Proc. of Urban Cultural Heritage Disaster Mitigation, Vol.3 (June 2009)

Improved GPS-free Ad-hoc Network Positioning for Urban Disaster Response

Pham Doan Tinh[1] and Makoto Kawai[2]

[1]Student, [2]Professor, Ritsumeikan University, Graduate School of Science and Engineering
(1-1-1 Noji-Higashi, Kusatsu, Shiga 525-8577, Japan)

Wireless Ad-hoc networks have exposed their strength of self-organizing, self-healing suitable for communications in disaster cases where the infrastructure devices might be damaged. Knowing the physical location of people or important objects is very important in supporting people from self-protection, searching and rescuing. This paper introduces an Improved GPS-free Ad-hoc Network Positioning Algorithm Based on Self-Organizing Maps to support the response in urban disasters. The method is free from GPS and is less expensive and suitable for indoor environments. The simulation results show the improvement of the proposed algorithm. In addition, it is also noted that by attaching small wireless intelligent sensors running the algorithm on cultural heritage objects, we can monitor their physical location as well as the environment information such as temperature and humidity, protect them from any harm to the object like thieves and find them in early stages of disasters.
Key Words : *Disaster, Ad-hoc network, GPS-free, Positioning, Self-Organizing Maps.*

1. Introduction

(1) Ad-hoc networks and disaster scenarios

Many historical cities around the world now are being protected and preserved their original characteristics. This preservation leads to the fact that some very ancient cities are more and more prone to damaging in disasters. From generation to generation, people are living in these cities. Preserving the life of people and precious objects is an important issue to be solved.

Wireless Ad-hoc networks do not rely on fixed and preinstalled devices such as wireless access points or base stations. They have a dynamic network topology and can be formed quickly from wireless devices. These features of wireless ad-hoc networks are very suitable with scenarios like during disaster or recovery and restoration after disaster. In such scenarios, infrastructure-based network might be severely damaged. Ad-hoc networks play as communication system to link people together, to broadcast important information and to guide the people about the current status of the environment in disaster. Among other useful information, location information plays a very important role in disaster cases telling people where they are and where the safe place to go or warning the rescuers for fast searching.

This work is an extension of an existing work[1)] aiming at improving the accuracy and robustness of the positioning algorithm.

(2) Technology investigation to the disaster mitigation

At present, the advances in science and technology let us think about applying more ideal solutions in disaster prediction, control, and mitigation to the historical cities. This paper will focus on the technological aspect of the disaster mitigation problem. All of us have an imagination about the scenes after each disaster: houses are damaged, and people are injured or killed or got stuck somewhere in the ruin. At any cost, we have the duty to help affected people from death or find out them even though they died already.

To achieve the goal of locating the people or objects, there are many methods ranging from manually locating the people by the first aid men to using modern technologies. From recent disasters, we can realize that if we had high technological devices, then victims would have been rescued immediately, and would not have suffered so much from thirst or pain.

In most natural disasters, electricity is cut off, so when applying any method we will consider the power supply for such high technological devices. But, thanks to the advances in science and technology, many wireless devices now consume very little power, and they can last for 5 years of idle state, and several days to a week of active state. An example of such devices are the Zigbee 802.15.4 devices with cheap price, long range communication, and very little power consumption. We can think about the case when all people and precious objects are equipped with these wearable devices together with the similar devices from the first aid men or centrally controlled devices will form a wireless ad-hoc network as showed in Fig.1. Then we apply a positioning algorithm so that all devices can know their and others' location. At that time, the first aid men will easily find out the location of the people or object to rescue. Besides that, the people need help also know about the locations around them and help guiding them to escape from the danger.

As we know that GPS positioning system is not effective in indoor environments and GPS devices draws lot of power sources. So, we considered the GPS-free Ad-hoc network positioning system which can be applied to the circumstances like disaster cases. Besides knowing the position, wireless devices can be equipped with other sensors to capture the surrounding environmental parameters such as temperature, humidity, and other information for other system such as Sensor/Actor networks. These information will help people in monitoring, preserving the cultural heritage sites effectively.

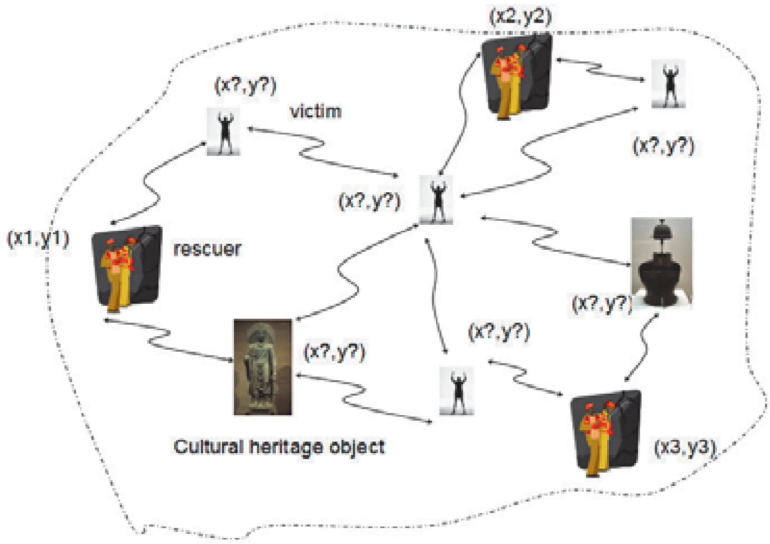

Fig.1 Ad-hoc network implementation at a disaster site.

2. Positioning Technologies

(1) Advances in Ad-hoc Positioning

Recently, mobile ad-hoc network localization has received attention from many researchers[2]. Many algorithms and solutions have been presented so far. These algorithms are ranging from simple to complicated schemes, but they can be categorized as range-based and range-free algorithms. Range-free algorithms utilize only connectivity information and the number of hops between nodes. The others utilize the distance measured between nodes by either using the Time-Of-Arrival (TOA) [3], Time-Differential-Of-Arrival (TDOA)[4], Angle-Of-Arrival (AOA)[5] or Received-Signal-Strength-Indicator[6,7] technologies. However, they usually need extra hardware to achieve such measurement. When calculating the absolute location, most schemes need at least three anchors (nodes that are equipped with Global Positioning System or know their location in advance).

DV-HOP is a typical range-free algorithm. It was proposed by Niculescu and Nath[8] as an Ad-hoc Positioning System (APS). DV-HOP uses distance-vector forwarding technique to get the minimum hop count from a node to heard anchors. By using corrections calculated by anchors (average hop-distance between anchors), nodes estimate their location by using lateration (triangulation) method. Besides DV-HOP, some other algorithms seem to be more complicated, but have better accuracy. The Multi-Dimensional Scaling Map (MDS-MAP) proposed by Yi Shang et al. [9] is an example. MDS-MAP is originated from a data analytical technique by displaying distance-like data in geometrical visualization. It computes the shortest paths between all pairs of nodes to build a distance matrix and then applies the classical Multi-Dimensional Scaling (MDS) to this matrix to retain the first two largest eigenvalue and eigenvector to a 2-D relative map. After that, with three given anchors, it transforms the relative map into an absolute map based on anchors' absolute location. This method is implemented in centrally controlled manner by the authors. Tran et al. [10] proposed a new localization scheme based on Support Vector Machine (SVM). The authors have contributed another machine learning method to the localization problem, and proved the upper bound error of this method.

Regarding the localization based on Self-Organizing Maps, some researchers have employed SOM directly or with some modification. The method presented by G. Giorgetti et al. [11] employed the classical SOM to the localization. This method uses centralized implementation and requires thousands of learning steps in convergence of network topology. The authors also realize that this method is good for small and medium size networks of up to 100 nodes. S. Asakura et al. proposed a distributed localization scheme[12] based on SOM. Jie Hu et al. [13] also proposed another version of distributed localization based on SOM. In this work, the authors employed a deduced SOM version[14]. But, this method still needs too many iterations (at least 4000) to make the topology to be converged with a relatively low accuracy. In another work[15], the authors use SOM to track a mobile robot with the utilization of surrounding environments from readings of sensor data. In the work presented by Ertin et al. [16], another version of SOM was used to implement the localization in wireless sensor networks.

(2) Motivation for SOM Based Positioning

Suppose that we have a network of connected nodes, in which only a small number of nodes know their location in advance (anchor nodes). Now we have to determine the location of the remaining nodes that do not know their location, especially in distributed manner. In our proposed scheme, one can think that a mobile ad-hoc network itself is an SOM network, in which each neuron is a node in that network, and these neurons are connected to their 1-hop neighboring nodes (nodes have direct radio links). The topological position and the weight of each neuron are associated with its estimated location. The learning process includes two phases. The first phase takes place locally at each node, where the input pattern is its estimated location (this input is dynamically changed over time except that the anchors use their known location) and neighborhood nodes are its 1-hop neighboring nodes. It is obvious that each node becomes the BMU at its local region. So when updating weights at the BMU, only its 1-hop neighbors' weights are updated. The BMU node also receives updates from other nodes when it becomes 1-hop neighbor of other nodes. In the second phase, if the network has some nodes know their location in advance (anchors), then each node will utilize the information from these anchors by adjusting its location towards the estimated absolute location based on the information from these heard anchors. At the end of the learning process, the weight at each node (SOM neuron) is its estimated location.

In wireless ad-hoc networks, making use of the resources of all nodes is a crucial problem for any application service, and localization is not an exception. We know that wireless ad-hoc networks have capability of self-organizing, so it is practical if nodes can do the localization themselves. From that point of view, we propose a Distributed Range-free Localization Algorithm Based on Self-Organizing Maps, which utilizes only connectivity information as well as information from some heard anchors. In an existing work[12], the authors modified the original SOM to solve the localization problem. Through out this paper, we call this existing method as SOM and our new proposed method as LS-SOM (Localization Scheme-Self-Organizing Maps) for the direct comparison. In comparison with this existing method, our proposed method modified the SOM updating function by utilizing the intersection area between neighboring nodes, thus makes the convergence more accurate and much faster.

3. Distributed GPS-free SOM based positioning algorithm

In this section, we will introduce about our proposed Distributed Range-free Localization Algorithm (LS-SOM). The following sections describe about initialization and learning stages of the main algorithm.

(1) Initialization stage

At a predefined interval, each anchor in the network broadcasts a packet to its neighboring nodes. This packet contains the anchor's location and a hop count initialized to one. When a node receives a packet contains anchor information, node then decides to discard or forward the packet to its neighboring nodes or not with the following rules.

a) If the packet is already in the cache, the node then compares the hop count of the packet with that of the cached packet. If the hop count of the arrival packet is less than that of the cached packet, then the cached packet is replaced with a new arrival packet, and forwarded to its neighboring nodes with hop count modified to add one hop. If the hop count of the arrival packet is greater than or equal to that of cached packet, then it is dropped.

b) If the packet is not in the cache, then it is added to the cache and forwarded to its neighboring nodes with hop count modified to add one hop.

Having information from some anchors, the nodes now initialize their location ready for SOM learning process. As proved by Mu-Chun Su et al.[23], the initialization is important in the convergence and preserving of the topology. In our proposed method, the initial location of a node is calculated based on either randomized value (if node does not receive enough three anchors) or a value calculated using a trilateral method. In this initialization stage, nodes also exchange information (using short "HELLO" message broadcast) so that each node has information about its neighboring nodes (1-hop neighbors). Nodes also exchange information about 1-hop neighbors with its neighboring nodes, so that all nodes in the network have information about both 1-hop and 2-hop neighboring nodes.

(2) Learning stage

Before going into our algorithm details, let us formulate the mathematical notations which will be used in this paper. We represent a wireless ad-hoc network as an undirected connected graph. The vertices are nodes' locations, and edges are the connectivity information (direct connection between neighboring nodes). The target wireless ad-hoc network is formed by G anchors with known locations Ω_i(i=1, 2, ..., G) and N nodes with unknown locations. The unknown nodes have actual locations denoted as ω_i (i=1, 2, ... , N) and estimated locations denoted as ϖ_i (i=1, 2,..., N).

a) Estimated location exchange

At this step, each node forwards its estimated location to all of its neighbors, so that it also knows the estimated location of its neighbors as $\varpi_{i,j}$ (j=1, 2, ..., N_i) with N_i is the number of nodes within its communication range.

b) Local update of relative location

We will now shape the topology at each region formed by the node with location ϖ_i together with all of its neighboring nodes. The node ϖ_i plays as the input vector and becomes the winning neuron for that region. Consequently, the neighboring nodes of ϖ_i will receive the updating vector from node ϖ_i.

Suppose that the node with the estimated location ϖ_i has N_i neighbors. The locations of these neighbors are denoted as $\varpi_{i,j}$ (j=1, 2, ..., N_i). Based on classical SOM, neighboring nodes of the node with location ϖ_i will update their weight with the following formula.

$$\varpi_{i,j}(m+1) = \varpi_{i,j}(m) + \Delta(m) \quad (1)$$

Where Δ(m) is calculated using (2).

$$\Delta(m) = \alpha(m).(\varpi_i(m) - \varpi_{i,j}(m)) \tag{2}$$

in which α(m) is the learning rate exponential decay function at iteration m-th defined in (3).

$$\alpha(m) = \exp\left(-\frac{m+1}{T}\right) \tag{3}$$

where m denotes the m-th time step of the total T learning steps.

But, updating by using (1) means that the neighboring nodes will move toward the location determined by ϖ_i. This will lead to the problem as showed in Fig.2. From Fig.2, the nodes with location ϖ_j and ϖ_k are the neighbors of the node with location ϖ_i, but ϖ_j is not the neighbor of ϖ_k. In the worst case, the estimated location of the node with location ϖ_j falls into the radio range of the node with location ϖ_k, then the node with location ϖ_j may not escape from that wrong location throughout the learning process (dead location). In this paper, we propose an algorithm to solve this problem as follows.

Suppose that at the node with location ϖ_i, we have to update location for the neighbor node with location $\varpi_{i,j}$ (j=1,...,N_i). First, we find out other $L_{i,j}$ neighbor node $\varpi_{i,j,k}$ (k=1,...,$L_{i,j}$) of the node with location ϖ_i that are not the neighbor of the node with location $\varpi_{i,j}$ (this is done easily because each node knows its neighbors' neighbors). Now we calculate the vector that has the direction towards the intersection area (the dashed area) in Fig.3. As illustrated in Fig.4, this vector is calculated using (4).

$$\xi_{i,j} = \frac{1}{L_{i,j}} \sum_{k=1}^{L_{i,j}} \frac{r - |\varpi_{i,j} - \varpi_{i,j,k}|}{|\varpi_{i,j} - \varpi_{i,j,k}|} (\varpi_{i,j} - \varpi_{i,j,k}) \tag{4}$$

where r denotes the maximum communicable range between ϖ_i and $\varpi_{i,j,k}$ (k=1,...,$L_{i,j}$). We use vector $\varpi_{i,j}$ as a guidance to update the location of the node with location $\varpi_{i,j}$ by changing (1) to (5).

$$\varpi_{i,j}(m+1) = \varpi_{i,j}(m) + \Delta(m) + |\Delta(m)|.\frac{\xi_{i,j}}{|\xi_{i,j}|}.\beta \tag{5}$$

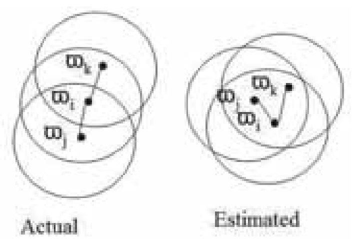

Fig.2 The case where node ϖ_j has wrong estimated location.

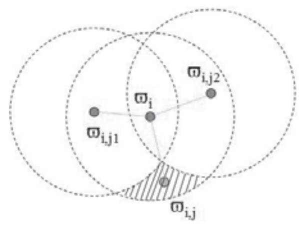

Fig.3 Possible location of neighboring node $\varpi_{i,j}$.

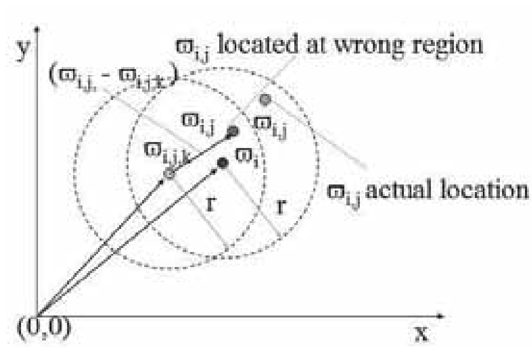

Fig.4 The case where neighboring node $\varpi_{i,j}$ located at wrong position.

The update by (5) makes each node move toward the intersection area as showed in Fig.3. This update also maximizes the correlation between the neighboring nodes that is the key problem for the speed and accuracy of topological convergence using SOM. In (5), β is a learning bias parameter calculated using (6).

$$\beta = \begin{cases} 0 & m <= \tau \\ 1 & m > \tau \end{cases} \tag{6}$$

with τ is a learning threshold. This threshold determines the step to apply this modification. Basically, we can apply this modification after several steps of SOM learning. At the end of this step, the node with location ϖ_i transmits its neighbor location updates based on (5) to all of its neighbors. As a result, it also receives the similar updates from its N_i neighboring nodes as $\varpi_{i,j}(j=1,...,N_i)$. Node with location ϖ_i now calculates its newly estimated location by averaging its current location and the updates from the neighboring nodes using (7).

$$\varpi_i = \frac{1}{N_i + 1}\left(\sum_{j=1}^{N_i} \varpi_{j,i} + \varpi_i\right) \tag{7}$$

4. Simulation Evaluations

To evaluate the performance of our proposed method, we use the average error ratio in comparison with the radio range of the nodes presented in (8).

$$Error(r) = \frac{1}{N}\sum_{i=1}^{N}\frac{|\varpi_i - \omega_i|}{r} \tag{8}$$

(1) Simulation parameters

We conducted the simulation for static scenarios by using our written Java program. In simulation, each experiment is done on thousands of randomly generated networks on an area of 1 by 1. The common parameters used in simulation are presented in Table 1.

TABLE I. SIMULATION PARAMETERS

Parameter	Value
Number of SOM learning steps T	100
Adjusting parameter a	1
Learning bias threshold τ	1

To ease the comparison, we call the method in the existing work[12] as SOM, and our proposed method as LS-SOM. We will evaluate our proposed method with static networks which consist of 100 nodes randomly generated with variances in connectivity and number of anchors.

(2) Simulation Results

With wireless ad-hoc networks, we study how the accuracy is influenced by the connectivity level (the average number of neighbor nodes that a node has direct communication with), and the number of anchor nodes deployed. Fig.5 shows the average error with different connectivity levels. The result indicates that LS-SOM achieves very good accuracy over the SOM and DV-HOP from sparse to dense networks. Especially with very sparse networks, LS-SOM still performs better than SOM. If we compare this result to the result of semi-definite programming (SDP)[17] and MDS-MAP, then we can see that at the connectivity level of 10 with 4 anchors, SDP and MDS-MAP have average error around 45%, and that of our method is just around 20%. The performance with the variance of anchors is showed in Fig.6. We find that LS-SOM increases accuracy when the number of anchors increases. When the number of anchors is 8, then the average error is only 15%. Fig.7 shows the average error through each SOM learning step. LS-SOM needs only 15 to 30 learning steps to achieve a stable result. Comparing to thousands of learning steps in the traditional SOM, LS-SOM decreases network overhead and computational cost. Fig.8(a) shows one of the actual topology that is generated during the simulation. Fig.8(b), Fig.8(c), and Fig.8(d) show the topologies estimated with DV-HOP, SOM and LS-SOM, respectively. In these figures, the rectangles and the circles denote the anchor nodes and the unknown nodes, respectively. From the figures, one can realize that LS-SOM outperforms the topology regeneration. It is resistant to the perimeter effect that other schemes encounter.

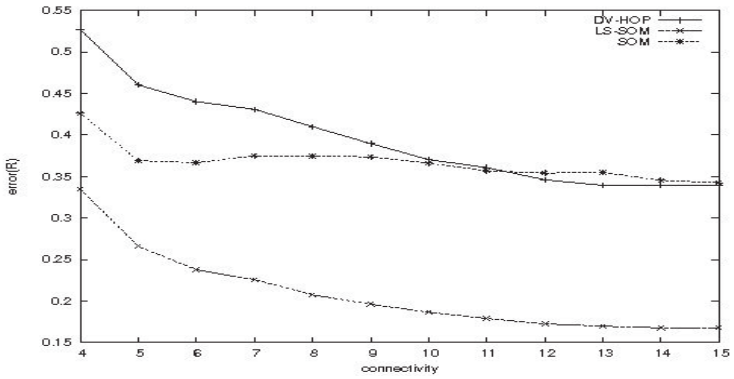

Fig.5 Performance by connectivity (N=100, G=4).

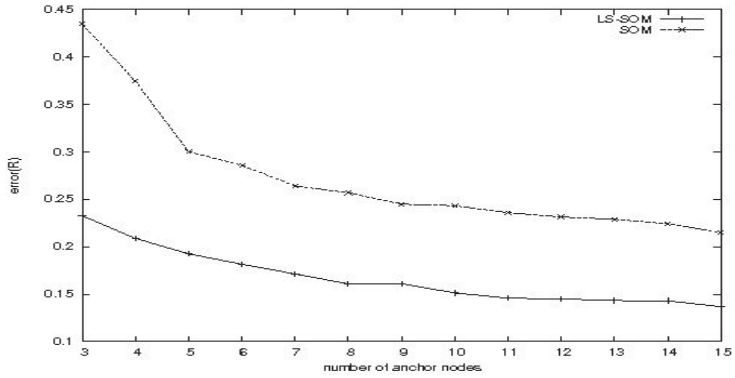

Fig.6 Performance by anchor ratio (N=100, connectivity=8).

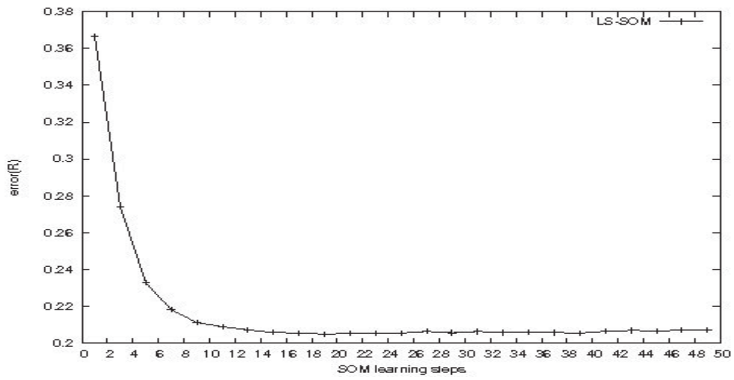

Fig.7 Performance by SOM learning steps.

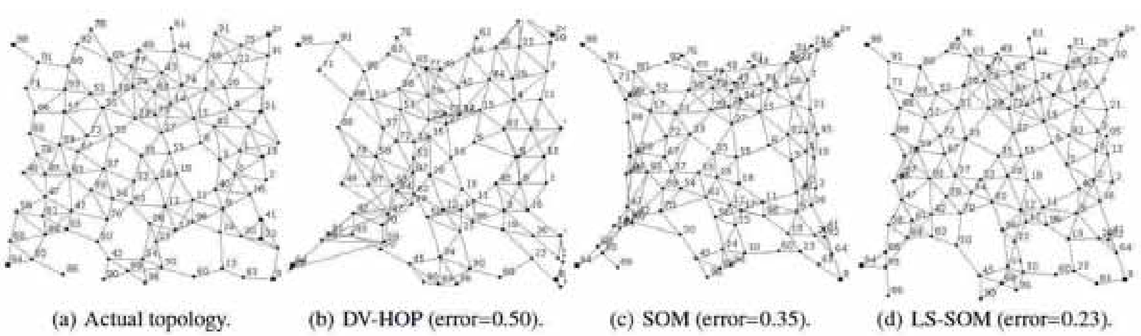

Fig.8 Topology regeneration (N=100, G=4, connectivity=4.88).

4. Conclusions

We have presented our Improved GPS-free Ad-hoc Network Positioning Algorithm Based on Self-Organizing Maps (LS-SOM) in this paper. By introducing the utilization of intersection areas between radio coverage of neighboring nodes, the algorithm maximizes the correlation between neighboring nodes in distributed SOM implementation. From intensive simulations, the results show that LS-SOM has achieved good accuracy over the original SOM and other algorithms. The method is free from GPS and needs only connectivity information (no special hardware is needed) thus makes it low cost and energy saving. With our proposed solution, LS-SOM is capable of monitoring the positions of many cultural heritage objects simultaneously and efficiently even in emergency cases like disasters. Future work will investigate in a more precise distance measurement method and integrate LS-SOM with other protocols to make LS-SOM to be more flexible.

References

1. Pham Doan Tinh, Taku Noguchi, Makoto Kawai. *Ad-hoc network positioning system for large scale disaster mitigation*. Proc. of Urban Cultural Heritage Disaster Mitigation, Vol. 2,pp. 49-54, 2008.
2. Poovendran, R., Wang, C.L. and Roy, S.: *Secure Localization and Time Synchronization for Wireless Sensor and Ad Hoc Networks*, Springer, 2007.
3. Wellenhoff,B. H., Lichtenegger,H. and Collins, J.: *Global Positioning System: Theory and Practice*, Fourth Edition, Springer Verlag, 1997.
4. Savvides, A., Han, C. C. and Strivastava, M.B.: *Dynamic fine-grained localization in ad-hoc networks of sensors*, Proc. The 7th annual international conference on Mobile computing and networking (MobiCom), pp. 166-179, 2001.
5. Niculescu, D. and Nath, B.: *Ad hoc positioning system (APS) using AoA*, Proc. INFOCOM, pp. 1734-1743, 2003.
6. Bischoff, U., Strohbach, M., Hazas,M. and Kortuem, G.: *Constraint-based distance estimation in ad-hoc wireless sensor networks*, Proc. Third European Workshop on Wireless Sensor Networks (EWSN),pp. 54-68, 2006.
7. Patwari, N., Hero III, A., Perkins, M., Correal, N. and O'Dea, R.: *Relative Localization Estimation in Wireless Sensor Networks*, IEEE Trans. Signal Processing, Vol. 51, No. 8, pp. 2137-2148, 2003.
8. Nicolescu, D. and Nath, B.: *Ad-Hoc Positioning Systems (APS)*, Proc. IEEE GLOBECOM, 2001.
9. Shang, Y., Ruml, W., Zhang, Y. and Fromherz, M.: *Localization from mere connectivity*, Proc. 4th ACM international Symposium on Mobile ad hoc networking \& computing, pp. 201-212, 2003.
10. Tran, D.A. and Nguyen, T.: *Localization In Wireless Sensor Networks Based on Support Vector Machines*, IEEE Trans. Parallel and Distributed Systems, Vol. 19, No. 7, pp. 981-994, 2008.
11. Giorgetti, G., Gupta, S.K.S. and Manes, G.: *Wireless localization using self-organizing maps*, Proc. IPSN'07, 2007.
12. Asakura, S., Umehara, D. and Kawai, M.: *Distributed Location Estimation Method for Mobile Terminals Based on SOM Algorithm*, IEICE Trans. Commun, Vol. J85-B, No. 7,pp. 1042-1050, 2002.
13. Hu, J. and Lee, G.: *Distributed localization of wireless sensor networks using self-organizing maps*, Proc. Multisensor Fusion and Integration for Intelligent Systems, pp. 284-289, 2008.
14. Sum, J., Leung, C.S., Chan, L.W. and Xu, L.: *Yet another algorithm which can generate topograph map*, IEEE Trans. Neural Networks, Vol. 8, pp. 1204-1207, 1997.
15. Janet, J., Gutierrez, R., Chase, T., White, M. and Sutton, J.: *Autonomous mobile robot global self-localization using Kohonen and region-feature neural networks*, Journal of Robotic Systems, Vol. 14, pp. 263–282, 1997.
16. Ertin, E. and Priddy, K.: *Self-localization of wireless sensor networks using self-organizing maps*, Proc. SPIE'05, Vol. 5803, pp. 138-145, 2005.
17. Shi, Q. and He, C.: *A SDP Approach for Range-Free Localization in Wireless Sensor Networks*, Proc. IEEE ICC'08, 2008.
18. Tinh, P.D., Noguchi, T., Kawai, M.: *Localization scheme for large scale wireless sensor networks*, Proc. ISSNIP 2008, pp.25-30, 2008.