

# Gait Analysis for Fall Prediction Using Hierarchical Textile-based Capacitive Sensor Arrays

Rebecca Baldwin, Stan Bobovych, Ryan Robucci, Chintan Patel, Nilanjan Banerjee  
CSEE University of Maryland, Baltimore County  
{rebald1,sb9,robucci,cpatel2,nilanb}@umbc.edu

**Abstract**—Falls are a major cause of injuries in adults above the age of sixty-five. The economic aftermath of falls and their consequent hospitalization can be huge, totaling more than 30 billion dollars in 2010 alone. A plausible way of mitigating this problem is accurate prediction of future falls and taking proactive remedial action. Spatio-temporal variation in gait is a reliable indicator of a future fall, however, existing systems focus on gait analysis in clinical settings and are not tuned towards continuous gait analysis. In this paper, we present the design of a novel textile capacitive sensor array-based system built into clothing that can reliably capture spatio-temporal gait attributes in a home setting. A key novel research contribution of our work is a context-aware hierarchical signal processing architecture that breaks down the signal processing algorithm into a hierarchy of processing elements. The lower power processing components perform generic feature extraction using observations derived from the capacitor plates, while the higher-level processors aggregate features to infer gait attributes such as stride speed and inter-leg spacing. The system activates the higher power processing elements only when it detects walking. We have prototyped our system using textile capacitive plates built into an ace-bandage and a custom FPGA-based system and show that our system can accurately detect gait attributes that have high correlation with falls, while consuming minimal energy as estimated for a multi-clock-domain 180-nm IC.

## I. INTRODUCTION

In 2010 alone, 2.3 million fall injuries and 662 thousand hospitalizations due to falls were reported in the United States [1]. Today one in every three adults above the age of 65 fall [1]. The healthcare costs associated with the aftermath of falls are gargantuan, totaling 30 billion dollars in 2010 alone [1]. While fall detection methods can help provide immediate remedy to the patient, fall prediction and prevention are the only plausible ways to minimize the economic and health impact of falls. While observable factors such as improper medication can be a valid predictor of a future fall, the spatio-temporal variation in the gait of an individual can reliably predict future falls [2], [3]. For instance, researchers in Gerontology have observed that increase in stride-to-stride variability, stride speed, and spatio-temporal variation in inter-leg spacing have a high correlation to falls [4]. Unfortunately, these variations are usually small and often occur over a long period of time [2], so capturing these variabilities solely in a clinical setting creates several difficulties [3].

To reliably evaluate the risk that an individual may have a future fall, his gait must be monitored continuously in a home setting. Designing systems that can continuously monitor spatio-temporal variations in an individual's gait, however, faces unique challenges. First, for long-term adoption, the system should ideally be minimally obtrusive and invisible to the user. At the same time, it should be able to accurately

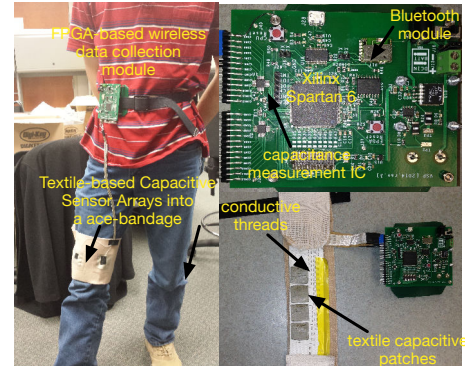


Fig. 1. The figure shows the prototype of our system. Capacitive Sensor Arrays built out of conductive textile are sewn into an elastic bandage. The data is collected and processed using a FPGA-based wireless module. The right subfigure shows the zoomed-in version of our custom-defined FPGA board, and the sensor array.

collect gait attributes such as variation in inter-leg spacing when the user is walking and changes in stride speeds over time. Second, the system should have a long battery lifetime, minimizing the need to recharge batteries. Existing systems rely either on camera-based techniques or accelerometers built into smartphone to perform such an analysis (<http://microberts.nl>). While depth cameras such as the Kinect (<http://www.xbox.com/en-US/kinect>) can help collect data on an individual's gait when the camera is facing the user, it cannot reliably collect data at all possible locations. Smartphone sensors such as accelerometers on the other hand can infer certain gait attributes such as stride speed, but cannot determine more fine-grained gait features such as the variation in the distance between the legs when the user is walking. Moreover, accuracy of using the smartphone sensors is predicated on where the user carries his smartphone, i.e., whether it is in the user's pocket or in his hands.

To address the above challenges, we present a system that uses textile-based capacitive sensor arrays built into wearable ace bandages that can continuously collect data on a user's gait. Figure 1 illustrates the prototype of our system. The textile sensors can be built into clothing of the user and worn at all times. The system uses a novel hierarchical low-power processing subsystem to accurately detect two attributes of gait—stride speed and inter-leg spacing of a walking individual—that have a high correlation with falls [2].

**Research Contributions:** The design, implementation, and evaluation of our prototype system presents three novel research contributions.

(a) **Wearable textile-based capacitive sensor arrays (CSAs) for gait analysis:** We present the use of flexible

textile-based capacitive sensor arrays built into items of daily use, e.g. an ace bandage determining gait analysis for fall prediction. Our sensors measure changes in capacitance due to motion in the proximity of the sensors. An array of these capacitive sensors help cancel noise in the environment, provide high spatial resolution, and sense modality for fine-grained motion detection.

(b) **Context-aware hierarchical signal processing for low-power wearable sensors:** We present a hierarchical signal processing architecture that breaks down the signal processing algorithm for calculating the gait attributes into a hierarchy of processing stages. The low power processing units calculate observations from a collection of capacitive sensors and extract features from the data relevant for motion detection. The more centralized processing unit aggregates features from multiple low power processing units to determine gait attributes. The processing stages are activated based on relevant context. For instance, the low power processing units perform feature extraction only when walking is detected and the centralized processor performs analysis only when relevant features are extracted by the lower level processors. We demonstrate that the hierarchical architecture can accurately detect gait attributes while consuming minimal energy.

(c) **Prototype development and evaluation:** We have prototyped our system using an array of rectangular capacitive textile sensors on an ace-bandage and a FPGA-based processing unit. Our evaluation using the prototype shows that the system can measure stride speeds with 85% accuracy and inter-leg spacing with an accuracy of 98%. We further evaluate the power consumption of our hierarchical signal processing architecture by synthesizing the FPGA design in a 180-nm CMOS process. We show that the power consumption of our hierarchical system is four times lower than a system where a single centralized processor is used for all the processing.

## II. RELATED WORK

Our work on using hierarchical capacitive sensing for gait analysis and fall prediction builds on previous work on gait and fall detection sensors, hierarchical signal processing, and capacitive sensors. Here we compare and contrast our work with the most relevant literature.

**Gait and fall detection sensors:** In the past, gait analysis for predicting falls used manual techniques such as using absorbent paper [5], talcum powder [6] and ink-saturated pads [2]. Modern commercial products today use cameras (vicon.com), force-based sensors (kistler.com), accelerometers (mroberts.nl), and electronic walkways (www.gaitrite.com). The commercial systems that are used for nomadic gait monitoring only capture limited gait attributes. For example, accelerometers can measure stride speed but cannot measure variations in inter-leg spacing. Portable electronic walkways such as GaitRite (www.gaitrite.com) use pressure mats that can be used for clinical trials but is difficult to use for continuous gait monitoring. Our wearable capacitive sensors are portable and can be used for remote and continuous gait monitoring.

**Hierarchical signal processing:** Another novel contribution of our work is the design and implementation of a hierarchical, context-aware signal processing architecture to extract gait attributes from capacitive sensor arrays. The hierarchical architecture can help minimize the energy consumption of the

system by selectively keeping the higher power subsystem switched off when not required. We use peak-valley detection and IIR filtering in our algorithm, however, our architecture can support more complex machine learning algorithms such as Hidden Markov Models [7], [8] and higher order learning techniques [9], [10].

**Capacitive sensors:** Our system builds on previous work on capacitive sensing [11] applied to industrial, automotive, and healthcare applications [12]. For instance, capacitive sensors have been used in positioning [13], humidity sensing [14], tilt sensing [15], pressure sensing [16], and MEMS-based sensing [16]. The proximity sensor is a critical device with applications to robotics, industrial monitoring, mobile phone touch screens and healthcare [17], [18]. Our approach extends this body of work in two ways. First, we use textile based capacitive arrays to design unobtrusive sensors embedded into clothing to gather fine-grained movement data, an application where capacitors have been minimally utilized. Second, we innovate in the concerted use of capacitive sensor arrays and hierarchical signal processing algorithms that can continuously detect gait attributes at low power.

## III. TEXTILE-BASED CAPACITIVE SENSOR ARRAYS

To capture all attributes of an individual's gait, a continuous 3D model of the leg movements while the user is walking is required. However, capturing this 3D model is infeasible without an elaborate imaging and 3D modeling system. Ultimately, we aspire to build such a sufficient 3D model of a individual's limb movements using low-cost minimally-obtrusive wearable sensors. As a first step towards that goal, this paper focuses on measuring a subset of gait attributes such as stride speed and inter-leg spacing. To capture these attributes, we use textile and conductive-thread based capacitive sensor arrays (CSA). While capacitive sensing is used in multi-touch screens [19] and touch sensors built of wearable fabrics [20], our approach explores more robust *proximity* sensing using wearable fabrics. To understand how these sensors work, Figure 2(a) diagrams an example sensor array built into an ace bandage on one leg. The effective capacitance between the plates change as the other leg moves into the vicinity of the plates. As the leg moves close to the plates it forms a set of series capacitors with the body. This also works at greater distances, enabling more-distant proximity sensing, though there are some challenges. For instance, stray capacitances exist that divert current away from the series measurement path affecting capacitance measurement. At close proximity, the sensor-body capacitances are large and the stray capacitances can be ignored. For larger distances, we mitigate the effect of stray capacitances by using arrays of sensors and analyzing changes over time rather than instantaneous individual values. Our hierarchical signal processing algorithm builds observations by combining data from capacitor plates using an observation model (described in §IV) to determine the approximate location, velocity, and acceleration of a body part with respect to the plates.

The use of capacitive sensor arrays (CSA) built into clothing has several advantages over conventional sensors such as pressure sensors and accelerometers used for gait analysis. First, pressure sensors and accelerometers can capture a narrow set of gait movements localized to the part of the body where they are placed. Because the CSAs are more

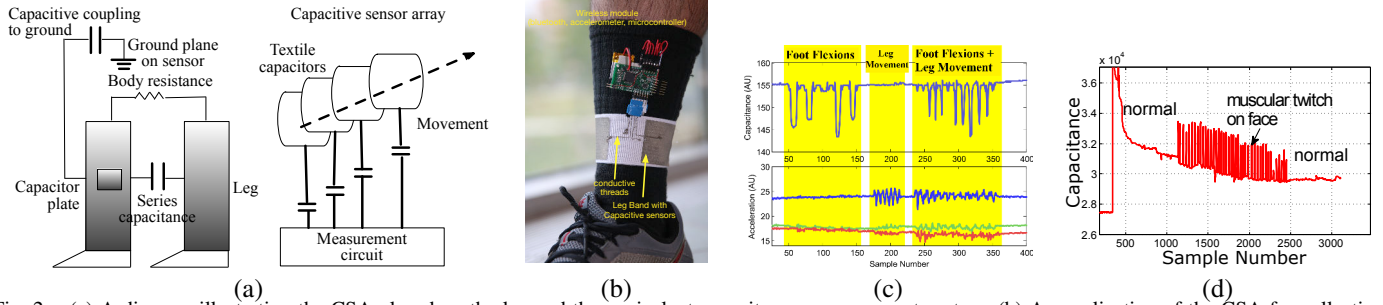


Fig. 2. (a) A diagram illustrating the CSA placed on the leg and the equivalent capacitance measurement system. (b) An application of the CSA for collecting foot flexion movements. (c) Data from the CSA and the accelerometer shows that the CSA can reliably capture foot flexions while the accelerometer can capture leg movement. (d) The sensitivity of the CSA is demonstrated using an experiment where the CSA is placed as a band aid on the face of an individual. The sensors are able to capture subtle muscular twitching on the cheek.

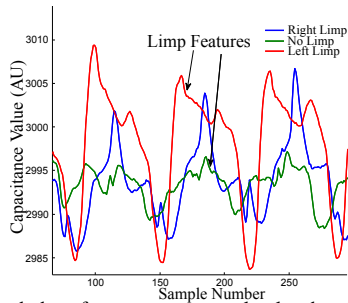


Fig. 3. Processed data from our sensor clearly showing the limp in the subject’s movement

versatile, they can capture subtle movements that occur in the proximity of the sensors. To demonstrate a scenario where the accelerometers cannot capture a specific leg movement, we present Figure 2 (b) and (c). Figure 2(b) shows a prototype band, similar to our ace bandage prototype worn around the ankle of an individual. The band consists of a capacitor sensor array and an accelerometer. Figure 2(c) presents data captured using the band when a subject performed three types of leg movements: (a) ankle flexions without intentionally moving the leg; (b) moving the leg without ankle flexions; and (c) moving the leg and performing ankle flexions. The figure shows that the capacitive sensors are able to reliably capture ankle flexions while the accelerometer can capture leg movements<sup>1</sup>. Therefore, by combining the two sensors we can reliably capture both ankle flexions and leg movements. Secondly, the CSAs have high sensitivity, are low cost (less than a cent a piece), and consume low power. To demonstrate the sensitivity of motion capture using a CSA, we present Figure 2(d). The figure plots the capacitance measured using textile-based capacitive sensors built into a band-aid placed on the cheek when a subject performed facial muscular twitching. The figure shows that the capacitive plate is able to reliably capture this subtle motion. Further, Figure 3 shows data from our ace-bandage prototype worn by an individual with a limp. The figure illustrates that our sensors can capture the limp reliably, demonstrating the sensitivity and versatility of the CSAs. While in this paper, we target monitoring leg movements, our overarching goal is to build a methodology for using this new type of sensor measurement on various areas of the body for a variety of body motion sensing applications. The effort requires complementary signal processing algorithm

<sup>1</sup>Ankle flexions during sleep have a high correlation with determining restless leg syndrome [21]

and hardware implementation framework. We next describe our signal processing framework that analyzes the data from the capacitive sensor array.

#### IV. HIERARCHICAL SIGNAL PROCESSING

Each textile capacitor plate measures the capacitance between one leg and the plate (depicted as the capacitor between the two legs in Figure 2(a)). The capacitance increases as one leg comes closer to the plate, or inversely, the capacitance value provides a measure of the location of the leg with respect to the plate. Our goal, however, is to extract other motion features such as velocity in addition to position and consequently convert these features into gait attributes such as stride speed and distance between legs. To accomplish this, we use an array of capacitor plates. Data from this array is processed using a hierarchical signal processing architecture illustrated in Figure 4(a). Several of the processing elements in our architecture are activated selectively to conserve computational effort and energy; and the data through communication channels between the processing stages is dynamically varied according to context to conserve power.

At the lowest level of the hierarchy, data from multiple capacitive plates are aggregated into *observations*, as shown in Figure 4(a). Aggregating capacitor channels into observations is important for extracting important motion attributes such as location, direction of motion, and speed of movement. These observations are described using the following observation model.

$$\begin{bmatrix} y_1 \\ \dots \\ y_k \end{bmatrix} = \begin{bmatrix} W_1^1 & \dots & W_n^1 \\ \dots & \dots & \dots \\ W_1^k & \dots & W_n^k \end{bmatrix} \cdot \begin{bmatrix} c_1 \\ \dots \\ c_n \end{bmatrix} \quad (1)$$

where  $y_1, \dots, y_n$  are the observations,  $W_i$ ’s can take a value of 0, 1, or -1, and  $c_1, \dots, c_n$  are the outputs from the  $n$  capacitor plates. To understand how these observations work, let us assume four capacitor plates  $C_1, \dots, C_4$ . Let us assume that there are two observations  $y_1 = c_2 + c_3$  and  $y_2 = c_2 - c_3$ . The observation  $y_1$  is roughly equivalent to data from a capacitor whose plate is the union of the capacitor plates  $c_2$  and  $c_3$ .  $y_1$  is maximized when the second leg is in the center of the two plates, hence can be used to localize a limb with respect to the plates.  $y_2$  is calculated from the differential between two plates and can be used to track motion as the leg moves from plate  $c_2$  to  $c_3$  and vice-versa. These observations, therefore can be used to track location and movement of objects in

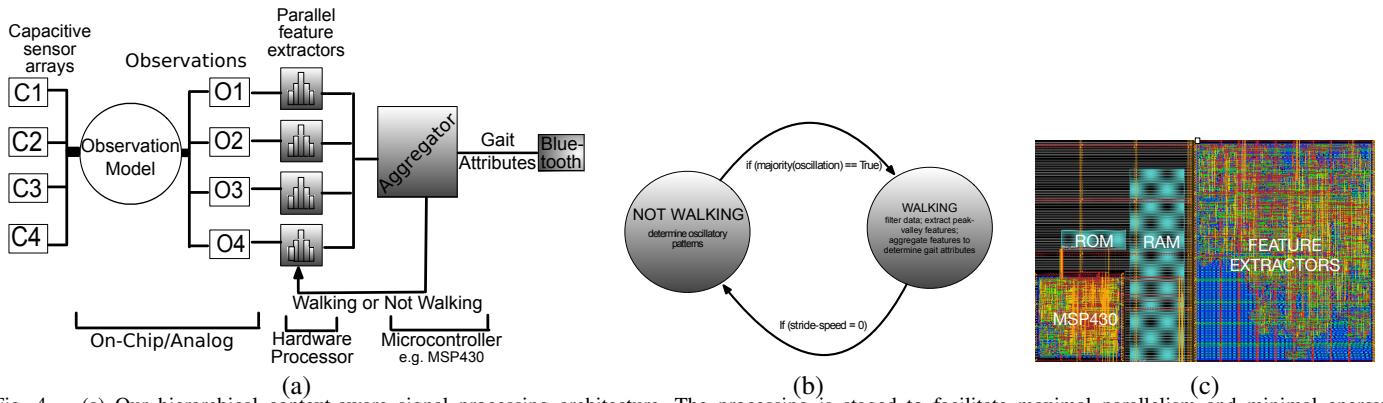


Fig. 4. (a) Our hierarchical context-aware signal processing architecture. The processing is staged to facilitate maximal parallelism and minimal energy consumption. Higher levels of the hierarchy can be switched off till the lower stages determine interesting features. The feature extraction is optimized based on the underlying context (walking or not walking in our prototype system). (b) The processing is driven by the underlying context. If the system is in the “not walking” state it looks for oscillatory patterns to determine if the user is walking and if the system is in the “walking state” the feature extraction and aggregation is tuned to extracting gait attributes such as stride width and stride speed. (c) Our synthesized cores place in an IC floor plan .

the proximity of the plates. A collection of these observations help us reliably determine strides and inter-leg spacing—gait attributes whose temporal variations have high correlation with imminent falls. The observation model is implemented on the capacitance measurement IC in hardware, and in the future, we plan to implement these observations using ultra-low power analog sub-threshold circuits.

The next level of hierarchy takes data from observations and extracts features such as waveform peak and valley locations. This level is implemented as a parallel array of processing stages. The feature extraction, however, at this stage is context-aware. In our prototype, the underlying context that determines the feature extraction is whether a user is walking or not walking. As illustrated in Figure 4(b), if the system determines that the user is walking, the features are targeted towards collecting gait attributes, otherwise the system tries to determine when the user has started walking. Since a user in a home setting would not be walking often, the context-aware feature extraction helps us implement low-power, coarse-grained feature extraction to determine walking and more fine grained filtering and feature extraction to calculate gait attributes, leading to large energy savings.

Walking is established through detection of an ongoing oscillatory pattern. The method was motivated by a zero-crossing detection approach, but modified according to the data features already available – peak-valley data. The algorithm uses the occurrence of a positive-valued peak followed by a negative-valued valley (or vice versa) as a zero crossing. For each parallel processor analyzing an observation, a binary vote for walking is cast when enough zero crossings occur in a specified time frame. This vote (if true) is sent to the highest level of the processing hierarchy and a simple majority vote determines that the context has now changed to walking. The higher level processor then triggers the low level processing units to capture features tuned to collecting gait attributes. A key design factor here is that enough sensors and measurements must be made to confidently establish the walking context, a question of *observability*. In our system the number of sensors was established through experimental trial-error-adjustment alongside design of the processing stages. We perform two optimizations to minimize computation and energy consumption when the underlying context is “not walking”. First, the feature extractors do not filter the data.

This saves energy in the system and filtering is not necessary for coarse feature extraction like the zero-crossing for detection of oscillations. Even if a step is missed in this process the only penalty is latency in detecting walking. Second, we reduce the sampling rate to minimize processing.

**Determining stride speed and inter-leg spacing:** When a state of walking is established by the system, the data from the observations undergo a filtering process. We have implemented an IIR filter [22] that approximates a low-order windowed-integrator. This pre-filtering of the data presents the peak-valley detector with a measure of area under/above the peak/valley, and hence the size of the peak, rather than just an instantaneous maxima/minima. Peak-valley data (value and time) is extracted and transmitted. The peak valley detection unit employs a strategy to look for a series of increasing samples followed by a series of decreasing samples for peak detection and vice versa for valley detection. Rapid glitches are rejected as noise so as not to cause unnecessary data flow and wake-up signals in the system. The extracted peaks and valleys are then used by the aggregator (a higher power processor in Figure 4(a)) to determine stride speed and spacing between legs.

To understand how the aggregator determines stride speed and leg spacing, it is important to understand the observations derived from the capacitor data. In our prototype, we use four capacitor plates  $\{c_1, c_2, c_3, c_4\}$ , and derive six observations from these four sensors  $y_1 = c_1 + c_2, y_2 = c_1 - c_3, y_3 = c_1 - c_4, y_4 = c_2 + c_3, y_5 = c_2 - c_4, y_6 = c_3 + c_4$ . The observations that sum capacitor outputs (*summed observations*) emulate a larger capacitor plate and a peak in the data stream from the observation can be used to locate and place the object with respect to the plates. The observations that take the differential between the capacitor plates (*differential observations*) can help find the direction of motion by analyzing peaks and valleys. In our algorithm at the aggregator, depicted in Figure 4(a), a majority vote is used to determine a walking state. Peaks from summation channels and valleys from differential channels are counted as votes. In a detected stride, the valleys and peaks coincide in time to when the opposite leg’s position is close to the capacitive sensors as the leg swings forward or backward. The time between strides is then used to determine the stride speed. Similarly, to determine the inter-leg spacing we first determine the start of a stride using the summed and

TABLE I. THE TABLE SHOWS THE ACCURACY OF CLASSIFYING TWO AND THREE DISCRETE INTER-LEG SPACING AS A FUNCTION OF THE AMOUNT OF CALIBRATION (I.E., CALIBRATING WITH ONE SPEED AND NORMAL INTER-LEG SPACING DATA AND CALIBRATING WITH THREE SPEED AND NORMAL INTER-LEG SPACING DATA).

Classifier	3-speed calib.	1-speed calib.
narrow, wide	98.1%	94.5%
narrow, normal, wide	92.5%	80%

differential observations described above, and then we use the average of the amplitude of the first peak (after the stride is determined) in the differential stages to calculate the inter-leg space. In our experiments, we found that the use of multiple observations improves the reliability of determining the gait attributes.

Our context-aware hierarchical architecture facilitates aggressive duty-cycling of higher power processing elements. The aggregator for instance, can be kept switched off, if the feature extractors do not see relevant features. While we demonstrate the efficacy of our system using two gait attributes that are related to falls, the system can be extended to other gait attributes and can be generally used to design low-power signal processing on data collected from an array of sensors.

**Prototype Implementation:** We have prototyped a fully functional sensing unit that comprises of four 1-inch by 1-inch capacitive sensor plates sewn into an ace bandage (Figure 1). The signal processing architecture is prototyped on a Spartan 6 FPGA. Additionally, we have designed a custom PCB that houses the FPGA module, a Bluetooth module, and a capacitance measurement IC. The feature extractor processors are build as hardware modules on the FPGA, and we use the micro blaze core as the aggregator. As the microblaze is a proprietary Xilinx processor implementation, we intend to use the TI msp430 open core implementation in our final system. Additionally, since an FPGA module is not designed for low power consumption, to demonstrate the efficacy of our hierarchical architecture, we have synthesized, placed, and routed ASIC cores for our signal processing architecture using 180-nm CMOS technology. The cores were synthesized using Cadence RTL Compiler and placed and routed using Cadence Encounter. We used the low power synthesis and place and route options in both the tools. The synthesized cores placed in an IC floorplan is illustrated in Figure 4(c).

## V. SYSTEM EVALUATION

The goal of our system is to capture gait attributes that have high correlation with falls in a home setting at minimal energy consumption. The evaluation of our system, therefore, focusses on the following key questions. (1) How accurately does our system measure stride speed and inter-leg spacing when a user is walking and can it capture temporal variations in gait? (2) What is the energy consumption of our system when compared to an architecture that does not use the hierarchical context-aware design?

**Experimental Setup:** We evaluate our system using a setup similar to clinical evaluation of fall prediction. In our setup, we mark the floor to indicate stride lengths and inter-leg spacing (width). We recruited six subjects and had them walk with our sensor system worn on the leg. For ground truth measurements, we collected videos of the experiments with

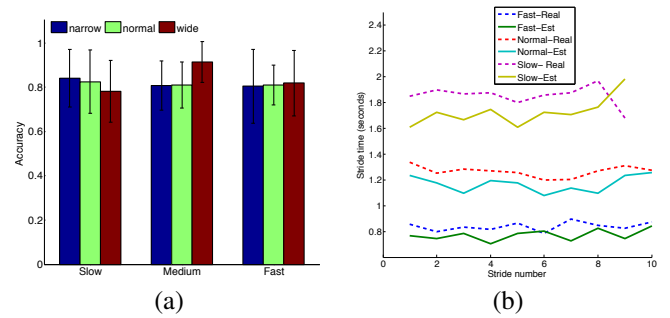


Fig. 5. (a) Accuracy of measuring stride speed for slow, normal, and fast walking with narrow, normal, and wide leg spacing. Our system can measure stride speed with an accuracy of 85% (with a positive deviation of 13%). (b) The figure shows that our system can accurately capture temporal variations in stride speed.

a web camera mounted on a wheelchair. We then manually analyzed the video frames to determine the ground truth stride speed and used the floor markers to determine the spacing between legs. In our experiments, we asked the subjects to walk at three distinct speeds: (1) normal; (2) faster than normal; and (3) slower than normal. Moreover, for these three walking speeds we also asked the subjects to walk with three distinct leg spacing: (1) normal; (2) wider than normal; and (3) narrower than normal. However, the subjects were asked to walk at a comfortable stride length, or step size. Overall we performed 54 trial experiments using our six subjects. Note that similar walking parameters are used in experiments performed in a clinical setting to analyze gait for fall prediction [3]. In addition to the experiments performed using our prototype, we derive energy consumption values using dynamic power analysis on our synthesized ASIC. We use these energy numbers to compare the performance of our system with a system that does not use the hierarchical context-aware architecture.

**System Accuracy:** In our first set of experiments, we evaluate the accuracy of using capacitive sensor arrays and our hierarchical signal processing algorithm. Figure 5 (a) graphs our system accuracy in measuring stride speed for fast, normal, and slow walking and for wide, normal, and narrow leg spacing. The bar graphs are the average accuracy values calculated across our subjects and the error bars represent the standard deviation from the average when all subjects are considered. The figure shows that our system can measure stride speeds with an average accuracy of 85% with a positive deviation of 13%. Table I shows the accuracy of our system in classifying inter-leg spacing into two classes (narrow or wide) and classifying inter-leg spacing between three disjoint classes (narrow, normal, and wide). The columns present results for two cases—when the system is calibrated using one walking speed (normal speed) and when it is calibrated using the three walking speeds (normal, fast, and slow). The table shows that the system is able to accurately distinguish three inter-leg spacing with an accuracy of 92% and two inter-leg spacing with an accuracy of 98% when calibrated with three speeds. The system must be calibrated once per day, but there is no need for trained personnel or proper training environment. Because of this calibration, changes in the positioning of the sensors does not negatively affect the system. In addition to average accuracy, another key aspect of fall prediction is reliably capturing variations in gait attributes. To demonstrate

TABLE II. POWER CONSUMPTION OF DIFFERENT COMPONENTS IN THE HIERARCHICAL DESIGN USING DYNAMIC POWER ANALYSIS ON OUR SYNTHESIZED ASIC. THE OBSERVATION COMPUTATION POWER NUMBERS ARE FOR THE IC USED IN OUR PROTOTYPE

Process	180-nm CMOS
msp430 (8 MHz, 20% switching activity) power	986 $\mu$ W
Hardware feature extractors (1 KHz)	2 $\mu$ W
Capacitance and Observation computation	313 $\mu$ W

that our system can capture gait variation (over time) we present Figure 5 (b) which compares the time per stride (for slow, fast, and medium speeds) measured by our system and the ground truth, over a number of strides. The figures show that CSAs augmented with our signal processing algorithm can reliably capture temporal variations in gait attributes, and hence can be used to collect data for fall prediction.

**Energy Consumption:** A key advantage of using our context-aware hierarchical signal processing architecture is that high power processing elements can be switched off if the feature extractors do not detect useful features. The system also runs at a low power state if the user is not walking. To demonstrate the power savings for the hierarchical architecture, we estimate the power consumption of each hierarchy using dynamic power analysis on our synthesized ASIC cores. The energy consumption for different modules in the hierarchy—calculating observations, parallel feature extraction, the feature aggregation—is illustrated in Table II. Assuming that a user spends 30 minutes (2% of time) in a day walking (<http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2894114/>) when gait attributes will be analyzed at the aggregator, our system would consume 8 mWh of energy per day. On the other hand, a system that does not use a hierarchical design and the feature extraction and observation calculation occurs at the higher power processor, would consume 31 mWh of energy per day. Hence, our hierarchical signal processing architecture can accurately capture gait attributes while consuming 4X lower energy.

## VI. CONCLUSION

In this paper, we present a system that uses textile-based capacitive sensor arrays built into clothing for capturing gait attributes that have a high correlation with falls. The system uses an array of capacitive sensors to detect movement in the proximity of the capacitor plates. Our novel hierarchical and context-aware signal processing algorithm converts data from the capacitive sensor array into high-level gait attributes such as stride speed and inter-leg spacing. We have prototyped our sensor on an ace bandage and our processing unit on an FPGA, and show that the system can accurately measure gait attributes with high accuracy and low energy consumption.

## VII. ACKNOWLEDGMENT

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] Center of Disease Control and Prevention. Falls among older adults: An overview. <http://www.cdc.gov/homeandrecreationalafety/falls/adultfalls.html>, 2010.

[2] Brian E Maki. Gait changes in older adults: predictors of falls or indicators of fear. *Journal of the American geriatrics society*, 45(3):313–320, 1997.

[3] Stephanie A Bridenbaugh and Reto W Kressig. Laboratory review: the role of gait analysis in seniors mobility and fall prevention. *Gerontology*, 57(3):256–264, 2010.

[4] Jorunn L Helbostad and Rolf Moe-Nilssen. The effect of gait speed on lateral balance control during walking in healthy elderly. *Gait & posture*, 18(2):27–36, 2003.

[5] Bertha H Clarkson. Absorbent paper method for recording foot placement during gait suggestion from the field. *Physical therapy*, 63(3):345–346, 1983.

[6] MJ Wilkinson and HB Menz. Measurement of gait parameters from footprints: a reliability study. *The Foot*, 7(1):19–23, 1997.

[7] Shinji Sako and Tadashi Kitamura. Subunit modeling for japanese sign language recognition based on phonetically depend multi-stream hidden markov models. In *Universal Access in Human-Computer Interaction*, pages 548–555. Springer, 2013.

[8] Haibo Pang and Youdong Ding. Dynamic hand gesture recognition using kinematic features based on hidden markov model. In *GCN 2012: Volume 5*, pages 255–262. Springer, 2013.

[9] Liang Yin, Mingzhi Dong, Ying Duan, Weihong Deng, Kaili Zhao, and Jun Guo. A high-performance training-free approach for hand gesture recognition with accelerometer. *Multimedia Tools and Applications*, pages 1–22, 2013.

[10] Tao Lu. A motion control method of intelligent wheelchair based on hand gesture recognition. In *ICIEA 2013*, pages 957–962. IEEE, 2013.

[11] 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.

[12] 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 *IEEE ISCAS*, pages 1395–1398. IEEE, 2012.

[13] Pavel Ripka and Alois Típek. *Modern sensors handbook*. Wiley. com, 2013.

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

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

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

[17] 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 *IEEE MEMS*, pages 640–643. IEEE, 2012.

[18] Pei-Hsuan Lo, Chitsung Hong, Sung-Cheng Lo, and Weileun Fang. Implementation of inductive proximity sensor using nanoporous anodic aluminum oxide layer. In *TRANSDUCERS*, pages 1871–1874. IEEE, 2011.

[19] SK Lee, William Buxton, and K. C. Smith. A multi-touch three dimensional touch-sensitive tablet. In *ACM CHI*, CHI ’85, pages 21–25, New York, NY, USA, 1985. ACM.

[20] T Scott Saponas, Chris Harrison, and Hrvoje Benko. Pockettouch: through-fabric capacitive touch input. In *ACM UIST*, pages 303–308. ACM, 2011.

[21] L Tyvaert, E Houdayer, H Devanne, JL Bourriez, P Derambure, and C Monaca. Cortical involvement in the sensory and motor symptoms of primary restless legs syndrome. *Sleep medicine*, 10(10):1090–1096, 2009.

[22] Rainer Storn. Differential evolution design of an iir-filter. In *Evolutionary Computation, Proceedings of IEEE International Conference on*, pages 268–273. IEEE, 1996.