DOI: 10.2298/CSIS110310053L

On the Efficiency of Cluster-based Approaches for Motion Detection using Body Sensor Networks

Kun-chan Lan, Chien-Ming Chou, Tzu-nung Wang, and Mei-Wen Li

Computer Science and Information Engineering
National Cheng Kung University
No.1, University Road, Tainan City 701, Taiwan
klan@csie.ncku.edu.tw

Abstract. Body Sensor Networks (BSN) are an emerging application that places sensors on the human body. Given that a BSN is typically powered by a battery, one of the most critical challenges is how to prolong the lifetime of all sensor nodes. Recently, using clusters to reduce the energy consumption of BSN has shown promising results. One of the important parameters in these cluster-based algorithms is the selection of cluster heads (CHs). Most prior works selected CHs either probabilistically or based on nodes' residual energy. In this work, we first discuss the efficiency of cluster-based approaches for saving energy. We then propose a novel cluster head selection algorithm to maximize the lifetime of a BSN for motion detection. Our results show that we can achieve above 90% accuracy for the motion detection, while keeping energy consumption as low as possible.

Keywords: body sensor network, motion detection, energy conservation, KNN.

1. Introduction

In recent years Body Sensor Networks (BSNs) have attracted increasing attention for their use in remote healthcare such as remotely monitoring the elderly and children. A BSN is composed of a number of tiny wireless sensors placed on the user's body. Sensor nodes can transmit data to a remote server for online or offline analysis via various wireless technologies (e.g., IEEE 802.15.4 [1] or Bluetooth) so that doctors can monitor the patient's physiological states in real-time. BSNs have been widely used for various healthcare applications [2, 3, 4, 5], motion detection/analysis [6, 7, 8, 9], and gaming [10].

A critical issue when designing a BSN is the energy consumption, since wireless sensors typically employ small batteries for ease of deployment. A number of works have studied the issue of reducing the energy consumption of BSN. For example, Xiao *et al.* [3] proposed a method to reduce the transmission power at the physical layer without sacrificing the reliability for a single-hop body area network. Lamprinos *et al.* [11] and Omeni *et al.* [12] employed a TDMA-based approach to avoid collisions. In addition, with TDMA a node only needs to turn on its radio in its assigned time slot to save energy. Furthermore, Xia *et al.* [2] proposed a prediction-based data transmission scheme for static data (e.g., blood pressure and heartbeat). In their method, both the sensor node and the base station (BS) periodically and independently execute an identical predictor (i.e., dual prediction) and thus they will obtain the same estimate. If the error between real and predicted data is acceptable, the sensor node just updates the state of the system using the predicted data without actually transmitting data to the base station. Similarly, the BS also simply updates the system using the predicted data unless it receives the real data from the sensor node. However, this approach only considered an individual sensor node instead of the entire BSN.

Cluster-based algorithms have recently been proposed for sensor networks, and showed promising results. In these approaches, nodes are organized into clusters and one node is then chosen as the cluster head (CH). The task of the CH is to aggregate sensor data and then transmits this to BS while the other cluster nodes (CNs) communicate only with the CH. Given that it typically consumes more power to communicate with BS than with other cluster nodes, a CH will consume more energy than a CN does. Therefore, each sensor node will take turns to serve as the CH in order to evenly distribute the energy load. Abbasi and Younis [13] have surveyed clustering algorithms and classified them based on clustering attributes such as the cluster property, cluster head capability, and clustering process. Most prior approaches randomly selected their CHs. For instance, LEACH [14] and DMCLUSTER [15] choose CHs probabilistically or based on the nodes' remaining energy. However, these prior works cannot guarantee to uniformly distribute the energy load and minimize the total intra-cluster energy consumption at the same time.

The objective of this work is to prolong the lifetime of a body sensor network for motion detection by distributing the energy load evenly among sensors while, at the same time, minimizing the total intra-cluster energy consumption. In this paper we focus on an application of BSN: posture detection. We develop a novel CH selection algorithm to maximize the lifetime of the BSN. We utilize the k-nearest neighbors (KNN) method to detect the motion of the user. In this work, we classify nodes in a BSN into two types: fixed and moving. Fixed nodes are nodes whose positions remain relatively static (i.e. unchanged) when the user changes to a new posture. We analyze different parameters such as the ratio between the number of moving and fixed nodes and the effect of CH selection on the energy-saving. We propose an algorithm for selecting a CH that can minimize the intra-cluster energy consumption by considering the relationship between inter-node distance and allowable power levels. In addition, a threshold-based method is developed to avoid a certain node from depleting its energy quickly by being frequently selected as the CH. Our contributions are threefold: 1) We discuss and analyze the amount of energy that can be saved using a cluster-based approach. 2) We propose a novel cluster head selection algorithm by considering the required power level for sending a packet from the cluster nodes to the cluster head for a given posture. 3) We implement our cluster-head selection algorithm on a testbed for motion detection using KNN and show that we can achieve high accuracy (above 90%) while saving energy for a body sensor network.

The remainder of this paper is organized as follows. In Section II we review the related work. We briefly describe different motion detection methodologies in Section III. In section IV, we propose a methodology to minimize the transmission cost in a BSN while balancing each node's energy consumption. The results of the experiment are shown in section V and we conclude this paper in Section VI.

2. Related Work

In this section, we describe the related work. Our work builds on prior research on body sensor networks, motion detection, and energy conservation.

2.1. Body Sensor Network

Numerous studies have proposed the use of body sensor networks (BSN) for healthcare applications [3, 4, 6]. Earlier works also pointed out that QoS [5] and energy conservation [2, 16] are key research issues for the BSN, since the former could affect life-or-death matters while the latter decides the lifetime of the network especially for those sensors embedded in a patient's body. In addition, BSN can be also applied to sports applications using inertial sensors to monitor the trainee's posture during actions such as walking/running [7], golf swings [8], and hand swings [9]. However, most of these prior studies either can only identify basic postures (sitting, standing, walking, and running) or can only detect single motions. While using some techniques such as machine learning [9] to detect more complex motion is possible, this might increase the latency or computation cost of the BSN.

2.2. Motion/Posture Detection

Various kinds of sensors can be used to capture the motion of a user, and these are briefly described as follows: Acoustic trackers can use high-frequency sound to triangulate a source [17, 18, 19]. These systems rely on line-of-sight (LOS) between the source and the sensors, and may therefore suffer from interference when surrounded by hard walls or other acoustic signal/noise. Inertial systems employ devices such as accelerometers or gyroscopes to measure positions and angles. They are often used in conjunction with other systems to provide updates and improvements of measurements, since they only measure relative changes instead of absolute positions. Most prior works focused on analyzing the characteristics of inertial sensors to capture some basic postures or daily behaviors [20, 21, 22]. Image-based systems use cameras to capture the movements of a subject who is attached with retro-reflective markers. The number of cameras used depends on what type of motions are being captured [23, 24, 25]. However, these image-based approaches are limited by the location of cameras because of the LOS requirement [26, 27, 28]. Magnetic systems measure changes in the magnetic field to estimate the position and orientation of an object [29, 30, 31], but these can be affected by any metallic material nearby, and thus are easily influenced by electromagnetic interference. Some hybrid systems have been proposed [32, 33, 34] that combine two or more of the above techniques to improve accuracy. In our work, we use G-sensors to capture the motions of the users.

2.3. Energy Conservation

While most of the prior studies focus on energy conservation at the physical layer ([35], [36]), data link layer (PAMAS [37], EAR [38], DBTMA [39], S-MAC [40]), and network layer (LEACH [14], DMCLUSTER [15]), our work employs an energy-efficient cluster-based routing system to reduce energy consumption in data transmission. The basic idea of using cluster-based routing is to choose a CH that will aggregate data from other CNs and then communicate with the remote BS. Given that the CH is generally much closer to CNs than the BS, the transmission power required to send a packet from the CN to the CH will be lower than that needed to send one from the CN to the BS. Therefore, if CNs only communicate with the CH, CNs will consume less energy (as compared to then every CN sends its packets directly to the BS), and the overall energy consumption will be reduced to prolong the lifetime of the sensor nodes and the entire network. Note that, given that CHs will consume more energy than other sensor nodes, in such a cluster-based architecture these need to be periodically re-selected to ensure the energy load is evenly distributed. Otherwise, some nodes which are frequently chosen as the CHs might quickly run out of power quickly, thus rendering the BSN useless.

In LEACH [14], clusters of the sensor nodes are formed based on the received signal strength. Each node in a cluster randomly elects itself as the CH depending on a pre-defined probability, which is based on the desired percentage (P) of CHs required. In 1/P rounds, all nodes will become CHs once so that the energy load can be evenly distributed. However, selecting CHs based on probability cannot guarantee that the selection is always optimal since these CHs might not be chosen uniformly over time. PEGASIS [41] is a near-optimal chain-based protocol that presents an improvement of LEACH. In this protocol, each node only communicates with its adjacent node and takes turns becoming the cluster leader that receives at most two packets before transmitting the aggregated data to a BS. This approach distributes the energy load more evenly among sensor nodes, but also causes a long delay. A hierarchical PEGASIS [42] was thus proposed to overcome this problem. Both PEGASIS and hierarchical PEGASIS operate with the following assumptions, which might be difficult to realized in some cases. First, all sensor nodes are static (i.e. there is no mobility). Second, every node has global knowledge of the network. Bandyopadhyay and Coyle [43] analyzed the optimal parameters p (the optimal probability of becoming a CH) and k (the maximum number of hops allowed from a sensor to its CH) to minimize the energy consumption, and extended the cluster from one level to multiple hierarchies. They focused on the minimization of energy without considering evenly distributing the energy load among nodes. In addition, the computational cost of obtaining p and k might increase the end-to-end latency. HEED [44] is a distributed clustering protocol in which tentative CHs are periodically selected based on the residual energy of sensor nodes. Intra-cluster communication cost, cluster properties (e.g., cluster size) and cluster power levels are considered for selecting the CH. However, HEED cannot guarantee an optimal CH selection, because it relies on secondary parameters to resolve conflicts. Furthermore, selecting temporary CHs could increase the energy consumption.

Generally speaking, all the above techniques aim to save energy by either minimizing the intra-cluster communication or balancing the energy load among nodes, which is also the goal of our work.

3. Motion Detection

In this work, we employ a KNN algorithm [45] to detect the motions of the subject. In this section, we briefly describe the KNN process, which can be divided into the training and detection phases, as follows.

3.1. Training Phase

Each accelerometer sensor has triaxial sensing data, and each axis represents a dimension. We adopt the concept of a Bit-code and Distance based index (BD) [46] to implement the KNN algorithm. In the training phase, the training data of each motion is separately measured, collected and combined together (i.e., $d(d_1,d_2,\ldots,d_h)$, h = number of the axes (i.e. 3) × number of sensor nodes), in the multi-dimensional data space, as shown in Fig. 1. We then decide a reference point $O(o_1,o_2,\ldots,o_h)$ to split the data space into $2^{h'}$ partitions represented in bit code (h' is a user-defined positive integer and $h' < h$). Thus, the partition/bit code $S(s_1,s_2,\ldots,s_{h'})$ of each set of training data $d(d_1,d_2,\ldots,d_h)$ is defined as

$$s_i = \begin{cases} 1, & d_i > o_i, \\ 0, & \text{otherwise}, \end{cases} \quad 1 \le i \le h'. \tag{1}$$

For instance, if h'=2, the space will be divided into 2^2 partitions, encoded by ‘00’, ‘01’, ‘10’, and ‘11’ respectively, as depicted in Fig. 2(a) [46]. The purpose of using the reference point is to set a central point in the multi-dimensional space to evenly distribute the data density of each partition. For non-uniform data distributions, reference points can be acquired by using some existing techniques, such as K-means, BIRCH or CURE [47]. An efficient KNN search can thus be used by exploiting these partitions. In other words, the KNN algorithm only compares and computes the data in the intersected partitions rather than the entire space.

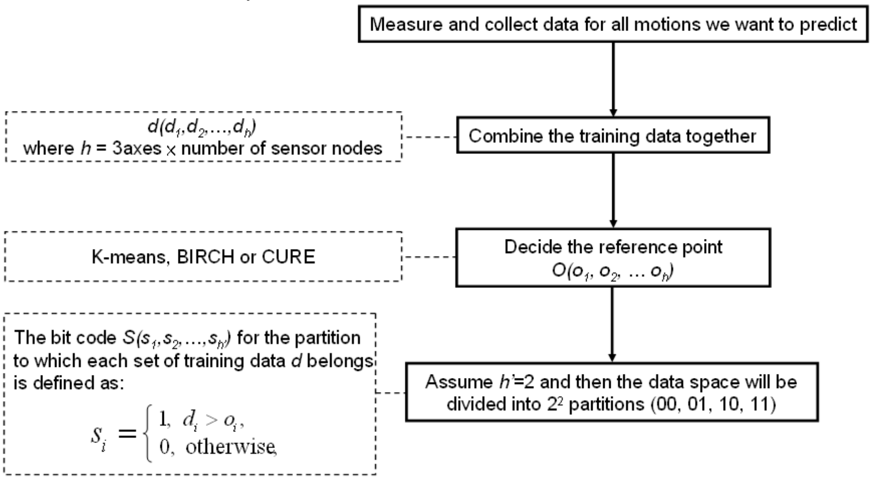


Fig. 1. Flow chart of the training phase

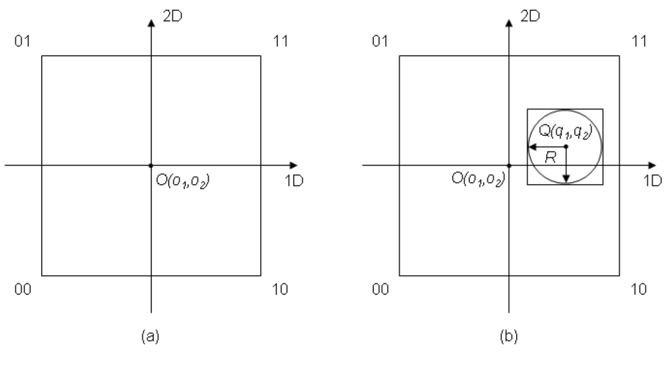


Fig. 2. (a) four-partition data space; (b) in this case, when |q1-o1|≧R, s1=1; when |q2-o2|<R, s2=0,1 so intersected partitions are (10) and (11)

3.2. KNN Detecting Phase

When the training phase is completed, motions can be detected in the KNN detecting phase, the flow chart of which is shown in Fig. 3. First of all, when the BS receives all sensing data as a query Q($q_1,q_2,\ldots,q_h$) via a BS, the bit code S of the query Q can be defined as in equation (1), but using q_i instead of d_i. Then, based on the information obtained above, the intersected partitions T($t_1,t_2,\ldots,t_h$), the number of intersected partitions (≧ 1) are computed by the equation (2) [46], so that only data in the intersected partitions needs to be compared/computed.

$$t_i = \begin{cases} s_i\ , & |q_i - o_i| \geq R, \\ 0 \text{ and } 1, & \text{otherwise,} \end{cases} \quad 1 \leq i \leq h', \tag{2}$$

where R is the radius for building the desired search field, as the example in Fig. 2(b).

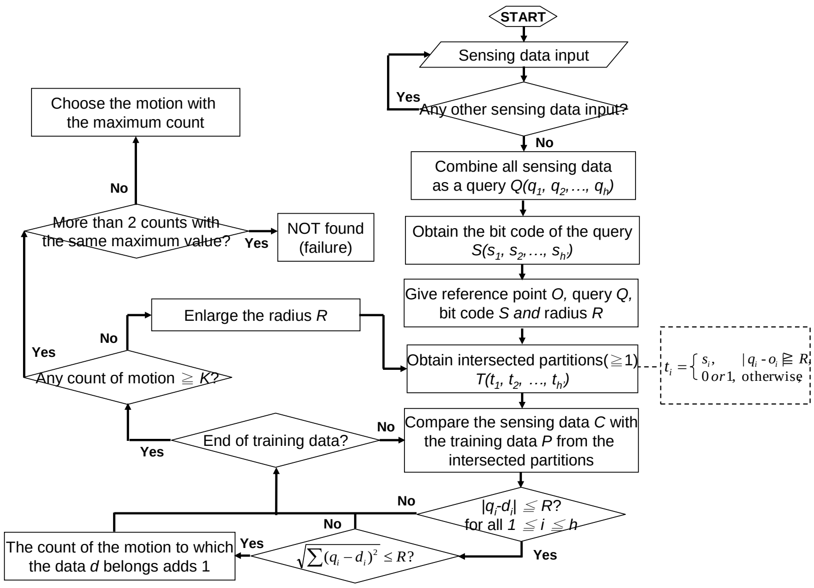


Fig. 3. Flow chart of the KNN detection phase

Next, in order to save computation time by first comparing rather than calculating the square and radical expressions for all data in the intersected partitions, the radius R becomes a pre-filter to filter out the data with any i, $|q_i - d_i| > R$ for $1 \leqq i \leqq h$, e.g., filtering out the data which is out of the smaller square in Fig. 2(b). Alternatively, one can calculate the Euclidean distance,

$$\sqrt{\sum_{i=1}^{n}(q_i - d_i)^2} \quad (3)$$

between the training data and the query to check if the sum is equal to or smaller than R, to ensure that the desired data is inside the circle. When that happens, the counter for the classified motion will be increased by one. This process will be repeated until all data in the intersected partitions are computed. KNN will then choose the motion that has the maximum count which is equal or larger than *k*. Here *k* is a user-defined threshold. However, if none of the motions has a count that is equal to or larger than *k*, then the radius R will be increased to enlarge the desired field for searching and the aforementioned process will be repeated. If more than two motions have the same maximum count or the count of every motion is smaller than *k* when R is already over the upper bound, then the detection is considered as a failure.

4. Energy Conservation

In this section, we first discuss the amount of energy that can be saved using a cluster-based approach. We then propose a novel cluster-head selection algorithm for energy-efficient data transmission. In the context of motion detection, nodes are placed at different places on the body (e.g. arms and legs). During any motion, some nodes might remain at the same positions between two consecutive postures, while others might change their locations. Here we consider two cases. Full-transmission (FT): all nodes periodically send their sensing data directly to the BS, for any motion. Partial-transmission (PT): Only nodes changing their locations will send their data to the CH. We use a threshold-based method to decide whether the nodes have moved from the last sampling time.

4.1. The Benefit of Using a Cluster-based Approach for Motion Capture

In this section, we provide an analysis of the amount of energy that can be saved using a cluster-based approach. We first define some parameters that will be used throughout our analysis. E_{Tx}(BS) is the energy consumption for sending data from a CN to a BS. E_{Tx}(CH) is the energy consumption for sending data from a CN to a CH. E_{Rx} is the energy consumption for receiving a packet. *n* is total number of sensor nodes and *m* is the number of static nodes. To simplify our analysis, here we assume that every CN has the same E_{Tx}(CH) and all nodes have the same E_{Tx}(BS) and E_{Rx}. One common practice to save energy is to compress the aggregated data before sending it to the BS [48, 49]. Here we assume that *r* is the compression ratio.

If all sensor nodes transmit their data directly to the BS without using clusters, the energy consumption E to detect one posture would be:

$$E1 = E_{Tx}(BS) \times n \times s \quad (4)$$

where *s* is the size of sensor data.

In a cluster-based approach, a CH receives and integrates CNs' sensing data and then transmits the aggregated data to a BS. Here we consider two situations when using clusters for the case of PT.

Case 1: CH is Moving at The Time of Sampling

$$E2 = (E_{Tx}(CH) + E_{Rx}) \times (n - 1 - m) \times s + E_{Tx}(BS) \times k \quad (5)$$

Case 2: CH is Static at The Time of Sampling

$$E2 = (E_{Tx}(CH) + E_{Rx}) \times (n - m) \times s + E_{Tx}(BS) \times k \quad (6)$$

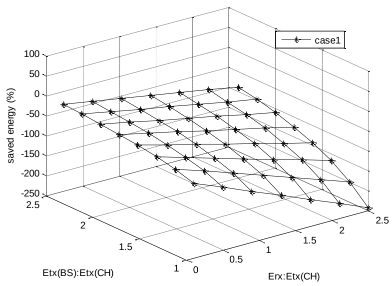
where *k* is the aggregated packet size. For example, in case 2, $k = (n - m) \times s \times r$.

Here (*n-m*) nodes have sensing data to be sent out. In case 1 the CH is one of the moving nodes, so it will receive packets from the other (*n-m-1*) moving nodes.

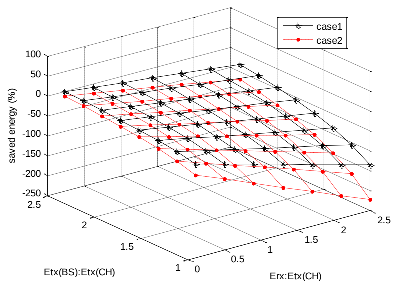
Different kinds of transceivers could have different energy consumption for the transmission and reception of packets [50]. Here we consider the case for different ratios of E_{Tx}(BS):E_{Tx}(CH) from 1.0 to 2.4 (the distance between BS and CH is always longer than that between CH and CNs) and different ratios of E_{Rx}:E_{Tx}(CH) from 0.1 to 2.5. To understand the effect of compression on the aggregated data, we also look at two different compression ratios, *r*=0.2 and 0.4.

In general, the combination of highest E_{Tx}(BS):E_{Tx}(CH) and lowest E_{Rx}:E_{Tx}(CH) can achieve the best performance in energy saving (i.e. E1 – E2), since it represents the situation when a cluster-based approach is the most efficient. This can be quickly observed from equations (4), (5) and (6).

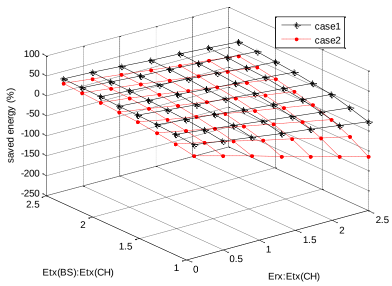
Fig. 4 and Fig. 5 show that the ratio of *n* and *m* (i.e. n/m) could strongly affect the amount of energy saved. In general, more energy can be saved when there are more static nodes (i.e. *m*) in the network. Using clusters does not guarantee that energy can always be saved, especially for the case of low E_{Tx}(BS):E_{Tx}(CH) and high E_{Rx}:E_{Tx}(CH). In addition, the compression ratio *r* could also affect the results. Based on the above insight, a network protocol designer might need to be careful when employing a cluster-based approach by taking the transceiver characteristics, the motion patterns and the network topology into consideration.



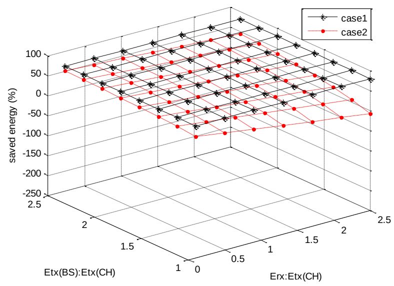
(a) r=0.2, n=4, m=0



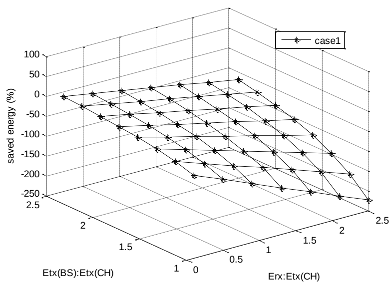
(b) r=0.2, n=4, m=1



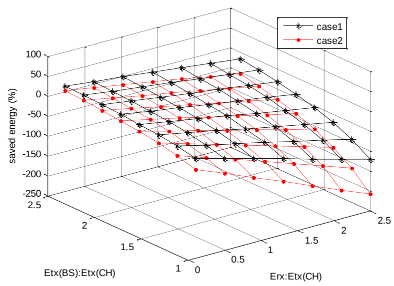
(c) r=0.2, n=4, m=2



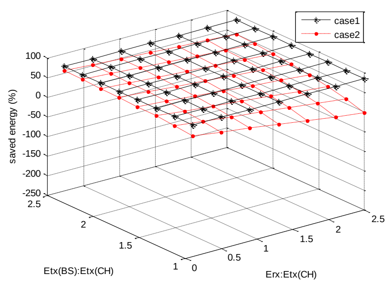
(d) r=0.2, n=4, m=3



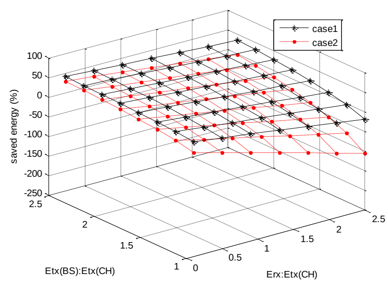
(e) r=0.4, n=4, m=0



(f) r=0.4, n=4, m=1

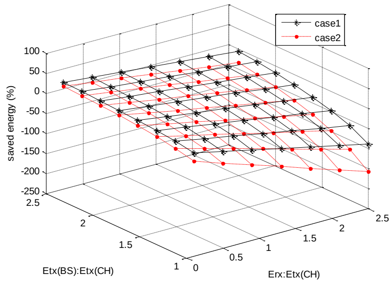


(g) r=0.4, n=4, m=2

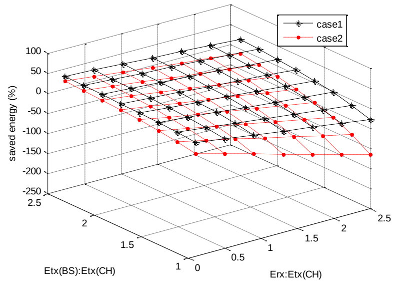


(h) r=0.4, n=4, m=3

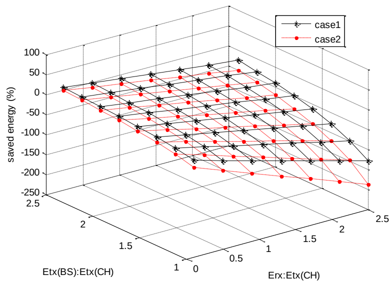
Fig. 4. The percentage of saved energy when n and r (4 graphs as a set) are fixed



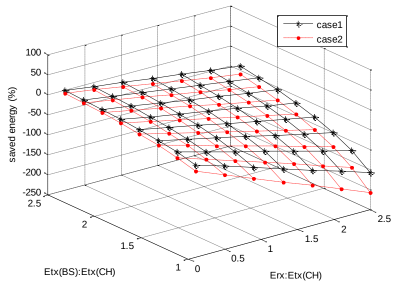
(a) r=0.2, n=4, m=2



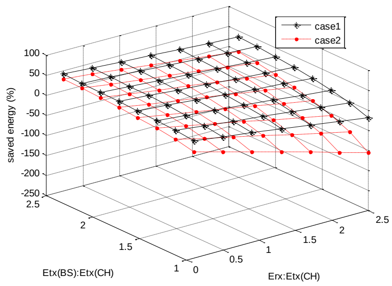
(b) r=0.2, n=5, m=2



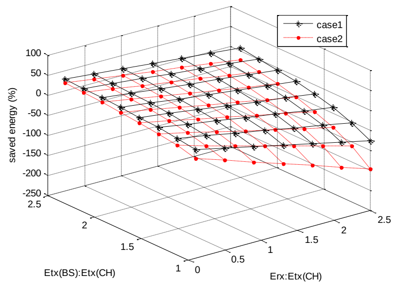
(c) r=0.2, n=6, m=2



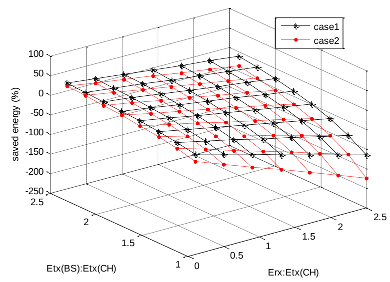
(d) r=0.2, n=7, m=2



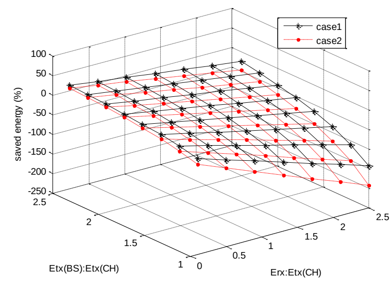
(e) r=0.4, n=4, m=2



(f) r=0.4, n=5, m=2



(g) r=0.4, n=6, m=2



(h) r=0.4, n=7, m=2

Fig. 5. The percentage of saved energy when m and r (4 graphs as a set) are fixed

4.2. Cluster Head (CH) Selection

In most previous studies, the CH is typically selected randomly or probabilistically. Some prior works chose the CH based on the nodes' residual energy. In this work, we develop a novel algorithm by minimizing the required energy for intra-cluster communication and preventing certain nodes from serving as the CH too often in order to prolong the lifetime of the entire network.

1) Minimization of Intra-cluster Communication Energy. Here we assume that all postures can be predefined. Therefore, the possible network topologies can be estimated in advance so that the maximum distance between any node in the BSN can be measured. Given that the required transmission power is a function of the distance, we can infer the minimum power level required if the maximum distance is known. Fig. 6 shows the relation between the power levels and node distances based on the measurement collected using Chipcon CC2420 RF transceivers [51]. As shown in Fig. 6, the transmission range using power level 3 is probably sufficient to cover the entire BSN including the BS. Therefore, we first measure all postures with different CHs to collect the total distances between all nodes and the CHs by

$$\sum_{i=1}^{n} Dis(N_i, CH) \quad, CH \in (N_1, N_2, ..., N_n) \tag{7}$$

where n is the total number of sensor nodes and $Dis(N_i, CH)$ is the distance between node i and CH. Based on the above insight, we select the CH by choosing the one which has the smallest distance from all CNs as the CH. Therefore, CNs can use the smallest allowable power level when transmitting their data to the CH. In other words, for each posture we find a CH that can minimize intra-cluster energy consumption by

$$\sum_{i=1}^{n} E_{Tx}(N_i, CH) \quad, CH \in (N_1, N_2, ..., N_n) \tag{8}$$

where n is the total number of sensor nodes and $E_{Tx}(N_i, CH)$ is the energy consumption of transmitting data from node i to a CH. We assign a unique priority number, ranging from 1 to n (where n is the number of nodes in the BSN), for each node. A smaller number means a higher priority. The node that consumes the least energy in equation (8) will have the highest priority (i.e. priority number=1). Therefore, by assigning the node with the highest priority to be the CH for every posture, we can minimize the intra-cluster energy consumption of the network.

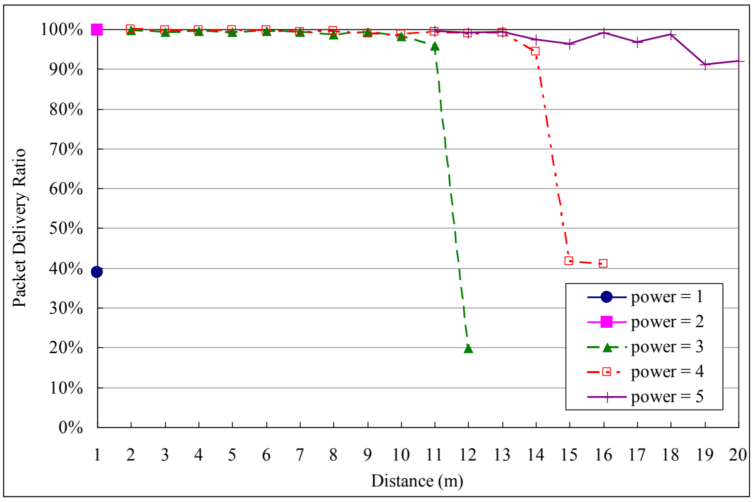


Fig. 6. Power level versus Distance. The range of power level is 0 to 31 for CC2420 RF transceiver

2) Energy Load Distribution. Assume x_i is the number of times node *i* serves as the CH and y_i is the number of times node *i* is a CN during the lifetime of a BSN. For a given set of motions we can estimate the maximum amount of energy a node might consume when it is a CH, denoted as *E*(CH). Similarly, we can also estimate *E*(CN), which is the maximum amount of energy a node might consume when a node is a CN. To simplify our analysis, here we assume that every node has the same *E*(CH) and the same *E*(CN). If the initial energy of each node is **I**, then the residual energy, *rE*(N_i), for node *i* after (x_i + y_i) runs can be computed by

$$rE(N_i) = I - E(CH) \times x_i - E(CN) \times y_i \tag{9}$$

The objective here is to maximize the sum of all x_i for $1 \leq i \leq n$ while keeping *rE*(N_i) greater than zero.

$$Maximize: \sum_{i=1}^{n} x_i \tag{10}$$

$$\begin{cases} rE(N_i) \ > \ 0 & ,1 \le i \le n \\ s.t. \ y_i \ge \sum_{j=1, j \ne i}^{n} x_j & ,1 \le i, j \le n \\ \quad 0 \le x_i, y_i & ,1 \le i \le n \end{cases} \tag{11}$$

Based on equation (11), the server can compute the threshold x_i that maximizes the objective function for each node *i*.

When selecting a CH, a node with the highest priority will be considered first. This node will then check if the number of times it has acted as the CH exceeds the assigned threshold. If not, it will be elected as the next CH. Otherwise, the node with the second highest priority will be considered. In addition, the server will recompute new thresholds based on each node's residual energy and broadcast them to each node. This process will be repeated until a node is finally selected as the CH.

In some situations, one node might accidentally be selected as the CH one more time than its assigned threshold, which leads to

$$y_i < \sum_{j=1, j \ne i}^{n} x_j \tag{12}$$

When this happens, all the other nodes need to readjust their thresholds. Specifically, all x_j (except $j = i$) are proportionally decreased by

$$new\ x_j = y_i \times \frac{old\ x_j}{\sum_{k=1, k \ne i}^{n} x_k} \quad ,1 \le j \le n \ and \ j \ne i. \tag{13}$$

An example of cluster head selection is shown in Fig. 7. There are four sensor nodes in this scenario. We assume that each round represents a different posture and CH is selected form the node with the highest priority. In this example, N1 is first selected to be the CH in round i and then N3 is selected as the CH in round i+1. In every round, when the server receives the aggregated data from the CH, it also records the number of times each node has acted as the CH. Once the number of times of that a node has acted as the CH exceeds its assigned threshold (in Figure 7, N3 exceeds its threshold in round i+k so that N2, the node with the second highest priority, is chosen as the CH instead), server will compute, according to equation (11), new thresholds for all sensor nodes based on the residual energy of each node, and then transmits the re-computed thresholds to all nodes.

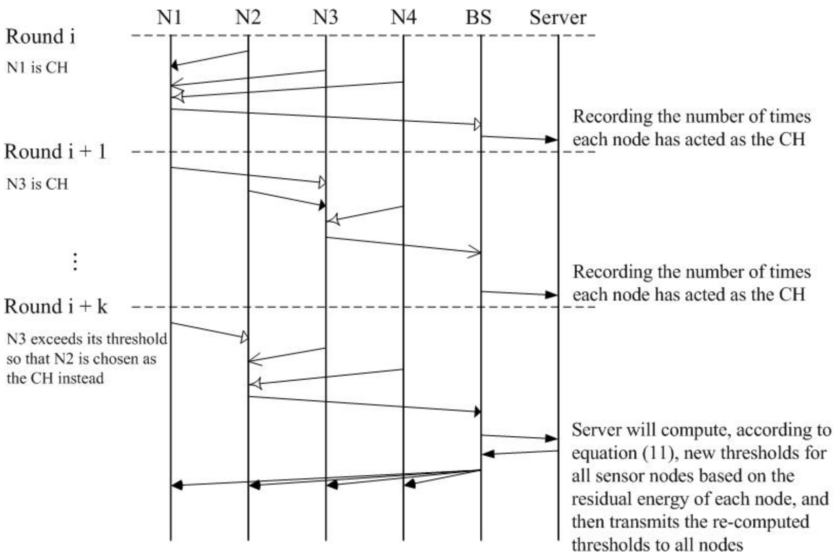


Fig. 7. Example of energy load distribution

5. Implementation

To evaluate the performance of our protocol, we compare it with the case of PT, as in Section 4, (in which all nodes use the same power level, one that is sufficient reach the entire BSN). We look at the real packet format in TinyOS in which sensing data is a small part (6 Bytes) of the whole packet (26 Bytes) and thus conservatively choose a compression ratio r=0.5 in our analysis. In addition, in our analysis we model the energy consumption for transmission and reception based on the specifications of a CC2420 transceiver. As shown in Fig. 8, our protocol can achieve more energy saving when the ratio of m/n increases. Here *n* is the number of nodes in the network and *m* is the number of static (non-moving) nodes. For a 10-node network, our protocol can achieve up to 90% less energy consumption as compared to PT. Here the saved energy is defined as $(E_{PT} - E_{our\ protocol})/E_{PT}$. In other words, in the best case, our protocol can prolong the lifetime of the network to be ten times longer as compared to PT.

We used KNN algorithm for the motion detection and implemented a testbed using the TelosB mote [52] which has a TI MSP430 [53] processor and a Chipcon CC2420 RF transceiver [51] that supports the IEEE 802.15.4 for communication. A pair of AA batteries are used for power supply. The data rate is 250 kbps when operating at the frequency of 2.4 GHz. We employed four triaxial MEMS accelerometer sensors placed on the wrists and ankles of a person to capture the subject's motions. The BS receives sensing data from the BSN and forwards it to a PC server where KNN is executed. We developed our program under TinyOS [54], which is a component-based and embedded operating system written in the nesC programming language.

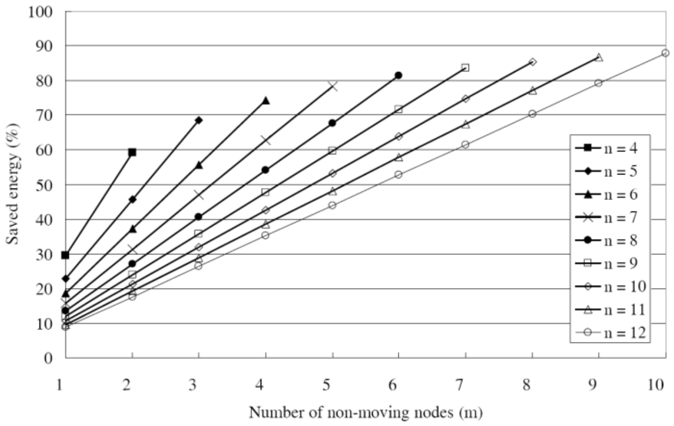


Fig. 8. Comparison between PT and our protocol for saving energy

To evaluate the accuracy of our cluster-based approach and compare it with the FT case, we used five pre-defined motions from boxing, namely Right Straight Punch (RSP), Right Hook Punch (RHP), Right Uppercut Punch (RUP), Right Front Kick (RFK), and Right Side Kick (RSK) in our experiments.

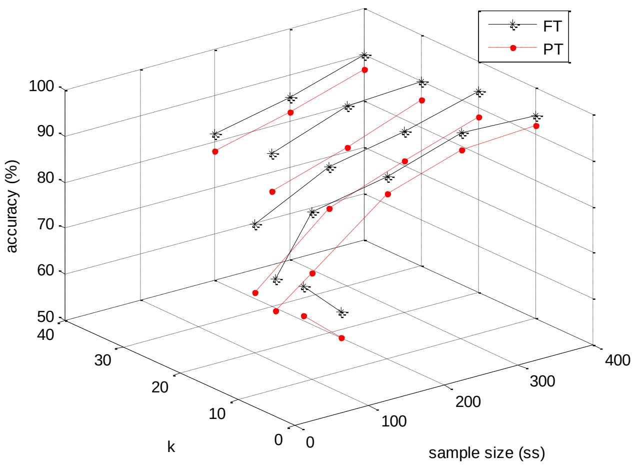


Fig. 9. Detecting accuracy based on varying combinations of *ss* and *k*

Our experiments were run in real time. In Fig. 9, each point represents the results from 50 repetitions of each motion (250 times in total). Sample size (*ss*) and *k* are important parameters for the KNN algorithm, so both of them are dynamically tuned to find the highest accuracy. Generally, when *k* is fixed, the accuracy will go up with the increase in sample size, but the reverse is not true. There is a tradeoff between (*ss*, *k*) and the accuracy, since we need to consider the computation cost/time depending on the characteristics of the applications. The results indicate our cluster-based algorithm (known as PT) can achieve similar performance as that of FT. When *ss*=300 and *k*=10, the accuracy is 94.8% for FT and 91% for PT respectively. The performance is slightly better when we increase the sample size to 400. The accuracies for FT and PT are 94% and 92%, respectively. Generally, the accuracy of FT is higher than that of PT, since it provides more information to the KNN.

6. Conclusion

In this paper, we provide a detailed analysis of how different parameters in a cluster-based algorithm can affect the amount of energy saving. Additionally, we propose a novel CH selection algorithm by considering the required power level for sending a packet from the CN to the CH. Furthermore, we use a threshold-based approach to evenly distribute the energy load among nodes by considering the number of times a node has acted as a CN and a CH. We implement our cluster-based algorithm on a testbed using the KNN algorithm and show that it can achieve high accuracy (above 90%). We are currently working on a prediction mechanism to reduce the number of transmissions required when motions of CNs can be predicted.

References

1. IEEE 802.15 WPAN™ Task Group 4 (TG4). [Online]. Available: http://www.ieee802.org/15/pub/TG4.html (current August 2011)
2. Xia, F., Xu, Z., Yao, L., Sun, W., Li, M.: Prediction-based Data Transmission for Energy Conservation. In Proceedings of the 5th Annual International Conference on Wireless Internet, Singapore. (2010)
3. Xiao, S., Dhamdhere, A., Sivaraman, V., Burdett, A.: Transmission Power Control in Body Area Sensor Networks for Healthcare Monitoring. IEEE Journal on Selected Areas in Communications, Vol. 27, No. 1, 37-48. (2009)
4. Park, S., Jayaraman, S.: On Innovation: Quality of Life and Technology of BodyNets. In Proceedings of the 3rd International Conference on Body Area Networks, Tempe, Arizona, USA. (2008)
5. Otal, B., Alonso, L., Verikoukis, C.: Novel QoS Scheduling and Energy-saving MAC Protocol for Body Sensor Networks Optimization. In Proceedings of the 3rd International Conference on Body Area Networks, Tempe, Arizona, USA. (2008)

6. Lorincz, K., Chen, B.-R., Challen, W. G., Chowdhury, A. R., Patel, S., Bonato, P., Welsh, M.: Mercury: A Wearable Sensor Network Platform for High-fidelity Motion Analysis. In Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems, Berkeley, California. (2009)
7. Biswas, S., Quwaider, M.: Body Posture Identification using Hidden Markov Model with Wearable Sensor Networks. In Proceedings of the 3rd International Conference on Body Area Networks, Tempe, Arizona, USA. (2008)
8. Ghasemzadeh, H., Jafari, R.: Sport Training using Body Sensor Networks: A Statistical Approach to Measure Wrist Rotation for Golf Swing. In Proceedings of the 4th International Conference on Body Area Networks, Los Angeles, California, USA. (2009)
9. Kwon, D. Y., Gross, M.: Combining Body Sensors and Visual Sensors for Motion Training. In Proceedings of the ACM SIGCHI International Conference on Advances in computer Entertainment Technology, Valencia, Spain, 94-101. (2005)
10. Vlasic, D., Adelsberger, R., Vannucci, G., Barnwell, J., Gross, M., Matusik, W., Popovic, J.: Practical Motion Capture in Everyday Surroundings. In Proceedings of the 34th International Conference on Computer Graphics and Interactive Techniques, San Diego, California, USA. (2007)
11. Lamprinos, I. E., Prentza, A., Sakka, E., Koutsouris, D.: Energy-efficient MAC Protocol for Patient Personal Area Networks. IEEE Engineering in Medicine and Biology Society, 3799-3802. (2005)
12. Omeni, O. C., Eljamaly, O., Burdett, A. J.: Energy Efficient Medium Access Protocol for Wireless Medical Body Area Sensor Networks. In Proceedings of the 4th IEEE/EMBS International Summer School and Symposium on Medical Devices and Biosensors, 29-32. (2007)
13. Abbasi, A.A., Younis, M.: A Survey on Clustering Algorithms for Wireless Sensor Networks. Elsevier Computer Communications, Vol. 30, 2826-2841. (2007)
14. Heinzelman, W.R., Chandrakasan, A., Balakrishnan, H.: Energy-efficient Communication Protocol for Wireless Microsensor Networks. In Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, Hawaii, USA, 1-10. (2000)
15. Ci, S., Guizani, M., Sharif, H.: Adaptive Clustering in Wireless Sensor Networks by Mining Sensor Energy Data. Computer Communications, Vol. 30, 2968-2975. (2007)
16. Yan, L., Zhong, L., Jha, N.: Energy Comparison and Optimization of Wireless Body-area Network Technologies. In Proceedings of the Second International Conference on Body Area Networks, Florence, Italy. (2007)
17. Foxlin, E., Harrington, M., Pfeifer, G.: Constellation: A Wide-Range Wireless Motion-Tracking System for Augmented Reality and Virtual Set Applications. In Proceeding of ACM SIGGRAPH 1998, New York, 371-378. (1998)
18. Foxlin, E., Harrington, M.: Wear Track: A Self-referenced Head and Hand Tracker for Wearable Computers and Portable VR. In Proceedings of the 4th International Symposium on Wearable Computers, Atlanta, Georgia, USA. (2000)
19. Smith, A., Balakrishnan, H., Goraczko, M., Priyantha, N.: Tracking Moving Devices with the Cricket Location System. In Proceedings of the Second International Conference on Mobile Systems, Applications, and Services, Boston, MA, 190-202. (2004)
20. Bussman, J., Martens, W., Tulen, J., Schasfoort, F., Van Den Bergemons, H., Stam, H.: Measuring Daily Behavior using Ambulatory Accelerometry: The

Activity Monitor. Behavior Research Methods, Instruments & Computers, 349-356. (2001)

21. Mayagoitia, R., Nene, A., Veltink, P.: Accelerometer and Rate Gyroscope Measurement of Kinematics: An Inexpensive Alternative to Optical Motion Analysis Systems. Journal of Biomechanics, Vol. 35, 537-542. (2002)
22. Sakaguchi, T., Tsutomu, K., Katayose, H., Sato, K., Inokuchi, S.: Human Motion Capture by Integrating Gyroscopes and Accelerometers. In Proceedings of the 1996 IEEE/SICE/RSJ International Conference on Multisensor Fusion and Integration for Intelligent Systems, Washington, DC, USA, 470-475. (1996)
23. KIRK, A. G., OBRIEN, J. F., FORSYTH, D. A.: Skeletal Parameter Estimation from Optical Motion Capture Data. In Proceeding of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 782-788. (2005)
24. Kurihara, K., Hoshino, S., Yamane, K., Nakamura, Y.: Optical Motion Capture System with Pan-tilt Camera Tracking and Tealtime Data Processing. In Proceedings of the 2002 IEEE International Conference on Robotics and Automation, Vol. 2, Washington, DC, USA, 1241-1248. (2002)
25. Dorfm¨uller-Ulhaas, Klaus: Robust Optical User Motion Tracking using a Kalman Filter. In Proceedings of the 10th ACM Symposium on Virtual Reality Software and Technology, Osaka, Japan. (2003).
26. Chen, Y., Lee, J., Parent, R., Machiraju, R.: Markerless Monocular Motion Capture using Image Features and Physical Constraints. In Proceedings of the Computer Graphics International, New York, USA, 36-43. (2005)
27. Sidenbladh, H., Black, M. J., Fleet, D. J.: Stochastic Tracking of 3D Human Figures using 2D Image Motion. In Proceedings of the 6th European Conference on Computer Vision, Dublin, Ireland, 702-718. (2000)
28. Uchinoumi, M.; Joo Kooi Tan; Ishikawa, S.: A Simple-Structured Real-Time Motion Capture System Employing Silhouette Images. In Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, The Hague, Netherlands, 10-13. (2004)
29. O'brien, J., Bodenheimer, R., Brostow, G., Hodgins, J.: Automatic Joint Parameter Estimation from Magnetic Motion Capture Data. In Proceedings of the Graphics Interface, Québec, Canada, 53-60. (2000)
30. Yu Su, Allen, C.R., Geng, D., Burn, D., Brechany, U., Bell, G.D., Rowland, R.: 3-D Motion System (Data-Gloves): Application for Parkinson's Disease. IEEE Transactions on Instrumentation and Measurement, Vol. 52, 662-674. (2003)
31. Hoshino, K.: CGA Synthesizer Interpolating and Extrapolating Motion Data. In Proceedings of the IFIP First International Workshop on Entertainment Computing, Makuhari, Japan, 339-346. (2002)
32. Roetenberg, D., Luinge, H.J., Veltink, P.H.: Inertial and Magnetic Sensing of Human Movement near Ferromagnetic Materials. In Proceedings of the IEEE/ACM International Symposium on Mixed and Augmented Reality, Tokyo, Japan, 268-269. (2003)
33. Nguyen, K. D., Chen, I-M., Yeo, S. H., Duh, B.-L.: Motion Control of a Robotic Puppet through a Hybrid Motion Capture Device. In Proceedings of the 3rd Annual IEEE Conference on Automation Science and Engineering Scottsdale, AZ, USA. (2007)
34. Klein, G., Drummond, T.: Tightly Integrated Sensor Fusion for Robust Visual Tracking. Image and Vision Computing, Vol. 22, 769-776. (2004)

35. Kawadia, V., Kumar, P.R.: Power Control and Clustering in Ad Hoc Networks. In Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communications Societies, San Francisco, USA. (2003)
36. Narayanaswamy, S., Kawadia, V., Sreenivas, R.S., Kumar, P.R.: Power Control in Ad-Hoc Networks: Theory, Architecture, Algorithm and Implementation of the COMPOW protocol. In Proceedings of the European Wireless 2002. Next Generation Wireless Networks: Technologies, Protocols, Services and Applications, Florence, Italy, 156-162. (2002)
37. Singh, S., Raghavendra, C.: PAMAS: Power Aware Multi-Access Protocol with Signalling for Ad Hoc Networks. ACM Computer Communication Review, Vol. 28, No. 3, 5-26. (1998)
38. Sohrabi, K., Gao, J., Ailawadhi, V., Pottie, G. J.: Protocols for Self-Organization of a Wireless Sensor Network, IEEE Personal Communications Magazine, Vol. 7, No. 5, 16-27. (2000)
39. Haas, Z., Tabrizi, S.: On Some Challenges and Design Choices in Ad-Hoc Communications. In Proceedings of the IEEE Military Communications Conference, Boston. (1998)
40. Ye, W., Heidenmann, J., Estrin, D.: An Energy-Efficient MAC Protocol for Wireless Sensor Networks. In Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies, New York, NY. (2002)
41. Lindsey, S., Raghavendra, C. S.: PEGASIS: Power Efficient GAthering in Sensor Information Systems. In Proceedings of the IEEE Aerospace Conference, Big Sky, Montana. (2002)
42. Lindsey, S., Raghavendra, C. S., Sivalingam, K.: Data Gathering in Sensor Networks using the Energy*Delay Metric. In Proceedings of the 2001 International Parallel and Distributed Processing Symposium, Workshop on Issues in Wireless Networks and Mobile Computing, San Francisco, CA. (2001)
43. Bandyopadhyay, S., Coyle, E.: An Energy-Efficient Hierarchical Clustering Algorithm for Wireless Sensor Networks. In Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communications Societies, San Francisco, USA. (2003)
44. Younis, O., Fahmy, S.: HEED: A Hybrid, Energy-Efficient, Distributed Clustering Approach for Ad Hoc Sensor Networks. In Proceedings of the 23rd Conference of the IEEE Communications Society, Hong Kong. (2004)
45. Manning C. D., Schutze H.: Foundations of Statistical Natural Language Processing. Cambridge, MIT Press. (1999)
46. Liang, J. J., Feng, Y. C.: Indexing the Bit-code and Distance for Fast KNN Search In High-dimensional Spaces. Zhejiang University: Science A, Vol. 8, 857-863. (2007)
47. Guha, S., Rastogi, R., Shim, K.: Cure: An Efficient Clustering Algorithm for Large Databases. In Proceedings of the ACM SIGMOD International Conference on Management of Data, Seattle, Washington, USA, 73-84. (1998)
48. Krishnamachari, B., Estrin, D, Wicker, S.: The Impact of Data Aggregation in Wireless Sensor Networks. In Proceedings of the 22nd International Conference on Distributed Computing Systems Workshops, Vienna, Austria. (2002)
49. Maddes, R., Szewczyk, M., Franklin, J., Culler, D.: Supporting Aggregate Queries over Ad-hoc Wireless Sensor Network. In Proceedings of the 4th IEEE Workshop on Mobile Computing Systems and Applications, Callicoon, NY, USA. (2002)
50. Shnayder, V., Hempstead, M., Chen, B.-R., Werner-Allen, G., Welsh, M.: Simulating the Power Consumption of Large-scale Sensor Network Applications.

In Proceedings of the Second ACM Conference on Embedded Networked Systems, Baltimore, Maryland, USA. (2004)
51. CC2420 Datasheet, http://focus.ti.com/docs/prod/folders/print/cc2420.html
52. TPR2400 Datasheet. [Online]. Available: http://www.willow.co.uk/html/telosb_mote_platform.html (current August 2011)
53. Microcontrollers (MCU). Online]. Available: http://focus.ti.com/mcu/docs/mcuhome.tsp?sectionId=101 (current August 2011)
54. Tinyos. [Online]. Available: http://www.tinyos.net/ (current August 2011)

Kun-chan Lan received his Ph.D degree from Computer Science at University of Southern California in 2004, advised by Professor John Heidemann. From 2004 to 2007, Kun-chan joined the Network and Pervasive Computing Program at NICTA (National ICT Australia) in Sydney as a researcher. He is currently an assistant professor at the Department of Computer Science and Information Engineering of National Cheng Kung University.

Chine-Ming Chou received his B.S. degree from Computer Science at Vanung University of technology in 2005 and M.S. degree from Information and Communication Engineering at Chaoyang University of technology in 2007. He is currently working for his Ph.D degree in Department of Computer Science and Information Engineering, National Cheng Kung University. His research interests include Network Simulation, Network measurement, Opportunistic routing, WiMAX and DSRC protocol, and mobility model.

Tzu Nung Wang received his BS (Industrial Engineering and Management) from National Taipei University of Technology and MS (Medical Informatics) from National Cheng Keng University in Taiwan in 2003 and 2010 respectively. His research interest is related to wireless sensor networks especially for energy saving issue. Now he works as an engineer of software development in a LCD company.

Mei-Wen Li received her M.S degrees from Chaoyang University of Technology in Taiwan in 2006. Her research interests encompass issues relating to Cellular Network, Vehicular network and Cognitive network. Currently, she works an camera development company as an engineer.

Received: March 10, 2011; Accepted: May 4, 2011.