Inviz: Low-power Personalized Gesture Recognition using Wearable Textile Capacitive Sensor Arrays

Gurashish Singh, Alexander Nelson, Ryan Robucci, Chintan Patel, Nilanjan Banerjee
Department of Computer Science and Electrical Engineering
University of Maryland, Baltimore County
{singhg1, alnel1, robucci, cpatel2, nilanb}@umbc.edu

Abstract*—Home automation and environmental control is a key ingredient of smart homes. While systems for home automation and control exist, there are few systems that interact with individuals suffering from paralysis, paresis, weakness and limited range of motion that are common sequels resulting from severe injuries such as stroke, brain injury, spinal cord injury and many chronic (guillian barre syndrome) and degenerative (amyotrophic lateral sclerosis) conditions. To address this problem, we present the design, implementation, and evaluation of Inviz, a low-cost gesture recognition system for paralysis patients that uses flexible textile-based capacitive sensor arrays for movement detection. The design of Inviz presents two novel research contributions. First, the system uses flexible textile-based capacitive arrays as proximity sensors that are minimally obtrusive and can be built into clothing for gesture and movement detection in patients with limited body motion. The proximity sensing obviates the need for touch-based gesture recognition that can cause skin abrasion in paralysis patients, and the array of capacitive sensors help provide better spatial resolution and noise cancellation. Second, Inviz uses a low-power hierarchical signal processing algorithm that breaks down computation into multiple low and high power tiers. The tiered approach provides maximal vigilance at minimal energy consumption. We have designed and implemented a fully functional prototype of Inviz and we evaluate it in the context of an end-to-end home automation system and show that it achieves high accuracy while maintaining low latency and low energy consumption.

I. Introduction

An estimated 1.5 million individuals in the United States are hospitalized each year because of strokes, brain injuries and spinal cord injuries [1], [2], [3]. Severe impairment such as paralysis, paresis, weakness and limited range of motion are common sequels resulting from these injuries requiring extensive rehabilitation. Changes in healthcare reimbursement over the past decade have resulted in shorter lengths of stay at hospitals and limitations on the amount of therapy that patients can receive post acute care. These changes present medical rehabilitation practitioners with a challenge to do more for patients with less time and resources. It is imperative that practitioners implement assistive technologies efficiently and effectively to help patients maximize independence as early in the rehabilitation process as possible and provide methods to augment and supplement direct care that can be utilized over time to support recovery. This is particularly true for patient conditions where physical recovery can be a slow process over many years [4], [5].

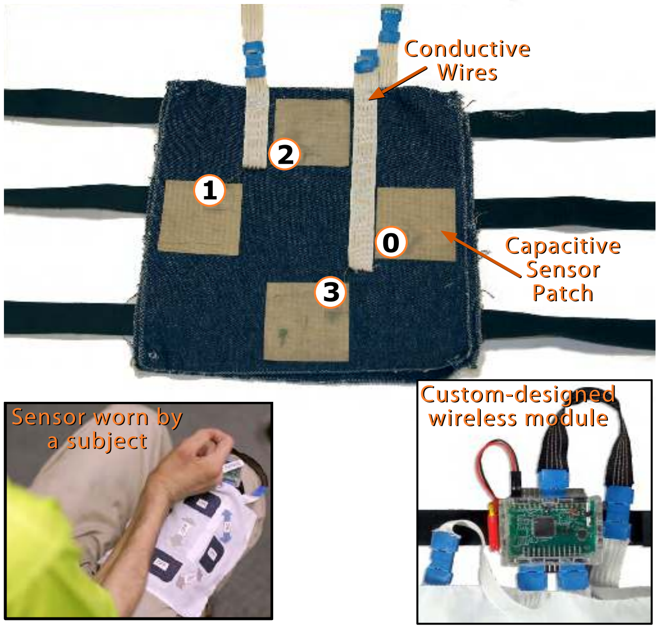


Fig. 1. The figure shows a prototype Inviz system worn by a patient with a spinal cord injury. The top figure demonstrates the capacitive sensor array sewn into the denim fabric using conductive wires. The data from the sensors is analyzed using our custom-designed wireless module which uses capacitance measurement ICs, an MSP430 micro-controller, and Bluetooth Low Energy (BLE) wireless module.

Gesture recognition-based environmental control systems are capable of allowing patients with mobility impairments greater control over their environment. Several techniques such as the use of inertial sensors, vision systems, and other forms of tracking can be used to capture body gestures [6], [7], [8], [9], [10]. Gesture recognition systems for individuals with mobility impairments, however, present a set of fundamental challenges that typical gesture recognition systems often fail to address. First, sensors for gesture recognition are intrusive, bulky, or expensive [11]. Eye tracking systems necessitate the use of mounted cameras while evoked-potential or touch-based systems use electrodes that can cause skin irritation and abrasion, conditions that can have a deleterious effect if unnoticed due to diminished sensation in the extremities. Second, existing systems are often not suitable for mobility impairments as they assume certain motions which a person may not be able to complete. There is a need, therefore, of systems that require minimal set-up and maintenance, and cause minimal fatigue and intrusiveness.

In order to address the above challenges, we have designed, implemented, and evaluated Inviz which uses wearable sensors built from textile-based capacitive-sensor arrays (CSA). These CSAs work on the principle of change in capacitance when there is movement in the proximity of the fabric capacitor plates. These plates can be sewn into clothing fabrics. Figure 1 illustrates a prototype Inviz system built using the capacitive plates and conductive threads sewn into the denim fabric. We have designed a low-power hierarchical signal processing algorithm that converts signals from a CSA to reliable gestures. Our prototype then uses these gestures to control appliances in the home. In Inviz, we support two broad categories of gestures (1) Swipes: moving the hand from one plate to another; and (2) Hovers: protruding the hand over a textile sensor plate and then retracting the hand from the plate. Through conversations with patients suffering from partial paralysis (e.g., C-6 spinal cord injuries), and their physical therapists, we have determined that swipes and hovers are gestures that are comfortable to perform.

Contributions: The design, implementation, and evaluation of Inviz present the following research contributions:

(1) **Textile-based Wearable Capacitive Sensor Arrays for Proximity Movement Detection:** We present textile-based CSAs as a novel and versatile sensing modality for remote movement and gesture recognition. The capacitive plates are built out of conductive textile and can be integrated as-needed into clothing or otherwise worn as a wearable garment, or integrated into the environment (e.g. furniture, wheelchair, car seats), so it is unobtrusive to the user. The CSA measures movement in the proximity of the capacitor plates, hence they render irrelevant any concerns of skin abrasion caused by touch-based sensors. Additionally, the integrity of the fabric plates is preserved by avoiding wear from continual touch. An array of plates reduce noise due to stray capacitance and electrical interference and can be reliably used for capturing attributes such as speed and direction of motion; and are sensitive enough to capture subtle body movements.

(2) **Hierarchical Low-Power Signal Processing for Gesture Recognition:** We present a hierarchical gesture recognition algorithm that distributes the processing into several tiers. These tiers include generating observations from the analog differences between plates, feature extraction from the generated observations, and using these features in a machine learning algorithm to determine the gesture with high accuracy. The computation involved is distributed among on-chip calculation using low-power hardware and general purpose micro-controllers, where the high-power controller is woken up only as-needed when interesting features are detected. The hierarchical processing provides accurate and low-power continuous gesture recognition.

(3) **Prototype Development and Evaluation:** We have designed, implemented, and evaluated an Inviz prototype that combines our CSA based system with a home automation system for environmental control. We evaluate the system on five subjects and demonstrate that the system can reliably detect gestures while consuming minimal power.

II. Related Work

Inviz builds on previous work on capacitive sensing, gesture recognition systems and signal processing algorithms for gesture recognition. Here we compare and contrast our work with the most relevant literature.

Capacitive Sensing: Inviz builds on previous work on capacitive sensing [12] applied to industrial, automotive, and healthcare applications [13], [14]. For instance, capacitive sensors have been used in positioning [15], [16]; humidity sensing [17], [18]; tilt sensing [19], [20]; pressure sensing [21]; and MEMS-based sensing [21]. Capacitors have also been applied as proximity sensors with applications to robotics, industrial monitoring and healthcare applications [22], [23], [24]. Products like Microchip's GestIC [**?**] have been introduced which allow for 3D gesture tracking using capacitive sensing using a rigidly-defined array shape. Inviz extends the use of capacitive sensing gesture recognition through two key novel contributions. First, Inviz uses an *array* of *textile-based* capacitive plates embedded into clothing to collect fine-grained movement data for paralysis patients, an application where capacitors have not been utilized, with unique challenges in remote sensing using imprecisely fabricated sensor elements. Typically precisely fabricated capacitor arrays (such as touch screens) are used for complex movement measurements, while previous work in textile capacitive sensors rely on touch and pressure sensing paradigms which do not present as many challenges as proximity detection. Second, Inviz uses a low-power hierarchical signal processing algorithm allows customization while dividing computation into low-power and high-power computational tiers. The tiered design allows the higher-power tier in the hierarchy to be switched off most of the time, leading to large energy savings, while still allowing us to invoke additional just-in-time computational resources. A key goal of this effort is a complex customizable recognition system with continuous monitoring capability.

Gesture Recognition Systems: The recent increase in adoption of home-automation technologies has spawned the development of new platforms for environmental control based on gestures. Vision based gesture-recognition systems [25], for instance, allow environmental control without physical contact with a controlling device. Vision-based systems, however, require blanket coverage of cameras that can be prohibitively expensive. Home automation techniques have recently been adapted to enable persons with limited mobility to have greater autonomy in environmental control. These mobility-limited targeted platforms include voice-activated systems [26], head-tracking [27], EOG based eye-tracking and inertial sensors [28]. Paramount to assistive technology in the home-automation space is the ability to customize the application to the specific needs and abilities of the user. The system should also be able to function regardless of user orientation or position within the home. Our CSA based approach allows for subtle and arbitrary gestures within any reference frame as Inviz is built into the patient's environment—in clothes, sheets, or wheelchairs.

Signal Processing for Gesture Recognition: There is a large body of work on applying signal processing techniques to gesture recognition. Learning techniques such as Hidden Markov Models [29], decision trees [30], and Bayesian inference [31] have been applied to converting data from sensors such as accelerometers to movement activities and gestures. Inviz uses machine learning techniques such as nearest-neighbor, Bayesian inferences, and decision trees in the highest tier of its signal processing hierarchy. The innovation in the design of Inviz is not in a particular signal processing algorithm but in combining feature extraction algorithms and learning techniques in a hierarchy of processing tiers such that energy consumption of the system can be minimized and gestures

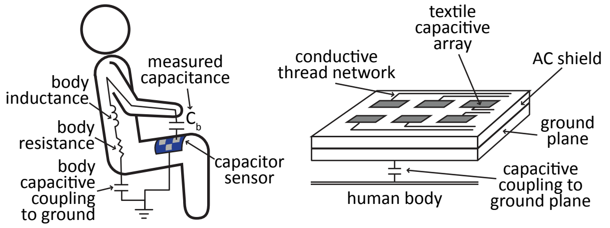


Fig. 2. The left figure shows the equivalent electrical circuit when a capacitive plate is placed on the leg and the user performs gestures using his hand. The body is capacitively coupled to the sensor ground using a sensor ground plane. The right figure illustrates a longitudinal cross sectional view of the sensor. The CSA is sewn into fabric. Data from the capacitive plates are collected using a network of conductive threads sewn into the fabric. The sensor array consists of two additional layers – an AC shield layer and a ground plane that comprise of conductive textile plates sewn into fabric.

can be recognized accurately using an array of textile-based capacitive plates.

III. System Goals

For long term adoption and continuous usage, the implementation of Inviz aspires to the following two goals:

Inviz must be minimally obtrusive and must require minimal battery recharges. One of the primary deterrent to continuous long-term use of existing environmental control systems used by patients with paralysis is their bulkiness [11]. To address this challenge, one of our primary goals is to make Inviz minimally obtrusive and possibly invisible to the patient. To this end, we use textile-based plates that can be embedded into items of daily use. Moreover, we design our signal processing and wireless system to consume as little power as possible, to minimize the number of battery recharges—a huge and sometimes under-appreciated impediment to the adoption of systems used by patients with limited mobility [11].

Inviz must avoid *precise* and touch-based gestures. Patients with paralysis have reduced skin sensation. Hence, touchpads [32] that require the users to perform precise touch gestures can cause skin abrasion. Moreover, precise gestures are difficult to perform for paralysis patients, and capabilities vary widely among patients. Our subject who suffers from type C-6 spinal cord injury (in Figure 1), for instance, is unable to perform precise touch or movement gestures. Hence, Inviz allows imprecise swipe and hover gestures that do not require physical contact. The proximity gesture recognition is enabled by CSAs, described in detail below.

IV. Capacitive Sensor Arrays for Gesture Recognition

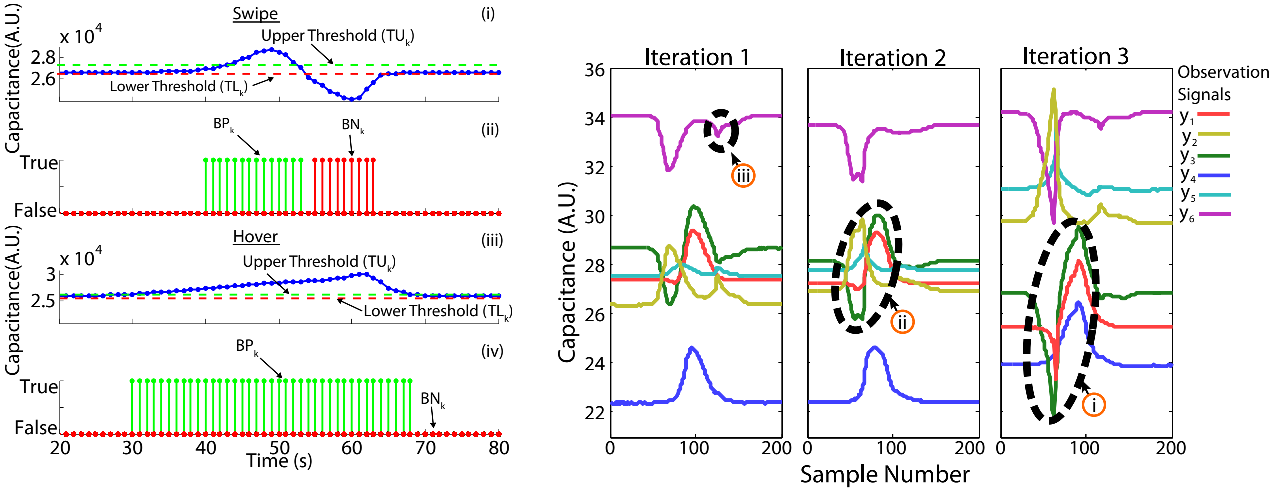
As described in the previous section, minimal obtrusiveness and avoiding precise touch-based gestures is key to adoption of environmental control systems for patients with limited mobility. While camera-based gesture recognition systems are a plausible remedy, blanket coverage of an area with cameras can be prohibitively expensive and would not be able to move with the patient. Our solution addresses this problem through the use of wearable textile-based capacitive plates built into clothing fabric. In our system conductive textile plates are sewn into fabric such as denim using conductive threads. Unlike touch-based sensing, a popular sensing modality for capacitive sensors, we focus on proximity sensing using an array of these capacitive sensor plates.

Capacitive sensors work on the principle of change in capacitance due to perturbation in the electric fields between the plates of the capacitor, making them highly *versatile*. Unlike accelerometers and gyroscopes that measure movement of the body to which they are attached, capacitive sensors can sense movement of remote bodies. Figure 2 illustrates the principle using an example where an array of capacitive plates is worn by a patient and gestures are performed by moving the hand in the vicinity of the array without touch. The body of the patient is capacitively coupled to the ground of the sensor system. When the hand is moved close to the capacitor plates, the capacitance C_b increases. Inversely, the value of capacitance C_b can be used to localize the hand with respect to the plates. The range of our capacitive plate of size 2 inch by 2 inch is close to 3 inches (7.6 cm). This range sufficient to prevent accidental touch and skin abrasion and can be adapted as the range is variable with plate size and shape. The longitudinal cross-section of our designed sensor is also illustrated in Figure 2. The top layer is a network of capacitive sensor plates connected via conductive threads. An AC shield plane minimizes parasitic capacitance and noise coupling. The ground plane capacitively couples the human body to the ground of the sensor and provides a common reference for the capacitance measurements.

One of the novel contributions of Inviz is in the use of an *array* of capacitive plates. An array of plates has several advantages over a single large plate. First, taking differentials between capacitor plates helps minimize noise due to stray movements in the vicinity of the plates. Secondly, an array of capacitor plates can help capture rich motion attributes such as velocity, and can be used to distinguish gestures. For example, Figure 3(a) graphs the analog difference between two capacitor plates when a hand is moved from one plate to the other with the subtraction of the two plates creating a peak, followed by a zero crossing, followed by a valley. Features such as width of the peak-valley pair can determine the speed of movement of the hand, and their causal order can determine direction of motion. Similarly, Figure 3(a) also graphs the analog difference in capacitance between plates when the user has their hand above a plate (termed a hover gesture). The width of the peak in this case can be used to determine the time of the hover. An array of capacitors can therefore be used to capture movement features such as time, velocity, and position of the hand with respect to the plate. Finally, an array of these sensors provide different vantage points for the same movement and can increase the reliability of gesture recognition. In Inviz, data from the CSA is converted into a reliable gesture using a hierarchical signal processing algorithm described below. Figure 3(b) illustrates the fundamental challenges in processing the raw capacitance data in the Inviz system. The figure shows the same gesture performed by the same user at three different times. As illustrated by the regions of interest i and ii, there is high irregularity in the signal produced by the user trying to produce the same gesture.

V. Hierarchical Signal Processing

While the evaluation of Inviz focuses on two gesture types, hovers and swipes, the goal of our hierarchical signal



(a) (b)

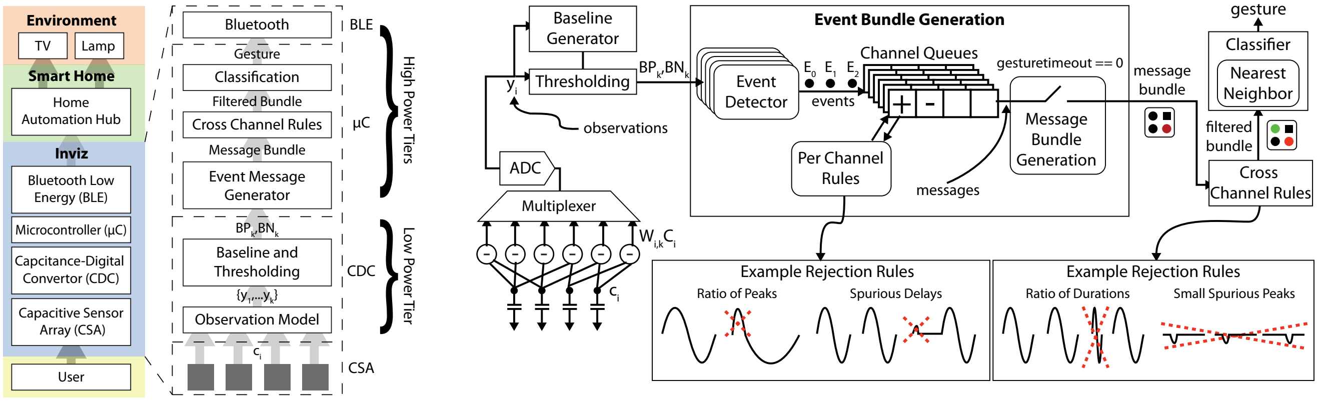
Fig. 3. (a) The figure plots the analog difference between capacitance measurements taken from two plates as the hand is swiped from one plate to the other and the hand is hovered on top of a single plate. It also plots threshold detection signals when the values are above a high threshold and below a low threshold. (b) The figure illustrates one of the fundamental challenges for Inviz. The figure shows the irregularity in the signals collected by our sensors when a subject performs the same gesture at three different times. For instance, in iteration 3 observation signal y_3 closely follows observation signal y_1, making the two almost indistinguishable. However, in iteration 2 that is not the case. Inviz mitigates this challenge using a hierarchical processing system that spans from sensor data collection to a machine learning algorithm.

processing algorithm, described below, is to present a general framework to determine gestures from an array of wearable plates while consuming minimal energy. Our signal processing architecture can be extended to support more complex gestures such as sign language alphabets [33].

Figure 4(a) illustrates our end-to-end gesture recognition system and Figure 4(b) illustrates how the data from the capacitor plates are transformed into gestures using our hierarchical signal processing architecture. Our system can train on imprecise swipe and hover gestures performed by a user and can be personalized to a specific user. Another key insight in our design is to break down the computation to a set of low- and high-power tiers. The low-power tier continuously processes data while waking up the high-power tier only for feature extraction and gesture classification. Such a hierarchical design provides high diligence and system availability at minimal energy consumption. Below we describe the different tiers in our processing hierarchy, as illustrated in Figure 4(b).

Observation Calculation: The observations are innately calculated at the lowest tier of the hierarchy. These observations are measurements from the capacitors taken as linear combinations of capacitance values from the plate array. These observations $\{y_1,..,y_k\}$ each follow an observation model $\sum_{i=1}^{i=n} W_{i,k} \cdot c_i$ where $W_{i,k} \in \{0,1,-1\}$, c_i represents the equivalent capacitance between a plate and ground and n is the number of sensor plates. Note that the measurements cannot be made concurrently in our system, since only a single data converter is used. Instead, the measurements are taken in a periodic sequential pattern to create a round of measurements, $[y_1,..,y_k]$. We control the linear combinations that are computed in the analog domain through low-power multiplexors. We employ, as demonstrated in this system, a pattern of differential measurements whereby analog subtractions between plates is calculated. The particular *ordering* of the measurements does not matter if the gestures are slow compared to the sampling rate. Furthermore, since we employ a machine learning approach in the higher-power tier in our system, as long as the same ordering is used for both training and testing, the measurement order is of minimal importance. The use of differential measurements rejects transient environmental noise including common noise among the plates. These differential measurements also form a receptive field most sensitive to motions in the proximity of the plates while being more insensitive to motions at a distance as compared to a single-ended measurement. Therefore, the differential measurements can also cancel noise due to stray movements far from the plates and can capture subtle movements close to the plates.

A characteristic response of a differential pair from a hand swipe over the two capacitor plates is shown in Figure 3(a) (i). In this case the hand passed successively over each plate. Likewise, when the hand only passes over a single plate, the characteristic differential response is shown in Figure 3(a) (iii). These two characteristic responses are detected in our system using a pair of threshold detectors capturing positive and negative *events* illustrated as the low and high thresholds in Figures 3(a) ((ii) and (iv)). The thresholds for the *events* are established relative to a baseline capacitance for each observation channel which is continually recalculated while there is only minimal changes in the capacitive data. The separation of the thresholds from the baseline was determined in our system using experimental data analysis using our prototype and programmed manually. Alternative designs and applications would require this threshold to be manually adjusted. The threshold detection is implemented in hardware using our ultra-low power measurement IC which supported threshold-crossing detection as well as an automatically adjusted baseline offset. Typically this threshold-detection functionality is provided for capacitance-based touch determination. We however exploit this generated signal for robust proximity motion detection using a challenging textile sensor. In addition to the irregularity of the sensors, the motions (gestures) themselves are much more irregular than a simple touch and cannot be defined



(a)

(b)

Fig. 4. (a) The figure illustrates our end-to-end home automation system. It demonstrates the data flow from the user to the environment through our system. The Inviz system encompasses two power tiers which includes low-power and high-power as represented in the figure. (b) The figure shows how data from modules are processed to generate gestures using a hierarchical signal processing architecture. The insets show examples of how anomalies that represent spurious events that are generated due to noise in the data are filtered, eliminating extraneous information that complicates classification.

as easily from capacitance signals. As shown in Figure 3(b), we are sensing more complex signals generated due to the conglomeration of hand, forearm, and wrist movements, as opposed to single-point finger touch. We next describe the feature extraction algorithm which uses the binary outputs of the digital threshold detectors to build higher-level features used in the final stages of machine learning-based classification.

Event Message Generators: The threshold signals serve two key purposes for event detection in the next level of our processing hierarchy. First, the temporal binary threshold signals are themselves the only representation of the signal passed to the event detectors. This simple compact representation of the signal minimizes the memory requirements for capturing the signal history and feeds into the simplicity of the real-time high-level feature extraction algorithm. Additionally, the binary signals serve a dual purpose as wake-up (interrupt) signals for the higher-level processor which remains in a low-power sleep mode until activity is detected. For each linear *observation* signal y_k, we define an upper and lower threshold, TU_k and TL_k respectively, on opposite sides of the baseline. We define two signals, the positive-peak binary signal BP_k and the negative-peak binary signal BN_k as follows:

$$\mathrm{BP}_k[n] = \begin{cases} \text{TRUE} & \text{if } y_k[n] > \mathrm{TU_k} \\ \text{FALSE} & \text{otherwise} \end{cases} \quad (1)$$

$$\mathrm{BN}_k[n] = \begin{cases} \text{TRUE} & \text{if } y_k[n] < \mathrm{TL_k} \\ \text{FALSE} & \text{otherwise} \end{cases} \quad (2)$$

where n is the sample number. We note that the first occurrence of a TRUE value for any BP_k or BN_k, after a period of inactivity, triggers the high-power processor to wake up. At the *event* detectors, the binary threshold signals' characteristics are analyzed to extract *event* features to form an *event* message. An *event* is signified as a period of a continuous TRUE value for BP_k or BN_k. The three *event* features generated for each *event* are (1) *arrival time*: defined as the delay from the first threshold crossing on any observation signal, (2) *duration*: length of time that a binary signal is TRUE; (3) *event polarity*: a binary symbol indicating which of BP_k or BN_k is TRUE. Additionally, a flag is set at the end of each *event* to signal the higher-level stage to process the *event* message.

Algorithm 1 CaptureEventMessages ($\Delta\mathrm{T} = \frac{1}{Sampling\ rate}$)

```
Event $E_1$ = first event.
$gesturetimeout$ = eventduration ($E_1$)
for every $\Delta T$ seconds
        IF $gesturetimeout$ == 0
            break
        end IF
        decrement $gesturetimeout$
        IF Event $E_i$ is collected from observation $i$.
            $gesturetimeout$ += eventduration($E_i$)
        end IF
end for
return TRUE
```

A critical challenge in online processing of multiple observation stages is to determine the amount of time that the high-power processor should remain awake to gather all *events*. This time determination is important since it is proportional to the energy consumed by the high-power processor. In our system, we maintain a counter called *gesturetimeout* that reflects this time. When *gesturetimeout* reaches 0, the *event* messages are propagated to an aggregation and filtering stage called *Message Bundle Generation* in Figure 4(b). Algorithm 1 describes how the value of *gesturetimeout* is updated. The key step in Algorithm 1 is to determine the duration of an *event* in an observation channel and increase *gesturetimeout* proportional to this duration. The duration of an event, described above, is indicative of two parameters: (1) the speed of performing the gesture and (2) when another *event* might occur on a different observation channel. The intuition behind Algorithm 1, therefore, is that the event message generator must wait for events at least for that duration of time. Once the events are determined, they are propagated to the aggregation and filtering module that performs domain-specific filtering.

In our prototype, we have found several instances where spurious events are generated due to noise in the data caused by undesired user actions and sensor displacement. Figure 3(b) (*iii*) illustrates four such spurious events that are an outcome of improperly performing gestures and are filtered by the aggregation and filtering stage, using the per-channel rules. (1) An *event* with much larger *duration* than all previous *durations* in the same channel erases all previous *event* messages from the channel queue; (2) When two *events* with the same *polarity* exist in the same channel, only the longer *event* is kept in the channel queue; (3) In a channel, if two newer messages exist with significantly higher *arrival time* and longer *duration* than an earlier message, the earlier message is deleted (4) If two *event* messages exist in a channel queue with a significant ratio of their *durations*, the shorter message is deleted. The goal of these filtering rules is to ensure that the spurious events do not reach the machine learning-based gesture classifier. Some of these rules are illustrated in Figure 4(b).

Once the messages are aggregated across observation channels, the *cross-channel rules* are applied to the aggregated *events* to filter noise and generate features for the machine learning algorithm. (1) If the total *event duration* of one channel, as defined by the sum of *event durations* in messages left in the queue at the time of gesture reporting, is much shorter than the average total *duration* of the other channels then the messages in the former channel are deleted; (2) If the max total *duration* across all channels is small, the gesture is ignored and the messages are purged; (3) *Events* are labeled as P (positive) or N (negative) *events* depending on if they are generated from a BP or BN signal respectively, otherwise they are labeled as NAE (not an event). If a channel has a complimentary pair of *events* positive-then-negative or negative-then-positive, they are combined to one message labeled PN or NP with a single *duration* calculated as the sum of the pair of *durations*. In case of NAE, the *duration* and *arrival times* are zero.

The resulting labeled *events'* features are passed to a gesture classifier. The features reported for each observation channel are: *Event* Label (P, N, PN, NP, NAE), *duration*, and *arrival time* with respect to the first *event*. The combination of *event* signatures on different observation channels is unique for a gesture, hence, the *event* features are important to distinguish between gestures. The *duration*, for instance, is representative of the speed of performing a gesture which is user-specific, and can help the machine learning algorithm distinguish between gestures among users. The *arrival time* for an *event* on an observation channel encodes the velocity of the gesture—the speed and direction of motion. Together these features help the machine learning algorithm, described below, infer the gestures accurately. The feature extraction and the machine learning classifying algorithm run in real-time on the micro-controller. We have experimentally verified that the above mentioned rule-sets are mandatory for a tractable embedded classification.

Machine Learning Algorithm: The final level in the hierarchy is a machine learning algorithm that takes as input a filtered message bundle and classifies the gestures. The machine learning algorithm is trained using gestures performed by individual subject. We have experimented with several machine learning algorithms such as Nearest Neighbor Classifier, Decision Tree Classifier, and Naive Bayesian Classifier. We compare the accuracies and complexity trade-offs in §VII. Once the algorithm determines a gesture, the Bluetooth low energy (BLE) module is woken up and the gesture is transmitted to a computer which controls a set of appliances using a custom home automation system.

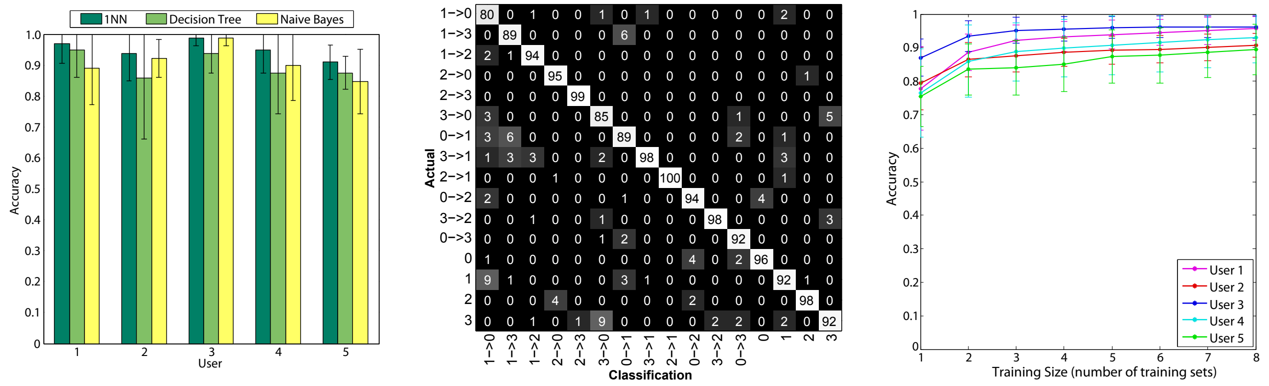
VI. Prototype Implementation

We have implemented a fully functional prototype of Inviz (illustrated in Figure 1) as an end-to-end cyber-physical system for home automation. Gestures recognized by the Inviz system are transmitted to a personal computer over BLE, and the PC device then controls appliances over Zwave connection using a Micasaverde Vera gateway. The Inviz prototype consists of a custom-designed PCB board with the capacitance measurement circuit, observation calculation and thresholding circuit (built into the capacitance measurement IC), an MSP430 micro-controller, and a BLE wireless module. The capacitive sensor plates were sewn into the denim fabric and attached to the data collection module using 4-ply conductive threads with a linear resistance of 50 Ω/meter. In our prototype implementation, we faced two challenges unique to designing textile-based wearable plates. First, the conductive threads are built by weaving silver-plated threads and non-conductive threads. Unfortunately, this leads to fraying on the ends of the thread and can cause microscopic shorts between adjacent threads which are difficult to diagnose, especially when vampire connectors were used to connect the thread to the data collection board. The second challenge was soldering onto the conductive threads which we mitigated using vampire FCC connectors.

VII. System Evaluation

The goal of Inviz is to provide accurate real-time gesture recognition for patients with limited mobility at minimal energy consumption. To this end, our evaluation of Inviz focuses on the following key questions. (1) How accurately does Inviz determine gestures across subjects? (2) What is the energy consumption of Inviz compared to a system that does not use the hierarchical signal processing architecture? (3) What are the trade-offs between accuracy of gesture recognition and training size and type of training data used? While answering these key questions, we also present micro-benchmarks on the energy consumption of different subsystems of Inviz and latency associated with different components of the Inviz prototype.

Experimental Setup: We performed our experiments on five adult subjects. While these subjects do not suffer from paralysis, we believe that they act as a baseline for evaluating the accuracy of our gesture recognition system. As future work (described in §VIII), we plan to perform a usability study with our prototype on patients suffering from paralysis. In our experimental setup, the subject wore the capacitive sensor on their thigh and performed swipe and hover gestures. Each subject performed an average of 180 gestures. The subjects performed between 9-12 gesture sets with each gesture set consisting of 16 gestures. With four plates in our prototype illustrated in Figure 1, the swipe gestures performed were the following: all combinations of $i \rightarrow j$, where $i \neq j$, and i and j are the plates numbered from 0 through 3. Similarly the gesture set included four hovers denoted by the plate numbers {0, 1, 2, 3}. Each subject was trained on how to perform the gestures before the experiments were performed. For all our accuracy results, we performed cross-validation. Below, we first present results on micro-benchmarks using our prototype followed by results on accuracy, energy consumption, and system trade-offs.



Actual \ Classification	1->0	1->3	1->2	2->0	2->3	3->0	0->1	3->1	2->1	0->2	3->2	0->3	0	1	2	3
1->0	80	0	1	0	0	1	0	1	0	0	0	0	0	2	0	0
1->3	0	89	0	0	0	0	6	0	0	0	0	0	0	0	0	0
1->2	2	1	94	0	0	0	0	0	0	0	0	0	0	0	0	0
2->0	0	0	0	95	0	0	0	0	0	0	0	0	0	0	1	0
2->3	0	0	0	0	99	0	0	0	0	0	0	0	0	0	0	0
3->0	3	0	0	0	0	85	0	0	0	0	0	1	0	0	0	5
0->1	3	6	0	0	0	0	89	0	0	0	0	2	0	1	0	0
3->1	1	3	3	0	0	2	0	98	0	0	0	0	0	3	0	0
2->1	0	0	0	1	0	0	0	0	100	0	0	0	0	1	0	0
0->2	2	0	0	0	0	0	1	0	0	94	0	0	4	0	0	0
3->2	0	0	1	0	0	1	0	0	0	0	98	0	0	0	0	3
0->3	0	0	0	0	0	1	2	0	0	0	0	92	0	0	0	0
0	1	0	0	0	0	0	0	0	0	4	0	2	96	0	0	0
1	9	1	0	0	0	0	3	1	0	0	0	0	0	92	1	0
2	0	0	0	4	0	0	0	0	0	2	0	0	0	0	98	0
3	0	0	1	0	1	9	0	0	0	0	2	2	0	2	0	92

(a) (b) (c)

Fig. 5. (a) The figure compares the average accuracy of gesture recognition across our five subjects for three machine learning algorithms: *Nearest Neighbor Classifier*, *Decision Tree Classifier*, and *Naive Bayesian Classifier*. The figure shows that the accuracies are comparable but the Nearest Neighbor classifier performs the best. (b) Confusion matrix illustrating the accuracy over all subjects expressed as a percentage. We use the Nearest Neighbor Classifier in this experiment. (c) The effect of training size on the accuracy of gesture recognition for the Nearest Neighbor Classifier for the five subjects. Each training set consists of 16 distinct gestures.

TABLE I. THE TABLE PRESENTS MICRO-BENCHMARKS ON POWER CONSUMPTION OF DIFFERENT COMPONENTS ON OUR CUSTOM DESIGNED DATA COLLECTION MODULE.

Module	Sleep power	Active Power
micro-controller	2.2 μ A	505.0 μ A
cap./observation meas. IC	152.0 μA (low power)	1148.0 μA
BLE	0.5 μA	8.1 mA (idle), 20.7mA (tx)

TABLE II. THE LATENCY OF EACH COMPONENT IN INVIZ.

Module	Latency
micro-controller wakeup	36 μs
micro-controller (ML algo)	286 ms
Bluetooth wakeup	450 μs

Micro-benchmarks: Table I and Table II presents the power consumption and the latency of different Inviz subsystems. The power consumption table motivates the need for a hierarchical signal processing architecture. The Bluetooth module consumes an order of magnitude more power than the micro-controller when active which in turn consumes four times more power than the capacitance/observation calculation hardware. Hence, by keeping the micro-controller and the Bluetooth module off until the event generation module generates interesting events can save a substantial amount of energy. The hierarchical design, however, is useful only if the transition cost associated with wakeup times of different modules is low. As illustrated in Table II, the wakeup latency associated with the Bluetooth and micro-controller wakeup is 450 μs and 36 μs respectively, demonstrating that the overhead of transition in the system is low. Additionally, it takes only 286 ms to execute the machine learning algorithm on the micro-controller, illustrating the efficiency of our implementation.

System Accuracy: Our first set of experiments focuses on evaluating the accuracy of Inviz in recognizing gestures. Figure 5(a) graphs the accuracy of recognizing the gestures for three machine learning algorithms (Nearest Neighbor Classifier, Decision Tree Classifier, and Naive Bayesian Classifier) across our five subjects. We chose these three classifiers since they represent algorithms with a wide range of computational needs. The Bayesian classifier and the decision tree classifier require training that may not be feasible on a micro-controller. However, once these classifiers are trained using them on the micro-controller is computationally feasible. The Nearest Neighbor classifier can be completely implemented on the micro-controller. The result shows that the Nearest Neighbor classifier performs the best with an average accuracy of 93%. Hence, in our prototype system we use the Nearest Neighbor classifier. Figure 5(b) is a confusion matrix that illustrates the percentage of gestures classified and misclassified for all the sixteen gestures. The experiment presents data collected from all our subjects. The figure shows that the lowest accuracies are for the swipe gestures performed when plate 1 was involved. Swipe gestures, 1 → 0, 1 → 3, and 1 → 2, have accuracies of 80%, 89%, and 94% respectively. In our experiments, our prototype was strapped onto the right leg and the subjects used their right hand to perform the gestures. Based on the orientation of the sensor plates illustrated in Figure 1 swipes over plate 1 from any other plate will cause the subject to pass over other plates, causing the misclassifications. If the orientation is changed this miss-classifications will occur on gestures performed on other plates. However, even with this interference with neighboring plates, our system is able to infer the gestures with an average accuracy of close to 93%.

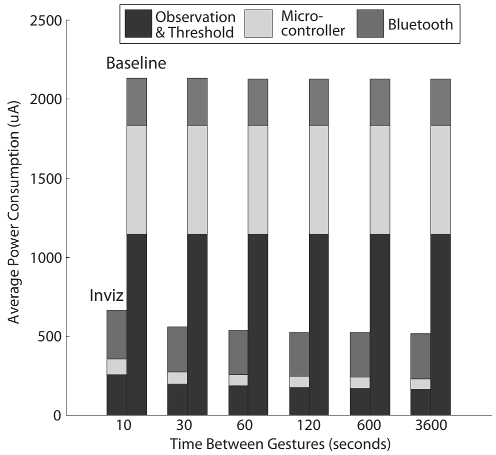


Fig. 6. The figure shows the average power consumption of the Inviz system as the frequency of performing gestures is changed. The bars are also broken down into the energy consumption of different components of the system.

Energy Consumption: The next set of experiments explores the energy consumption of Inviz. Figure 6 compares the energy consumption of Inviz with a system that does not

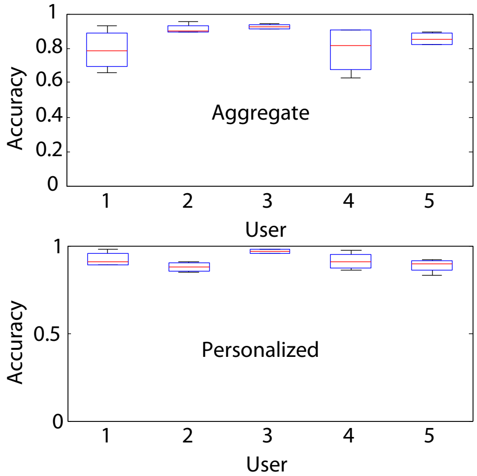


Fig. 7. The figure illustrates the need for personalized training for high accuracy. When an *aggregated* training set is used, there is high variance in accuracy for two subjects. The variance is lower and the accuracy higher when the training set is generated per-user (personalized).

use the hierarchical signal processing algorithm. The system (termed Baseline) processes all the data on the micro-controller and wakes up the Bluetooth module only when a gesture is detected. The figure illustrates the average power consumption of Inviz when gestures are performed at the rate of once every 10 seconds, 30 seconds, 1 minute, 2 minutes, 10 minutes , and 60 minutes (last two extrapolated from measurements). The figure also shows the breakdown of the power consumed by different components of Inviz, namely the observation and threshold calculation hardware, the micro-controller, and the Bluetooth module. We can draw three conclusions from the figure. First, the absolute power consumption of Inviz is low and is close to 525 μ A (1.7 mW) when gestures are performed once every 2 minutes, which is a very high gesture performing frequency. On a 1000 mAh battery, the system would last for 2000 hours (83 days) on a single charge. Second, the system consumes 4X lower power than a system that does not use the hierarchical architecture. Third, we find that the two primary energy consumers in our system are the Bluetooth module and the observation threshold calculation hardware. To address this problem, as future work, we plan to implement the observation and thresholding algorithms using low power analog sub threshold circuits.

System Trade-offs: In our last set of experiments, we evaluate the tradeoffs associated with Inviz. Specifically, we study the accuracy of recognizing gestures in Inviz as the training size increases. Figure 5(c) graphs the change in accuracy of the Nearest Neighbor classifier as the training set is increased for the five subjects in our study. One training set comprises of 16 gestures in this experiment. From the figure, we find that as the number of training sets increases the average accuracy improves, however, the accuracy saturates after five training sets. The figure demonstrates that the amount of training required for our system is low. This is a consequence of the intelligent event generation and filtering that we perform on the sensor data that reduces the complexity of the machine learning classifier. The next trade-off that we study is the type of training used. We study two cases: *personalized training* where the classifier is trained per subject and *aggregate training* where a single training set that is generated by randomly selecting five training sets across subjects. Figure 7 compares the accuracy of gesture recognition for the two training approaches. The figure shows that there is high variance in the accuracy of subject 1 and subject 4 when *aggregate training* is used. However, the variance is low and accuracy is higher when *personalized training* is used. In our subject pool, subjects 1 and 4 have shorter forearms and legs compared to the other subjects. Hence, if the aggregate training set did not include data sets from these subjects, the classifier is unable to capture gesture attributes unique to these subjects. This problem is addressed by using personalized per-subject training—a fundamental design principle in Inviz.

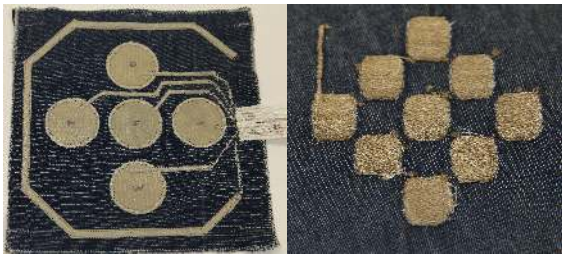


Fig. 8. As future work, we are exploring different capacitor plate shapes and placement. The left figure shows circular plates with a boundary plate that can be used to determine whether the user's is performing a gesture. The right figure demonstrates a future prototype where we plan to design the capacitive array using embroidered patches built using conductive threads.

VIII. Future Work

We are working on several future extensions to the design and evaluation of Inviz. We have underlined some of the avenues of future work that we are pursing.

Embroidering capacitive patches: We have tested Inviz on a small set of textile capacitive patches. As future work, we are working on automating the process of creating the network of capacitive patches using embroidery. Figure 8 illustrates an example network of capacitive patches generated using embroidery machine. Our goal is to study the scalability issues with processing a large number of patches for fine-grained movement and gesture recognition. Moreover, we are experimenting with different shapes and sizes of capacitive patches. For example, Figure 8 illustrates a prototype that uses circular patches instead of rectangular patches.

Miniaturization and self-sustainable sensing: One of our future goals is to minimize the size of our data collection module and further reduce the power signature of the system. To this end, we are working on fabricating a custom IC that will replace the micro-controller and the capacitance/observation calculation modules. Further, with a custom design we conjecture that the power signature can be further reduces and it is possible to power the system by harvesting indoor light.

Medical data collection: As future work, we plan to test the efficacy of our system on real paralysis patients. We have build a collaboration with the University of Maryland, Medical School to help recruit subjects with varying degree of paralysis to test our capacitive sensors for environment control.

IX. Conclusion

In this paper we present Inviz, a wearable capacitive sensor-based gesture recognition system for environmental control in individuals with limited mobility. Inviz uses textile-based capacitive plates sewn into fabric such as denim as proximity sensors. The Inviz system uses a novel hierarchical signal processing algorithm where the computation is broken down into several low power and high power tiers. The low power tiers maintain high availability of the system at low power and the higher power tiers are woken up only when required to perform sophisticated signal processing. We have prototyped a fully functional Inviz system and evaluated it in the context of a home automation system, and show that it can infer gestures with high accuracy and low energy consumption.

ACKNOWLEDGEMENT

This material is based upon work supported by the National Science Foundation under awards CNS-1305099 and IIS-1406626, CNS-1308723, CNS-1314024, and the Microsoft SEIF Awards. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF or Microsoft.

REFERENCES

[1] Margaret Jean Hall, Shaleah Levant, and Carol J DeFrances. Hospitalization for stroke in us hospitals, 1989–2009. *Diabetes*, 18(23):23, 2012.

[2] Jean A Langlois, Wesley Rutland-Brown, and Marlena M Wald. The epidemiology and impact of traumatic brain injury: a brief overview. *The Journal of head trauma rehabilitation*, 21(5):375–378, 2006.

[3] National spinal cord injury statistical center.

[4] Sarah Steinbaum, J Duncan Harviel, Jonathan H Jaffin, and Marion H Jordan. Lightning strike to the head: case report. *The Journal of Trauma and Acute Care Surgery*, 36(1):113–115, 1994.

[5] Marilyn Seif Workinger and Ronald Netsell. Restoration of intelligible speech 13 years post-head injury. *Brain Injury*, 6(2):183–187, 1992.

[6] Holger Junker, Oliver Amft, Paul Lukowicz, and Gerhard Tröster. Gesture spotting with body-worn inertial sensors to detect user activities. *Pattern Recognition*, 41(6):2010–2024, 2008.

[7] Ari Y Benbasat and Joseph A Paradiso. An inertial measurement framework for gesture recognition and applications. In *Gesture and Sign Language in Human-Computer Interaction*, pages 9–20. Springer, 2002.

[8] Sung-Jung Cho, Jong Koo Oh, Won-Chul Bang, Wook Chang, Eunseok Choi, Yang Jing, Joonkee Cho, and Dong Yoon Kim. Magic wand: a hand-drawn gesture input device in 3-d space with inertial sensors. In *IWFHR-9 2004*, pages 106–111. IEEE, 2004.

[9] Ying Wu and Thomas S Huang. Vision-based gesture recognition: A review. In *Gesture-based communication in human-computer interaction*, pages 103–115. Springer, 1999.

[10] Yi Xia, Zhi-Ming Yao, Xian-Jun Yang, Sheng-Qiang Xu, Xu Zhou, and Yi-Ning Sun. A footprint tracking method for footprint tracking method for gait analysis. *Biomedical Engineering: Applications, Basis and Communications*, 26(01), 2014.

[11] Susan Koch Fager and Judith M Burnfield. Patients' experiences with technology during inpatient rehabilitation: opportunities to support independence and therapeutic engagement. *Disability and Rehabilitation: Assistive Technology*, 9(2):121–127, 2013.

[12] Gabe Cohn, Sidhant Gupta, Tien-Jui Lee, Dan Morris, Joshua R Smith, Matthew S Reynolds, Desney S Tan, and Shwetak N Patel. An ultra-low-power human body motion sensor using static electric field sensing. In *ACM Ubicomp*, pages 99–102. ACM, 2012.

[13] Jingyuan Cheng, Oliver Amft, and Paul Lukowicz. Active capacitive sensing: Exploring a new wearable sensing modality for activity recognition. In *Pervasive Computing*, pages 319–336. Springer, 2010.

[14] Hyun Kyu Ouh, Jungwoo Lee, Sangyun Han, Hyunjip Kim, Insik Yoon, and Soonwon Hong. A programmable mutual capacitance sensing circuit for a large-sized touch panel. In *ISCAS*, pages 1395–1398. IEEE, 2012.

[15] Yang Piao, Xiaohui Xiao, Xiaozhao Wang, Quan Zhou, Kun Fan, and Pinkuan Liu. Conditioning circuit for capacitive position sensor with nano-scale precision based on ac excitation principle. In *IEEE EIT*, pages 1–6. IEEE, 2011.

[16] Pavel Ripka and Alois Tipek. *Modern sensors handbook*. Wiley. com, 2013.

[17] Lion Precision. Capacitive sensor operation and optimization. Technical report, Tech. rep. http://www. lionprecision. com/tech-library/technotes/cap-0020-% sensor-theory. html, 2006.

[18] Peter Gründler. Conductivity sensors and capacitive sensors. *Chemical sensors: An introduction for scientists and engineers*, pages 123–132, 2007.

[19] L Zhao and EM Yeatman. Micro capacitive tilt sensor for human body movement detection. In *BSN 2007*, pages 195–200. Springer, 2007.

[20] Edin Terzic, Romesh Nagarajah, and Muhammad Alamgir. A neural network approach to fluid quantity measurement in dynamic environments. *Mechatronics*, 21(1):145–155, 2011.

[21] Jithendra N Palasagaram and Ramesh Ramadoss. Mems-capacitive pressure sensor fabricated using printed-circuit-processing techniques. *Sensors Journal, IEEE*, 6(6):1374–1375, 2006.

[22] Pei-Hsuan Lo, Chitsung Hong, Shih-Hsiung Tseng, Jen-Hao Yeh, and Weileun Fang. Implementation of vertical-integrated dual mode inductive-capacitive proximity sensor. In *MEMS 2012*, pages 640–643. IEEE, 2012.

[23] Pei-Hsuan Lo, Chitsung Hong, Sung-Cheng Lo, and Weileun Fang. Implementation of inductive proximity sensor using nanoporous anodic aluminum oxide layer. In *Solid-State Sensors, Actuators and Microsystems Conference (TRANSDUCERS), 2011 16th International*, pages 1871–1874. IEEE, 2011.

[24] Se Dong Min, Jin Kwon Kim, Hang Sik Shin, Yong Hyeon Yun, Chung Keun Lee, and Myoungho Lee. Noncontact respiration rate measurement system using an ultrasonic proximity sensor. *Sensors Journal, IEEE*, 10(11):1732–1739, 2010.

[25] Prashan Premaratne and Q Nguyen. Consumer electronics control system based on hand gesture moment invariants. *IET Computer Vision*, 1(1):35–41, 2007.

[26] Hua Jiang, Zhenduo Han, Peter Scucces, Sean Robidoux, and Ying Sun. Voice-activated environmental control system for persons with disabilities. In *Bioengineering Conference, 2000*, pages 167–168. IEEE, 2000.

[27] Sing Bing Kang. Hands-free interface to a virtual reality environment using head tracking, December 28 1999. US Patent 6,009,210.

[28] A Nelson, J Schmandt, P Shyamkumar, W Wilkins, D Lachut, N Banerjee, S Rollins, J Parkerson, and V Varadan. Wearable multi-sensor gesture recognition for paralysis patients. In *Sensors, 2013 IEEE*, pages 1–4. IEEE, 2013.

[29] Andrew D Wilson and Aaron F Bobick. Parametric hidden markov models for gesture recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(9):884–900, 1999.

[30] Gaolin Fang, Wen Gao, and Debin Zhao. Large vocabulary sign language recognition based on hierarchical decision trees. In *Proceedings of the 5th international conference on Multimodal interfaces*, pages 125–131. ACM, 2003.

[31] Pedro Domingos and Michael J. Pazzani. On the optimality of the simple bayesian classifier under zero-one loss. *Machine Learning*, 29(2-3):103–130, 1997.

[32] Wayne Westerman, John G Elias, and Alan Hedge. Multi-touch: A new tactile 2-d gesture interface for human-computer interaction. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 45, pages 632–636. SAGE Publications, 2001.

[33] Rung-Huei Liang and Ming Ouhyoung. A real-time continuous gesture recognition system for sign language. In *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*, pages 558–567. IEEE, 1998.