An Energy-Efficient Computational Model for Uncertainty Management in Dynamically Changing Networked Wearables

Ramyar Saeedi, Ramin Fallahzadeh, Parastoo Alinia, Hassan Ghasemzadeh
School of Electrical Engineering and Computer Sciecne
Washington State University, Pullman, WA 99164–2752
{rsaeedi, rfallahz, palinia, hassan}@eecs.wsu.edu

ABSTRACT

The utility of wearables is currently limited to lab experiments and controlled environments mainly because computational algorithms embedded in wearables fail to produce accurate measurements in uncontrolled, dynamically changing, and potentially harsh environments. With the exponentially growing adoption of these systems in human-centered Internet-of-Things (IoT) applications, development of resource-efficient solutions to enhance the accuracy of this systems remains a considerable research challenge. In this paper, we introduce an energy-efficient framework for uncertainty management of networked wearables. The core components of our framework are anomaly screening units for detecting anomalies that require handling, thus resulting in one order of magnitude less energy consumption compared to the conventional frameworks. Furthermore, our screening approach achieves 98.3% accuracy in detecting anomalies based on real data collected with wearable motion sensors.

CCS Concepts

•General and reference → Reliability; Performance; •Human-centered computing → Ubiquitous and mobile computing systems and tools; •Computing methodologies → Feature selection; •Computer systems organization → Embedded software; •Hardware → Emerging architectures;

Keywords

Networked wearables, accelerometer, activity monitoring, power optimization, reliability, uncertainty management, feature selection

1. INTRODUCTION

Networked wearables have gained tremendous attention recently due to their potential to enable emerging applications in wellness, health-care, fitness as one of the main focus in Internet-of-Thing (IoT) domains [1–3]. The utilization of these systems, however, is currently limited to controlled environments, lab settings, and small field trials. This limitation has created major obstacles in scaling these systems up and advancing their utility in real-world settings.

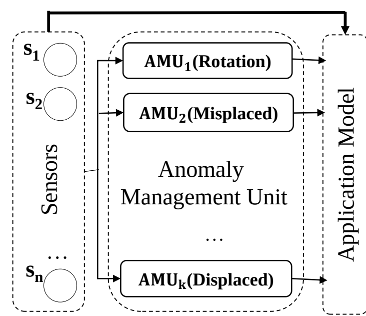


Figure 1: Current uncertainty management approaches in networked wearables. This approach combines detection and mitigation of anomalies into one module called anomaly management unit (AMU).

Wearables collect sensors data and monitor/predict events of interest (e.g., daily activities), which often requires development of machine learning algorithms for real-time execution. Examples of such algorithms are classification, template matching, and clustering. When used in real world scenarios, wearables face different types of anomaly, such as mis-orientation, displacement, and misplacement of sensors, which dramatically impact performance. For example, one can train a classifier, based on a training dataset, to perform activity recognition. The classification algorithm, however, fails to detect physical activities, if the location of the on-body sensors changes [4]. Therefore, it is crucial to develop uncertainty management frameworks that detect and mitigate the effect of such anomalies on the performance of the machine learning algorithms.

Current approaches for uncertainty management events are significantly energy-inefficient: (1) these techniques aim to mitigate the effects of a particular anomaly (e.g., mis-orientation) on a specific application of wearables (e.g., activity recognition) [1, 5, 6]. As a result, as shown in Figure 1, these techniques require separate unit for each anomaly; (2) although anomalies are sparse, occurring at most 10% of the times [7], current algorithms require constant execution of several anomaly management units (AMUs), an approach that demands extremely high computational power, given the stringent constraint resources of wearables.

The aforementioned drawbacks with the current uncertainty management approaches warrant development of a new class of models that not only are accurate in detecting anomalies and mitigating the effect of such events on the performance but also are energy-efficient. To the best of our knowledge, development of energy-

ISLPED '16 August 08–10, 2016, San Francisco Airport, CA, USA

DOI: http://dx.doi.org/10.1145/2934583.2934617

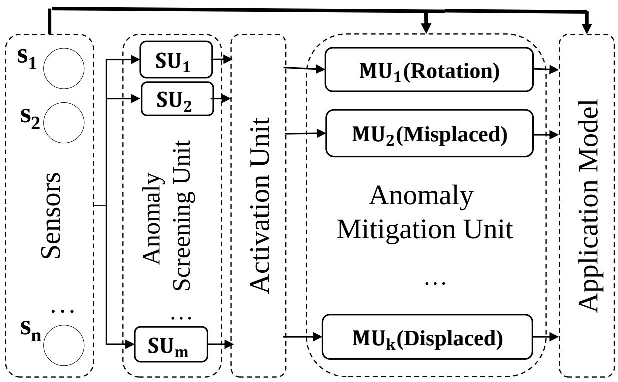


Figure 2: Energy-efficient uncertainty management framework with the anomaly screening unit (ASU).

efficient uncertainty management frameworks has remained a virtually unexplored area to date.

Our contribution in this paper can be summarized as follows: (1) we present a novel energy-efficient uncertainty management framework that uses computationally simple screening units for detection of sparse anomalies; (2) a novel approach is introduced to capture the collective contribution of various signal attributes (i.e., features) in predicting existence of an anomaly in the network as well as the costs associated with the features; this method is used to develop simple yet effective *Anomaly Screening Module* (ASU); (3) we introduce a prediction graph model to capture inter- and intra-sensor energy consumption for anomaly detection; (4) we pose an optimization problem, based on our graph model, to find minimum-cost feature sets capable of predicting anomalies within the network; (5) we validate the efficiency of our approach using real data collected with motion sensors for physical activity monitoring.

2. SYSTEM OVERVIEW

A unique attribute of our uncertainty management framework, as shown in Figure 2, is an *Anomaly Screening Unit* (ASU) which is used to detect different types of anomaly. ASU is composed of several screening units (SUs) which each one is used to detect one or several types of anomalies (e.g. misplacement, displacement).

Followed by the ASU is a set of mitigation units (MUs) which each one is responsible to address a specific types of anomaly. As soon as an anomaly is detected, the screening unit activates the corresponding MU(s) through activation unit. Because of sparsity of anomalies, decoupling anomaly screening and anomaly mitigation results in energy saving given that screening units are designed to be computationally inexpensive and communicate infrequently.

Furthermore, in our framework, management is independent of the application computational model (e.g., activity recognition, energy expenditure estimation, etc). In other words, uncertainty management is performed on the sensor data without considering the underlying application. The main benefit is that the ASU is transferable to different types of application with the same set of sensors.

ASU is the core component of our framework in delivering highly energy-efficient management of anomalies. The goal is to design SUs such that they are energy-efficient capable of remaining active all the time, but also accurately activate the MUs upon occurrence of an anomaly. Our approach to development of each screening unit (SU) is based on the concept of information redundancy. It involves selecting an optimal set of features (in a training phase) and using the selected features during real-time execution of the system.

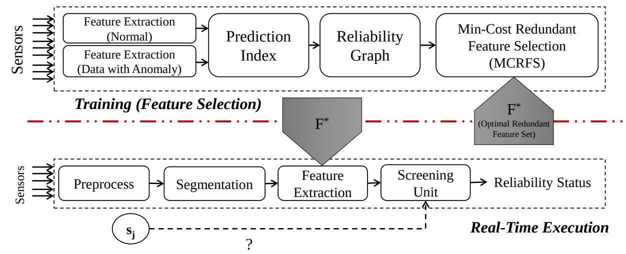


Figure 3: The process of developing the anomaly screening module.

A high-level illustration of the two phases is shown in Figure 3. In real-time execution, a sensor node within the network acquires a set of redundant features for screening. These features may be retrieved from another node, upon request, or extracted locally. Clearly, it is more desirable to use local features to avoid data transmission between sensor nodes. The set of minimum-cost redundant features are used to construct a computationally simple classifier to determine anomaly occurrence. In Section 3, the problem formulation for building an individual screening unit in training phase is provided.

3. ANOMALY SCREENING UNIT

In this section, we present our cost-efficient feature selection approach to construct the computational model of SUs to detect anomaly. This approach identifies a set of redundant features with minimum computation and communication cost. We first introduce a new metric, called *prediction index*, which uses the inter-feature correlation and estimates the potential power of two features for predicting an anomaly. For this part, it is needed to have training controlled data and data in presence of anomaly as shown in Figure 3. The prediction index is then used to construct a graph model, called *Prediction Graph*, which is further employed to find a set of feature pairs in constructing SUs. For each anomaly type, we construct a specific prediction graph. Finally, a set of energy efficient features are selected from each prediction graph to build individual screening units which together construct the anomaly screening unit (ASU).

3.1 Problem Statement

Let $S = \{s_1, s_2, \ldots, s_n\}$ be a set of 'n' sensor nodes forming a networked wearable system. Furthermore, let $F_i = \{f_{i1}, f_{i2}, \ldots, f_{im}\}$ be a set of 'm' features extracted from sensor node s_i.

DEFINITION 1 (PREDICTION INDEX). *Given two features f_{ij} and f_{kl}, prediction index is defined as*

$$\lambda_{ij}^{kl} = \frac{U_R(f_{ij}, f_{kl})}{U_A(f_{ij}, f_{kl})} \quad (1)$$

where U_R denotes correlation between the two features during normal operation of the network while U_A represents inter-feature correlation associated with presence of an anomaly. In this paper, we use symmetric uncertainty for correlation measurement. Symmetric uncertainty captures non-linear correlation between variables and therefore, is a safe measure for feature analysis studies [8, 9].

DEFINITION 2 (SYMMETRIC UNCERTAINTY). *The symmetric uncertainty between two discrete random variables X and Y is given by:*

$$U(X, Y) = \frac{2 \times I(X, Y)}{H(X) + H(Y)} \quad (2)$$

where $H(X)$ and $H(Y)$ represent the entropy of random variables X and Y, respectively, and $I(X,Y)$ denotes the information gain between the two variables. $I(X,Y)$ is defined by:

$$I(X,Y) = H(X) - H(X|Y) \tag{3}$$

Symmetric uncertainty is the normalized information gain and is always between 0 and 1, where U=1 means that knowing the value of either variable can completely predict the other variable, and U=0 indicates that the two variables are completely independent.

DEFINITION 3 (STRONG PAIR FEATURES). *Two features f_{ij} and f_{kl} are considered as a strong pair, if the prediction index associated with them exceeds a given threshold, Δ:*

$$\lambda_{ij}^{kl} \geq \Delta \tag{4}$$

DEFINITION 4 (PREDICTION GRAPH). *The prediction graph G = (V,E,C_V,W_E) is an undirected graph with the following specifications. The variable V denotes the set of $m \times n$ vertices associated with the features extracted from the sensor nodes in S; E represents the set of edges such that an edge e_{ij}^{kl} exists if features f_{ij} and f_{kl} form a strong pair as defined previously. The set C_V represents vertex weights. A weight c_{ij} is assigned to each vertex v_{ij} to represents the cost of feature f_{ij} computation. Furthermore, W_E represents edge weights. Each edge e_{ij}^{kl} has a cost w_{ij}^{kl} given by*

$$w_{ij}^{kl} = \begin{cases} \beta, & \text{if } i \neq k \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

where β is the cost of data transmission between two different sensor nodes s_i and s_k.

PROBLEM 1 (MIN-COST REDUNDANT FEATURE SELECTION). *Min-Cost Redundant Feature Selection (MCRFS) is to select a subset of features such that the total cost in terms of computation and communication is minimized, while we have at least one feature from each sensor node.*

Given that wearables are used in a highly dynamic environments, the measurements are likely to be noisy, which may reduce the effectiveness of feature pairs in predicting anomalies. In order to further enhance robustness of the system, our proposed framework provides flexibility in reliability level of feature selection approach, which select different set of features. Motivated by K-connectivity in wireless sensor networks, we define K-reliable feature selection as follows [10]:

PROBLEM 2 (K-RELIABLE FEATURE SELECTION). *Given an exhaustive set of features, K-reliable feature selection is defined as finding K set of features which meet the conditions of our framework. In each iteration, the selected edges are removed from prediction graph for the next round.*

3.2 Problem Formulation

The MCRFS problem can be formulated as follows. Assume a_{ij}^{kl} is a binary variable indicating whether or not the edge e_{ij}^{kl} exists in the prediction graph G. That is:

$$a_{ij}^{kl} = \begin{cases} 1, & \text{if } e_{ij}^{kl} \in \text{E} \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

The goal is to choose a subset of the edges such that all sensor nodes have at least one feature candidate in the final feature set. By choosing an edge from the prediction graph, we can perform a cross-feature analysis to determine whether or not the system has encounter an uncertainty event. Let x_{ij}^{kl} be a binary variable indicating whether or not the edge e_{ij}^{kl} is selected by our feature selection algorithm. Thus,

$$x_{ij}^{kl} = \begin{cases} 1, & \text{if } e_{ij}^{kl} \text{ is selected} \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

Thus, the optimization problem can be written as follows.

$$Z = \sum_{i,k=1}^{n} \sum_{j,l=1}^{m} a_{ij}^{kl}(c_{ij} + w_{ij}^{kl} + c_{kl})x_{ij}^{kl} \tag{8}$$

$$\text{Minimize} \quad Z \tag{9}$$

$$\text{Subject to:} \quad \sum_{k=1}^{n} \sum_{j,l=1}^{m} x_{ij}^{kl} \geq 1, \quad \forall i \in \{1, \ldots, n\} \tag{10}$$

$$x_{ij}^{kl} \in \{0,1\} \quad , \quad \forall i,k,j,l \tag{11}$$

THEOREM 1. *cc MCRFS problem is NP-hard,and can not be approximated for any time computable function $\alpha(t)$.*

Proof A reduction from the Weighted Set Cover (WSC) problem to MCRFS is used to proof NP-hardness. consider the universe U = {$s_1, s_2, \ldots, s_n$} and a collection of sets S = {(e_1, w_1), (e_2, w_2), ..., (e_k, w_k)} along with their weights. If we repeatedly add several copies of each set to the collection of sets, the WSC problem will not change in terms of complexity. Also, this new problem is correspond with the MCFRS, if we consider each edge as a set. Each set contains those nodes in the network which are connected by the corresponding edge. Several replica for each set is because of having super-nodes(network nodes), which have multiple connections between them. As this problem is a reduction of WSC problem, the MCFRS problem is an NP-Hard problem.

In the next section, we present a greedy approximation algorithm for solving the problem in polynomial time.

3.3 Greedy Algorithm

Algorithm 1 shows the proposed greedy algorithm. The algorithm keeps track of selected features in F. The function $h(F)$ outputs the sensor nodes that are associated with a given feature set F. During each iteration, the algorithm chooses an edge with minimum cost per each new covered sensor node by that edge. The set of newly covered sensor nodes by edge e_{ij}^{kl} is computed by $h(F \cup \{s_i, s_k\}) - h(F)$. The algorithm iterates until all sensor nodes are covered (i.e., $h(F) = h(V)$). For the case of K-Reliability, the algorithm will repeat the procedure on the graph without the edges that has been selected in previous rounds.

THEOREM 2. *The greedy algorithm returns a feature set of at most 1.5 times the minimum cost of any feature set covering all the sensor nodes.*

Proof When the greedy algorithm chooses an edge e_{ij}^{kl}, imagine that it charges the price per sensor node for that iteration to each sensor node newly covered by e_{ij}^{kl}. Then the total cost of the edges chosen by the algorithm equals the total amount charged, and each element is charged only once. Since features are chosen in pairs, consider any pair of features f_{ij} and f_{kl} in the optimal feature set, $F^\star$ associated with edge e_{ij}^{kl}.

During the iteration that e_{ij}^{kl} is chosen, one (i.e., either s_i or s_j) or two (both s_i and s_j) are uncovered yet. If only one sensor node is

Algorithm 1 Greedy solution for MCRFS

Require: Reliability graph G=(V,E,W_V,W_E), Degree of reliability (K)
Ensure: Set of minimum cost redundant features, F
  Initialize $F \leftarrow \emptyset$.
  for all $r \in 1...K$ **do**
    Initialize $F_r \leftarrow \emptyset$, $E_r \leftarrow \emptyset$. Define $h(F_r) = \{s_i | f_{ij} \in F_r\}$
    repeat
      $e_{ij}^{kl} = \arg\min \frac{(c_{ij}+w_{ij}^{kl}+c_{kl})}{|h(F_r \cup \{s_i,s_k\})-h(F_r)|}$
      $F_r \leftarrow F_r \cup \{f_{ij}, f_{kl}\}$
      $E_r \leftarrow E_r \cup \{e_{ij}^{kl}\}$
      $c_{ij} = 0; c_{kl} = 0$
    until $h(F_r) = h(V)$
    $F = F_r \cup F$
    $E = E - E_r$
  end for
  $return\ F$

uncovered, then the greedy algorithm would pay a cost per sensor node of at most W_{ij}^{kl} charged to the sensor node being covered during this iteration ($W_{ij}^{kl} = c_{ij} + w_{ij}^{kl} + c_{kl}$). If, however, two sensor nodes are uncovered at the beginning of the iteration, then the cost per sensor node would be it would pay a cost per sensor node of at most $W_{ij}^{kl}/2$. Summing over all sensor nodes associated with e_{ij}^{kl}, the total amount charged is at most W_{ij}^{kl} + $W_{ij}^{kl}/2$ = $1.5 \times W_{ij}^{kl}$. Summing over all edges in $F^\star$ and noting that every sensor node is covered by some edge in $F^\star$, the total amount charged to sensor nodes overall is given by

$$\sum_{W_{ij}^{kl} \in F^\star} 1.5 \times W_{ij}^{kl} = 1.5 \times OPT \tag{12}$$

4. VALIDATION

4.1 Energy Model

We define energy models for both conventional and our framework as shown in Figure 1 and Figure 2 respectively. The energy consumption of a wearable sensor node consists of three components: (1) data sampling; (2) feature extraction; and (3) machine learning algorithms. According to Figure 1, the total energy consumption can be calculated by

$$E(conv) = E(ml) + E(s) + \sum_{k=1}^{N} E(AMU_k) \tag{13}$$

where $E(conv)$ represents energy consumption of the conventional models, $E(ml)$ represents energy consumption of the application algorithm (e.g., activity recognition), $E(s)$ denotes energy consumption due to data sampling, and $E(AMU_k)$ is energy consumption of the k-th anomaly management unit.

The total energy consumption of our uncertainty management framework, shown in Figure 2, can be written as:

$$E(scr) = E(ml) + E(s) + \sum_{k=1}^{N} \alpha_k \times E(MU_k) + \sum_{i=1}^{M} E(SU_i) \tag{14}$$

where $E(scr)$ denotes energy consumption of the proposed model based on anomaly screening, $E(SU_i)$ represents energy consumption of the i-th screening unit, and α_k indicates the activation frequency for SU_k and is given by

$$\alpha_k = (F_k \times tp_k + (1 - F_k) \times fp_k) \tag{15}$$

where tp_k and fp_k denote true positive rates and false positive rates of the SU_k classifier respectively, and F_k represents the frequency of occurrence of the k-th anomaly type. These rates indicates how often a mitigation unit (MU_k) need to be activated by corresponding SU_k. We note that, in this formulation, the energy consumption due to feature extraction is considered as a part of the processing energy consumption for AMU, SU and MU units.

Table 1: Energy consumption of computing individual features.

Feature	Description	Energy (nJ)
Amp	Amplitude of signal segment	16386
Med	Median of signal segment	405159
Mean	Mean value of signal segment	8126
Max	Maximum amp of signal segment	8103
Min	Maximum amp of signal segment	8129
P2P	Peak to peak amplitude	16291
Var	Variance of signal segment	38846
Std	Standard deviation	40431
RMS	Root mean square power	29705
S2E	Start to end value	83
Morph	Morphological Samples	41

4.2 Experimental Setup

To demonstrate the effectiveness of the proposed framework, we constructed a network of three wearables placed on 'left wrist', 'right ankle', and 'waist'. We used TelosB motes to measure acceleration and angular velocity of different body segments using a triaxial accelerometer and a biaxial gyroscope. Three subjects were asked to perform 30 different daily activities such as 'sit to stand', 'sit to lie', 'bend and grasp', 'kneeling', 'step forward', 'step backward', etc. Each transitional movement was repeated 10 times by each subject. The sensor nodes were programmed to sample data at 50Hz and transmit the data wirelessly to a laptop computer. In addition to these locations, movement data were collected for several other network configurations. These configurations resemble anomaly (i.e. misplacement) that can happen in real scenarios. we here includes sensor misplacement and displacement for our case study.

We extracted 10 statistical features as well as 10 morphological features from each activity trial. The features are shown in Table 1 where the energy consumption of computing each feature is also reported. We calculated instruction level energy consumption of MSP430 micro-controller, which is available on the TelosB motes used in our experiments. The details of energy calculations is based on an instruction level energy model provided in [9, 11]. Also, information on power consumption and specifications of the sensor nodes can be found in [12]. Morphological features were equally-spaced samples of the activity signal. Overall, we extracted 300 features from three sensors.

For allocating the communication costs in the prediction graph, we assume that the communication cost is zero for edges that connect feature of the same sensor node which runs the uncertainty management. The communication costs of inter-node data transmission was set to be a fixed number computed based on the energy cost of transmitting on data packet using ZigBee protocol. That is, the communication power is based on ZigBee protocol specifications [13].

Table 2: On-body locations used for data collection in normal, and in presence of anomaly, and corresponding accuracy drops.

ID	Uncertainty	Type	Drop in Recall	Drop in Precision
1	Right Ankle $\rightarrow$ Left Forearm	Misplacement	36.1%	33.2%
2	Left Wrist $\rightarrow$ Right Knee	Misplacement	33.4%	28.2%
3	Waist $\rightarrow$ Neck	Misplacement	25.1%	28.1%
4	Right Ankle $\rightarrow$ Right Shin	Displacement	32.1%	34.5%
5	Left Wrist $\rightarrow$ Right Forearm	Displacement	14.2%	16.4%
6	Waist $\rightarrow$ Left Hip	Displacement	15.8%	16.3%

Table 3: Power saving of feature selection compare to the baseline for various reliability levels ($K \in \{1, 2, 3, 4\}$).

Power Saving/Reliability Level	K=1	K=2	K=3	K=4
Computation Power (Misplacement)	97.1%	95.7%	90.6%	55.1%
Computation Power (Displacement)	97.1%	96.9%	90.2%	84.5%
Communication Power	99.0%	97.5%	97.0%	96.0%

4.3 Results

In this section, we present our results on (1) the impact of uncertainty events on accuracy performance of activity recognition; (2) performance of individual SUs; and (3) energy efficiency of the proposed uncertainty management framework.

4.3.1 Impact of Anomaly on Accuracy

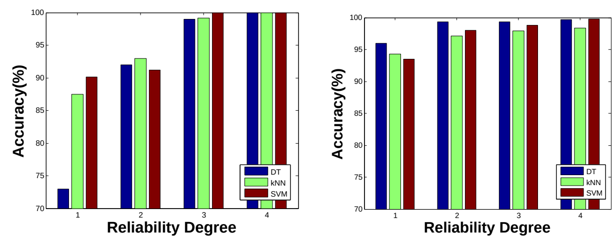
To show the impact of anomaly on the performance of networked wearables, we used the collected dataset for activity recognition where we develop a machine learning classifier to detect physical activities of daily life. Given that k-Nearest Neighbor (kNN) classification has shown promising results in activity recognition [4], we chose kNN for activity classification. First, we developed a machine learning classifier using the perfect data, and then tested this model using ten-fold cross validation, which provided us with an average accuracy of 97.8% for detecting the 30 activities. later, we test the activity recognition model on data with different types of anomalies as listed in Table 2. In each case, the assumption is that there is just one anomaly in the network. Table 2 shows the amount of accuracy drop due to these anomaly occurrences, namely sensor misplacement and sensor displacement. As shown in Table 2, the accuracy of activity recognition drops by 22.6% on average.

4.3.2 Performance of Anomaly Screening Unit

To assess the performance of the anomaly screening module, we constructed two prediction graph with 300 vertices associated with the extracted features for two anomaly type. The inter-feature correlations is needed for both perfect data and data in presence of anomaly (i.e. one sensor is misplaced) to build each prediction graph.

The number of edges in the prediction graphs is directly dependent on the value of Δ. Clearly, the number of edges decreases as the threshold increases. We experimentally determined a range of $3.5 \leq \Delta \leq 3.9$ to be an acceptable range. If we choose Δ outside of these range the graph would be either sparse, or contain many irrelevant features. As the level of reliability increased from K=1 to K=4, the number of selected features increased from 3 to 9.

To compute the accuracy performance of the screening units that reside in our anomaly screening module, we considered single-anomaly that are listed in Table 2. We used three classification algorithms including Decision Tree (DT), (kNN), and support vector machine (SVM) for this analysis. The accuracy of screening units ranged from 73.7% to 100% using different classifiers and different levels of reliability as shown in Figure 4. The average accuracy for misplacement, and displacement cases are 91.1%, and 98.5% using a decision tree classifier.



(a) Misplacement (b) Displacement

Figure 4: Accuracy of SUs using Decision Tree (DT), kNN, and SVM classifiers respectively; The results are shown for both misplacement (a) and displacement (b).

Table 3 shows the amount of energy savings obtained using our power-aware feature selection technique compare to the baseline case with no optimization. The energy savings ranged from 55.1% for the highest reliability level (K=4) to 97.1% for the lowest reliability level (K=1) in terms of feature computation power. The communication power saving is also based on the percentage of features that are needed to transmit using the proposed framework compared to the baseline scenario.

4.3.3 Energy Analysis of the Framework

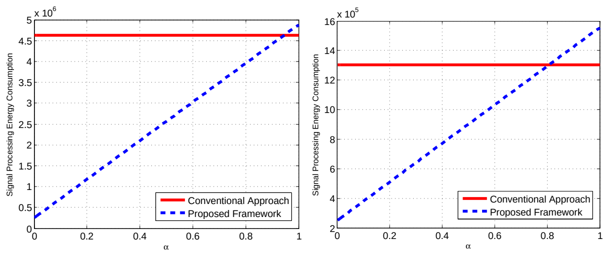
In this section, energy analysis of the propose framework compared to the conventional approach is presented. Given that we focused on two anomaly types, in this paper, our framework consists of two SUs and two MUs one for each anomaly type. Table 4 shows the energy consumption, and accuracy of anomaly detection and mitigation units that are used in our system. The MU units in Table 4 are node localization models which are used to mitigate the effects of misplacement, and displacement.

We measured the total computation energy of both approaches as a function of α, the activation rate of the MUs. In this comparison, we only considered energy consumption of signal processing and feature extraction as other energy components are used in both conventional model and proposed framework.

The energy consumption of the two models (i.e., conventional and proposed) is shown in Figure 5a and Figure 5b. With $\alpha = 0.1$, the amount of power savings of our framework is 88% and 69% for kNN and DT classifiers respectively. It is reasonable to assume $\alpha = 0.1$ as an uncertainty rate [7]. In short, our system outperforms the conventional model for $\alpha \leq 0.93$ using kNN, and $\alpha \leq 0.81$

Table 4: Power consumption and accuracy of different processing units.

Unit/Parameter	Precision (P)	Recall (R)	Power Consumption (μJ)	fp	tp
SU (DT)	99.1%	99.3%	250.045	0.01	0.99
MU1 (KNN)	98.1%	97.9%	4630.605	0.02	0.98
MU2 (DT)	86.6%	86.8%	1304.097	0.13	0.87



(a) kNN classifier (b) DT classifier

Figure 5: Energy consumption comparison of proposed framework and conventional approach using kNN (a) and DT (b) classifiers.

using DT, as shown in Figure 5b. We note that for extremely larger α (e.g., $\alpha > 0.93$ for kNN and $\alpha > 0.81$ for DT), the conventional approach outperforms ours. This situation, however, is unrealistic because anomalies are sparse in time that they occur at most 10% of the times.

5. CONCLUSION AND FUTURE WORK

We proposed an energy-efficient framework which is used to increase the robustness of wearable sensors. A feature selection algorithm was developed to exploit correlation variation of pair of features. The energy saving of framework is 88.2% and 69.7% compared to conventional methods considering anomaly occurrence of 10% for kNN-based, and DT-based MUs respectively. There is a trade off between power consumption and accuracy of MUs, the more accurate classifier, the lower amount of power saving. simple SU units detect anomaly types with 98.3% accuracy on average.

We focused on activity recognition as our pilot application and performed our analysis on motion sensors such as accelerometers and gyroscope sensors. Our ongoing research involves several expansions with regards to the framework validation. First, we are working on integrating additional sensor modalities such as audio signals and Electromyography (EMG) sensors into our wearable network. Second, we plan on creating a larger sensor network with an exhaustive set of sensors covering many body joints involved in building human skeleton model.

In this paper, we only considered single-fault model where we assumed that each sensor will experience at most one anomaly at a time. Given that anomalies are sparse, this assumption is intuitively reasonable. Yet, the simultaneous occurrence of multiple anomalies is theoretically possible(i.e. multiple sensors, multiple anomaly). Thus, we plan to enhance the system which can deal with multiple-anomaly scenarios as part of our future work.

6. REFERENCES

[1] H. Sagha, A. Calatroni, J. del R Millan, D. Roggen, G. Troster, and R. Chavarriaga, "Robust activity recognition combining anomaly detection and classifier retraining," in *2013 IEEE International Conference on Body Sensor Networks (BSN)*. IEEE, 2013, pp. 1–6.

[2] R. Fallahzadeh, M. Pedram, and H. Ghasemzadeh, "Smartsock: A wearable platform for context-aware assessment of ankle edema," in *the 38th Annual International Conference of Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2016.

[3] R. Fallahzadeh, M. Pedram, R. Saeedi, B. Sadeghi, M. Ong, and H. Ghasemzadeh, "Smart-cuff: A wearable bio-sensing platform with activity-sensitive information quality assessment for monitoring ankle edema," in *Pervasive Computing and Communication Workshops, 2015 IEEE International Conference on*. IEEE, 2015, pp. 57–62.

[4] R. Saeedi, N. Amini, and H. Ghasemzadeh, "Patient-centric on-body sensor localization in smart health systems," in *The Asilomar Conference on Signals, Systems, and Computers*. IEEE, 2014.

[5] Y. E. Ustev, O. Durmaz Incel, and C. Ersoy, "User, device and orientation independent human activity recognition on mobile phones: challenges and a proposal," in *Proceedings of the 2013 ACM conference on Pervasive and ubiquitous computing adjunct publication*. ACM, 2013, pp. 1427–1436.

[6] P. Alinia, R. Saeedi, B. Mortazavi, A. Rokni, and H. Ghasemzadeh, "Impact of sensor misplacement on estimating metabolic equivalent of task with wearables," in *IEEE 12th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*. IEEE, 2015, pp. 1–6.

[7] L. Eadicicco, "American Check Phones 8 Billion Times a Day," time.com/4147614/smartphone-usage-us-2015/, [Online; accessed 10-May-2016].

[8] D. J. MacKay, *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

[9] H. Ghasemzadeh, N. Amini, R. Saeedi, and M. Sarrafzadeh, "Power-aware computing in wearable sensor networks: An optimal feature selection," *Mobile Computing, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2014.

[10] A. Ghosh and S. K. Das, "Coverage and connectivity issues in wireless sensor networks: A survey," *Pervasive and Mobile Computing*, vol. 4, no. 3, pp. 303–334, 2008.

[11] R. Saeedi, B. Schimert, and H. Ghasemzadeh, "Cost-sensitive feature selection for on-body sensor localization," in *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*. ACM, 2014, pp. 833–842.

[12] J. Polastre, R. Szewczyk, and D. Culler, "Telos: enabling ultra-low power wireless research," in *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*. IEEE, 2005, pp. 364–369.

[13] J. Baviskar, A. Mulla, M. Upadhye, J. Desai, and A. Bhovad, "Performance analysis of zigbee based real time home automation system," in *Communication, Information & Computing Technology (ICCICT), 2015 International Conference on*. IEEE, 2015, pp. 1–6.