Low Power Programmable Architecture for Periodic Activity Monitoring

Mohammad-Mahdi Bidmeshki, Roozbeh Jafari
{bidmeshki, rjafari}@utdallas.edu
Department of Electrical Engineering
The University of Texas at Dallas
Richardson, TX 75080-3021

ABSTRACT

Body sensor networks (BSNs) are considered a great example for cyber-physical systems due to their close coupling with human body. Activity monitoring is one of the numerous applications of BSNs. Continuous and real-time monitoring of human activities has many applications in healthcare and wellness domains. BSNs utilizing light-weight wearable computers and equipped with inertial sensors are highly suitable for real-time activity monitoring. However, power requirement is a major obstacle for miniaturization of these wearable systems, due to the need for sizable batteries, and also limits the life time of the system. In this paper, we propose a low-power programmable signal processing architecture for dynamic and periodic activity monitoring applications which utilizes the properties of the physical world (*i.e.*, human body movements) to reduce the power consumption of the system. The significant power reduction is achieved by performing signal processing in a tiered-fashion and removing the signals that are not of interest as early as possible. Our proposed architecture uses wavelet decomposition and is favorable for the discrimination of periodic activities. The experimental results show that our architecture achieves 75.7% power saving while maintaining 96.9% sensitivity in the detection of target actions, compared with the scenario where the signal processing is not performed in tiered-fashion. This creates opportunities to enable the next generation of self-powered wearable computers.

Categories and Subject Descriptors

C.3 [**Computer Systems Organization**]: Special-Purpose and Application-Based Systems—*Signal processing systems, Real-time and embedded systems*

General Terms

Design, Algorithms, Experimentation

ICCPS '13, April 8-11, 2013, Philadelphia, PA, USA

Keywords

Wearable Computing, Body Sensor Networks, Signal Processing, Power Optimization, Activity Monitoring

1. INTRODUCTION

Long-term sensing and real-time monitoring of human body movements has numerous applications in healthcare and wellness assessment. This monitoring reveals important information about the quality of life, specially in those who suffer from diseases such as Parkinson [17], or going through rehabilitation, for example after a knee surgery [2], and provides the basis for home telecare paradigms. Using the real-time activity recognition and classification, special events or activities can be captured and the care provider can be notified to take the appropriate action.

Advances in technology introduce light-weight wearable computers which provide the essential sensing, computing and communication platform needed for these monitoring applications which are called Body Sensor Networks (BSNs). BSNs are excellent examples of cyber-physical systems in which BSN nodes are closely coupled with the physical world, *i.e.*, the human body. Several of these tiny nodes can be placed on different parts of the body and closely monitor and register every health related event on the patient as well as the patient's movements and activities. Sensor nodes equipped with inertial sensors provide a mechanism for the natural capturing of human body movements, with minimum external intervention [3]. However, the size of battery needed to power these nodes during the monitoring period is one of the major obstacles which prevents the miniaturization of these sensor nodes and limits their life time and wearability. Moreover, the limited power source restricts the processing power of the sensor nodes and imposes the use of signal processing algorithms with low complexity, for real-time classification.

Our ultimate objective in the design of BSNs is to create batteryless units which can use body heat or body movements as the source of energy. However, the power budget of these sources is in the order of μW. Current state of the art low-power microcontrollers still require the power budget of a few mW or a few hundreds of μW and are not suitable for this purpose. ASIC design can satisfy the power requirement, but it is not programmable and is restricted to a specific application. BSN applications, on the other hand, have specific properties that guide through a more efficient approach for signal processing.

Despite the necessity of continuous monitoring, many BSN applications are interested in specific events during the mon-

itoring period. Such events (*e.g.* walking) occur sparsely with a low duty cycle ($< 5\%$). Signals in these applications are governed by specific physical models of human body. For example, the user may walk slowly or quickly, that will affect the duration and the signal lengths, but we do not expect to observe a 50 g acceleration. Furthermore, the signals change slowly, thus, sampling rates in these applications are low ($< 100Hz$) which enables slow speed processing. Based on these physical properties, we observe that the majority of the signals that we collect may not be of interest and may significantly differ from the target signal of interest. We exploited these properties to propose the Granular Decision Making (GDM) architecture based on template matching [13].

The intuition behind the GDM is to perform less extensive but very low power (*e.g.*, using less number of bits) signal processing at the beginning. If a signal is an immediate *reject*, GDM will not activate the remaining signal processing modules. If a signal is *likely* of interest, the system increases the decision accuracy and the power to make more confident decisions. The processing modules for this architecture are called *Screening Blocks*. The decision accuracy and the power of *Screening Blocks* can be adjusted by several *tunable* parameters such as bit resolution, frequency of sampling, etc. In the last level of this architecture, a microcontroller can be used to throughly process the signal, or if no such processing is needed, GDM can be used to enable a data recording/forwarding mechanism. In either case, GDM can remove the non-target signals as early as possible while taking into account the physical properties of human body movements and prevent the higher cost processing of the non-target signals. This approach will provide a signal processing which satisfies the μW power budget, yet with the benefit of programmability (*i.e.*, *Screening Blocks* can be programmed according to the target signal(s) in the application, can work on various templates and the interconnections between the *Screening Blocks* would be fully reconfigurable). Parameters like bit resolution, sampling frequency and time duration for template matching have been considered for building the screening blocks in the prior work [9, 10, 13, 14], which are mainly applicable to non-periodic movements and time domain analysis. Template matching is not the most suitable method for the detection of periodic activities due to variations in the intensity or the speed of performing such movements.

Dynamic activities such as walking and running are an important part of our daily movements. Type and intensity of those activities provide important information about the treatment and rehabilitation process in some diseases [20]. Discrete wavelet transform can be computed efficiently [6] and is suitable for the discrimination of periodic activities. To provide a more robust GDM approach, compared to template matching, for the discrimination of dynamic and periodic activities, we introduce a GDM architecture which employs wavelet analysis. We design the screening blocks, based on features extracted using various levels of discrete wavelet packet decomposition. Therefore, the tunable parameters for our screening blocks are the *number of features* and the *level* of the decomposition for extracting features. We also formulate the problem of finding the optimal path of the screening blocks with minimum energy cost and provide an optimal solution.

Our experimental results for a set of activities including two different walking speeds, running, ascending stairs and a combination of other non-periodic movements illustrate that our architecture can maintain 96.9% sensitivity while reducing the power consumption of the overall architecture by 75.7%.

2. PREVIOUS WORK

Several low power wearable systems are proposed for specific BSN applications which are pre-customized for specific signal processing, and are not generalizable or programmable. Examples are ultra low-power wearable biopotential sensor nodes for ECG monitoring [25], an implantable blood pressure ECG sensing with adaptive RF powering [5], and a batteryless MEMS implant for cardiovascular applications [18]. A batteryless accelerometer device powered by RFID mechanism was introduced [22]. It consumes ≈15mW, mainly by its microcontroller. It is unclear if ≈15mW is suitable for batteryless operation.

Various architectural and circuit level approaches have been applied to design low power general processors for sensor network applications. Using near threshold operation and multi-threshold CMOS, authors in [11] introduced a simple processor for ambient temperature sensing applications which consumes a few nanowatts, and only samples temperature readings without significant signal processing. To reduce the power cost of frequently used tasks, *e.g.*, filtering, [12] uses hardware accelerators in conjunction with a general purpose microcontroller for processing rare events. A sub-threshold processor for sensor applications is introduced in [26]. Asynchronous techniques has also been used to design processors for sensor network applications [7, 15]. Our approach is complementary to those techniques and considers the specific properties of signal processing in body movement monitoring applications. Realization of our architecture using those circuit technologies and design techniques will result in much higher power savings for BSN applications and wearable computers.

Several studies [16, 19, 23, 24] have used wavelet analysis to discriminate dynamic activities such as walking, running, ascending and descending stairs. There is not a unified method to use wavelets for this task in the literature. A comparison of frequency domain and wavelet features for the classification of dynamic activities is done in [20].

GDM modules for BSN applications has been proposed for time domain analysis using multiple bit resolutions [10], mini templates [9] and dynamic time warping (DTW) [14]. Significant amount of power saving is achieved by reducing the microcontroller's active cycle. Those architectures use template matching and are useful for the discrimination of non-periodic actions. In this paper, we propose GDM architecture using features which are extracted from wavelet packet decomposition for frequency domain analysis. This architecture is mainly applicable to periodic signals and dynamic activities. Our proposed architecture is fully programmable, while it operates with significantly lower power budget (orders of magnitude smaller) compared to microcontroller based solutions. The significant power saving is achieved by exploiting specific properties that are unique to the applications of movement monitoring.

3. PROPOSED ARCHITECTURE

Many BSN applications consider a specific or a small set of movements or events, *e.g.*, gait analysis only deals with

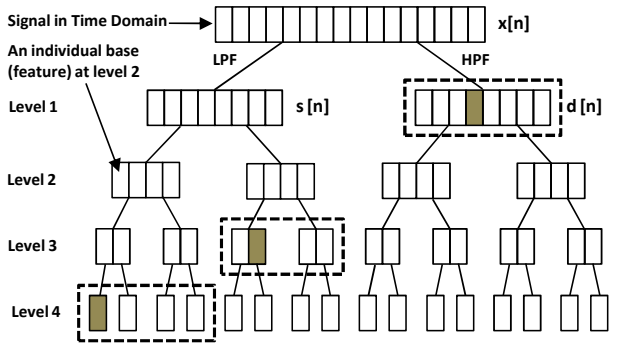

Figure 1: Wavelet packet decomposition tree of a signal of length 16 up to level 4, typical Local Discriminant Bases (dashed boxes), and 3 most discriminant individual bases at each level (gray).

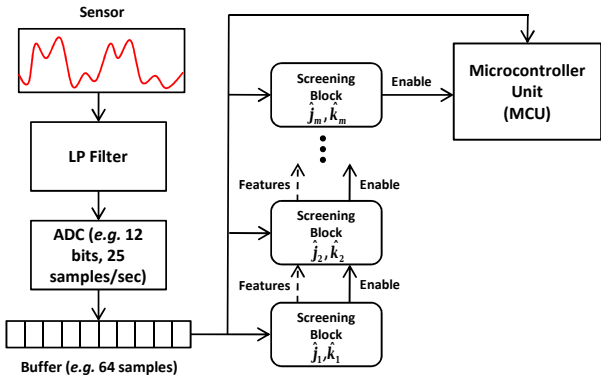

Figure 2: Proposed Granular Decision Making architecture, using optimal path of screening blocks.

walking. Based on this property, we propose an architecture to reject non-target activities with a very low power cost. Our architecture is based on wavelet extracted features and is mainly applicable to dynamic and periodic activities. *Tunable* parameters of our signal processing are the number of features and the level of wavelet packet decomposition in which those features are computed. The cost of processing (power consumption) has a direct relation to these two parameters. We briefly introduce the use of wavelet features for classification and our proposed architecture, in this section.

Assume that we are sampling a quantity, *e.g.*, acceleration on a fixed position of the human body, continuously and process it using a window (buffer) of size n. We refer to this window as the signal, and use wavelet packet transform to decompose it up to $J = \log_2 n$ levels. Fig. 1 shows wavelet packet decomposition tree for signals of length 16 up to level $\log_2 16 = 4$. Local discriminant bases (LDB) [21] was introduced to provide the best representation for the discrimination of signals of various classes using wavelet decomposition (*e.g.*, dashed boxes in Fig. 1). This representation is of the same length as the original signal. To reduce the number of features for the discrimination task, *i.e.*, individual bases, authors in [21] use statistical measures such as Fisher's class separability measure (or its robust version) to find the strength of each feature. Then, a few most powerful individual bases (features) in LDB are selected for the discrimination task. Our experiments on real inertial data from dynamic activities show that often using more features of higher levels in the decomposition can produce more accurate results. However, higher levels in the decomposition have higher computation cost, because to compute an individual base at level $j+1$, corresponding bases at level j are required. We use this property to build a hierarchical architecture which aims at rejecting non-target actions at the lowest possible computational cost. Instead of finding LDB, we treat each level of the decomposition separately. Using robust Fisher's class separability measure, we find up to K_j most powerful individual bases at level $j = 1, 2, \ldots, J$. We build decision making modules at level j which use $k = 1, 2, \ldots, K_j$ most powerful individual bases for acceptance or rejection of the signal. We call these decision making modules *Screening Blocks* denoted by $B_{j,k}$. Screening blocks have different costs (*i.e.*, power consumptions) because they use different number of features and extract features from various levels. We propose a methodology to select a path of screening blocks that will reduce the overall cost.

Fig. 2 shows our proposed GDM architecture that uses features extracted using wavelet packet decomposition and an optimal path of screening blocks. To remove the redundancy in the computation of features, a screening block may also get some features from the previous blocks, if it is using features of the same level. Each screening block processes the signal and if it can confirm that the signal is likely to be the event of interest, it triggers the next screening block for further processing. Using this approach, we can reduce the cost of processing of non-target signals by removing them from signal processing chain as early as possible. In the next two sections, we formulate the problem for finding the optimum path of screening blocks and propose a solution.

4. PROBLEM FORMULATION

To formulate the problem, we model the GDM architecture using a directed acyclic graph (DAG) $G = (V, E)$. Each screening block is represented by a node in this graph. We also have two dummy nodes. First dummy node represents the sensor, shown with $B_{0,1}$, and the other, $B_{J+1,1}$ that represents the microcontroller (or its substitution device). Thus, we have

$$V = \{B_{j,k} | j = 0, 1, \ldots, J+1; k = 1, 2, \ldots, K_j\} \quad (1)$$

where K_j is the maximum number of individual bases which will be considered in level j and $K_0 = K_{J+1} = 1$. We assume a predefined order for the screening blocks: by increasing level (j), and at each level by increasing the number of features (k). Thus we can define the set of edges in graph G as:

$$\begin{aligned} E = \{ & e_{j,k \to j',k'} | \\ & j = 0, 1, \ldots, J+1; k = 1, 2, \ldots, K_j; \\ & j' = j, j+1, \ldots, J+1; \\ & k' = \begin{cases} k+1, k+2, \ldots, K_j & \text{if } j' = j \\ 1, 2, \ldots, k_{j'} & \text{if } j' > j \end{cases} \\ & \} \end{aligned} \quad (2)$$

We assume that we do not need to further process signals which are rejected by the earlier screening blocks. Therefore,

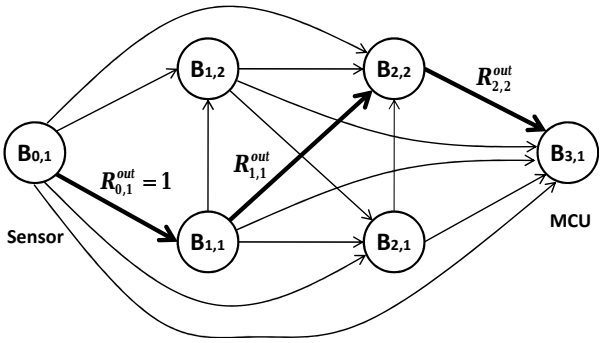

Figure 3: A model of the architecture up to level 2 of wavelet packet decomposition, considering up to 2 features at each level. Thick edges show a typical path from sensor to the microcontroller.

the output rate of $B_{j,k}$ would be a combination of its true positive rate ($tpr_{j,k}$) and its false positive rate ($fpr_{j,k}$). The output rate of $B_{j,k}$ is defined as

$$R^{out}_{j,k} = tpr_{j,k} + fpr_{j,k} \tag{3}$$

and it represents the percent of the signals which are accepted by $B_{j,k}$. The output rate of the sensor node to every other node is 1, *i.e.*,

$$R^{out}_{0,1} = 1 \tag{4}$$

To build our architecture, we need to choose a path in graph G, which starts at the sensor ($B_{0,1}$) and terminates in the microcontroller ($B_{J+1,1}$), *e.g.*, the path shown with thick edges in Fig. 3. Suppose a path starting from sensor node as:

$$P = \{B_{\hat{j}_0,\hat{k}_0}, B_{\hat{j}_1,\hat{k}_1}, B_{\hat{j}_2,\hat{k}_2}, \ldots, B_{\hat{j}_m,\hat{k}_m}\} \tag{5}$$

where $\hat{j}_0 = 0 \quad \text{and} \quad \hat{k}_0 = 1$. Assuming $Cost(j,k)$ as the cost of processing at block $B_{j,k}$, and using the fact that the output rate of a screening block can not be greater than its input, the total cost of processing for path P can be computed as:

$$W_P = \sum_{i=1}^{m} Cost(\hat{j}_i, \hat{k}_i) * R^{in}_{\hat{j}_i,\hat{k}_i} \tag{6}$$

where

$$R^{in}_{\hat{j}_i,\hat{k}_i} = \min\{1, R^{out}_{\hat{j}_1,\hat{k}_1}, R^{out}_{\hat{j}_2,\hat{k}_2}, \ldots, R^{out}_{\hat{j}_{i-1},\hat{k}_{i-1}}\} \tag{7}$$

The goal is to find the minimum cost (minimum power consumption) path from $B_{0,1}$ to $B_{J+1,1}$. However, this is not a straight forward shortest path problem because the input rate of each block depends on the preceding blocks on the path.

Another issue which adds to the complexity of finding the optimum path, comes from the processing cost of the signal in every screening block, which also depends on the selected path. The processing in each screening block has two parts: 1. Computing missing features, 2. Classification (making decision to accept or reject the signal). The computational cost of the classification is deterministic and is directly related to the number of features (k for block $B_{j,k}$). But the cost of computing missing features is not predetermined. This cost depends on the selected path and candidate (most discriminant) features at previous blocks. One type of dependency occurs between the screening blocks of the same level. In Fig. 3, consider block $B_{2,2}$ as an example. Arriving at this block from $B_{1,1}$ or $B_{1,2}$ requires the computation of two features, while arriving from $B_{2,1}$ requires computing only one feature. Dependency might also exist between the features (or bases) from different levels. This happens because of the binary tree structure of wavelet packet decomposition. Computing bases at level j, requires bases from level $j-1$, which depending on the path and selected features, may already have been computed. Due to the complexity that arises from this type of dependency, we assume that screening blocks at each level compute their features independent from other levels.

5. COST OF SCREENING BLOCKS

5.1 Cost of Features

An efficient way of computing discrete wavelet transform of a signal is based on the lifting method [6]. In this work, we use Haar wavelet, which is the simplest wavelet transform and has the lowest computational complexity. One level of decomposition of $x[n]$ using (unnormalized) Haar wavelet can be computed in the following steps:

$$\begin{aligned} d[n] &= x[2n+1] - x[2n] \\ s[n] &= x[2n] + \tfrac{1}{2}d[n] = \tfrac{1}{2}(x[2n] + x[2n+1]) \end{aligned} \tag{8}$$

where $d[n]$ and $s[n]$ are detail coefficients (high passed) and approximation coefficients (low passed) respectively. These computations can be repeated to find bases at the desired level of the decomposition. As these equations show, depending on the position of an individual base, we need one addition, or one addition and one multiplication (or in this special case, a right shift operation) to compute it from the previous level of decomposition. Since the computation of an individual base at level j requires 2 bases from level $j-1$, we will need at most $2^j - 1$ additions and $2^j - 1$ shift right operations to compute an individual base at level j, *e.g.*, 63 additions and 63 shift right operations for a base at level 6. With a sampling frequency of 25 Hz, this translates to 1575 additions/sec and 1575 right-shifts/sec for computing a single base at level 6.

5.2 Classification Using Quadratic Discriminant Analysis

To build light-weight screening blocks, we need to have a simple classification rule for the extracted features. Discriminant analysis is a good candidate for this purpose. It computes the discriminant score of the observed features for each one of classes and chooses the class with the best score for that observation. Assuming normal distribution for those observations, in Quadratic Discriminant Analysis [8] the discriminant score for c-th class, and using k features, is defined as:

$$d_c(\boldsymbol{X}) = (\boldsymbol{X} - \boldsymbol{\mu}_c)^T \boldsymbol{\Sigma}_c^{-1} (\boldsymbol{X} - \boldsymbol{\mu}_c) + \ln|\boldsymbol{\Sigma}_c| - 2\ln \pi_c \tag{9}$$

where $\boldsymbol{X}$ is k dimensional observation vector. $\boldsymbol{\mu}_c$ and $\boldsymbol{\Sigma}_c$ are the mean vector and covariance matrix for c-th class and are computed using training data. π_c is the prior probability for c-th class. The classification rule is to choose the class with minimum $d_c(\boldsymbol{X})$, *i.e.*, maximum posterior probability. Although this method assumes normal distribution for data,

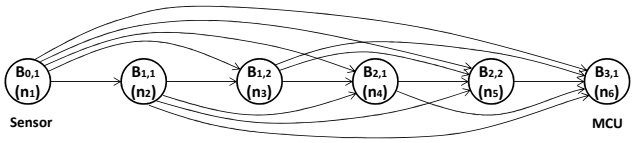

Figure 4: Graph of Fig. 3 with topological ordering.

it is also a useful approximation for non-normal distributions [1]. Since we have only one target movement, we have two classes: target and non-target. However, we do not need to compute the discriminant score of Eq. 9 for both classes. We use a threshold based decision in each block and the decision can be made only using the discriminant score of the target class. By absorbing the constants of Eq. 9 into the threshold value, screening block $B_{j,k}$ will accept $\boldsymbol{X}_{j,k}$ if:

$$(\boldsymbol{X}_{j,k} - \boldsymbol{\mu}_{j,k})^T \boldsymbol{\Sigma}_{j,k}^{-1} (\boldsymbol{X}_{j,k} - \boldsymbol{\mu}_{j,k}) < \hat{thr}_{j,k} \tag{10}$$

$\hat{thr}_{j,k}$ is a design parameter and is determined based on the desired sensitivity (true positive rate) of the system (F). This measure shows what fraction of the target actions are being accepted as the target. The true positive rate of $B_{j,k}$ increases as $\hat{thr}_{j,k}$ increase. However, this also increases the number of non-target actions which are being accepted and increases the false positive rate. Therefore, $\hat{thr}_{j,k}$ is set to the smallest value which satisfies the required true positive rate of the system, *i.e.*,

$$\hat{thr}_{j,k} = \underset{thr_{j,k}}{\arg\min}\, tpr_{j,k} \geq F \tag{11}$$

Note that the target class does not need to represent a single action. A combination of several actions can be considered as the target. However, if those movements do not appear to have similar representations, we may introduce several (independent) decision paths. This does not change the complexity of the proposed solution and will not affect the power performance. Each target action may be viewed separately and its decision path would be formulated individually. Our experimental analysis considers the multiple target-action scenario.

According to Eq. 10, $B_{j,k}$ needs k additions for the computation of $(\boldsymbol{X}_{j,k} - \boldsymbol{\mu}_{j,k})$. Since the covariance matrix is a k-by-k matrix, $k(k+1)$ multiplications and $(k+1)(k-1)$ additions are required for the computation of matrix multiplication. Considering one addition for comparison, the total computation cost of classification for $B_{j,k}$ will be $k(k+1)$ additions and $k(k+1)$ multiplications. For example, a decision using 5 features needs 30 multiplications and 30 additions, which means at 25 Hz sampling rate and making a decision on every new sample, we need 750 additions/sec and 750 multiplications/sec.

6. SHORTEST PATH SOLUTION

As the first part of the solution, we assume that the computational cost of features at each block is deterministic and only the output rates of the screening blocks depend on the path (Eq. 7). Despite this dependency, the solution for our shortest path problem can be formulated by applying the shortest path algorithm for directed acyclic graphs using topological ordering. Fig. 4 shows the sample graph of Fig. 3 with topological ordering representation. To improve

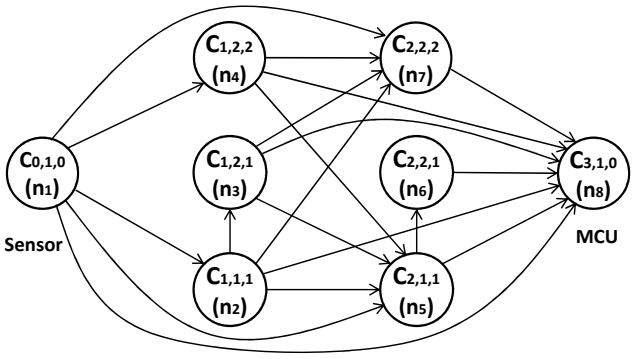

Figure 5: The converted graph of Fig. 3 to consider the computation cost based on missing features in screening blocks of the same level.

the readability, nodes are renamed to $n_1, n_2, \ldots$ based on the order they appear in this representation. According to the definition of the edges in the graph (Eq. 2), it is evident that nodes which appear after n_i does not have any effect on the path from sensor node (n_1) to n_i. Because if a signal is processed by a screening block with higher decision accuracy, it does not need to be processed by screening blocks of lower accuracy. For example, the only path from sensor to n_2 is the path which directly comes from n_1 and that node is the only one which can change the output rate of n_2. Similarly, there are only two paths from n_1 to n_3: directly from n_1, and n_1, n_2, n_3. Once the shortest path to n_2 is found, we can compare the cost of these two paths and find the best path from n_1 to n_3. This best path to n_3 also determines the output rate of n_3. Continuing the same procedure will lead us to the shortest path from sensor node to the microcontroller.

To take into account the dependency for the cost of feature computation in screening blocks of the same level, we build a new graph G' based on G. Node $B_{j,k}$ in G is converted to k nodes $C_{j,k,m}; m = 1, 2, ..., k$ in G'. m in node $C_{j,k,m}$ represents the number of missing features this node needs to compute. In G', we have an edge from $C_{j,k,m}$ to $C_{j',k',m'}$ when one of the following conditions holds:

$$\begin{cases} j' = j \text{ and } k' > k \text{ and } k' - m' = k \\ j' > j \text{ and } k' = m' \end{cases} \tag{12}$$

Fig. 5 shows the converted graph of Fig. 3 using these rules. The computational costs of the nodes in this new graph are deterministic. Using the similar topological ordering for traversing the nodes as the first part of the solution, we can find the shortest path from sensor to the microcontroller.

7. EXPERIMENTAL RESULTS

To measure the power consumption of our proposed architecture, we not only have to consider the implementation details of the architecture but also have to take into account the characteristics of the data and sensor readings obtained through BSNs and human subject studies. The latter in particular is essential to assess the activation frequency of the screening blocks, which will have a significant effect on the overall power consumption. Chip implementation and fabrication of the proposed architecture and *in-situ* valida-

Table 1: Power consumption of screening block $B_{2,5}$ (or $C_{2,5,5}$)

Leakage	Dynamic	Total
0.53 μW	6.85 μW	7.38 μW

Table 2: Periodic movements in the experiments

No.	Type	Specification
1	Ascend stair step	45 stairs/min
2	Ascend stair step	60 stairs/min
3	Ascend stair step	75 stairs/min
4	Walk on treadmill	3 miles/hour, incline: 0%
5	Walk on treadmill	3 miles/hour, incline: 1.5%
6	Walk on treadmill	3 miles/hour, incline: 3%
7	Walk on treadmill	4 miles/hour, incline: 0%
8	Walk on treadmill	4 miles/hour, incline: 1.5%
9	Walk on treadmill	4 miles/hour, incline: 3%
10	Run on treadmill	6 miles/hour, incline: 0%
11	Run on treadmill	6 miles/hour, incline: 1.5%
12	Run on treadmill	6 miles/hour, incline: 3%

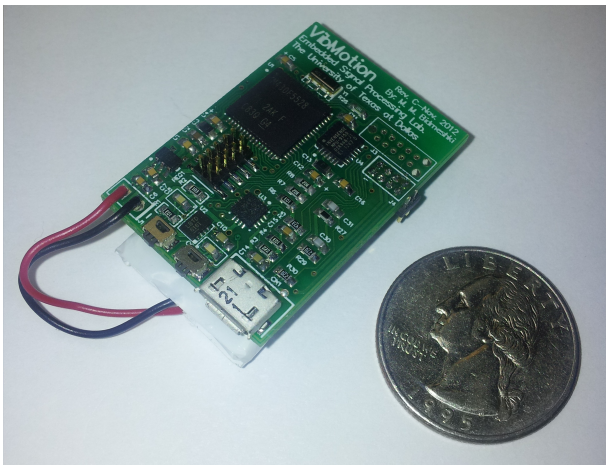

Figure 6: A sensor node equipped with a triaxial accelerometer and gyroscope used for experimental studies and data collection.

Table 3: Decision path for the detection of each action w/ and w/o GDM for the desired sensitivity of 99%

Target Action	w/o GDM	w/ GDM
Ascending Stairs	$C_{6,1,1}$	$C_{3,1,1} \rightarrow C_{3,2,1} \rightarrow C_{6,1,1}$
Walking at 3 mph	$C_{4,4,4}$	$C_{4,1,1} \rightarrow C_{4,2,1} \rightarrow C_{4,4,2}$
Walking at 4 mph	$C_{3,8,8}$	$C_{3,1,1} \rightarrow C_{3,4,3} \rightarrow C_{3,8,4}$
Running at 6 mph	$C_{3,7,7}$	$C_{3,2,2} \rightarrow C_{2,4,2} \rightarrow C_{3,7,3}$
Walking	$C_{3,8,8}$	$C_{3,1,1} \rightarrow C_{3,3,2} \rightarrow C_{3,8,5}$

tion while the proposed architecture is worn on-body was not feasible due to the cost prohibitive issues. Therefore, we synthesized the proposed architecture using CAD tools and measured the power consumption of individual screening blocks. We then applied the sensor readings acquired using BSNs to the screening blocks to measure the power consumption of the proposed architecture *in-situ*. This step was performed using MATLAB and data obtained from subject studies, considering IRB approval and subjects' permission.

To measure the power consumption of screening blocks, we synthesized a simplified 8051 microcontroller using a 180 nm standard cell library working at nominal voltage of 1.8 V, which was deactivated upon finishing the processing required for each screening block. We used the switching activity annotations to get the power consumption of each screening block using Synopsys PrimeTime, which is shown for $B_{2,5}$ (or $C_{2,5,5}$) in Table 1 as an example. There is further room to improve the power efficiency of each screening block. However, this method provides a basis to illustrate the effectiveness of the proposed GDM approach. There are several techniques and technologies such as sub-threshold CMOS [4] that can be applied to our architecture to further reduce the power consumption and create more efficient circuits.

In order to get the inertial data of the activities, we performed a set of experiments in which 4 subjects were asked to perform a set of periodic movements in Table 2 and a set of non-periodic movements and postures such as sitting, sit-to-stand, standing, etc. Since those periodic movements do not occur much in our daily living (*e.g.*, <5% of time), we constructed our data set such that 5% of its movements are the movements of Table 2 and the rest are non-periodic movements and postures. In reality, the portion of perodic movements is much less than 5% and our architecture can provide even higher power efficiency. Each subject wore 5 sensor nodes (Fig. 6) at the waist, knees and ankles which utilize a low power TI MSP430 microcontroller and Bluetooth radio and were equipped with a triaxial accelerometer and gyroscope sensor. The data of each movement were collected for 30 seconds at 25 Hz sampling rate and 12 bits resolution (for non-periodic movements, this duration has been extended with more trials to provide the suitable balance in the data set). We used half of the data for training, tuning the screening blocks and finding the best decision path, and the remaining data for testing and verifying the performance.

In our experiments, we used a window of 64 samples which moves one sample upon arrival of a new sample. This window size covers at least one period of the slowest periodic movement. We used 6 levels of decomposition and considered 10 most powerful bases at each level. The screening blocks were fully designed and tested using MATLAB. All computations were done using fixed point operations by multiplying the coefficients by appropriate constants. Results were obtained using 12 MSBs of the square of the magnitude of the acceleration vector ($\|acc\|^2 = a_x^2 + a_y^2 + a_z^2$) from left knee node.

Because the screening blocks have much less power consumption than a typical microcontroller (which is a few mW), comparing the results to a microcontroller would show significant power savings of orders of magnitude. In our pilot study, the signal processing exhibited suitable accuracy so that a microcontroller for further processing may not be required. Thus, to provide a fair comparison and highlight the benefits of the GDM technique, we use the screening block with the highest accuracy as the point of reference. We find the optimal path of the screening blocks considering the highest accuracy screening block at the end of the path (in place of the microcontroller in Fig. 2 and 3). In the results, we compare two following scenarios: **Scenario one** where the architecture only uses the most accurate screening

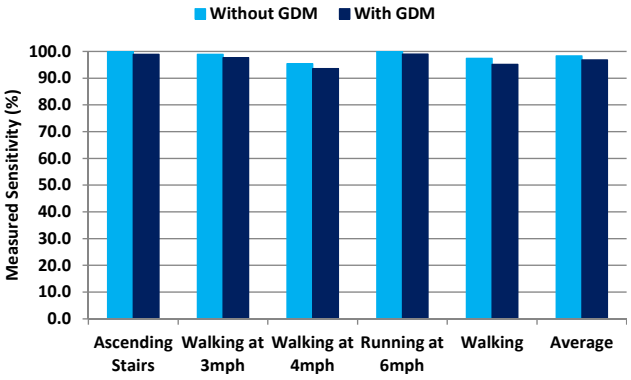

Figure 7: Measured sensitivity w/ and w/o GDM.

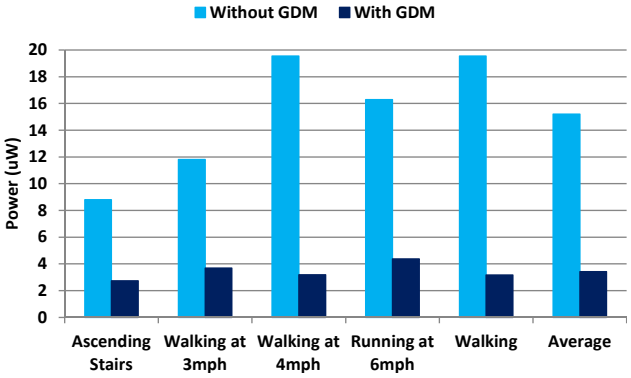

Figure 8: Power consumption w/ and w/o GDM.

block for signal processing and essentially does not exploit removing signals that are not of interest as early as possible (*i.e.*, without GDM), and **Scenario two** where a path of the screening blocks are used (*i.e.*, with GDM). Regardless of the detailed implementation of the screening blocks, this comparison can provide insight in the power benefits of the GDM architecture. Table 3 shows the selected screening blocks in these two scenarios for the detection of each periodic activity. In this table, walking is a combination of two walking speeds and the desired sensitivity was set to 99%. Note that in case of using a single screening block (without GDM - scenario one), it is required to compute all featues for making a decision, so the number of missing features is equal to the number of features that the screening block needs for classification. Fig. 7 compares the measured sensitivity for detection of each movement with and without GDM. By applying the GDM, in scenario two, we lost at most 2.3% of sensitivity in the detection of the target action (when walking is used as the target action) because screening blocks are not completely matched on the rejection decision. However, at the expence of higher false positive rate, the threshold of the screening blocks can be increased to improve the sensitivity. Fig. 8 shows the power consumption in these two scenarios which varies from 8.8 to 19.5 μW for the case of without GDM and 2.7 to 4.4 μW for the case of using GDM. As Fig. 9 shows, our proposed GDM architecture could provide 68.8% to 83.8% power saving.

To elaborate further, Fig. 10 shows the decision path for the detection of running along with the output rates of each

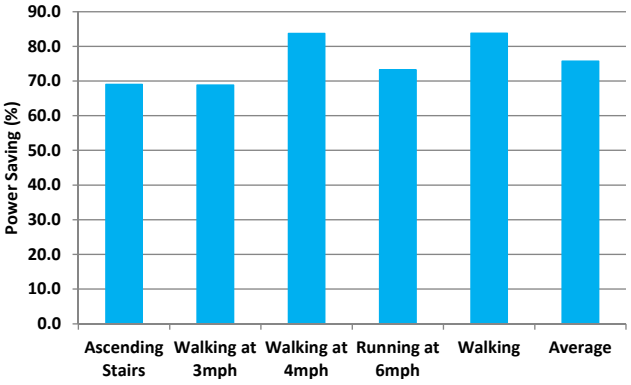

Figure 9: Power saving using proposed GDM.

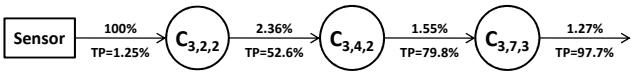

Figure 10: The decision path and output rates for the detection of Running and desired sensitivity of 99%.

screening block. Among sensor readings and data provided to the first screening block on the decision path ($C_{3,2,2}$), 1.25% represents the running activity. This screening block accepts 2.36% of the incoming signals, where 52.6% of them are true positives and represent our target action. The second screening block, $C_{3,4,2}$, rejects another 0.81% of non-target actions (over all incoming signals at the sensor level). The last block on the path accepts only 1.27% of all samples, where 97.7% of them truly represent the target activity.

On average, our GDM architecture could provide 75.7% power savings while maintaining 96.9% sensitivity on the detection of target actions, compared to an architecture that does not perform the decision making in a granular fashion. We also report an average power consumption of 3.7μW for our archirecture as shown in Fig. 8.

8. CONCLUSION AND FUTURE WORK

In this paper, we proposed a GDM architecture to discriminate periodic activities for use in BSN applications. This architecture uses wavelet extracted features to reject non-target actions as early as possible, reducing the need for higher cost signal processing. On average, we could obtain 75.7% power saving while maintaining 96.9% sensitivity on real motion data from several activities. Detection of some actions may require data form multiple nodes. Data fusion from multiple nodes will be considered in our future work. We will also investigate the effect of other parameters such as bit resolution, sampling frequency, window size and wavelet type on the accuracy, complexity and attainable power savings.

9. ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation, under grants CNS-1150079 and CNS-1138396, the National Institute of Health, under grant R15AG037971 and the STARnet, a Semiconductor Research Corporation program sponsored by MARCO and DARPA. Any opinions,

findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding organizations.

10. REFERENCES

[1] E. Alpaydin. *Introduction to machine learning*. The MIT Press, 2004.

[2] L. Atallah, G. Jones, R. Ali, J. Leong, B. Lo, and G. Yang. Observing recovery from knee-replacement surgery by using wearable sensors. In *Body Sensor Networks (BSN), 2011 International Conference on*, pages 29–34. IEEE, 2011.

[3] A. Avci, S. Bosch, M. Marin-Perianu, R. Marin-Perianu, and P. Havinga. Activity recognition using inertial sensing for healthcare, wellbeing and sports applications: A survey. *ARCS 2010*, 2010.

[4] B. Calhoun, D. Daly, N. Verma, D. Finchelstein, D. Wentzloff, A. Wang, S. Cho, and A. Chandrakasan. Design considerations for ultra-low energy wireless microsensor nodes. *IEEE Transactions on Computers*, pages 727–740, 2005.

[5] P. Cong, N. Chaimanonart, W. Ko, and D. Young. A wireless and batteryless 10-bit implantable blood pressure sensing microsystem with adaptive rf powering for real-time laboratory mice monitoring. *Solid-State Circuits, IEEE Journal of*, 44(12):3631–3644, 2009.

[6] I. Daubechies and W. Sweldens. Factoring wavelet transforms into lifting steps. *Journal of Fourier analysis and applications*, 4(3):247–269, 1998.

[7] V. Ekanayake, C. Kelly IV, and R. Manohar. Bitsnap: Dynamic significance compression for a low-energy sensor network asynchronous processor. In *Asynchronous Circuits and Systems, 2005. ASYNC 2005. Proceedings. 11th IEEE International Symposium on*, pages 144–154. IEEE, 2005.

[8] J. Friedman. Regularized discriminant analysis. *Journal of the American statistical association*, 84(405):165–175, 1989.

[9] H. Ghasemzadeh and R. Jafari. An ultra low power granular decision making using cross correlation: Minimizing signal segments for template matching. In *Cyber-Physical Systems (ICCPS), ACM/IEEE International Conference on*, 2011.

[10] H. Ghasemzadeh and R. Jafari. Ultra low power granular decision making using cross correlation: Optimizing bit resolution for template matching. In *Real-Time and Embedded Technology and Applications Symposium (RTAS), 17th IEEE*, pages 137–146, 2011.

[11] S. Hanson, M. Seok, Y. Lin, Z. Foo, D. Kim, Y. Lee, N. Liu, D. Sylvester, and D. Blaauw. A low-voltage processor for sensing applications with picowatt standby mode. *Solid-State Circuits, IEEE Journal of*, 44(4):1145–1155, 2009.

[12] M. Hempstead, D. Brooks, and G. Wei. An accelerator-based wireless sensor network processor in 130 nm cmos. *Emerging and Selected Topics in Circuits and Systems, IEEE Journal on*, 1(2):193–202, 2011.

[13] R. Jafari. Tiered low power wake-up modules for lightweight embedded systems. In *Body Sensor Networks (BSN), International Conference on*, 2011.

[14] R. Jafari and R. Lotfian. A low power wake-up circuitry based on dynamic time warping for body sensor networks. In *Body Sensor Networks (BSN), International Conference on*, 2011.

[15] C. Kelly IV, V. Ekanayake, and R. Manohar. Snap: A sensor-network asynchronous processor. In *Asynchronous Circuits and Systems, 2003. Proceedings. Ninth International Symposium on*, pages 24–33. IEEE, 2003.

[16] N. Lovell, N. Wang, E. Ambikairajah, and B. Celler. Accelerometry based classification of walking patterns using time-frequency analysis. In *Engineering in Medicine and Biology Society (EMBS), 29th Annual International Conference of the IEEE*, pages 4899–4902. IEEE, 2007.

[17] S. Mitchell, J. Collin, C. De Luca, A. Burrows, and L. Lipsitz. Open-loop and closed-loop postural control mechanisms in Parkinson's disease: increased mediolateral activity during quiet standing. *Neuroscience Letters*, 197(2):133–136, 1995.

[18] N. Najafi and A. Ludomirsky. Initial animal studies of a wireless, batteryless, mems implant for cardiovascular applications. *Biomedical microdevices*, 6(1):61–65, 2004.

[19] M. Nyan, F. Tay, K. Seah, and Y. Sitoh. Classification of gait patterns in the time-frequency domain. *Journal of biomechanics*, 39(14):2647–2656, 2006.

[20] S. Preece, J. Goulermas, L. Kenney, and D. Howard. A comparison of feature extraction methods for the classification of dynamic activities from accelerometer data. *Biomedical Engineering, IEEE Transactions on*, 56(3):871–879, 2009.

[21] N. Saito and R. Coifman. Local discriminant bases and their applications. *Journal of Mathematical Imaging and Vision*, 5(4):337–358, 1995.

[22] S. Sasaki, T. Seki, and S. Sugiyama. Batteryless accelerometer using power feeding system of rfid. In *SICE-ICASE, 2006. International Joint Conference*, pages 3567–3570. IEEE, 2006.

[23] M. Sekine, T. Tamura, M. Akay, T. Fujimoto, T. Togawa, and Y. Fukui. Discrimination of walking patterns using wavelet-based fractal analysis. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 10(3):188–196, 2002.

[24] M. Sekine, T. Tamura, T. Togawa, and Y. Fukui. Classification of waist-acceleration signals in a continuous walking record. *Medical engineering & physics*, 22(4):285–291, 2000.

[25] R. Yazicioglu, T. Torfs, J. Penders, I. Romero, H. Kim, P. Merken, B. Gyselinckx, H. Yoo, and C. Van Hoof. Ultra-low-power wearable biopotential sensor nodes. In *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*, pages 3205–3208. IEEE, 2009.

[26] B. Zhai, L. Nazhandali, J. Olson, A. Reeves, M. Minuth, R. Helfand, S. Pant, D. Blaauw, and T. Austin. A 2.60 pj/inst subthreshold sensor processor for optimal energy efficiency. In *VLSI Circuits, 2006. Digest of Technical Papers. 2006 Symposium on*, pages 154–155. IEEE, 2006.