Ultra Low-Power Signal Processing in Wearable Monitoring Systems: A Tiered Screening Architecture with Optimal Bit Resolution

HASSAN GHASEMZADEH, University of California, Los Angeles
ROOZBEH JAFARI, University of Texas at Dallas

Advances in technology have led to the development of wearable sensing, computing, and communication devices that can be woven into the physical environment of our daily lives, enabling a large variety of new applications in several domains, including wellness and health care. Despite their tremendous potential to impact our lives, wearable health monitoring systems face a number of hurdles to become a reality. The enabling processors and architectures demand a large amount of energy, requiring sizable batteries. In this article, we propose a granular decision-making architecture for physical movement monitoring applications. The module can be viewed as a tiered wake-up circuitry. This decision-making module, in combination with a low-power microcontroller, allows for significant power saving through an ultra low-power processing architecture. The significant power saving is achieved by performing a preliminary ultra low-power signal processing, and hence, keeping the microcontroller off when the incoming signal is not of interest. The preliminary signal processing is performed by a set of special-purpose functional units, also called screening blocks, that implement template matching functions. We formulate and solve an optimization problem for selecting screening blocks such that the accuracy requirements of the signal processing are accommodated while the total power is minimized. Our experimental results on real data from wearable motion sensors show that the proposed algorithm achieves 63.2% energy saving while maintaining a sensitivity of 94.3% in recognizing transitional actions.

Categories and Subject Descriptors: C.3 [**Computer Systems Organization**]: Special Purpose and Application-Based Systems—*Real-time and embedded systems*; J.3 [**Computer Applications**]: Life and Medical Science—*Health*; H.1.2 [**Models and Principles**]: User/Machine Systems—*Human information processing, Human factors*.

General Terms: Design, Algorithms, Performance

Additional Key Words and Phrases: Medical embedded systems, body sensor networks, signal processing, wearable monitoring systems, power optimization.

ACM Reference Format:
Ghasemzadeh, H. and Jafari, R. 2013. Ultra low-power signal processing in wearable monitoring systems: A tiered screening architecture with optimal bit resolution. ACM Trans. Embedd. Comput. Syst. 13, 1, Article 9 (August 2013), 23 pages.
DOI: http://dx.doi.org/10.1145/2501626.2501636

1. INTRODUCTION

Long-term pervasive sensing and monitoring can aid in diagnosis and tracking of certain diseases, such as Parkinson's [Mitchell et al. 1995; Patel et al. 2007], or extracting biokinematic characteristics of the human body, such as gait parameters [Morris and Paradiso 2002; Hagler et al. 2010]. Advances in technology have led to development of

Authors' addresses: H. Ghasemzadeh, UCLA Wireless Health Institute, 3531A Boelter Hall, UCLA, Los Angeles, CA 90095; email: hassan@cs.ucla.edu; R. Jafari, Department of Electrical Engineering, University of Texas at Dallas, Richardson, TX 75080.

DOI: http://dx.doi.org/10.1145/2501626.2501636

wearable sensing, computing, and communication devices that can be woven into the physical environment of our daily lives, enabling a large variety of new applications in several domains, including wellness and health care. These systems, also called body sensor networks (BSNs), enable real-time monitoring of the human body. A BSN consists of several nodes placed on the human body that provide sensing, processing and communication capabilities. BSNs offer the unprecedented ability to monitor patients in a natural setting for an extended period of time.

Despite their tremendous potential to impact our lives, wearable health monitoring systems face a number of hurdles to become a reality. The enabling processors and architectures demand a large amount of energy, requiring sizable batteries. This creates challenges for further miniaturization of the wearable units.

In addressing energy concerns in wearable monitoring platforms, this article introduces an ultra low-power *granular decision-making* methodology based on coarse-to-fine-grained signal processing techniques requiring low to slightly higher power. This architecture is presented in the context of physical movement monitoring which aims at detecting target human actions, such as 'walking', 'sit to stand', 'kneeling', or 'lie to sit'. The granular decision-making module (GDMM) will remove actions that are not of interest as early as possible from the signal processing chain, deactivating all remaining signal processing modules, including the microcontroller. The granular decision-making module can be viewed as a tiered wake-up circuitry. It is composed of hundreds or thousands of choices of screening blocks, although in this article, we consider only a specific case where bit resolution of sensor readings is considered for optimizing power consumption of decision making architecture. Each screening block is essentially a classifier with several tunable parameters, by which power versus classification accuracy can be adjusted.

Emerging applications of health care monitoring have unique properties motivating the proposed processing architecture: signals and events observed from the human body are slowly changing [Nelson 1991]. They are governed by the physics of the human body (e.g., kinematics, dynamics) which constrains the variations and reduces the randomness in the signals. In addition, *events of interest*, which may require the microcontroller's attention, often occur with a low duty cycle [Cuddihy et al. 2007]. We exploit these properties to propose novel programmable multilevel information-driven decision-making techniques that are highly power optimized.

The contributions of this article can be summarized as follows. (1) We present a novel programmable architecture for detecting low duty-cycle actions mainly for physical movement monitoring applications; (2) we first model the signal processing tightly with the architecture and HW and then propose an optimization problem for minimizing the number of functional units used for preliminary signal processing; (3) we propose algorithms for solving the optimization problem and present experimental results on experimental data collected from our BSN platform and demonstrate the energy efficiency of the proposed granular decision making architecture.

2. RELATED WORK

Several ultra low-power wearable systems with a power budget of less than 1 mW and with signal processing capabilities have been proposed. The proposed systems, however, are either not programmable (except that they may provide a few tunable parameters), or the programmability is handled completely by a microcontroller. An intraocular CMOS pressure sensor system implant was proposed which contains an on-chip micro mechanical pressure sensor array, a temperature sensor, a microcontroller-based digital control unit, and an RF transponder [Stangel et al. 2001]. An interface chip for implantable neural recording was proposed with tunable band-pass filters and adjustable gain [Liew et al. 2009]. A batteryless accelerometer system with 3D

loop antenna was proposed that utilizes the radio wave for power feeding and signal communication as RFID. However, the control unit of the system is a microcontroller and is unclear how it can be powered up by energy scavenging [Sasaki et al. 2006].

Several other systems were suggested that are primarily tailored towards specific applications. Examples include a machine-learning-based patient-specific seizure detector [Shoeb et al. 2009], an implantable blood pressure, ECG-sensing microsystem with adaptive RF powering [Cong et al. 2009a, 2009b, 2010; Chaimanonart and Young 2009], an implantable batteryless telemetric microsystem for EMG recording [Parramon et al. 1997], and a batteryless MEMS implant for cardiovascular applications [Najafi and Ludomirsky 2004]. An ultra low-power sensing device for measuring pulse oximetry is presented in Tavakoli et al. [2010]. Other work [Mandal et al. 2010; Sarpeshkar 2006] introduces a battery-free sensing device with radio frequency energy-harvesting from wireless module.

There have been efforts towards creating ultra low-power semiconductor components. Multithreshold CMOS (MTCMOS) circuits is an example [Douseki 2004]. A wireless system with MTCMOS/SOI circuit technology was suggested which lowers the supply voltage of the LSIs 0.5 V and reduces the power dissipation to 1 mW [Douseki et al. 2003]. One mW, however, is still larger than the energy budget of the energy harvesting circuits. The power budget of energy harvesting circuits is often tens of μWs. For example, a batteryless vibration-based energy harvesting system was proposed for ultra low-power ubiquitous applications that can generate 36.79 μW [Chao et al. 2007].

Our power reduction approach employs a sequence of lightweight built-in classifiers and operates in a chain. The idea of using a cascade of simple classifiers for performance enhancement is studied in a variety of image processing subfields, including detection of handwritten digits [Zhang et al. 2007], face detection [Ou et al. 2005], and many other object detection areas [Viola and Jones 2001]. The approach proposed by Viola and Jones for object detection [2001] is a widely used technique in which a cascade architecture is proposed to boost the classification performance. This object detection framework is primarily used for face detection applications [Viola and Jones 2004a] and employs an AdaBoost-like scheme to perform both feature selection and classifier training. A set of classifiers are arranged in a cascade in order of complexity, where each successive classifier is trained only on those selected samples which pass through the preceding classifiers. If at any stage in the cascade a classifier rejects the current subwindow, no further processing is performed, and it continues on searching the next subwindow [Viola and Jones 2004b]. The idea of combining simple classifiers to achieve higher accuracy is also discussed [Ravi et al. 2005], where a single accelerometer is used for activity monitoring and combining classifiers using plurality voting. Much work has been done on selecting only relevant pieces of information for classification. Selecting only individual features for each activity can improve the performance, as demonstrated in Huynh and Schiele [2005]. In Lester et al. [2006] AdaBoost is used to select a small number of features in order to ensure fast classification. The algorithm automatically selects the best features and ranks them based on their classification performance. Given the maximum number of features that the activity recognition system can use, the system automatically chooses the most discriminative subset of features and uses them to learn an ensemble of discriminative static classifiers for activity recognition.

As mentioned previously, our power-saving approach is motivated by sparsity of the events that occur in health monitoring applications. Sparsity of events has inspired many studies in the past in other monitoring applications. An example of these studies is Lucid Dreaming [Jevtic et al. 2007] that is proposed for use in structural crack monitoring. The system performs high resolution sampling of sensor data in bursts upon occurrence of an event of interest. This wake-up circuit consumes only 16.5 μW. CargoNet

Table I. Notations

Term/Abbreviation	Description
BSN	Body sensor network
GDMM	Granular decision-making module
MSPC	Main signal processing chain
MCSP	Minimum cost screening path
$\hat{a}$	target action
A	set of m nontarget actions
T	template generated for target action $\hat{a}$
$\gamma(T, S)$	similarity score between template T and signal segment S
n	maximum number of quantization bits provided by ADC
B_i	ith screening block
thr_i	threshold value for screening block B_i
b_i	bit resolution of screening block B_i
tp_i	true positive rate, percentage of target actions accepted by B_i
fp_i	false positive rate, percentage of non-target actions accepted by B_i
λ	desired true positive rate
w_i	per-action energy consumption of B_i

[Malinowski et al. 2007] is another system aimed at environmental monitoring and is capable of asynchronous wake-up upon occurrence of exceptional events. The wake-up module uses adjustable thresholds to adapt to dynamic environments. The power consumption of the system is reported to be less than 25 μW. Benbasat and Paradiso [2007] employe a decision tree model to activate sensor nodes and adjust their sampling rates. The performance of the decision tree classifier is comparable with that of the support vector machines, while the decision tree consumes significantly less energy. Another example is the system presented in Rumberg et al. [2010] which uses an ultra low-power front-end analog circuitry to enable/disable digital component of the system by examining the power spectrum of the signal. Furthermore, an acoustic surveillance system is presented [Goldberg et al. 2004], which implements a digital VLSI periodicity detector (with a core power consumption of 835nW) to wake up the system. The detection criterion is based on the degree of low-frequency periodicity in the acoustic signal.

Our approach is different from previous researches. (1) Our granular decision-making module is composed of extremely low-power functional units that perform preliminary signal processing (template matching) on incoming signals; (2) the proposed architecture takes into account specific properties of BSN applications and their signal processing requirements; (3) our signal screening module is reprogrammable in the sense that it can be embedded with the microcontroller to provide signal screening for monitoring low duty cycle events.

3. PRELIMINARIES

The problem addressed by this article is energy savings in BSNs through a preliminary signal screening block, called the granular decision-making module. Before presenting more details of our ultra low-power architecture, we discuss major components of a typical BSN platform, including sensing hardware and signal processing flow. Throughout this article, we use notations listed in Table I.

3.1. Sensing Platform

A BSN is composed of several sensor nodes mounted on the patient's body, embedded with the clothing, or implanted in the human body. Physical movement monitoring uses inertial information acquired by motion sensors, such as accelerometers, gyroscopes, and magnetometers. The sensor node we use has a processing unit, TelosB

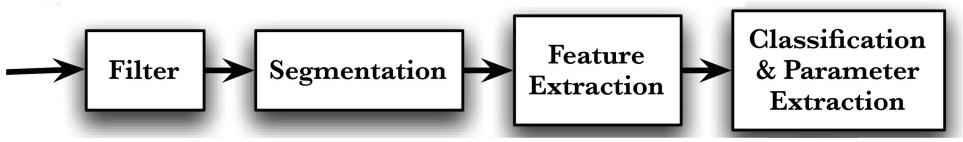


Fig. 1. MSPC (main signal processing chain) for action recognition.

mote [Polastre et al. 2005] from XBow, with several analog-to-digital (ADC) channels responsible for acquiring and digitizing analog signals. The MSP430 microcontroller used for our experiments has an ADC unit with 12-bit resolution. The sensor node has also a radio module for communication with other nodes in the network or with a gateway such as a cell phone. Note that this sensing platform is purely used for data collection. In Section 7, the data collected from this platform will be used to validate the architecture presented in Section 4.2.

3.2. Main Signal Processing Chain

The goal of the main signal processing chain (MSPC) is to extract useful information from sensor data. Frequently, this data is a high-level observation, such as "Is the subject running?" or "What is the stride length when the subject is walking?" In other words, the purpose of main signal processing is to provide a fully SW-programmable environment for development of highly reliable signal processing techniques for action detection/verification and extracting details from the signals (e.g., balance during 'sit to stand' when it occurs).

Out of all possible actions, only a few are of interest to main signal processing, for example, 'walking' or 'sit to stand'. Therefore, the main signal processing needs to classify actions of interest prior to extracting any further details about actions. The overall goal of the classifier is labeling actions of interest, also called target actions. Figure 1 shows a typical signal processing model commonly used for movement monitoring applications and algorithms appearing in literature [Yang and Yacoub 2006]. In this model, signals are processed in real time by a series of processing blocks to arrive at a classification result. These processing blocks include filtering, segmentation, feature extraction, and classification and parameter extraction. The filtering is generally applied to remove sensor artifacts and noise. In the context of action recognition, segmentation determines portions of the signal that represents a complete action—segregating activity versus rest. Features are functions run on the segmented data to decrease dimensionality of the signal without significantly reducing the relevant information. Statistical features are frequently used for action recognition [Ghasemzadeh et al. 2009]. Finally, each node uses the feature vector generated during feature extraction to determine the most likely action by utilizing some classification algorithm, such as k-nearest neighbor (k-NN) [Duda et al. 2000].

4. SCREENING APPROACH FOR POWER SAVING

This section presents different components of our energy-efficient architecture. We describe motivation for signal screening first and present a top-level view of our system followed by more detailed information on each component of the system in subsequent sections.

4.1. Motivation

Most BSN applications are only concerned with a very small subset of human actions. For instance, gait analysis only is concerned with 'walking', fall detection with 'falls', Parkinson's disease monitoring with certain movements such as 'tremors' [Patel et al.

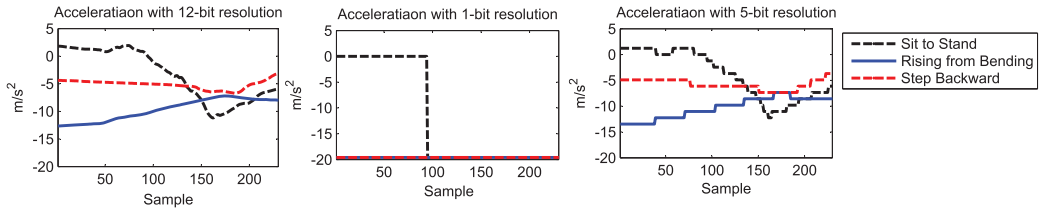


Fig. 2. Motivation.

2007], sleep apnea with 'restless leg syndrome' and 'periodic limb movements' [Ohayon and Roth 2002]. In real-time continuous patient monitoring, these target actions occur infrequently. Considerable energy is wasted processing non-target actions. As a result, efficiently rejecting non-target actions with a screening classifier could lead to a significant increase in system lifetime through deactivating the main processor which provides the full signal processing for classification. This way, an ultra low-power screening block activates the MSPC only when a target action is observed. Clearly, one requirement of such a screening classifier is to achieve a significantly high precision, *true positive rate*, in detecting target actions (activating main processor due to occurrence of a target action). To obtain high true positive rates, the screening architecture may accept some non-target actions. Such actions determine the *false positive rate*. For the actions that the screening block cannot reject reliably, the MSPC will be activated. The main advantage of this method is the power saving due to removing non-target actions from the signal processing chain, thus deactivating the remaining modules in the signal processing chain. We note that the false positives may not generate problems as they can be rejected by the MSPC. Furthermore, a chain of classifiers, which performs coarse- to fine-grained preliminary processing, can be used. In this article, we focus on using a sequence of classifiers that differ in the level of bit resolution of sensor readings. Lower-level classifiers operate at lower precision and consume less power, while higher-level classifiers have higher percision at the cost of higher power consumption.

Figure 2 illustrates the motivation behind using a sequence of screening blocks with low-to-high bit resolution. This figure shows real signals collected with our wearable sensors where only three actions are used for visualization. The graphs show raw sensor readings from the Z-axis accelerometer of a node placed on the waist of a subject. Assume 'Rising from Bending' (bold blue plot) is the action of interest and the other two actions, 'Sit to Stand' (dashed black plot) and 'Step Backward' (dashed red plot) may occur as non-target. The graph on the left shows sensor readings with 12 bits of ADC resolution. Clearly, the target action ('Rising from Bending') can be distinguished from the two other actions on the 12-bit graph. This can be accomplished by a template matching function, as will be discussed in Section 5.1. Let us assume that all actions are equally likely. Therefore, a 12-bit template matching needs to be active all the time, monitoring incoming signals. It would activate the main signal processing chain only when a target action occurs (33.3% of the time). Now assume that two classifiers with 1-bit and 5-bit resolutions are used, as shown in middle and right graphs, instead of a 12-bit classifier. As shown in the figure, a 1-bit resolution classifier can reject 'Sit to Stand' as not being the target action. However, it cannot reject a 'Step Backward' action, as the signal has the same value as the target action (see blue and red graphs in the 1-bit resolution scenario). A not-rejected action, however, can be further processed by the next classifier in the chain (a 5-bit resolution classifier), where 'Step Backward' and 'Rising from Bending' actions are distinguishable. Therefore, an incoming signal can be reliably classified as either 'target' or 'non-target' after passing through a sequence

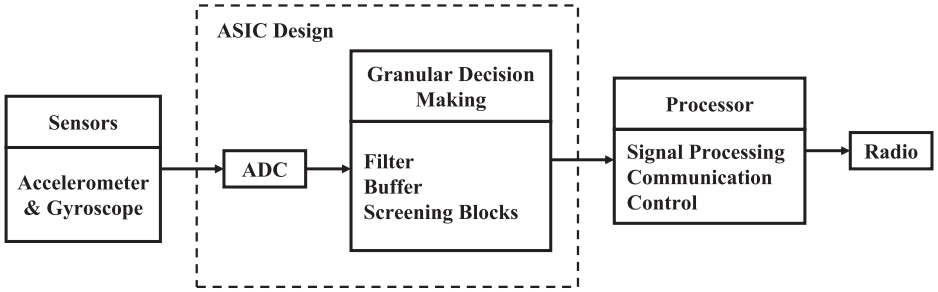


Fig. 3. Architecture of the proposed system with GDMM.

of two classifiers with lower bit resolutions than the 12-bit classifier. In this example, a 1-bit resolution classifier is active all the time, while a 5-bit classifier is active only 66.7% of the time. We note that a sequence of lower-resolution classifiers can potentially consume a lot less power than a classifier with full-scale quantization.

4.2. System Architecture

An overview of our system architecture for low-power signal processing is shown in Figure 3. There are four main portions of the platform: the sensors, the proposed special purpose functional unit or granular decision-making module (GDMM), a low-power general-purpose processor, and the radio. Human actions can be examined using motion sensors, such as accelerometers and gyroscopes. The sensor readings are sent through an ASIC architecture, including an analog-to-digital converter (ADC) and GDMM, which digitizes the reading and performs screening tests. The ADC is an essential component which acquires and digitizes analog signals for further analysis. The MSP430 microcontroller used for our experiments has an ADC unit with 12-bit resolution. Any action that is accepted by the GDMM will be forwarded to the MSPC for further processing. In the following two sections, we present details of our signal screening architecture. The MSPC presented in Section 3.2 is implemented on the main processor where the results can be transmitted through the radio.

4.3. Granular Decision-Making Module

Our power saving model is based on a set of screening blocks performing template matching on incoming signal. Each screening block can be adjusted to perform preliminary signal processing at different precision rates at the cost of power usage. An example of tunable parameters is the number of quantization bits or bit resolution of the sampled data. Specifically, optimizing the screening architecture with respect to bit resolution is the main focus of this article. Figure 4 illustrates GDMM in connection with other components of the system, where screening blocks operate at different bit resolutions. The module includes digital pre-filtering, a buffer, and a chain of screening classifiers as described previously. The sensor data from body-mounted motion sensors is frequently noisy. A moving average filter is enough to filter the signal and remove high frequency noise [Ghasemzadeh et al. 2008].

As previously discussed, each screening block in the chain is applied in a sequence that will be detailed in Section 5. The processing stops as soon as a screening block in the chain rejects the incoming action. Activating screening blocks in serial introduces a time delay for each subsequent block. In order to allow each block to operate on the proper signal segment, a single buffer is used.

The lowest-level screening block (i.e., B_1) has the lowest precision (e.g., 1-bit resolution) but is also the least energy-consuming block. An active screening block makes

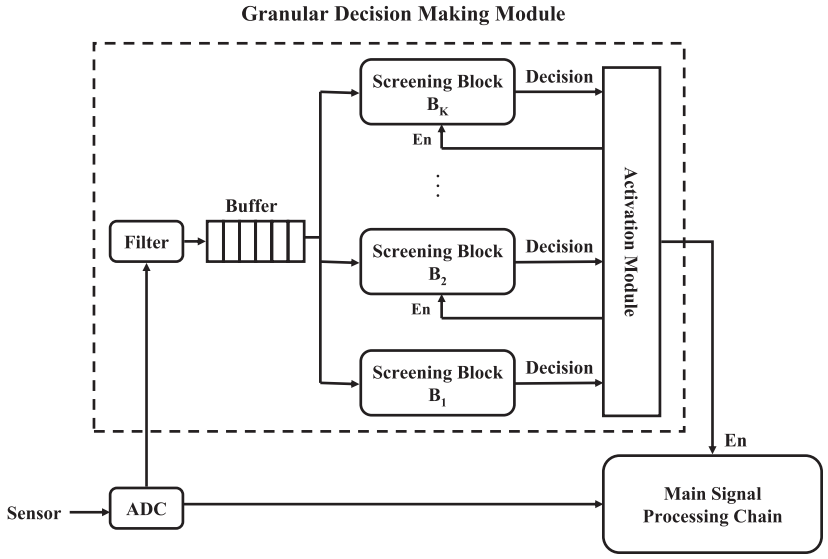


Fig. 4. GDMM (granular decision-making module) composed of several screening blocks, each having a different bit resolution.

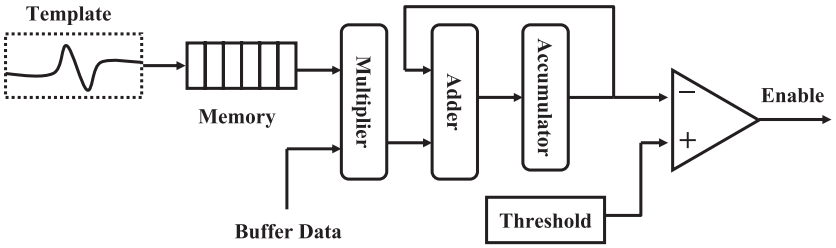


Fig. 5. Screening block performing template matching operations.

a preliminary binary decision (Accept/Reject) on incoming signal. A higher-level block (e.g., B_2) is activated only if the incoming action is accepted by the preceding block (e.g., B_1). Clearly, the block at the lowest processing level (e.g., B_1) needs to be active all the time, processing incoming signals. In Section 5, we present an optimization problem aiming to find the optimal sequence of the screening blocks where quantization bit is considered as the tunable parameter.

The *activation module* is responsible for turning on the next screening block or the MSPC. That is, activation of the screening blocks and the main processor are programmable thought the activation module. A higher-level screening block is activated only if the current action is accepted by its preceding block.

4.4. Screening Blocks

Each screening block compares the incoming signal against a predefined template over a fixed window. The comparison is made using template matching operations. The template matching, shown in Figure 5, is based on Normalized Cross Correlation (NCC) [Yang 2010], which will be discussed in Section 5.1. Template matching is implemented using a multiplier-accumulator (MAC) circuit. Real-time signal processing requires high speed and high throughput MAC units that consume low energy, which is always a key to achieving a high performance low-power computing system.

Each screening block is a binary template classifier based on the cross correlation score obtained by comparing the incoming action with a precomputed template of the target action. This comparison assigns a score value, γ, based on a similarity measure between the signal segment and the template. For classification decision, γ is compared against a threshold value, thr_i, and the action is classified as either *accept* or *reject*. A rejection causes processing to stop for that action.

5. MINIMUM COST SCREENING PATH

The GDMM in Figure 4 is composed of several screening blocks that form a decision path for classification. Each block is associated with a quantization bit level which affects performance of the classification. Finding the minimum set of screening blocks and their ordering is challenging because each block has a different operating point depending on the bit resolution and the threshold used for classification. In this section, we formulate an optimization problem in order to find the optimal decision path forming the best sequence of screening blocks for examining each incoming action.

5.1. Template Matching

Given a target action $\hat{a}$ and $A = \{a_1, a_2, \ldots, a_m\}$ a set of m non-target actions, we generate template T, associated with $\hat{a}$, from a set of training trials. Templates are generated, as shown in Definition 5.1, using a set of training trials. During real-time operation of the system, a classification decision is made by comparing the incoming action to the predefined template. The comparison is made based on the similarity score defined in Definition 5.2.

Definition 5.1 (*Template*). Given an action a_i with L number of training trials, a template T_i for a_i is the best representative trial with respect to the similarity score γ between all pairs of the trials. The trial with the highest summed similarity score between itself and the other trials is selected, as shown in Eq. (1).

$$T_i = \arg\max_{a_i^l} \sum_r \gamma\left(a_i^l, a_i^r\right). \tag{1}$$

Definition 5.2 (*Similarity Score*). Given two time series signals f and g of length N, the similarity score $\gamma(f, g)$ between the two signals is defined based on their normalized cross correlation by

$$\gamma(f,g) = \frac{\sum_{t=1}^{N}[f(t) - \bar{f}][g(t) - \bar{g}]}{\sqrt{\sum_{t=1}^{N}[f(t) - \bar{f}]^2 \sum_{t=1}^{N}[g - \bar{g}]^2}}, \tag{2}$$

where $\bar{f}$ and $\bar{g}$ denote mean values of f and g.

5.2. Performance of Screening Blocks

As mentioned previously, each screening block performs preliminary classification based on the score associated with cross correlation between the template and the incoming action. A screening block B_i rejects the incoming action if the score is smaller than a certain threshold thr_i. Classification performance of a screening block B_i depends on thr_i and the bit resolution of the block, b_i. The threshold is set during the training to obtain a prespecified precision associated with a desired performance criterion. The larger the threshold is, the higher the likelihood of rejecting an incoming action. Therefore, the threshold value directly affects the true positive rates (tp_i) and false positive rates (fp_i). Our granular decision-making architecture aims to minimize power consumption of the system while maintaining a given lower bound on the true positive rate (λ). The module introduces a decision path, including a sequence of the

screening blocks. The power consumption of the module is determined by the *acceptance rate* of the screening blocks on the path (r_i) and the energy consumption of each block (w_i). We call this problem *minimum cost screening path* (MCSP) and study this problem by mapping the entire set of screening paths onto a graph model and formally formulating the problem on the proposed graph.

Definition 5.3 (*Block Acceptance Rate*). For each screening block B_i on a decision path, an acceptance rate r_i is defined by

$$r_i = tp_i + fp_i, \tag{3}$$

where tp_i and fp_i refer to the true positive rate and false positive rate of the block B_i. The acceptance rate r_i is clearly determining percentage of the actions (including target and non-target) accepted by B_i.

We note that the true positive rate and false positive rate concepts are absolute measures and do not carry the effect of one screening block on the other. That is, tp_i and fp_i values are calculated while all actions are fed to B_i. Similarly, the resulting acceptance rate, r_i, represents the percentage of the actions that pass through a block if all actions are used as input to that block.

5.3. Problem Formulation

In order to present the minimum cost screening path (MCSP) problem, we first map all possible decision paths onto a graph model called *screening graph*. We then use this model to find the optimal path, including a subset of screening blocks and their ordering for preliminary signal screening.

Definition 5.4 (*Screening Graph*). Given a set of screening blocks $\{B_1, \ldots, B_n\}$ with $b_i < b_{i+1}$, the screening graph $G = \{V, E, W\}$ is a direct acyclic graph defined by a set of vertices, V, a set of edges, E, and sets of weights, W associated with vertex set. The set of vertices, V, is $\{s, v_1, \ldots, v_n, t\}$, where s is a dummy node connected to all other nodes, and t is the destination node associated with the main signal processing chain. Thus, $|V| = n+2$. Furthermore, each vertex v_i $(1 \leq i \leq n)$ is associated with a screening block B_i. An edge e_{ij} $(i < j)$ connects a vertex v_i (corresponding to a lower-level block B_i) to vertex v_j (corresponding to a higher-level block B_j). Thus, $|E| = \frac{(n+1)(n+2)}{2}$. The set $W = \{w_1, \ldots, w_n\}$ denotes the cost of each vertex for processing a single incoming action and is associated with the energy consumption of corresponding screening blocks.

Definition 5.5 (*Outgoing Flow*). For each vertex $u_i \in V$ on a path $P = \{s, u_1, \ldots, u_l, t\}$ of length l from s to t, we define an outgoing flow (f_i) which represents the percentage of actions that are passed through the corresponding block B_i. The set of outgoing flows for vertices on the path P is denoted by $F = \{f_1, \ldots, f_l\}$.

We note that the outgoing flow, f_i, associated with B_i is different from the acceptance rate, r_i. Acceptance rates define percentage of actions that can be accepted by individual screening blocks, independent of how they are connected to other blocks. Outgoing flows, however, represent the percentage of actions that are accepted by a particular block while considered as part of the screening graph. Each f_i is a function of r_i and the acceptance rate of the preceding blocks on the path, as will be discussed later in this section.

For the dummy node s, $f_s = 1$ resulting in an outgoing flow of 1. The idea is to feed all actions to the dummy node first. A path from s to t determines what screening blocks are active during preliminary classification. Furthermore, $w_s = 0$, because the dummy node does not represent a physical component of the system. We assign a zero outgoing flow to the destination node ($f_t = 0$) because MSPC is the last processing

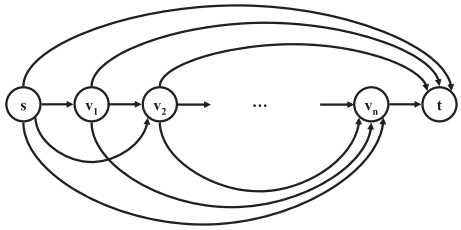


Fig. 6. Screening graph.

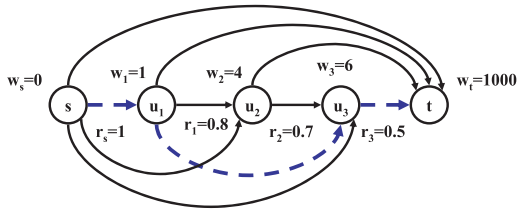


Fig. 7. An example of a screening graph with four vertices. A minimum cost decision path is $P = \{s, u_1, u_3, t\}$ with a total cost of 505 nW.

component of the system and does not convey actions to any subsequent module. The energy consumption of the destination node, w_t, is the amount of energy required for running MSPC on the main processor.

Figure 6 shows the screening graph with v_1 to v_n corresponding to n screening blocks. As mentioned previously, the energy cost of a screening block B_i is denoted by w_i per incoming action. Therefore, the overall cost of each screening block depends on the percentage of the incoming actions that are passed through the decision path to the screening block B_i. This is directly defined by the outgoing flows. Our objective is to find the decision path with minimum overall cost.

To better describe how the overall cost for a path from s to t is calculated, we present a synthetic screening graph (Figure 7) with three screening blocks indicated by u_1, u_2, and u_3. Each vertex has an outgoing flow, f_i, and a weight denoted by w_i. The weights are shown in nW with destination node (t) having a significantly larger amount of power consumption (e.g., $w_t = 1{,}000$) as it corresponds to the main signal processing chain. A path $P = \{s, u_1, u_3, t\}$, which is the minimum cost path with a cost of 505 nW. The edge (s, u_1) has a cost of $f_s \times w_1 = 1$ nW. The next edge, (u_1, u_3) has a cost of $f_1 \times w_3 = 4.8$ nW. We note that, in fact, f_1 is the minimum rate among all previously traversed vertices (s and u_1). The cost for the edge (u_3, t) is $f_3 \times w_t = 500$ nW. Similarly, f_t is the smallest acceptance rate among previously traversed nodes (s, u_1, u_3). In general, however, if r_i is not the smallest rate among all preceding nodes, the cost for an edge e_{ij} is a function of acceptance rates on all traversed nodes. In this example, acceptance rates are monotonically decreasing. Thus, the cost for each edge e_{ij} can be computed based on the rate and weight of adjacent vertices (i.e., r_i and w_j). In order to formally define the MCSP problem, we first define the cost for each decision path on a given screening graph.

Definition 5.6 (*Decision Path Cost*). Given a screening graph G, the total cost for a decision path $P = \{s, u_1, \ldots, u_l, t\}$ of length l is given by

$$\begin{aligned} \mathbf{w}^P &= f_s w_1 + f_1 w_2 + f_2 w_3 + \cdots + f_l w_t \\ &= r_s w_1 + min(r_s, r_1) w_2 + min(r_s, r_1, r_2) w_3 + \cdots + min(r_s, r_1, \ldots, r_l) w_t, \end{aligned} \tag{4}$$

where $u_i \in V$, r_i is the acceptance rate for vertex u_i and $r_s = 1$. Furthermore, each term $f_i w_j = min(r_s, r_1 \ldots, r_i)w_j$ represents the cost associated with the edge $e_{ij} = (u_i, u_j)$ on the path.

The intuition behind using the *min* function in the preceding formulation is that the acceptance rates, r_i, are all calculated with respect to the source node, s, rather than the adjacent node in the graph. Therefore, each r_i denotes the percentage of the actions that can pass through B_i if the node is placed adjacent to s (i.e., all actions are fed to the block). As it can be observed from Definition 5.6, the cost for each edge e_{ij} on the path depends on the cost of u_j and acceptance rate of all previously traversed nodes.

PROBLEM 1. *Given a screening graph G, the MCSP problem is to find a decision path, $\hat{P}$, with minimum cost.*

Definition 5.7 (*Path Acceptance Rate*). For a decision path P from s to t on the screening graph G, the acceptance rate R_P is defined as the percentage of actions that are accepted by all the nodes on the path, and is given by

$$R_P = min_{u_i \in P}(r_i). \tag{5}$$

5.4. Shortest Path Solution

The problem presented in Section 5.3 is different from the traditional shortest-path problem, because the contribution of an edge to path cost depends not only on the cost of that edge but also on the costs of the edges already traversed. A special case of this problem with applications in multimedia data transmission has been studied [Sen et al. 2000].

We transform the MCSP problem to the traditional shortest path by simplifying some of the assumptions on acceptance rate of our screening blocks. We show that under these realistic assumptions, the problem can be solved with computationally simple shortest path algorithms.

In our work, the classifiers use the same template and signal, but linearly quantized at different bit levels. From this model, several basic assumptions can be inferred.

(1) The target actions are rejected in approximately the same order by all the screening blocks on the decision path. Equivalently, if a target action is rejected by B_i, it is also rejected by B_j, while $j > i$. In other words, a higher-level block B_j may reject some target actions that are accepted by B_i. Therefore, compared to a lower-level block, a higher-level block may have a smaller or equal true positive rate ($tp_j \leq tp_i$).
(2) Similarly, the non-target events are rejected in approximately the same order by all the classifiers. Thus, a higher-level block B_j may reject some non-target actions that are accepted by a lower-level block B_i. Therefore, compared to a lower-level block, a higher-level block may have a smaller or equal false positive rate ($fp_j \leq fp_i$).
(3) Classifiers at higher quantization bit levels perform better or equal to classifiers at lower bit levels. That is to say, for two screening blocks with equal true positive rates $tp_i = tp_j = F$ and $j > i$, then $r_j \leq r_i$. In fact, in order to achieve the lower bound λ on true positive of the entire granular decision making, we set the threshold thr_i on each screening block such that the fixed true positive rate of λ is obtained.

THEOREM 5.8. *If $u_1, u_2, \ldots, u_k$ (associated with screening blocks $B_1 \ldots B_k$) form an optimal decision path, the cost of an edge e_{ij} is a function of w_j and r_i.*

PROOF. As shown in Eq. (4), the total cost associated with edge e_{ij} on path P is given by

$$w_{ij}^e = min(r_s, r_1, \ldots, r_i)w_j. \tag{6}$$

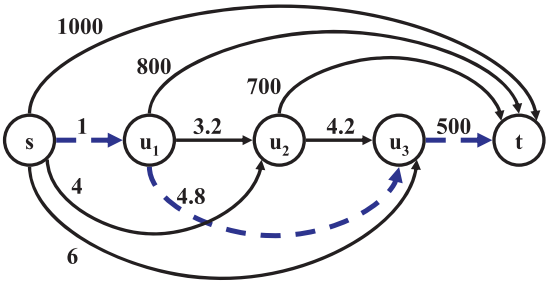


Fig. 8. Example of shortest-path calculation.

The assumptions on monotonically decreasing acceptance rate would result in u_i having smallest acceptance rate among all preceding nodes. That is $min(r_s, r_1, \ldots, r_i) = r_i$. Therefore,

$$w_{ij}^e = r_i w_j. \tag{7}$$

□

The immediate result of Theorem 5.8 is that the cost of each edge on the decision path is deterministic and can be computed before running any algorithm for computing the path. Therefore, the problem is transformed into a simple shortest-path problem [Moore 1959].

Calculating shortest path on directed acyclic graphs (DAGs) is computationally less expensive than on general graphs, based on the principal of topological ordering. It can be done by processing the vertices in a topological order and computing the path cost from each vertex to be the minimum cost obtained via any of its incoming links. Each screening graph in our modeling is a DAG with a unique topological ordering. This ordering is specified by the directed path that contains all the vertices. In the example shown in Figure 8, the topological ordering is $\{s, u_1, u_2, u_3, t\}$. In this example, costs associated with edges are calculated by applying Eq. (7) on the graph in Figure 7. The shortest path from s to t is $\{s, u_1, u_3, t\}$, which is calculated as follows. Vertex u_1 has only one incoming edge. Thus, there is only one path from s to u_1 with the cost of 1. The next vertex in the topological ordering is u_2, which has two incoming edges associated with two paths ($\{s, u_1, u_2\}$ and $\{s, u_2\}$ with the costs of 4.2 and 4, respectively. Therefore, minimum cost path from s to u_2 is $\{s, u_2\}$. Similarly, u_3 has three incoming edges which indicate three paths $\{s, u_2, u_3\}$, $\{s, u_1, u_3\}$, and $\{s, u_3\}$, introducing costs of 8.2, 5.8, and 6. Consequently, $\{s, u_1, u_3\}$ is the minimum cost path from s to u_3. And finally, destination t has four incoming edges representing four candidates for the shortest path. These paths include $\{s, u_1, u_3, t\}$, $\{s, u_2, t\}$, $\{s, u_1, t\}$, and $\{s, t\}$ with the costs values of 505.8, 704, 801, and 1,000 respectively. Thus, $\{s, u_1, u_3, t\}$ is the overall shortest decision path from s to t with a cost of 505.8.

6. DESIGN STEPS

Based on the design approach that we discussed in previous sections, we outline different steps that need to be taken in order to find an optimal decision path using the proposed GDMM. In describing these steps, we assume that a set of training examples associated with human actions are given and one action is specified as target. Furthermore, the system is aimed to achieve a minimum desirable sensitivity to the classification of the target action. Moreover, a set of screening blocks associated with different bit resolutions, the microcontroller and their corresponding power consumptions are given.

Table II. Experimental Actions

No.	Action
1	Stand to Sit
2	Sit to Stand
3	Sit to Lie
4	Lie to Sit
5	Bend and grasp
6	Rising from Bending
7	Kneeling
8	Rising from Kneeling
9	Look Back
10	Return from look back
11	Turn clockwise
12	Step forward
13	Step Backward
14	Jumping

—*Step 1.* Generate a template associated with the target action using the training examples and based on the approach described in Section 5.1.
—*Step 2.* Set the threshold for each screening block such that the block meets the minimum desirable sensitivity. In order to maintain this requirement, the threshold for a given screening block is set to the largest value that satisfies this requirement. This is mainly due to the fact that larger values of the threshold result in lower sensitivity rates. Thus, we start from a small value (e.g., close to '0' which would result in 100% sensitivity) and increase this value as long as the sensitivity rate is still above the desirable rate.
—*Step 3.* For each screening block, compute the acceptance rate as the percentage of actions that are being accepted by that block, as described in Section 5.2. We note that depending on the desirable sensitivity rate, different blocks will have different acceptance rates. Therefore, a given screening block will experience higher acceptance rates when the sensitivity increases. The acceptance rate is a function of the desirable sensitivity. In fact, the sensitivity determines the threshold for each block and similarly affects the acceptance rate.
—*Step 4.* Construct a screening graph as described in Section 5.3. The power consumption and acceptance rate of the individual screening blocks are used to calculate the parameters of the screening graph.
—*Step 5.* Execute a shortest-path algorithm on the screening graph to compute the minimum cost decision path as described in Section 5.4. In finding a decision path, we note that the power consumption of GDMM is a function of (1) the power consumption of the active screening blocks that are on the path, and (2) the percentage of actions that are passed to each block (acceptance rate of the preceding block on the path).

7. EXPERIMENTAL RESULTS

We evaluated performance of our granular decision-making architecture for identifying each one of the 14 target actions listed in Table II. In each phase, one of the actions was considered as the target action and the rest as non-target. A set of experiments was carried out on three male subjects, all between the ages of 25 to 35 and in good health condition. Subjects were asked to repeatedly perform each specific action ten times.

7.1. Data Acquisition Platform

We used nine motion sensor nodes for data collection. Each sensor node had a three-axis accelerometer, a two-axis gyroscope. The data were collected and processed in

MATLAB. For simplicity, we used only one sensor/node for detecting each target action (e.g., Z-axis of the node placed one the 'waist' for 'sit to stand'). For actions that require multiple sensors, the same methodology can be used. That is, the template matching multiple nodes/axes could be utilized to activate MSPC. In this case, a data fusion algorithm will be used to combine decisions made by different sensors and decide if the microcontroller needs to be turned on/off. The data fusion from multiple sensor nodes is out of the scope of this article (for brevity) and is the subject of our future work.

For the purpose of action recognition, we used TelosB motes [Polastre et al. 2005] which have an embedded MSP430 microcontroller, particularly used for executing main signal processing tasks, and consume 3 mW in active mode.

7.2. Template Generation

The sensor data recorded from each action were equally split into training and test datasets. The training dataset was used for template generation as well as construction of the optimal decision path, and the test dataset was used for validation of the results.

7.3. Parameter Setting

As discussed in Section 5, a screening graph has two set of parameters which are used for constructing optimal decision path. These parameters include weights (W) and incoming rates (R) associated with different screening blocks. Both parameters are calculated using training trials. Weights are calculated based on the amount of energy consumed by corresponding screening blocks. Incoming rates are calculated by examining percentage of training trials that are accepted by each screening block.

To estimate the energy consumption of each screening block, the screening blocks were implemented using template matching units, as described previously. The template matching function was modeled using Verilog. The cross-correlation was implemented by a series of MAC steps depending on the number of incoming samples. At each clock instant, the digitized template data and the incoming signal data were multiplied and added to the previous MAC value. The multiply-add operation repeated depending on the length of the template in order to calculate the cross correlation function. All the operations were carried out at a low frequency of 20 Hz. The design was synthesized using Synopsys with the 45nm NanGate Open Cell library. The switching activity was then considered, and the power numbers were computed in Synopsys. The power values ranged between 0.34 nW for the 1-bit block ($w_1 = 0.34$) and 1.45 nW for the 12-bit screening block ($w_{12} = 1.45$).

In order to calculate the incoming rates (R) on individual vertices of the screening graph, we set the threshold (thr_i) on each screening block such that the desired true positive rate (λ) is obtained. In fact, the threshold is set to guarantee the lower bound λ on the overall true positive rate of the system. Therefore, the threshold on each screening block B_i is given by

$$\hat{thr}_i = \arg\max_{thr_i} \quad tp_i \geq F. \tag{8}$$

We note that tp_i (sensitivity) decreases as thr_i grows. Thus, thr_i is set to the largest possible value that meets the desired sensitivity requirement. Thus, we start with a small value (e.g., close to '0' which would result in 100% sensitivity) and increase this value as long as the sensitivity rate is above the desirable rate.

7.4. Decision Paths

Table III shows decision path reported by our optimization technique while the desired true positive rate (λ) varies from 50% to 100%. In this table a node 's' denotes the source

Table III. Shortest Paths for Detecting 'Sit to Stand'

Sensitivity (%)	Decision Path	Path Activation (%)	Classifier Threshold
50	$s \rightarrow B_1 \rightarrow B_{12} \rightarrow t$	4.76	0.95
55	$s \rightarrow B_1 \rightarrow B_{12} \rightarrow t$	5.71	0.94
60	$s \rightarrow B_1 \rightarrow B_{12} \rightarrow t$	5.71	0.94
65	$s \rightarrow B_1 \rightarrow B_{12} \rightarrow t$	7.62	0.93
70	$s \rightarrow B_1 \rightarrow B_{12} \rightarrow t$	9.05	0.92
75	$s \rightarrow B_2 \rightarrow B_{12} \rightarrow t$	12.9	0.86
80	$s \rightarrow B_2 \rightarrow B_{12} \rightarrow t$	12.9	0.86
85	$s \rightarrow B_2 \rightarrow B_{12} \rightarrow t$	14.3	0.85
90	$s \rightarrow B_2 \rightarrow B_{12} \rightarrow t$	16.7	0.84
95	$s \rightarrow B_2 \rightarrow B_{12} \rightarrow t$	22.9	0.80
100	$s \rightarrow B_2 \rightarrow B_{12} \rightarrow t$	22.9	0.80

and a 't' represents the destination node (associated with the main processor). In all cases, only two screening blocks are chosen by the algorithm. We note, however, that the total energy depends also on the bit resolution of the individual screening blocks. For example, B_{12} consumes more power than B_1 and B_2. An interesting observation is that more power hungry blocks (e.g., B_2) are chosen as the desirable sensitivity increases. This observation can be interpreted as follows. When sensitivity rate increases, lower-quality blocks (e.g., B_1 in this example) may not be able to provide the amount of granularity that is required for obtaining the desirable sensitivity. Thus, higher level blocks (i.e., B_2) get activated to provide sufficient performance. Cleary, this would result in more power consumption, in favor of higher accuracy.

The third column in Table III shows path acceptance rate, that is, percentage of the time that the main signal processing chain (MSPC) is activated by the algorithm. We note that the first screening block (e.g., three-bit or four-bit template matching block) is active all the time. However, the second screening block is activated based on the outcome of the previous template matching.

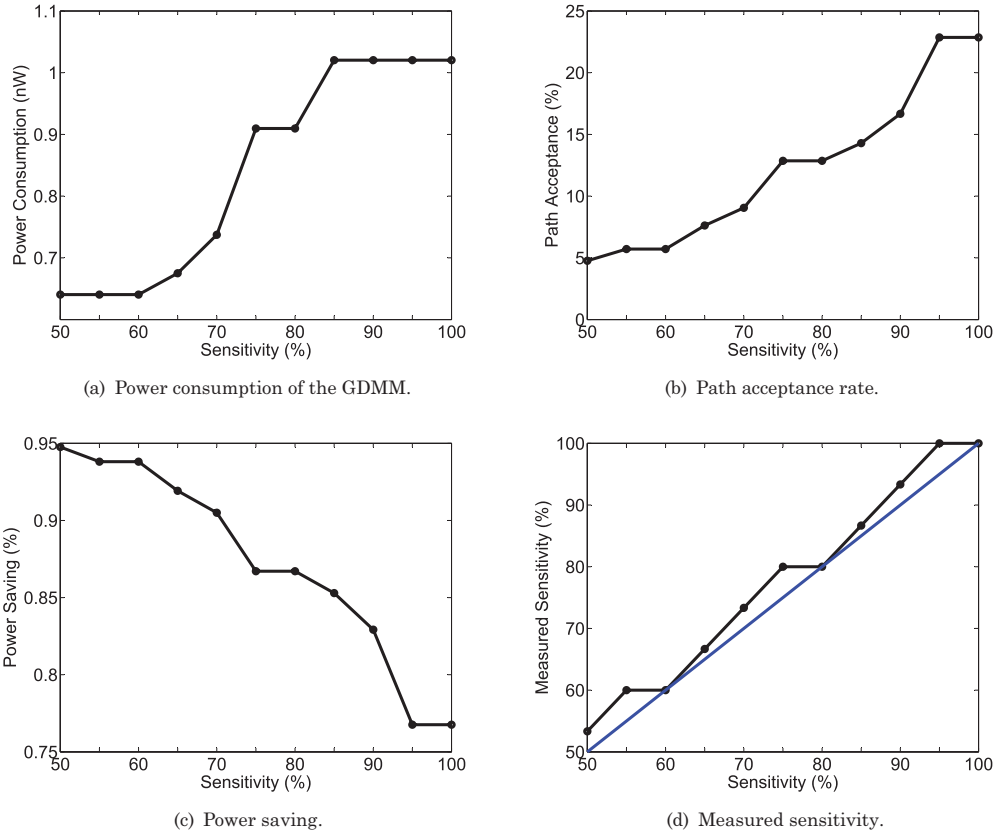
The last column in Table III shows the threshold values assigned to the last screening block (i.e., B_{12}) on the path. These values determine how similar an incoming signal and the target template need to be in order to classify the signal as target. Correspondingly, this indicates the amount of difference between a non-target action and the target action in order to properly classify them. We recall that the classification is based on the similarity score in Definition 5.2. If the similarity score between an incoming signal and the template exceeds the threshold, the signal is classified as target. Otherwise, the signal is classified as non-target, resulting in the main processor remaining inactive. Clearly, the threshold value for a particular GDMM depends on the sensitivity of the system, which is a design parameter. Higher sensitivities incur smaller thresholds, allowing more actions to be classified as target.

7.5. Power Analysis

The power optimization problem with a desired true positive rate would lead to a shortest-path problem, as explained in Section 5.4. Power and accuracy results for detecting 'sit to stand' through the granular decision-making module (GDMM) are shown in Figure 9. Figure 9(a) shows the power consumption of the GDMM. The power consumption ranges from 0.64 nW to 1.02 nW with an average of 0.84 nW.

As expected, Figure 9(b) confirms that MSPC is activated more often when a higher sensitivity is desired. The value of path activation ranges from 4.8% for the case of 50% sensitivity to 22.9% for the case of 100% sensitivity.

The power consumption of our decision-making module can be compared with that of an MSP430 microcontroller, which consumes 3 mW in active mode. Figure 9(c) shows



(a) Power consumption of the GDMM.

(b) Path acceptance rate.

(c) Power saving.

(d) Measured sensitivity.

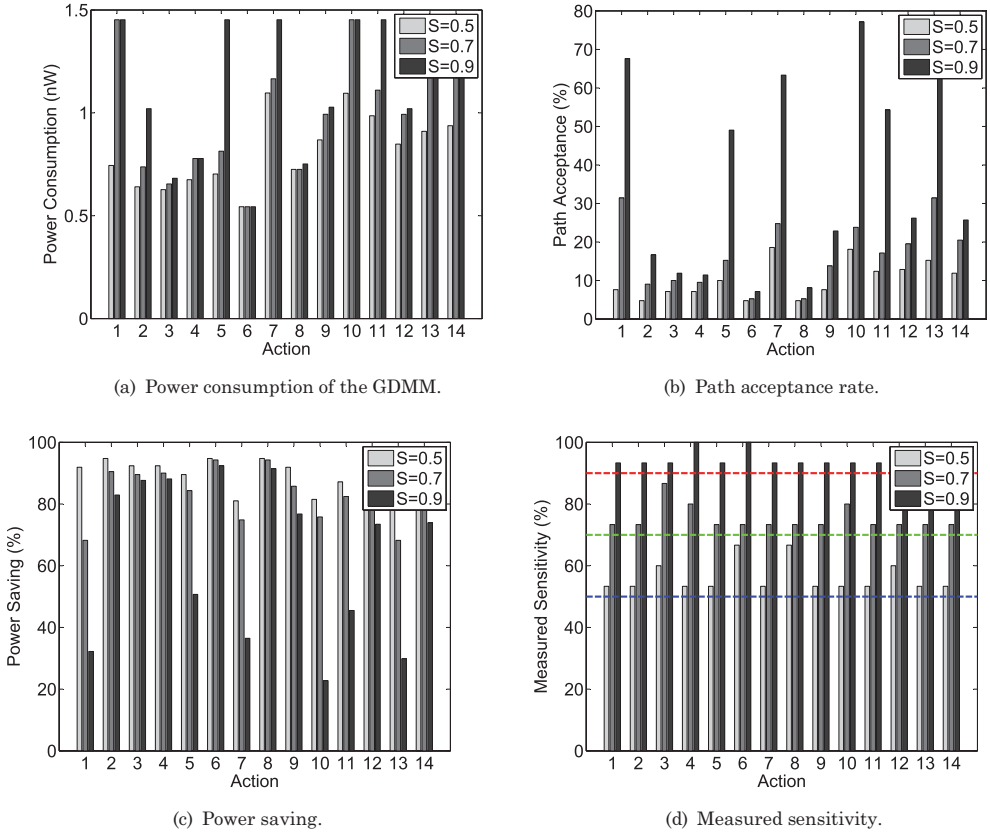
Fig. 9. Results for detecting 'Sit to Stand'.

the amount of power savings obtained by our system compared to a system with the microcontroller being instantly active (a direct connection between node s and t in the screening graph). As expected, the higher the true positive rate is, the lower energy saving that can be achieved. Depending on the desired sensitivity, the percentage of power saving ranges from 76.8% to 94.8%.

Once the optimal decision path is constructed, it can be used to measure its actual precision when test trials are applied. This simulates a real-time scenario where incoming signals are examined by the decision-making module for identification of a specific target action. For this purpose, we fed the sensor data to the optimal decision paths shown in Table III. The actual measured sensitivities are shown in Figrue 9(d). The values range from 53.3% to 100% with an average of 77.6%. We note that all values on the graph in Figure 9(d) are above the dashed line, which implies that the measured sensitivity is always higher than the desired lower bound (λ).

7.6. Single Action Detection

In order to establish the robustness of our granular decision-making architecture with respect to different target actions, we used the data collected from our sensing platform and considered each action in Table II to be the target action. In each case, the appropriate template was chosen and the specific action was considered as the target. For each target action, the optimal decision path was constructed from the screening graph, as discussed in Section 5. For extracting each decision path, the thresholds



(a) Power consumption of the GDMM.

(b) Path acceptance rate.

(c) Power saving.

(d) Measured sensitivity.

Fig. 10. Results when each action is considered as the target action. Results are shown for three different values of desired sensitivity (S = 50%, S = 70%, S = 90%).

were set on each template matching block to meet the desired true positive rate. The sequence of the screening blocks on the decision path was used to accept/reject each incoming signal and activate MSPC. The set of screening blocks were then employed to classify the target movement. The results obtained from this analysis are shown in Figures 10(a)–10(d) for three different sensitivity rates. In particular, our system achieves an average power saving of 63.1% while maintaining a sensitivity of 92.7%.

Figure 10 shows the details of this analysis. Figure 10(a) shows power consumption of GDMM for detecting each one of the 14 target actions. For the case of a 90% sensitivity, the power consumption of the GDMM veries depending on the target action. It ranges from 0.543 nW for 'Rising from Bending' to 1.45 nW for 'Stand to Sit' and 'Kneeling'. On average, the screening path consumes 1.14 nW. The amount of power saving ranges from 22.7% to 92.4% with an average of 63.1% while maintaining a minimum sensitivity of 90%.

7.7. Detecting Multiple Actions

Thus far, we have considered the case where our granular decision-making module is trained to monitor only one particular action (associated with the template) and our experimental analysis is presented accordingly to wake the main processor up upon occurrence of the target action. Yet, this methodology can be extended to detect a group of target actions. In case of multiple target actions, one can use one of the following

Table IV. Power Consumption and Savings for Detecting Multiple Actions

	S = 0.5	S = 0.7	S = 0.9
Power Consumption (mW)	2.7	3.6	3.9
Power Saving (%)	71.5	38.3	20.9

Table V. Similarity Scores for Irregular Walking

	$\gamma(a_1, a_1)$	$\gamma(a_2, a_2)$	$\gamma(a_1, a_2)$
Mean Value	0.81	0.79	0.68
Standard Deviation	0.17	0.18	0.10

Note: a_1 is 'touch right side' and a_2 is 'touch left side'.

approaches. (1) The most straightforward, but not essentially optimal, approach is to replicate the decision making module for each action of interest, where each module classifies one target template and activates the main processor independent of the others. This approach may result in less power saving because similarity among target actions is not taken into consideration, and therefore, some of the modules might be redundant. (2) Another approach is to generate a unique template that represents all actions of interest and use that template as target template. This method is more efficient, but may not be feasible if actions of interest are significantly different in terms of structural patterns, which makes generation of a common template challenging. (3) If a large number of target actions with significant structural variations are given, a combination of the two aforementioned approaches can be used. Similar actions can be grouped together to generate a unique template, and one granular decision-making module can be constructed per action group.

In order to provide insight into how detection of multiple actions can be addressed using the GDMM, we used the first approach (using one GDMM for each target action) to detect the first four actions in Table II. The results are illustrated in Table IV for different sensitivity values.

7.8. Detecting Abnormal Gaits

Practically, our systems can be used in different wearable monitoring applications, in particular for patients monitoring in healthcare domain. One specific area of interest is gait analysis, where signal processing algorithms are designed to detect abnormal gaits and extract detailed information from gait cycles. In such application, the GDMM can be utilized to activate the main processor when an abnormal walking pattern is observed. A more detailed analysis will be accomplished by the MSPC on the main processor. In this section, we demonstrate the effectiveness of the proposed screening approach in highlighting irregularities in walking movements. Irregular gait patterns are generated by applying some disturbance while walking under a controlled experimental environment.

A set of walking experiments was conduced, where a motion sensor node was placed on the ankle of the subject while walking at a speed of 3.5 mph (miles per hour) on a treadmill. The subject was asked to perform two types of irregular actions, 'touch right side' and 'touch left side' (denoted as a_1 and a_2). Each action was repeated ten times. The X-axis of the gyroscope sensor (along the medial-lateral direction) was used to calculate similarity scores between the two actions as well as different trials of the same action. Table V shows the mean and standard deviation of the similarity scores for different test categories. Although the ankle experiences only slight differences during the execution of these two actions, this minor variation is readily captured by the template matching function. While normalized cross correlation between different

trials of the same action is about 0.8, it is less than 0.7 when comparing trials of different actions.

8. DISCUSSION AND FUTURE WORK

The power consumption of our granular decision-making module is six orders of magnitude smaller than state-of-the-art low-power microcontrollers.

A unique property of our low-power wake-up circuitry module is the preliminary signal processing (template matching) that triggers activation of the main processor. To the best of our knowledge, our work is the first study on integrating signal processing within a wake-up circuitry block that both performs preliminary classification and activates main processor on events of interest. Thus, it is infeasible to compare power savings achieved by our systems with those reported by previous studies.

An important property of the proposed decision-making architecture is its programmability in specifying the target event as well as power/accuracy trade-offs. The screening blocks (i.e., decision path) obtained from the optimization problem can vary depending on the action of interest. One can program the module to enable the blocks that are needed for detecting any particular target action. Furthermore, the template matching threshold can be adjusted to meet specific power/accuracy requirements of the application. Thus, a decision path is fixed for the particular action recognition application and its desirable accuracy needs. The path, however, may change as the system is deployed in a different application scenario.

Given the fact that our GDMM requires small amount of resources and introduces negligible delay in decision making, the costs associated with including this module with the wearable sensor unit are negligible. We envision that the proposed GDMM be part of future microcontrollers used for wearable monitoring systems. It operates at a very low-power regime (nano Watts) and consumes a small amount of memory to store action templates and buffer incoming signal. Thus, the amount of increase in the form-factor of the wearable unit is insignificant. The module, however, introduces some delay, associated with the preliminary signal processing, in the overall signal processing. Yet, this delay is readily tolerable as human actions are slow (e.g., occurring in seconds), while our preliminary processing module is optimized to immediately trigger the activation of the main processor by checking incoming signals in a very low bit resolution.

The amount of power savings that can be achieved by our decision-making architecture highly depends on the frequency of occurrence of the target action. For our experiments, we assumed that all actions are equally likely, and therefore, 'sit to stand' occurs 10% of the time. In reality, however, human actions are sparse, occurring much more infrequently, which results in much higher power savings.

In our experiments, we used only a single sensor (e.g., Z-axis of accelerometer on the waist node) to detect target actions. In general, there might be actions that require information from multiple sensors. In such cases, the designer can replicate the granular decision making to accept/reject each incoming action. The microcontroller can be activated based on the decision made by a decision fusion module. The data fusion from multiple sensor nodes is out of scope of this article and is the subject of our future work.

In this article, we focused on optimizing screening path with respect to bit resolution. In the future, we will also investigate the effect of other tuning parameters, such as the sampling rate and window size on the accuracy and the complexity of the signal processing, as well as the energy saving.

9. CONCLUSION

We proposed a light-weight signal processing methodology for body sensor networks applications by early rejection of non-target actions. The proposed hardware-assisted

algorithm uses template matching blocks at different bit levels and finds an optimal order for their execution. Our experimental results demonstrate the effectiveness of the proposed architecture in reducing the power consumption of the system. In particular, we achieved an average energy saving of 63.1% while maintaining 94.3% true positive rate in detecting actions of interest.

REFERENCES

BENBASAT, A. Y. AND PARADISO, J. A. 2007. A framework for the automated generation of power-efficient classifiers for embedded sensor nodes. In *Proceedings of the 5th International Conference on Embedded Networked Sensor Systems (SenSys'07)*. ACM, New York, NY, 219–232.

CHAIMANONART, N. AND YOUNG, D. 2009. A wireless batteryless in vivo EKG and body temperature sensing microsystem with adaptive RF powering for genetically engineered mice monitoring. In *Proceedings of the 15th International Conference on Solid-State Sensors, Actuators and Microsystems*. 1473–1476.

CHAO, L., TSUI, C.-Y., AND KI, W.-H. 2007. A batteryless vibration-based energy harvesting system for ultra low power ubiquitous applications. In *Proceedings of the IEEE International Symposium on Circuits and Systems*. 1349–1352.

CONG, P., CHAIMANONART, N., KO, W., AND YOUNG, D. 2009a. A wireless and batteryless 10-bit implantable blood pressure sensing microsystem with adaptive RF powering for real-time laboratory mice monitoring. *IEEE J. Solid-State Circuits 44,* 12, 3631–3644.

CONG, P., CHAIMANONART, N., KO, W., AND YOUNG, D. 2009b. A wireless and batteryless 130mg 300w 10b implantable blood-pressure-sensing microsystem for real-time genetically engineered mice monitoring. In *Proceedings of the IEEE International Solid-State Circuits Conference-Digest of Technical Papers*.

CONG, P., KO, W., AND YOUNG, D. 2010. Wireless batteryless implantable blood pressure monitoring microsystem for small laboratory animals. *IEEE Sensors J. 10,* 2, 243–254.

CUDDIHY, P., WEISENBERG, J., GRAICHEN, C., AND GANESH, M. 2007. Algorithm to automatically detect abnormally long periods of inactivity in a home. In *Proceedings of the 1st ACM SIGMOBILE International Workshop on Systems and Networking Support for Healthcare and Assisted Living Environments*. 94.

DOUSEKI, T. 2004. Ultralow-voltage MTCMOS/SOI technology for batteryless mobile system. In *Proceedings of the International Conference on Solid-State and Integrated Circuits Technology (ICSICT)*. Vol. 2. 1242–1247.

DOUSEKI, T., TSUKAHARA, T., YOSHIDA, Y., UTSUNOMIYA, F., AND HAMA, N. 2003. A batteryless wireless system with MTCMOS/SOI circuit technology. In *Proceedings of the Custom Integrated Circuits Conference*. 163–168.

DUDA, R. O., HART, P. E., AND STORK, D. G. 2000. *Pattern Classification*. Wiley-Interscience Publication, Hoboken, NJ.

GHASEMZADEH, H., BARNES, J., GUENTERBERG, E., AND JAFARI, R. 2008. A phonological expression for physical movement monitoring in body sensor networks. In *Proceedings of the 5th IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS'08).* 58–68.

GHASEMZADEH, H., GUENTERBERG, E., AND JAFARI, R. 2009. Energy-efficient information-driven coverage for physical movement monitoring in body sensor networks. *IEEE J. Select. Areas Commun. 27*, 58–69.

GOLDBERG, D. H., ANDREOU, A. G., JULIÁN, P., POULIQUEN, P. O., RIDDLE, L., AND ROSASCO, R. 2004. A wake-up detector for an acoustic surveillance sensor network: algorithm and VLSI implementation. In *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks (IPSN'04)*. ACM, New York, NY, 134–141.

HAGLER, S., AUSTIN, D., HAYES, T., KAYE, J., AND PAVEL, M. 2010. Unobtrusive and ubiquitous in-home monitoring: A methodology for continuous assessment of gait velocity in elders. *IEEE Trans. Biomed. Eng. 57,* 4, 813–820.

HUYNH, T. AND SCHIELE, B. 2005. Analyzing features for activity recognition. In *Proceedings of the Joint Conference on Smart Objects and Ambient Intelligence: Innovative Context-Aware Services: Usages and Technologies*. 159–163.

JEVTIC, S., KOTOWSKY, M., DICK, R. P., DINDA, P. A., AND DOWDING, C. 2007. Lucid dreaming: Reliable analog event detection for energy-constrained applications. In *Proceedings of the 6th International Conference on Information Processing in Sensor Networks (IPSN'07).* ACM, New York, NY, 350–359.

LESTER, J., CHOUDHURY, T., AND BORRIELLO, G. 2006. A practical approach to recognizing physical activities. In *Proceedings of the 4th International Conference on Pervasive Computing*. 1–16.

LIEW, W.-S., ZOU, X., YAO, L., AND LIAN, Y. 2009. A 1-v 60-uw 16-channel interface chip for implantable neural recording. In *Proceedings of the Custom Integrated Circuits Conference*. 507–510.

MALINOWSKI, M., MOSKWA, M., FELDMEIER, M., LAIBOWITZ, M., AND PARADISO, J. A. 2007. Cargonet: A low-cost micropower sensor node exploiting quasi-passive wakeup for adaptive asychronous monitoring of exceptional events. In *Proceedings of the 5th International Conference on Embedded Networked Sensor Systems (SenSys'07)*. ACM, New York, NY, 145–159.

MANDAL, S., TURICCHIA, L., AND SARPESHKAR, R. 2010. A low-power, battery-free tag for body sensor networks. *IEEE Perv. Comput. 9,* 1, 71–77.

MITCHELL, S. L., COLLIN, J. J., LUCA, C. J. D., BURROWS, A., AND LIPSITZ, L. A. 1995. Open-loop and closed-loop postural control mechanisms in Parkinson's disease: Increased mediolateral activity during quiet standing. *Neurosci. Lett. 197,* 2, 133–136.

MOORE, E. 1959. The shortest path through a maze. In *Proceedings of the International Symposium on the Theory of Switching*. Vol. 2. 285–292.

MORRIS, S. AND PARADISO, J. 2002. Shoe-integrated sensor system for wireless gait analysis and real-time feedback. In *Proceedings of the 2nd Joint 24th Annual Conference on Engineering in Medicine and Biology and the Annual Fall Meeting of the Biomedical Engineering Society EMBS/BMES Conference*. 2468–2469.

NAJAFI, N. AND LUDOMIRSKY, A. 2004. Initial animal studies of a wireless, batteryless, MEMS implant for cardiovascular applications. *Biomed. Microdevices 6,* 1, 61–65.

NELSON, R. C. 1991. Qualitative detection of motion by a moving observer. *Int. J. Comput. Vision 7*, 33–46. 10.1007/BF00130488.

OHAYON, M. AND ROTH, T. 2002. Prevalence of restless legs syndrome and periodic limb movement disorder in the general population. *J. Psychosomatic Res. 53,* 1, 547–554.

OU, Z., TANG, X., SU, T., AND ZHAO, P. 2005. Cascade adaboost classifiers with stage optimization for face detection. In *Advances in Biometrics*, Lecture Notes in Computer Science, vol. 3832, Springer-Verlag, Berlin Heidelberg, 121–128.

PARRAMON, J., DOGUET, P., MARIN, D., VERLEYSSEN, M., MUNOZ, R., LEIJA, L., AND VALDERRAMA, E. 1997. ASIC-based batteryless implantable telemetry microsystem for recording purposes. In *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. Vol. 5. 2225–2228.

PATEL, S., LORINCZ, K., HUGHES, R., HUGGINS, N., GROWDON, J., WELSH, M., AND BONATO, P. 2007. Analysis of feature space for monitoring persons with Parkinson's Disease with application to a wireless wearable sensor system. In *Proceedings of the 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS'07)*. 6290–6293.

POLASTRE, J., SZEWCZYK, R., AND CULLER, D. 2005. Telos: Enabling ultra-low power wireless research. In *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN'05)*. 364–369.

RAVI, N., DANDEKAR, N., MYSORE, P., AND LITTMAN, M. 2005. Activity recognition from accelerometer data. In *Proceedings of the National Conference on Artificial Intelligence*. Vol. 20. 1541.

RUMBERG, B., GRAHAM, D. W., AND KULATHUMANI, V. 2010. Hibernets: Energy-efficient sensor networks using analog signal processing. In *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN'10)*. ACM, New York, NY, 129–139.

SARPESHKAR, R. 2006. Ultra low power electronics for medicine. In *Proceedings of the International Workshop on Wearable and Implantable Body Sensor Networks (BSN'06)*. 1–37.

SASAKI, S., SEKI, T., AND SUGIYAMA, S. 2006. Batteryless accelerometer using power feeding system of RFID. In *Proceedings of the SICE-ICASE International Joint Conference*. 3567–3570.

SEN, A., CANDAN, K., FERREIRA, A., BEAUQUIER, B., AND PERENNES, S. 2000. On shortest path problems with non-Markovian link contribution to path lengths. In *Proceedings of the IFIP-TC6/European Commission International Conference on Broadband Communications, High Performance Networking, and Performance of Communication Networks*. Lecture Notes in Computer Science, vol. 1815, Springer-Verlag, Berlin Heidelberg, 859–870.

SHOEB, A., CARLSON, D., PANKEN, E., AND DENISON, T. 2009. A micropower support vector machine based seizure detection architecture for embedded medical devices. In *Proceedings of the 31st Annual International Conference of the IEEE Engineering in Medicine and Biology Society: Engineering the Future of Biomedicine (EMBC'09)*. 4202–4205.

STANGEL, K., KOLNSBERG, S., HAMMERSCHMIDT, D., HOSTICKA, B., TRIEU, H., AND MOKWA, W. 2001. A programmable intraocular CMOS pressure sensor system implant. *IEEE J. Solid-State Circuits 36,* 7, 1094–1100.

TAVAKOLI, M., TURICCHIA, L., AND SARPESHKAR, R. 2010. An ultra-low-power pulse oximeter implemented with an energy-efficient transimpedance amplifier. *IEEE Trans. Biomed. Circuits Syst. 4,* 1, 27–38.

VIOLA, P. AND JONES, M. 2001. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'01)*. Vol. 1. I–511–I–518.

VIOLA, P. AND JONES, M. 2004a. Robust real-time face detection. *Int. J. Comput. Vision 57,* 2, 137–154.

VIOLA, P. AND JONES, M. J. 2004b. Robust real-time face detection. *Int. J. Comput. Vision 57*, 137–154.

YANG, G. AND YACOUB, M. 2006. *Body Sensor Networks*. Springer-Verlag New York Inc., New York, NY.

YANG, Z. 2010. Fast template matching based on normalized cross correlation with centroid bounding. In *Proceedings of the International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*. Vol. 2. 224–227.

ZHANG, P., BUI, T., AND SUEN, C. 2007. A novel cascade ensemble classifier system with a high recognition performance on handwritten digits. *Pattern Recog. 40,* 12, 3415–3429.

Received January 2011; revised August, December 2011, March 2012; accepted July 2012