Implementing Preserving Location Monitoring System for Wireless Sensor Networks

S.B. Swathi
Assistant Professor, Department of Information Technology KITS, Warangal, Andhra Pradesh, India

B.Surya Samantha
Assistant Professor, Department of Information Technology KITS, Warangal, Andhra Pradesh, India

Mahesh Kumar Thota
Assistant Professor, Department of Information Technology KITS, Warangal, Andhra Pradesh, India

Abstract- Customarily, personal locations are protected, monitored with potentially untrusted server poses privacy threats to the monitored individuals. To address this issue of privacy threat and secured location monitoring, a privacy-preserving location monitoring (PPLM) system for wireless sensor networks (WSN) is developed and has been effectively utilized. The intention of developing PPLM system is to provide secured privacy to each and every individual and also high quality location monitoring services for different locations. In this paper, the performance of PPLM system is evaluated in terms of generated aggregate locations, computational efficiency and cloaked area size. Two in network location anonymization algorithms, namely, resource and quality-aware algorithms are implemented, whose main aim and objective is to enable the proposed system, to provide a high quality location monitoring services for system users and to preserve the privacy of individual person or an object. The algorithms are implemented to generate the aggregate location information for a given location and to provide high quality monitoring service. The performance of proposed PPLM system is evaluated through simulated experiments and the experimental results proved the efficacy of implemented algorithms and the proposed system provides a high quality location monitoring services for system users and guarantees the location privacy of the monitored persons. To monitor a given location, aggregate location of the given location is obtained first and then the queries are answered based on the estimated distribution of the monitored persons or objects.

Keywords – Wireless Sensor Network (WSN), Privacy preserving location monitoring (PPLM) system, Resource- and Quality-aware algorithms, Estimation error, Computational efficiency and Cloaked area size

I. INTRODUCTION

With the spreading application of Wireless Sensor Networks (WSNs) [1]-[3] in various sensitive areas such as health-care, military, habitat monitoring, etc, the need to ensure security and privacy is becoming imperatively important. For example, in battlefield application scenario, the location of a soldier should not be exposed if he initiates broadcast query . In the meantime, query must be transferred to the destination in an encrypted manner via only trusted route nodes. Similarly, in habitat monitoring application scenarios, such as Great Duck Island or Save-the-panda application where large numbers of sensor nodes are deployed to observe the vast habitat of ducks and pandas, an adversary can try to capture the panda or duck by back-tracing the routing path until it reaches the source sensor nodes. Therefore, in order to prevent the adversary from back-tracing, the route, location and data privacy mechanisms must be enforced. With respect to these application scenarios, Network level privacy has often been categorized into four categories:

1. **Sender node identity privacy**: No intermediate node can get any information about who is sending the packets except the source, its immediate neighbors and the destination.
2. **Sender node location privacy**: No intermediate node can have any information about the location (in terms of physical distance or number of hops) about the sender node except the source, its immediate neighbors and the destination.
3. **Route privacy**: No node can predict the information about the complete path (from source to destination). Also, a mobile adversary gets no clue to trace back the source node either from the contents and/or directional information of the captured packet(s), and

4. **Data packet privacy**: No node can see the information inside in a payload of the data packet except the source and the destination. Existing privacy schemes that have specifically been proposed for WSNs only provide partial network level privacy. Providing a full network level privacy is a critical and challenging issue due to the constraints imposed by the sensor nodes (e.g., energy, memory and computation power), sensor network (e.g., mobility and topology) and QoS issues (e.g., packet reach-ability and trustworthiness). Thus, an energy efficient privacy solution is needed namely, Resource-aware and Quality-aware algorithms are implemented for efficient monitoring of the geographical areas and also to provide better monitoring services. Presently sensor nodes are being used for applications like knowing wind response, intruders detection, classification and tracking, healthcare, home security, industrial control, precision agriculture, social networking etc. The system ,wireless sensor and actuator network (WSAN) depicted in Fig. 1, is a collection of small randomly dispersed devices that provide three essential functions; the ability to monitor physical and environmental conditions, often in real time, such as temperature, pressure, light and humidity; the ability to operate devices such as switches, motors or actuators that control those conditions; and the ability to provide efficient, reliable communications via a wireless network. The implementation of this last capability is the most unique to WSAN [4],[6]. Since they are designed for low traffic monitor and control applications, it is not necessary for them to support the high data throughput requirements that data networks like Wi-Fi require. Typical WSAN over-the-air data rates range from 20 kbps to 1 Mbps. Consequently they can operate with much lower power consumption, which in turn allows the nodes to be battery powered and physically small. WSANs are typically self-organizing and self-healing. Self-organizing networks allow a new node to automatically join the network without the need for manual intervention. Self-healing networks allow nodes to reconfigure their link associations and find alternative pathways around failed or powered-down nodes. How these capabilities are implemented is specific to the network management protocol and the network topology, and ultimately will determine the networks flexibility, scalability, cost and performance. Implementing these privacy-preserving location algorithms provides monitoring service for the different geographical areas.

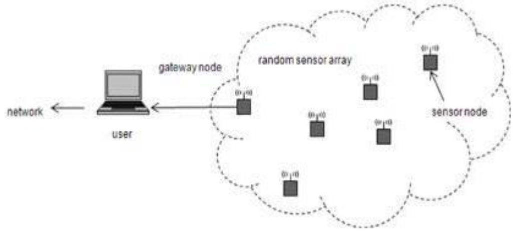


Figure 1: Wireless sensor/actuator network

The system relies on k-anonymity privacy concept i.e there must be minimum of k persons or objects in a given location. All the persons are treated with equal identity. By using the anonymity concept, all the persons are assigned the same identity belonging to a given location. To provide monitoring services, two algorithms namely, Resource-Aware and Quality-Aware algorithms are implemented. The former aims to minimize the communication and computational cost while the latter aims to minimize the size of cloaked areas and maximize the accuracy of aggregate locations. and sends it the requested users. The sensor nodes have to collaborate with each other to get the aggregate locations. The cloaked area for a given location is obtained if there at least k persons existing in that given location. The cloaked area is obtained by implementing Resource-Aware algorithm on a given location. The cloaked area is amplified by the Quality Aware algorithm. The information present in the amplified image is used to answer the queries of the user. The queries are stored in a table called Resource-Aware Location at the server .Whichever sensor node answers a given query, its identity and the location name which user has requested is stored in a table called Peer List table at the server. A sensor node monitors the geographic locations based on its values of latitude and longitude. The queries are answered based on the amplified maps generated by the Quality-Aware algorithm. A spatial histogram is used to analyze the aggregate locations which in turn give the estimated distribution of monitored persons in the system. The system is evaluated through simulated experiments. The results show that the communication and computational cost of the resource-aware algorithm is lower than the quality-aware algorithm, while the quality-aware algorithm provides more accurate monitoring services (the average accuracy is about 90%) than the resource-aware algorithm (the average accuracy is about 75%). Hence Resource-Aware algorithm is used to compute aggregate location which involves the activity of communication and computation and the Quality- Aware algorithm is used to obtain amplified and accurate images of the aggregate locations to monitor these locations. Both algorithms only reveal k-anonymous aggregate location information to the server which are suitable for different system settings. The resource-aware algorithm is suitable for the system, where the sensor nodes are provided with scarce communication and computational resources, while the quality-aware algorithm is suitable for the system, where accuracy is the most important factor in monitoring services.

II. WSN SYSTEM MODEL

The architecture of WSN system is shown in Fig. 2, where there are three major entities, *sensor nodes*, *server*, and *system users*. We will define the problem addressed by our system, and then describe the detail of each entity and the privacy model of the system. For a given set of sensor nodes *s1; s2; : : : ; sn* with sensing areas *a1; a2; : : : ; an*, respectively, a set of moving objects *o1; o2; : : : ; om*, and a required *anonymity level k*, for which an aggregate location is to be identified for each sensor node *Si* in a form of *Ri = (Areai ;Ni),* where *Areai* is a rectangular area containing the sensing area of a set of sensor nodes *Si* and *Ni* is the number of objects residing in the sensing areas of the sensor nodes in *Si*, such that *Ni ≥ k* and a spatial histogram is build to answer an aggregate query Q that asks about the number of objects in a certain area Q:Area based on the aggregate locations reported from the sensor nodes.

Sensor nodes*:* Each sensor node is responsible to find the number of objects in its sensing area, blur its sensing area into a cloaked area A, which includes at least k objects, and the object information is reported to A, as aggregate location information to the server. There is no assumption about having any assumption about the network topology. The system uses a distributed tree to acquire a communication path from each sensor node to the server.

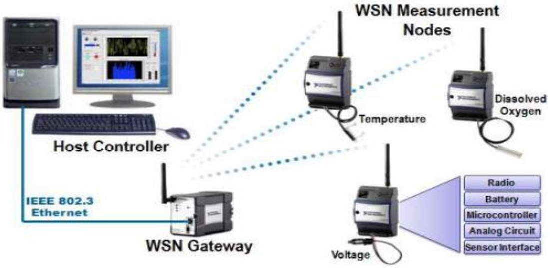


Figure 2: WSN System Architecture

Each sensor node is also aware of its location and sensing area.

Server: The server collects the aggregate locations reported from the sensor nodes and uses a spatial histogram to estimate the distribution of the monitored objects, and answers the range of queries based on the estimated object distribution. Furthermore, the administrator can change the anonymized level k of the system at anytime by disseminating a message with a new value of k to all the sensor nodes.

System users: Authenticated administrators and users can issue range queries to our system through either the server or the sensor nodes, as depicted in Fig. 2. The server uses the spatial histogram to answer their queries.

Privacy model: In our system, the sensor nodes constitute a trusted zone, where they behave as denoted in our algorithm and communicate with each other through a secure network channel to avoid internal network attacks, for example, eaves dropping, traffic analysis, and malicious nodes [7]-[10]. Since establishing such a secure network channel has been studied in the literature, the discussion of how to get this network channel is beyond the scope of this paper. However, the solutions that have been used in previous works can be applied to our system.

The WSN system also provides anonymous communication between the sensor nodes and the server by employing existing anonymous communication techniques [11]-[13]. Thereby for a given aggregate location R, the server only knows that the sender of R is one of the sensor nodes within R. Furthermore, only authenticated administrators can change the *k-anonymity* level and the spatial histogram size. In emergency cases, the administrators can set the k-anonymity level to a small value to get more accurate aggregate locations from the sensor nodes, or even set it to zero to disable our algorithm to get the original readings from the sensor nodes, in order to get the best services from the system. Since the server and the system user are outside the trusted zone, they are untrusted. In view of the privacy threat in existing location monitoring systems, in an identity-sensor location monitoring system, each sensor node reports the exact location information of each monitored object to the server through which an opponent can pinpoint each object's exact location. On the other hand, in a counting-sensor location monitoring system, each sensor node reports the number of objects in its sensing area to the server. The opponent can map the monitored areas of the sensor nodes to the system layout. If the object count of a monitored area is very small or equal to one, the opponent can infer the identity of the monitored objects based on the mapped monitored area. Since our system only allows each sensor node to report a k-anonymous aggregate location to the server, the opponent cannot infer an object's exact location. The larger the anonymity level, k, the more difficult for the opponent to infer the object's exact location. With the k-anonymized aggregate locations reported from the sensor nodes, the underlying spatial histogram at the server provides low quality location monitoring services for a small area, and better quality services for larger areas. This is a nice privacy-preserving feature, because the object count of a small area is more likely to reveal personal location information. The definition of a small area is relative to the required anonymity level, because our system provides lower quality services for the same area if the anonymized level gets stricter. For each sensor node, an attacker model is used which counts the number of objects with in a cloaked area.

III. SIMULATION RESULTS

The different geographical areas monitored by the sensor nodes are taken by the server. Resource Aware algorithm is implemented on these locations to get the cloaked areas and the aggregate locations and the histograms for the performance evaluation parameters are plotted in Fig.3 to Fig. 5; estimation error, computational complexity and cloaked area size after processing the in-network anonymization algorithms. The number of object count existing in a cloaked area is obtained from the attacker model. The value of k=10 is assumed. When the server receives the user requests, it implements the Quality-Aware Location algorithm on the user requested locations and amplifies its aggregate location image to answers the user queries. The data regarding the different aggregate location images, identities of the sensor nodes (whichever have answered the user queries) and the user requests are stored in three different database tables.

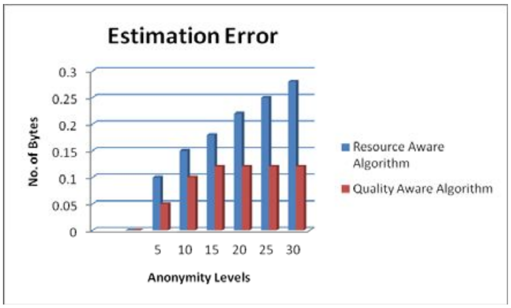


Figure 3: Histogram plot of Estimation Error

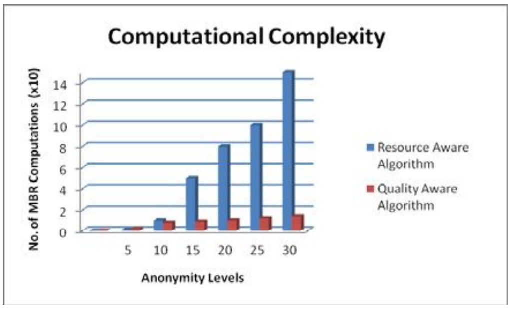


Figure 4: Histogram plot of Computational Complexity

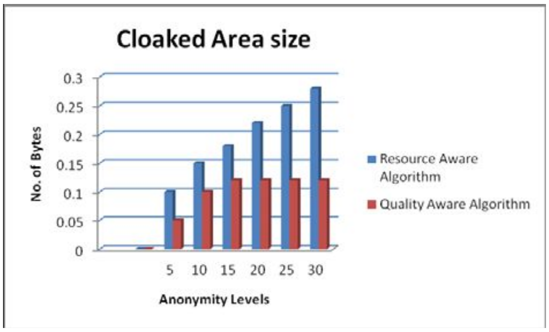


Figure 5: Histogram plot of Cloaked Area Size

3.1 The Resource-Aware Algorithm

IV. THE DISTRIBUTED LEAST SQUARES ALGORITHM

A. General Idea

DLS builds on the mathematical formulations introduced in the background section. By using the linearization tool the matrices in Equ. (5) have two important benefits. First, all elements in the coefficient matrix *A* are generated by beacon positions *B*1(*x, y*) . . . *Bm*(*x, y*) only. We assume in the first instance that all sensor nodes can establish communication links between all beacons, then matrix *A* is the same on every sensor node. Second, vector **b** contains distances between sensor nodes and beacons *r*1 . . . *rm*, which have to be estimated on every sensor node independently. The result is that the normal equations (QRF and SVD as well) can be split into two parts -, the *precalculation*:*Ap and* the *postcalculation AT*.

Here, the precalculation is executed on one high performance node, which additionally avoids high redundancy, because normally this precalculation has to be executed on all sensor nodes separately. It is very important to emphasize that the precalculation is identical on each sensor node. Thus, it is calculated only once, conserving expensive energy resources. The simple postcalculation is then executed on each sensor node with its individual distance measurements to all beacons. This approach complies with two important design strategies for algorithms in large sensor networks a **resource-aware** and **distributed** localization procedure. Finally, this can be achieved with less communication overhead required for other exact algorithms.

B. Algorithm Description

At this point we briefly describe the algorithm process. DLS is divided into three phases, which are shown in Fig. 1. In phase 1 all beacons send their position *Bi*(*x, y*) hop-by-hop over their beacon neighbors to the base station. In phase 2 the base station starts generating the initial matrices and computes *Ap*. The result is sent over beacons to all sensor nodes, which estimate their position after measuring the distances to all beacons and executing the postcalculation in phase 3. This leads to the following sequence:

Phase 1: Initialization

- All beacons send their position *B*(*x, y*) to the base station.

Phase 2: Complex Precalculation (central)

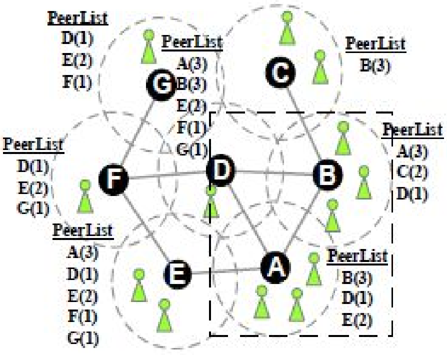
- Base station builds matrix *A* and vector **d***p*. - Starting the complex precalculation of matrix *Ap*.

Phase 3: Simple Postcalculation (distributed)

- Base station sends matrix *Ap* and vector **d***p* to all sensor nodes. - Sensor nodes determine the distance to every beacon *r*1..*rm*. - Sensor nodes receive matrix *Ap* and vector **d***p*, built vector **b** and estimate their own position *Pest*(*x, y*) autonomously.

*C.***Algorithm 1** Resource-aware location anonymization

```
1: function RESOURCEAWARE (Integer k, Sensor m, List R)
2: PeerList   f;g
// Step 1: The broadcast step
3: Send a message with m's identity m:ID, sensing area m:Area, and obj count m:Count to m's neighbor peers
4: if Receive a message from a peer p, i.e., (p:ID, p:Area, p:count) then
5:     Add the message to PeerList
6:     if m has found an adequate number of objects then
7:         Send a noti_cation message to m's neighbors
8:     end if
9:     if Some m's neighbor has not found an adequate number of objects th
10:        Forward the message to m's neighbors
11:    end if
12: end if
// Step 2: The cloaked area step
13: S :  fmg
14: Compute a score for each peer in PeerList
15: Repeatedly select the peer with the highest score from PeerList to S until the total number of objects in S is at least k
16: Area   a minimum bounding rectangle of the senor nodes in S
17: N :  the total number of objects in S
// Step 3: The validation step
18: if No containment relationship with Area and R 2 R then
19:     Send (Area;N) to the peers within Area and the server
20: else if m's sensing area is contained by some R 2 R then
21:     Randomly select a R0 2 R such that R0:Area contains m's sensing area
22:     Send R0 to the peers within R0:Area and the server
23: else
24:     Send Area with a cloaked N to the peers within Area and the server
25: end if
```



This approach is to find a cloaked area based on the information stored in *PeerList*. For each sensor node m, m initializes a set S = fmg, and then determines a score for each peer in its *PeerList* (Lines 13 to 14 in Algorithm 1). The score is defined as a ratio of the object count of the peer to the Euclidean distance between the peer and m. The idea behind the score is to select a set of peers from *PeerList* to S to form a cloaked area that includes at least k objects and has an area as small as possible. Then we repeatedly select the peer with the highest score from the *PeerList* to S until S contains at least k objects (Line 15). Finally, m determines the cloaked area (Area) that is a *minimum bounding rectangle* (MBR) that covers the sensing area of the sensor nodes in S, and the total number of objects in S (N) (Lines 16 to 17).

An MBR is a rectangle with the minimum area (which is parallel to the axes) that completely contains all desired regions, the dotted rectangle is the MBR of the sensing area of sensor nodes A and B (as in figure-6). The major reasons of our algorithms aligning with MBRs rather than other polygons are that the concept of MBRs have been widely adopted by existing query processing algorithms and most database management systems have the ability to manipulate MBRs efficiently.

Figure 3c illustrates the cloaked area step. The *PeerList* of sensor node A contains the information of three peers, B, D, and E. The object count of sensor nodes B, D, and E is 3, 1, and 2, respectively.We assume that the distance from sensor node A to sensor nodes B, D, and E is 17, 18, and 16, respectively. The score of B, D, and E is 3=17 = 0:18, 1=18 = 0:06, and 2=16 = 0:13, respectively. Since B has the highest score, we select B. The sum of the object counts of A and B is six which is larger than the required anonymity level k = 5, so we return the MBR of the sensing area of the sensor nodes in S, i.e., A

and B, as the resource-aware cloaked area of A, which is represented by a dotted rectangle.

Step 3: The validation step. The objective of this step is to avoid reporting aggregate locations with a containment relationship to the server. Let Ri and Rj be two aggregate locations reported from sensor nodes I and j, respectively.

⊂

If Ri's monitored area is included in Rj 's monitored area, Ri:Area   R j:Area or Rj:Area ⊂ Ri:Area, they have containment relationship. We do not allow the sensor nodes to report their aggregate locations with the containment relationship to the server, because combining these aggregate locations may pose privacy leakage. For example, if Ri:Area ≠ Rj:Area and Ri:Area 6= Rj:Area, an adversary can infer that the number of objects residing in the non-overlapping area, Rj:Area - Ri:Area, is Rj:N - Ri:N. In case that Rj:N - Ri:N < k, the adversary knows that the number of objects in the non-overlapping is less than k, which violates the k-anonymity privacy requirement. As this step ensures that no aggregate location with the containment relationship is reported to the server, the adversary cannot obtain any deterministic information from the aggregate locations. In this step, each sensor node m maintains a list R to store the aggregate locations sent by other peers. When a reporting period starts, m nullifes R. After m finds its aggregate location Rm, m checks the containment relationship between Rm and the aggregate locations stored in R. If there is no containment relationship between Rm and the aggregate locations in R, m sends Rm to the peers within Rm:Area and the server (Line 19 in Algorithm 1). Otherwise, m randomly selects an aggregate location Rp from the set of aggregate locations in R that contain m's sensing area, and m sends Rp to the peers within Rp:Area and the server (Lines 21 to 22). In case that no aggregate location in R contains m's sensing area, we find a set of aggregate locations in R that are contained by Rm, R0, and N0 is the number of monitored persons in Rm that is not covered by any aggregate location in R0. If N′ ≥ k, the containment relationship does not violate the k-anonymity privacy requirement;

Algorithm 2 Quality-aware location anonymization

```
1: function QUALITYAWARE (Integer k, Sensor m, Set init solution, List R)
2:   current min cloaked area   init solution
// Step 1: The search space step
3:   Determine a search space S based on init solution
4:   Collect the information of the peers located in S
// Step 2: The minimal cloaked area step
5:   Add each peer located in S to C[1] as an item
6:   Add m to each itemset in C[1] as the _rst item
7:   for i = 1; i _ 4; i ++ do
8:     for each itemset X = fa1; : : : ; ai+1g in C[i] do
9:       if Area(MBR(X)) < Area(current min cloaked area) then
10:        if N(MBR(X)) _ k then
11:          current min cloaked area   fXg
12:          Remove X from C[i]
13:        end if
14:      else
15:        Remove X from C[i]
16:      end if
17:    end for
18:    if i < 4 then
19:      for each itemset pair X=fx1;: : : ;xi+1g, Y =fy1;: : : ;yi+1g in C[i]
         do
20:        if x1 = y1; : : : ; xi = yi and xi+1 6= yi+1 then
21:          Add an itemset fx1; : : : ; xi+1; yi+1g to C[i + 1]
22:        end if
23:      end for
24:    end if
25:  end for
26:  Area   a minimum bounding rectangle of current min cloaked area
27:  N   the total number of objects in current min cloaked area
// Step 3: The validation step
28:  Lines 18 to 25 in Algorithm 1
```

therefore m sends Rm to the peers within Rm:Area and the server. However, if N0 < k, m cloaks the number of monitored persons of Rm, Rm:N, by increasing it by an integer uniformly selected between k and 2k, and sends Rm to the peers within Rm:Area and the server (Line 24). Since the server receives an aggregate location from each sensor node for every reporting period, it cannot tell whether any containment relationship takes place among the actual aggregate locations of the sensor nodes.

IV.CONCLUSION

We presented a privacy-preserving location monitoring (PPLM) system, implemented for wireless sensor networks without disturbing the privacy of the individual persons or objects. The two in-network location anonymization algorithms, namely, resource- and qualityaware algorithms are implemented one after other for improved secured privacy. The resource-aware algorithm aims to minimize communication and computational cost, while the quality-aware algorithm aims to minimize the size of cloaked areas in order to generate more accurate aggregate locations. A spatial histogram approach is used to provide location monitoring services through answering the range queries. The system is evaluated through simulated experiments. The experimental results proved that the presented system provides a high quality location monitoring services while preserving the monitored object's location privacy. In this paper, the performance of PPLM system is evaluated in terms of generated aggregate locations, computational efficiency and cloaked area size. As part of the future enhancement of presented work on wireless sensor networks, the same process of monitoring can be implemented by using mobile devices which is provided with server activation feature.

REFERENCES

[1] J. Kong and X. Hong, ANODR: Anonymous on demand routing with untraceable routes for mobile adhoc networks , in Proc. Of MobiHoc, 2003.

[2] P. Kamat, Y. Zhang,W. Trappe, and C. Ozturk, Enhancing sourcelocation privacy in sensor network routing , in Proc. Of ICDCS, 2005.

[3] S. Guo, T. He, M. F. Mokbel, J. A. Stankovic, and T. F. Abdelzaher, On accurate and efficient statistical counting in sensor-based surveillance systems , in Proc. of MASS, 2008.

[4] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, .The cricket location-support system, in Proc. of MobiCom, 2000.

[5] B. Son, S. Shin, J. Kim, and Y. Her, Implementation of the realtime people counting system using wireless sensor networks ,IJMUE, vol. 2, no. 2, pp. 63.80, 2007.

[6] Traf-Sys Inc., .People counting systems. http://www.trafsys.com/products/people-counters/thermal-sensor.aspx.

[7] M. Gruteser, G. Schelle, A. Jain, R. Han, and D. Grunwald, Privacy-aware location sensor networks , in Proc. of HotOS, 2003.

[8] G. Ghinita, P. Kalnis, A. Khoshgozaran, C. Shahabi, and K.-L. Tan, Private queries in location based services: Anonymizers are not necessary , in Proc. of SIGMOD, 2008.

[9] W. He, X. Liu, H. Nguyen, K. Nahrstedt, and T. Abdelzaher PDA: Privacy-preserving data aggregation in wireless sensor networks , in Proc. of Infocom, 2007.

[10] M. Shao, S. Zhu, W. Zhang, and G. Cao, PDCS: Security and privacy support for data-centric sensor networks , in Proc. of Infocom, 2007.

[11] B. Carbunar, Y. Yu, W. Shi, M. Pearce, and V. Vasudevan, Query privacy in wireless sensor networks , in Proc. of SECON, 2007.

L. Ghouti, A. Bouridane, M.K. Ibrahim, and S. Boussakta, Digital image watermarking using balanced multiwavelets , IEEE Trans. Signal Process., 2006, Vol. 54, No. 4, pp. 1519-1536.