carpus muscles [12, 18, 32, 42, 47]. These studies measure and calibrate data captured from different skin sources. For example, cameras [63], accelerometers, and gyroscope attached to the user’s wrist and finger [10] can be used to measure the changes of the outer skin. Other approaches such as WristWhirl [21], SkinWatch [51], SensIR [43] captures the change of muscles from the light reflected off the skin when a chain of multiple IR emitters and receivers are attached on. On the other hand, inner skin source method measures the muscle signal (EMG), i.e. Myo product [76]; or the impedance tomography (tomo [85] and tomo2 [86]). However, none of these techniques are battery-free or require minimum training as CapBand

The works that are most relevant to CapBand are capacitive-sensing wearable devices. Swedish glove [34] introduces a capacitive sensor attached to each finger and an on-wrist air pressure sensor to capture finger flexions. GestureWrist [62] measured capacitance caused by wrist-shape changes and forearm movements using a system of 6 receiver electrodes and 1 transmitter electrode placed on top of the user’s wrist. These devices and our previous works [77, 78] are different from CapBand in that (1) we have the dedicated sensors and wristband design that eliminate the influences of surrounding factors such as user’s sweat and ambient heat and (2) we measure the capacitance changes of those capacitors with an ultra-low-power sensing technique which makes battery-free possible.

Low power wristband. Times in operation is one of the key characteristics of a wearable device [56]. Users always have the demand to reduce charging time of their smart phone, smart watch or even a laptop. The development of wristband is also required to follow the trend of prolonging operation time by cutting down the power consumption. One of the relevant work on low-power consumption of wrist-worn device is Wristband Vital [6] which can track down human activity for approximately 60 hours with a lithium-ion coin battery. However, their device only has a very simple function of tracking. The other one, WristFlex [13], designed to recognize human hand gestures, can put in operation for up to one week. In our work, we demonstrate the capability of CapBand to operate with transient powering from both solar cell and RF-based energy harvesting module. In fact, most of COTS devices are not optimized to obtain ultra-low power consumption due to their use of sensors that are not specifically designed wearables.

11 DISCUSSION

Comparative studies. We have reported gesture classification accuracy of 0.95 using “motion+static” data and 0.93 using “static-only” data. Prior research has reported performance with inconsistent gesture set and their prototypes are lack of open-source materials, making it difficult to construct a direct comparison. As a result, we have focused primarily on a relative comparison with widely-used machine learning methods. In addition, the reported result in this paper is per-person trained. We are working on collect enough training data in order to build a generalized model and eliminate the per-person training process. Even though CapBand shows high recognition accuracy with the current set of 15 gestures, there are still other hand gestures that are not included, e.g. pinch gestures

and small fingers only gestures. It will be part of our future work to test our system with a larger set of gestures.

Future integration. CapBand includes the energy harvesting monitor, the custom sensing hardware, and the host device (either PC or BLE-enabled smartdevice) along with CapBand localization and hand gesture recognition software. One of our main goals is to reach low power consumption of the sensing mechanism on custom hardware. Therefore, we do not pay much attention to the optimization and evaluation of the energy consumption on the host device in this work, which can be an interesting topic for future studies. The current prototype is still at POC stage. Our future goal is to reduce the size and cost of CapBand and integrate all the modules into one piece. Moreover, the techniques in this paper, i.e. the capacitance sensing and customized sensor fabrication, along with the PCB fabrication improvement and the sensor size reduction, not only have great potential to be integrated into commercial products, such as smartbands or smartwatches. Some of these devices have already employed BLE component and battery. They are also promising to be utilized in other applications related to skin deformation detection such as respiration rate measurement, facial gesture recognition [48] and silent speech system.

12 CONCLUSIONS

In this work, we designed CapBand, a battery-free wristband system for hand gesture recognition. We analyzed human wrist and hand gesture anatomy to build a wrist muscles-to-gesture model, introduced a successive capacitance measurement technique for ultra-low-power and highly sensitive sensing of small skin movement on the human wrist. We designed a CNN-based model for classifying 15 hand gestures using both motion and static features. To help users adjust the wristband position, we also developed a kernel-based on-wrist localization method. A prototype CapBand system has been implemented to demonstrate its feasibility with a custom-built flexible electronic circuit, custom PCB hardware, and custom-built RF-based energy harvesting hardware module. Evaluations with 20 participants provide promising accuracy, with 95.0% for classifying 15 hand gestures and 95.3% for on-wrist localization. We also conducted a user study to understand the usability of the system in terms of its responsiveness, robustness and form factor. As our future work, we would like to investigate more effective use of the motion and static features to further improve classification accuracy and reduce system latency. We will investigate the robustness of the system manufacturing process and its impact on system performance. Since CapBand is battery-free and robust across users and per-use variations, we expect CapBand to be readily usable for many real-world settings for accurate hand gesture recognition. Our kernel-based on-wrist localization method may also prove to be generally applicable to other on-body sensing wearable solutions.

Acknowledgements. We thank the anonymous shepherd and reviewers for their insightful comments. We also thank Keith Graham, Timothy May, Rebecca McKain and Nhat Pham for their help and consultant during the project. This work is supported in part by the US National Science Foundation (NSF) through grant CNS 1619392 and 1528138.

REFERENCES

- [1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. Tensorflow: A system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283, 2016.
- [2] B. Abbott and D. Wilkie. The relation between velocity of shortening and the tension-length curve of skeletal muscle. *The Journal of Physiology*, 120(1-2):214–223, 1953.
- [3] S. Agarwal, A. Mondal, G. Joshi, and G. Gupta. Gestglove: A wearable device with gesture based touchless interaction. In *Proc. of the 8th Augmented Human International Conference*, AH '17, pages 3:1–3:8, 2017.
- [4] C. Alippi and C. Galperti. An adaptive system for optimal solar energy harvesting in wireless sensor network nodes. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 55(6):1742–1750, 2008.
- [5] ArcSoft. Arcsoft gesture recognition technology. <https://goo.gl/bJmRea>. [Online; accessed Apr 01, 2017].
- [6] K. Braam, T. C. Huang, C. H. Chen, E. Montgomery, S. Vo, and R. Beausoleil. Wristband vital: A wearable multi-sensor microsystem for real-time assistance via low-power bluetooth link. In *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*, pages 87–91, Dec 2015.
- [7] Broadcom. Hsms-285c detector diode. <https://goo.gl/iu3AZY>. [Online; accessed Apr 05, 2018].
- [8] T. S. Buchanan, D. G. Lloyd, K. Manal, and T. F. Besier. Neuromusculoskeletal modeling: estimation of muscle forces and joint moments and movements from measurements of neural command. *Journal of applied biomechanics*, 20(4):367–395, 2004.
- [9] T. S. Buchanan, M. J. Moniz, J. P. Dewald, and W. Z. Rymer. Estimation of muscle forces about the wrist joint during isometric tasks using an emg coefficient method. *Journal of biomechanics*, 26(4-5):547–560, 1993.
- [10] K.-Y. Chen, S. N. Patel, and S. Keller. Finexus: Tracking precise motions of multiple fingertips using magnetic sensing. In *ACM CHI*, pages 1504–1514, 2016.
- [11] C. Cornelius, R. Peterson, J. Skinner, R. Halter, and D. Kotz. A wearable system that knows who wears it. In *Proceedings of the 12th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '14, pages 55–67, 2014.
- [12] A. Dementyev and J. A. Paradiso. Wristflex: Low-power gesture input with wrist-worn pressure sensors. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pages 161–166, 2014.
- [13] A. Dementyev and J. A. Paradiso. Wristflex: Low-power gesture input with wrist-worn pressure sensors. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pages 161–166, New York, NY, USA, 2014. ACM.
- [14] Y. Du, W. Jin, W. Wei, Y. Hu, and W. Geng. Surface emg-based inter-session gesture recognition enhanced by deep domain adaptation. *Sensors*, 17(3), 2017.
- [15] EnOcean. Pushbutton transmitter module. <https://goo.gl/oIHnVj>. [Online; accessed Apr 05, 2018].
- [16] FCC. Fcc codes of regulation, part 15. <https://goo.gl/8mcBDI>. [Online; accessed Apr 05, 2018].
- [17] A. Ferrone, X. Jiang, L. Maiolo, A. Pecora, L. Colace, and C. Menon. A fabric-based wearable band for hand gesture recognition based on fiber Bragg strain sensors: A preliminary investigation. In *2016 IEEE Healthcare Innovation Point-Of-Care Technologies Conference (HI-POCT)*, pages 113–116, Nov 2016.
- [18] A. Ferrone, F. Maita, L. Maiolo, M. Arquilla, A. Castiello, A. Pecora, X. Jiang, C. Menon, A. Ferrone, and L. Colace. Wearable band for hand gesture recognition based on strain sensors. In *2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pages 1319–1322, 2016.
- [19] A. E. Flatt. Grasp. *Proceedings (Baylor University Medical Center)*, 13(4):343, 2000.
- [20] R. Fukui, M. Watanabe, T. Gyota, M. Shimosaka, and T. Sato. Hand shape classification with a wrist contour sensor: Development of a prototype device. In *Proceedings of the 13th International Conference on Ubiquitous Computing, UbiComp '11*, pages 311–314, 2011.
- [21] J. Gong, X.-D. Yang, and P. Irani. Wristwhirl: One-handed continuous smartwatch input using wrist gestures. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, UIST '16, pages 861–872, New York, NY, USA, 2016. ACM.
- [22] T. Grosse-Puppeldahl, C. Holz, G. Cohn, R. Wimmer, O. Bechtold, S. Hodges, M. S. Reynolds, and J. R. Smith. Finding common ground: A survey of capacitive sensing in human-computer interaction. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 3293–3315. ACM, 2017.
- [23] GSA. Lighting condition. <https://goo.gl/2SfKq4>. [Online; accessed Apr 05, 2018].
- [24] J. L. Hernandez-Rebolledo, N. Kyriakopoulos, and R. W. Lindeman. The AccelerGlove: A whole-hand input device for virtual reality. In *SIGGRAPH*, page 259, 2002.
- [25] Y. Huang, W. Guo, J. Liu, J. He, H. Xia, X. Sheng, H. Wang, X. Feng, and P. B. Shull. *Preliminary Testing of a Hand Gesture Recognition Wristband Based on EMG and Inertial Sensor Fusion*, pages 359–367. Springer International Publishing, Cham, 2015.
- [26] T. Instruments. Bluetooth low energy cc2650 module. <https://goo.gl/Ysc2dL>. [Online; accessed Apr 01, 2017].
- [27] T. Instruments. Bq25570. <https://goo.gl/XV8BzM>. [Online; accessed Apr 05, 2018].
- [28] T. Instruments. Chip msp430g2553. <https://goo.gl/szRjHL>. [Online; accessed Apr 01, 2017].
- [29] T. Instruments. Multiplexer cd74hc4067. <https://goo.gl/8KQzSm>. [Online; accessed Apr 01, 2017].
- [30] S. Jiang, B. Lv, X. Sheng, C. Zhang, H. Wang, and P. B. Shull. Development of a real-time hand gesture recognition wristband based on semg and imu sensing. In *2016 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 1256–1261, Dec 2016.
- [31] X. Jiang, L.-K. Merhi, Z. G. Xiao, and C. Menon. Exploration of force myography and surface electromyography in hand gesture classification. *Medical Engineering & Physics*, 41(Supplement C):63 – 73, 2017.
- [32] P. G. Jung, G. Lim, S. Kim, and K. Kong. A wearable gesture recognition device for detecting muscular activities based on air-pressure sensors. *IEEE Transactions on Industrial Informatics*, 11(2):485–494, 2015.
- [33] K. Kadirvel, Y. Ramadas, U. Lyles, J. Carpenter, V. Ivanov, V. McNeil, A. Chandrakasan, and B. Lum-Shue-Chan. A 330na energy-harvesting charger with battery management for solar and thermoelectric energy harvesting. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, 2012 IEEE International, pages 106–108. IEEE, 2012.
- [34] N. Karlsson, B. Karlsson, and P. Wide. A glove equipped with haptic feedback sensors as a command generator used in a fuzzy control system. In *IMTC/98 Conference Proceedings. IEEE Instrumentation and Measurement Technology Conference. Where Instrumentation is Going (Cat. No.98CH36222)*, volume 1, pages 441–445, 1998.
- [35] J. M. Kauer, A. de Lange, H. H. Savelberg, and J. G. Kooilos. The wrist joint: Functional analysis and experimental approach. In *Wrist Disorders*, pages 3–12. Springer, 1992.
- [36] F. Kerber, M. Puhl, and A. Krüger. User-independent real-time hand gesture recognition based on surface electromyography. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '17, pages 36:1–36:7, 2017.
- [37] D. Kim, O. Hilliges, S. Izadi, A. D. Butler, J. Chen, I. Oikonomidis, and P. Olivier. Digits: Freehand 3d interactions anywhere using a wrist-worn gloveless sensor. In *Proc. of the 25th Annual ACM Symposium on User Interface Software and Technology*, UIST '12, pages 167–176, 2012.
- [38] S. Kim, R. Vyas, J. Bito, K. Niotaiki, A. Collado, A. Georgiadis, and M. M. Tentzeris. Ambient rf energy-harvesting technologies for self-sustainable standalone wireless sensor platforms. *Proceedings of the IEEE*, 102(11):1649–1666, 2014.
- [39] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.
- [40] G. Laput, R. Xiao, and C. Harrison. Viband: High-fidelity bio-acoustic sensing using commodity smartwatch accelerometers. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, UIST '16, pages 321–333, 2016.
- [41] T. Li, X. Xiong, Y. Xie, G. Hito, X.-D. Yang, and X. Zhou. Reconstructing hand poses using visible light. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 1(3):71:1–71:20, 2017.
- [42] J.-W. Lin, C. Wang, Y. Y. Huang, K.-T. Chou, H.-Y. Chen, W.-L. Tseng, and M. Y. Chen. Backhand: Sensing hand gestures via back of the hand. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, UIST '15, pages 557–564, 2015.
- [43] J. McIntosh, A. Marzo, and M. Fraser. Sensir: Detecting hand gestures with a wearable bracelet using infrared transmission and reflection. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, UIST '17, pages 593–597, 2017.
- [44] J. McIntosh, A. Marzo, M. Fraser, and C. Phillips. Echoflex: Hand gesture recognition using ultrasound imaging. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 1923–1934, 2017.
- [45] J. McIntosh, C. McNeill, M. Fraser, F. Kerber, M. Löchtefeld, and A. Krüger. EmPress: Practical hand gesture classification with wrist-mounted emg and pressure sensing. In *Proc. of the 2016 CHI Conf. on Human Factors in Computing Systems*, CHI '16, pages 2332–2342, 2016.
- [46] Myo. Myo support. <https://goo.gl/TgBtBP>. [Online; accessed Apr 01, 2017].
- [47] H. W. Ng, X. Jiang, L.-K. Merhi, and C. Menon. Investigation of the Feasibility of Strain Gages as Pressure Sensors for Force Myography, pages 261–270, 2017.
- [48] P. Nguyen, N. Bui, A. Nguyen, H. Truong, A. Suresh, M. Whitlock, D. Pham, T. Dinh, and T. Vu. Tyth-typing on your teeth: Tongue-teeth localization for human-computer interface. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*, pages 269–282. ACM, 2018.
- [49] P. Nguyen, U. Muncuk, A. Ashok, K. R. Chowdhury, M. Gruteser, and T. Vu. Battery-free identification token for touch sensing devices. In *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM*, pages 109–122. ACM, 2016.
- [50] P. Nintanavongsa, U. Muncuk, D. R. Lewis, and K. R. Chowdhury. Design optimization and implementation for rf energy harvesting circuits. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 2(1):24–33, 2012.

- [51] M. Ogata and M. Imai. Skinwatch: Skin gesture interaction for smart watch. In *Proceedings of the 6th Augmented Human International Conference*, AH '15, pages 21–24, 2015.
- [52] U. Olgun, C.-C. Chen, and J. Volakis. Design of an efficient ambient wifienergy harvesting system. *IET Microwaves, Antennas & Propagation*, 6(11):1200–1206, 2012.
- [53] S. Ortega-Avila, B. Rakova, S. Sadi, and P. Mistry. Non-invasive optical detection of hand gestures. In *Proceedings of the 6th Augmented Human International Conference*, AH '15, pages 179–180, 2015.
- [54] G. K. Ottman, H. F. Hofmann, A. C. Bhatt, and G. A. Lesieutre. Adaptive piezoelectric energy harvesting circuit for wireless remote power supply. *IEEE Transactions on power electronics*, 17(5):669–676, 2002.
- [55] Panasonic. Panasonic - bsg am-5610car solar cell. <https://goo.gl/CtF6m6>. [Online; accessed Apr 05, 2018].
- [56] A. Pantelopoulos and N. G. Bourbakis. A survey on wearable sensor-based systems for health monitoring and prognosis. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 40(1):1–12, 2010.
- [57] J. K. Perng, B. Fisher, S. Hollar, and K. S. J. Pister. Acceleration sensing glove (ags). In *Digest of Papers. Third International Symposium on Wearable Computers*, pages 178–180, 1999.
- [58] M. Piñuela, P. D. Mitcheson, and S. Lucyszyn. Ambient rf energy harvesting in urban and semi-urban environments. *IEEE Transactions on Microwave Theory and Techniques*, 61(7):2715–2726, 2013.
- [59] Q. Pu, S. Gupta, S. Gollakota, and S. Patel. Whole-home gesture recognition using wireless signals. In *Proc. of the 19th International conference on Mobile computing & networking*, pages 27–38. ACM, 2013.
- [60] Pyralux. Flex pcb material. <https://goo.gl/1pWRbR>. [Online; accessed Apr 01, 2017].
- [61] S. S. Rautaray and A. Agrawal. Vision based hand gesture recognition for human computer interaction: a survey. *Artificial Intelligence Review*, 43(1):1–54, Jan 2015.
- [62] J. Rekimoto. Gesturewrist and gesturepad: Unobtrusive wearable interaction devices. In *Wearable Computers, 2001. Proceedings. Fifth International Symposium on*, pages 21–27. IEEE, 2001.
- [63] Z. Ren, J. Yuan, and Z. Zhang. Robust hand gesture recognition based on finger-earth mover's distance with a commodity depth camera. In *Proceedings of the 19th ACM international conference on Multimedia*, pages 1093–1096. ACM, 2011.
- [64] SAMSUNG. Gesture in smart tv. <https://goo.gl/36vPhx>, 2013. [Online; accessed Nov 21, 2017].
- [65] T. S. Saponas, D. S. Tan, D. Morris, and R. Balakrishnan. Demonstrating the feasibility of using forearm electromyography for muscle-computer interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '08*, pages 515–524, 2008.
- [66] T. S. Saponas, D. S. Tan, D. Morris, R. Balakrishnan, J. Turner, and J. A. Landay. Enabling always-available input with muscle-computer interfaces. In *Proceedings of the 22Nd Annual ACM Symposium on User Interface Software and Technology, UIST '09*, pages 167–176, 2009.
- [67] A. R. Sarkar, G. Sanyal, and S. Majumder. Hand gesture recognition systems: a survey. *International Journal of Computer Applications*, 71(15), 2013.
- [68] Shopify. Lunar solar smart watch. <https://goo.gl/nbzFDC>. [Online; accessed Apr 05, 2018].
- [69] Smooth-On. Dragon skin 10 very fast. <https://goo.gl/Uao6gP>. [Online; accessed Apr 01, 2017].
- [70] Smooth-On. Dragon skin 30. <https://goo.gl/uLMNhY>. [Online; accessed Apr 01, 2017].
- [71] Smooth-On. Ecoflex 10. <https://goo.gl/CjuJ8X>. [Online; accessed Apr 01, 2017].
- [72] H. H. Szu. Non-convex optimization. In *Proceedings of the SPIE*, volume 698, pages 59–65, 1986.
- [73] J. Taleisnik, R. H. Gelberman, B. W. Miller, and R. M. Szabo. The extensor retinaculum of the wrist. *The Journal of hand surgery*, 9(4):495–501, 1984.
- [74] V. Talla, B. Kellogg, S. Gollakota, and J. R. Smith. Battery-free cellphone. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(2):25, 2017.
- [75] V. Talla, B. Kellogg, B. Ransford, S. Naderiparizi, S. Gollakota, and J. R. Smith. Powering the next billion devices with wi-fi. In *Proceedings of the 11th ACM Conference on Emerging Networking Experiments and Technologies*, page 4. ACM, 2015.
- [76] Thalmic-Labs. Myo armband wearable guessture control. <https://www.myo.com/>, 2017. [Online; accessed Apr 04, 2017].
- [77] H. Truong, P. Nguyen, N. Bui, A. Nguyen, and T. Vu. Low-power capacitive sensing wristband for hand gesture recognition. In *Proceedings of the 9th ACM Workshop on Wireless of the Students, by the Students, and for the Students*, pages 21–21. ACM, 2017.
- [78] H. Truong, P. Nguyen, A. Nguyen, N. Bui, and T. Vu. Capacitive sensing 3d-printed wristband for enriched hand gesture recognition. In *Proceedings of the 2017 Workshop on Wearable Systems and Applications*, pages 11–15. ACM, 2017.
- [79] H. Truong, P. Nguyen, V. Nguyen, M. Ibrahim, R. Howard, M. Gruteser, and T. Vu. Through-body capacitive touch communication. In *Proceedings of the 9th ACM Workshop on Wireless of the Students, by the Students, and for the Students*, pages 7–9. ACM, 2017.
- [80] T. Vu et al. Distinguishing users with capacitive touch communication. In *ACM MobiCom*, pages 197–208, 2012.
- [81] R. J. Vyas, B. B. Cook, Y. Kawahara, and M. M. Tentzeris. E-wehp: A batteryless embedded sensor-platform wirelessly powered from ambient digital-tv signals. *IEEE Transactions on microwave theory and techniques*, 61(6):2491–2505, 2013.
- [82] B. Wan, R. Wu, K. Zhang, and L. Liu. A new subtle hand gestures recognition algorithm based on emg and fsr. In *2017 IEEE 21st International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, pages 127–132, 2017.
- [83] C. Xu, P. H. Pathak, and P. Mohapatra. Finger-writing with smartwatch: A case for finger and hand gesture recognition using smartwatch. In *Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications, HotMobile '15*, pages 9–14, 2015.
- [84] P. Xu. A real-time hand gesture recognition and human-computer interaction system. *CoRR*, abs/1704.07296, 2017.
- [85] Y. Zhang and C. Harrison. Tomo: Wearable, Low-Cost Electrical Impedance Tomography for Hand Gesture Recognition. In *ACM UIST*, pages 167–173, 2015.
- [86] Y. Zhang, R. Xiao, and C. Harrison. Advancing Hand Gesture Recognition with High Resolution Electrical Impedance Tomography. In *ACM UIST*, pages 843–850, 2016.