Mobile User Location Determination using Extreme Learning Machine

Teddy Mantoro[1], Akeem Olowolayemo[2], Sunday Olusanya Olatunji.[3]
[1]Dept of Computer Science, KICT, International Islamic University Malaysia, Kuala Lumpur, Malaysia
[2]Dept of Information Systems, KICT, International Islamic University Malaysia, Kuala Lumpur, Malaysia
[3]Dept of Information Systems, FICT, Universiti Teknologi Malaysia (UTM), Johor, Malaysia

***Abstract:-* There has been a rapid convergence to location based services for better resources management. This is made possible by rapid development and lower cost of mobile and handheld devices. Due to this widespread usage however, localization and positioning systems, especially indoor, have become increasingly important for resources management. This requires information devices to have context awareness and determination of current location of the users to adequately respond to the need at the time. There have been various approaches to location positioning to further improve mobile user location accuracy. In this work, we examine the location determination techniques by attempting to determine the location of mobile users taking advantage of signal strength (SS) and signal quality (SQ) history data and modeling the locations using extreme learning machine algorithm (ELM). The empirical results show that the proposed model based on the extreme learning algorithm outperforms k-Nearest Neighbor approaches.**

Categories and Subject Descriptors
H.5.2 [User Interfaces]

Keywords
Location Awareness, Artificial Neural Network, Extreme Learning Machines.

I. Introduction

The prime goal of pervasive computing is the idea of Information Technology (IT) services every time and everywhere. This makes IT enabled services available and responsive to user's needs in mobile environment taking advantages of their location and context information. This signifies that location and context aware computing has become very important concept that is contributive and very crucial to mobility services management.

There have been wide area of applications of context computing and location awareness and there is an increase demand and research for its applications in various new areas. Some of the application areas include resources optimization (e.g. power, bandwidth, etc), fleet tracking and shipping applications, security - tracking criminals by law enforcement officials, automotive applications, advertisement and marketing, aged and disabled support systems and a host of all others.

Through context computing, IT infrastructures and services can now be made available and better utilized by adapting to users' needs and thus facilitating enhanced resources management. Positioning systems accuracies can be tailored into three levels of orientation. These orientations are point, line and area-orientation [1]. Location positioning systems' accuracies is an issue of great concern especially when the positioning has to do with point-orientation, such as those services meant for disabled and aged citizens who require assistance. In such cases, the accuracy must be tailored to the "exact" position of the person concerned. However, in other orientations such as line or area-orientations, there is still need for improvement to give better confidence and better management of resources. A serious challenge in the case of wrong positioning of fleet management is good example [1]. Lately, location fingerprinting has been adjudged as better suited for indoor positioning. However, in location fingerprinting positioning, the appropriate selection of location determination algorithms and models for context enabled services has been very crucial for accuracy and enhancement of positioning systems and applications, which is one of the keys to future mobile infrastructures and services management and security . Various indoor approaches such as those used in location fingerprinting give some advantages over device-based method that uses GPS. This is due to two important reasons. One, GPS performs inadequately in indoor environment due to obstruction to signal and non line of sight, which lead to attenuation and poor signal quality while location fingerprinting on the other hand is not affected by non-line of sight. The other issue in devise-based positioning is that there is an added overhead of computation power as a result of processing of location information on the device while at the same time being used for network [1,9].

Prediction accuracies are usually affected by the techniques and devices used as well as the algorithms applied [9]. The appropriate management of these components of context determination is more often than not the major problem areas in location positioning systems.

This work is an attempt to further device a better positioning accuracy based on location fingerprinting taken advantage of two important mobile fingerprints, namely signal strength and signal quality and subsequently building a model based on extreme learning machine, a new learning algorithm for single-hidden-layer neural networks (SLFNs) , described in [3].The remaining part of this article is organized as follows. The next section gives a review of related work on key concepts in location and context awareness, location fingerprinting and neural networks (NN). Section III discusses the system's design techniques and the last section; conclusion evaluates the results of the study as well as the challenges and future directions.

II. Related Work

Location determination is the most important component of context awareness. There have been tremendous attempts previously to better approximate location of mobile users.

These are due largely to the ever growing need for better positioning for improved location based service management. Different approaches have been used and proposed in literatures. One such approach that has gained more recognition in the literature is location fingerprinting. This is to position users based on differential signal attributes at different location rather than computing the distance between the signal transmitting points, usually the access point, and mobile device terminal peculiar to other network-based approaches. The received signal strength is then compared to location history data stored in a radio map. Efforts at using location fingerprint to determine mobile user locations have been explored in several previous works such as in [2, 9, 10, 13]. Specifically in [3], a comparison of models neural networks using signal strength was developed. The neural networks models were based on the simulated-annealing (SABPN) and feedforward neural networks with back-propagation (BPN), showing better performance results of the latter over that of the models.

The approaches involve the applications of mobile positioning techniques, adequate and require up-to-date devices as well as appropriateness of algorithms in ways that will carefully analyze the existing techniques and methods to be able to develop sound strategies to further enhance context aware applications in the subsequent section each of the components of positioning systems are further elaborated.

A. Techniques

Positioning techniques can be classified into three; network-based, device-based and hybrid methods, comprising both first two approaches [1]. **Network-based** positioning often regarded as integrated approach since the network is used for communication and data transmission to other networks as well as determination of user location. In network-based positioning, user locations are determined by estimating the signal travelling to and from a set of transmitter base stations. The position is thereby measured through computation of the length and direction of the radio path of a mobile device from these base stations [1]. Network-based positioning methods do not require extra software or hardware to be installed in the mobile device since the network is used for the computation. Power is also conserved unlike in device-based methods that require high power for computation. There are many network-based positioning techniques that are used in indoor users' location techniques for measuring signals at a particular location to an Access Point. These include Cell of Origin (the earliest method of locating a wireless user by determining the transmitting range where the call is made), Angle of Arrival (AOA) – locating a user's location by the overlapping of two cell where the cell phone receive signal, Enhanced Observed Time Difference (E-OTD) – determining user's position by measuring the time travel between the phone and multiple towers, Time of Arrival (TOA), Time Difference of Arrival (TDOA) – locating a user by determining the time it takes the signal from the cell-phone to reach the transmitting tower and Location Fingerprinting, that is, recalling patterns (such as multipath) which mobile phone signals are known to exhibit at different locations in each cell. Out of these methods, Location Fingerprinting is considered most appropriate. The other techniques are affected by non-line-of-sight and multi-path in indoors space [9]. This leads to errors and inaccuracies in the training data-sets captured.

However, for outdoor users' location determination Global positioning System (GPS) is used. GPS uses satellites to track a user's latitude, longitude and altitude, it can be used virtually everywhere in the world, including on airplanes and ship [1]. The main drawback of GPS is that it is not effective for indoor location determination. Recently, A-GPS, that is, a combination of GPS with one or more of network-based positioning approaches, such as Cell-ID, TDOA, AOA, has been used as hybrid approaches to work around this shortcoming of GPS[1].

B. Algorithms

Furthermore, there are different algorithms that could be used to determine the user location. The major component of context awareness system that determines the accuracy of the predictions is the appropriateness and adequacy of the algorithm used. There are a number of major algorithms that have been adopted or presented in context awareness literatures. These include Markov models, Hidden Markov Models, Bayesian Networks, Neural Networks or Time Series. Several previous works location position has been estimated using a number of algorithms and techniques. In [11], the algorithms were classified as deterministic and probabilistic. Under deterministic approach, which is based on the traditional Euclidean distance approximation method, approaches with different norms were considered.

C. Proposed Methodology

The technique proposed in this work is to estimate users' location based location fingerprinting specifically collecting mobile devices location data, namely, the signal strength (SS) and signal quality (SQ), as first proposed in [4], based on proximity to a number of Access points (APs). Subsequently, a model based on extreme learning machine modeling is used to estimate the location of the mobile user.

D. Artificial Neural Network

The prediction technique used is discussed in this section. This involve the development and application of extreme learning machine , a new artificial neural network (ANN) framework derived from single-hidden-layer neural networks (SLFNs) to determination of location of mobile users based on the signal strength (SS) and signal quality (SQ) peculiar to the particular location. Artificial Neural Network is a mathematical model that tries to stimulate the functions of the brain [10]. Similar to the human brain, the neural network is composed of an interconnected assembly of nodes and directed links. It is known that the human brain learns by changing the strength of the synaptic connection between neurons upon repeated stimulation by the same impulse [12]. A generic structure of simple artificial neural network is found in Figure 1.

The aim is to compute the value of the computed output y close enough to the target output $f(x)$.

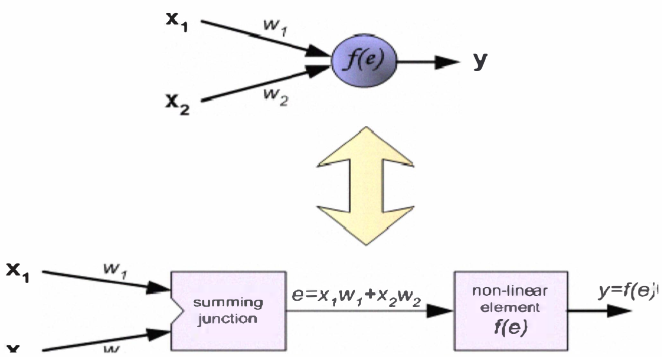


Fig. 1. A generic model of Neural Network [11].

The artificial neural network is defined as described thus.

Given a finite input nodes $x = x_i \ldots . x_k$, with corresponding weight vector, , $w = w_{i..} w_k$, the network's response, y is given by

$$y = g\left(\sum_k w_k . x_k\right) \in (0,1)$$

adjusting the weights sufficiently to estimate y, after a finite number of steps, such that

$$y \approx f(x)$$

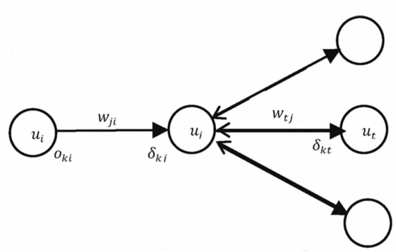
by minimizing the difference between the network's response, y and the desired output, f(x). The termination condition is achieved after a specified number of iterations, usually referred to as *epoch* is reached. The performance of artificial neural network is often specified in form of root mean square error, rms. The network is known to usually converge. However, an optimal convergence is not guaranteed.

In order to accelerate convergence in multilayered Feed-forward neural networks, Back-propagation is often used. The goal is to learn the weights thereby minimizing the mean squared error. This implies that we first compute

$$o_{kj} = g_j\left(\sum_i w_{ji} . o_{ki}\right)$$

for each hidden and output unit.

Subsequently, for each node u_j , the accumulated error δ_{kj}



$$\delta_{kj} = (t_{kj} - o_{kj})g'_j\left(\sum_i w_{ji} . o_{ki}\right)$$

is computed. The computed δ becomes the input of the reversed network, to adjust the weights vector, w. This is essentially the back-propagation. Thus we compute for each u_j

$$\delta_{kj} = f'_j\left(\sum_i w_{ji} . o_{ki}\right)\sum_t \delta_{kt} w_{tj}^{k-1}$$

and the weight update is then computed for each $w_{ji}^k \in$ w.

$$w_{ji}^k = w_{tj}^{k-1} + \mu \delta_{kj} o_{ki}$$

The multilayered feed forward neural network (MFN) with back-propagation is known to be far slower and a new learning algorithm based on the single-hidden layer feedforward neural networks, named extreme learning machine has been proposed and found to outperform the *k*-Nearest Neighbour previously used on the same dataset of signal strength and signal quality [1]. A brief description of extreme learning machine is given in the subsequent section.

III. EXTREME LEARNING MACHINES

The Extreme Learning Machine modelling scheme, a new framework based on the traditional feedforward neural network proposed recently, is composed of a single hidden layer. It randomly chooses the input weights and analytically determines the output weights of the single layer feedforward neural network [3]. Previous attempts have compared its performance with other variants of the traditional neural networks as well as other algorithms such as support vector machine (SVM), fuzzy neural network (FNN), fuzzy regression (FR), multiple linear regression (MLR), generalized regression neural network (GRNN) and other in different domain prediction.

Previous study of the learning rate of feed-forward neural networks (FFNN) have shown that it is time-consuming which impacted FFNN scalability thereby demanding new frameworks to formulate faster and better performing networks [1]. According to [1], there are two main reasons behind this behavior, one is the slow gradient based learning algorithms used to train neural network (NN) and the other is the iterative tuning of the parameters of the networks by these learning algorithms.

To overcome these problems [1, 3], proposed a learning algorithm called extreme learning machine (ELM) for single hidden layer feed-forward neural networks (SLFNs) which randomly selected the input weights and analytically determines the output weights of SLFNs. The authors in [5] stated that "In theory, this algorithm tends to provide the best generalization performance at extremely fast learning speed". This is extremely good because in the past, it seems that there exists an unbreakable virtual speed barrier which classic learning algorithms cannot break through and therefore feed-forward neural network implementing them take a very long time to train itself, independent of the application type, whether simple or complex.

The ELM has several interesting and significant features different from traditional popular gradient-based learning algorithms for feed forward neural networks: These include:

- The learning speed of ELM is extremely fast. In simulations reported in literatures, [5-6], the learning phase of ELM can be completed in seconds or less than seconds for many applications
- The ELM has better generalization performance than the gradient-based learning such as back propagation in most cases.
- The traditional classic gradient-based learning algorithms and some other learning algorithms may face several issues like local minima, improper learning rate, over fitting, etc. In order to avoid these issues, some methods such as weight decay and early stopping methods may need to be used often in these classical learning algorithms. The ELM tends to reach the solutions straightforward without such trivial issues. The ELM learning algorithm looks much simpler than most learning algorithms for feed-forward neural networks.

Unlike the traditional classic gradient-based learning algorithms which only work for differentiable activation functions, as easily observed, the ELM learning algorithm could be used to train SLFNs with many non-differentiable activation functions [5, 7].

A. How Extreme Learning Machine Algorithm Works

To appreciate ELM, a description of the standard SLFN (single-hidden layer feed-forward neural networks) is necessary. Given an N samples (x_i, t_i), where $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in R^n$ and $t_i = [t_{i1}, t_{i2}, \dots, t_{im}]^T \in R^n$,

The standard SLFN with $\tilde{N}$ hidden neurons and activation function g(x) is defined as:

$$\sum_{i=1}^{\tilde{N}} \beta_i g(w_i . x_j + b_i) = o_j, j = 1, \dots, N,$$

where $w_i = [w_{i1}, w_{i2}, \dots, w_{in}]^T$ gives the weight vector that connects the i^{th} hidden neuron and the input neurons, $\beta_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{im}]^T$ is the weight vector that connects the i^{th} neuron and the output neurons, and b_i is the threshold of the i^{th} hidden neuron. The "." in $w_i . x_j$ means the inner product of w_i and x_j.

In SLFN, just like the traditional feedforward neural network with back-propagation, the aim is to minimize the difference between, network response, o_j and the target output, t_j. That is,

$$\sum_{i=1}^{\tilde{N}} \beta_i g(w_i . x_j + b_i) = t_j, j = 1, \dots, N$$

Or, more compactly, as:

$$H\beta = T$$

where H can be represented as

$$\mathbf{H}(w_1, \dots, w_{\tilde{N}}, b_1, \dots, b_{\tilde{N}}, x_1, \dots, x_N) = \begin{bmatrix} g(w_1 . x_1 + b_1) & \dots & g(w_{\tilde{N}} . x_{\tilde{N}} + b_{\tilde{N}}) \\ . & & . \\ . & \dots & . \\ . & & . \\ g(w_1 . x_N + b_1) & \dots & g(w_{\tilde{N}} . x_N + b_{\tilde{N}}) \end{bmatrix}_{N \times \tilde{N}},$$

$$\beta = \begin{bmatrix} \beta_1^T \\ . \\ . \\ . \\ \beta_{\tilde{N}}^T \end{bmatrix}_{\tilde{N} \times m}, \quad \text{and } T = \begin{bmatrix} T_1^T \\ . \\ . \\ . \\ T_{\tilde{N}}^T \end{bmatrix}_{N \times m}$$

According to [8], H is referred to as the neural network output matrix. The ELM algorithm can be described as follows [3]:

Given a training set

$$N = \{(x_i, t_i) \mid x_i \in \mathbf{R}^n, t_i \in \mathbf{R}^m, i = 1, \dots, N\},$$

activation function *g(x)*, and hidden neuron number = $\tilde{N}$, do the following:

1. Assign random value to the input weight w_i and the bias b_i, $i = 1, \dots, \tilde{N}$
2. Find the hidden layer output matrix H.
3. Find the output weight β as follows:

$$\beta = H^{\dagger} T$$

where β, H and T are defined in the same way they were defined in the SLFN specification above.

IV. Experiments

A. Experimental data

The data was collected using Multi-Observers Method [10] by collecting the data in a particular location a number of times, specifically eight times for all experimental locations. The calibration was such that the distance between subsequent measurements is 1m to get yet a more accurate result. We collected values of multiple observed signal strength and quality in a database, such that the extreme learning machine algorithm can then estimate the signal strength and signal quality as a data instance from a single point/location which is then compared it to the training data-set stored in our database, thereby able to determine the distance value between training data and data sample. The location positioning model is as presented in Figure 2.

Subsequently, the extreme learning machine algorithm deduces the closest member based on the training data-set by classifying the closest member based on the the specified algorithm number of closeness. This results in the most user location that belongs to the member.

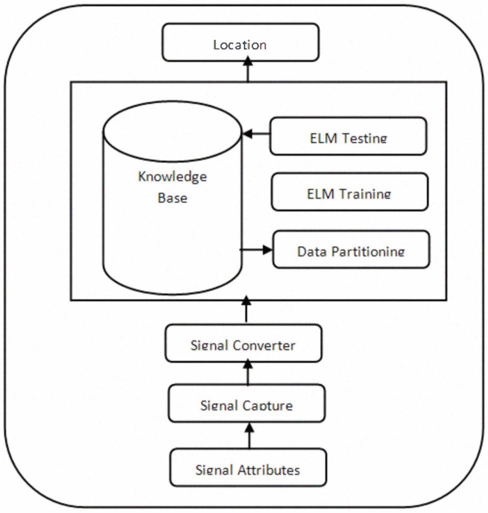


Fig. 2. The Location Positioning Model

The extreme learning machine algorithm was built in the following stages.

i. The Multi-observers training data was partitioned into training and testing data.
ii. Data preparation was carried out so that the data is well formatted and structured to conform to the requirements of the network.
iii. The network was trained with different parameters, 80% Training data with 20% test data and 85% training data and 15% test data to compare the performances of each model and thereby able to select the better model after validation. To select the most efficient network based on mean square error (MSE) and network simplicity, requiring less computation and memory, in the experiment, we differentiated based on only signal strength comparing with the results of only signal quality and later compare the results obtained based on these models with the one that used both predictors, signal strength and signal quality as data set.

An experimental procedure based on test-set-cross-validation was employed in our study. We used the stratified sampling approach to divide the data set into both training and testing data, such that the size of the training set is 80% of the available data and the testing is the rest in the first case, and 85% training data in the 2nd case.

B. *Performance Indicator*

Percentage of correctly classified samples These criteria also provide a more accurate evaluation of the classifiers.

C. *Simulation Setting*

In this work we carried out three separate forms of classification on the same dataset. Viz. (i) classification using the first six predictor variables (tagged SS1-SS8) (ii)) classification using the last six predictor variables (tagged SQ1-SQ8) (ii)) classification using all the 12 predictor variables (tagged SS1-SS8 and SQ1-SQ8).

For each of the three different cases, we divided the dataset into both training and testing set using the stratifying approach so that the data are randomly divided for effective representation. In all, six experiments were recorded as adequate representation among the different models.

V. Results and Discussion

We present here the results of the classification using 85% for training and 15% for testing, then followed by the classification using 80% for training and 20% for testing.

TABLE 1: RESULTS OF USING 85% FOR TRAINING & 15% FOR TESTING

	SS1-SS8 predictors		SQ1-SQ8 predictors		All predictors variable	
	Tr	Ts	Tr	Ts	Tr	Ts
ELM	94.72	92.49	93.89	92.96	94.22	95.31

From the results in Table 1 and 2, it was shown that the model outperforms the model based on k-Nearest Neighbor [9]. The best result from all the parameter variations was 93.26% in the k-Nearest Neighbor model using 80% training and 20% test data sets.

TABLE 2: RESULTS OF USING 80% FOR TRAINING & 20% FOR TESTING

	SS1-SS8 predictors		SQ1-SQ8 predictors		All predictors variable	
	Tr	Ts	Tr	Ts	Tr	Ts
ELM	95.09	91.55	93.95	93.31	94.47	93.66

In this model, using all predictors, both signal strength and signal quality, the model gives above 94% for training in both parameter variations of 80% to 20% training and test data respectively and also 85% to 15% variation. Using the later variation of 85% training data to 15% for test data was shown to have given the best result, overall, a little above 95%. Generally for the training data, 80% training data for signal strength alone, without the signal quality, (SS1-SS8) gives the best performance of slightly above 95%. This further confirms that the signal strength attribute is more predictive of location than signal quality. This can also be seen in the 85% to 15%, where the signal strength also outperforms the results from signal quality. However, the effect of combining with signal quality is obvious when compared with the all predictors' performance of above 94% for training and above 95% for test performance. It can be said also that the average performance for both training and test data is around 94% using all predictors. This is a more remarkable performance than the previous work [10].

VI. Conclusion

In this study, the following conclusions and recommendations could be drawn based on previous analysis, discussions, deep investigation, experiments, and comparative studies in the work.

A new computational intelligence modeling scheme, based on the Extreme learning machine has been investigated, developed and implemented, as an efficient and more accurate predictive solution for determining position of mobile users based on location fingerprint data (signal strength and signal quality). The new framework based on extreme learning machine has been compared with the k-Nearest Neighbor presented earlier in [10]. The empirical results have shown that the proposed model based on the extreme learning algorithm outperforms k-Nearest Neighbor approaches.

Further work is underway to compare the proposed model with other variants of Neural Networks such as back-propagation, simulated-annealing-based and further improve the accuracy with Kalman Filter.

REFERENCES

[1] Brimicombe, A and Li, C *Location-Based Services and Geo-Information Engineering*, UK: Wiley-Blackwell, 2009.

[2] Tsai, C.Y., Chou, S. Y. Lin, S. W. & Wang, W. H.. Location Determination of mobile Device for Indoor WLAN Application using Neural Networks. In Journal for Knowledge and Information Systems, Springer London, 2009.

[3] Huang, G.B., Q.Y. Zhu, and C.K. Siew, *Extreme learning machine: a new learning scheme of feedforward neural networks*, in *Proceedings of International Joint Conference on Neural Networks (IJCNN2004)*. 2004: Budapest, Hungary.

[4] Huang, G.B., Q.Y. Zhu, and C.K. Siew, *Extreme learning machine: Theory and applications 70 (2006) 489–501.* Neurocomputing, ELSEVIER, 2006. **70**: p. 489-501.

[5] Huang, G.B., Q.Y. Zhu, and C.K. Siew, *"Can threshold networks be trained directly?"* IEEE Trans. Circuits Systems II, 2006. **53**(3): p. 187-191.

[6] Olatunji, S.O., S. Ali, and A.-R. Abdul Azeez. *Modeling Permeability Prediction Using Extreme Learning Machines.* in Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation, AMS2010. 2010. Kota-Kinabalu, Malaysia.

[7] Mao, Y., F. Xuqian, and L. Xue, *A Class of Self-Stabilizing MCA Learning Algorithms.* IEEE Transactions on Neural Networks, 2006. **17**(6): pp. 1634-1638.

[8] Guang-Bin, H. and A.B. Haroon, *Upper bounds on the number of hidden neurons in feedforward networks with arbitrary bounded nonlinear activation functions.* IEEE Transactions on Neural Networks, 1998. **9**(1): p. 224-229.

[9] Mantoro, T and Olowolayemo, A(2009) *Mobile User Location Prediction Enabled Services in Ubiquitous Computing: Determining User Location Using MFN Algorithm*

[10] Mantoro, T., Azizan, A. Khairuzzaman, S. and M. A. Ayu *Multi-observers Instance-Based Learning Approach for Indoor Symbolic User Location Determination Using IEEE 802.11 Signals.* IEEE Symposium on Industrial Electronics and Applications (ISIEA 2009), Kuala Lumpur, Malaysia. 2009.

[11] Rumelhart, D.E., Hinton, G.E., & Williams, R.J. *Learning internal representations by error propagation.* In D.E. Rumelhart & J.L. McClelland (Eds.), Parallel distributed processing: Explorations in the microstructure of cognition (Vol. 1). Cambridge, MA: MIT Press, 1986.

[12] Bernacki, M. & Wlodarczyk, P. *Neural networks and artificial inteligence in electronics".* 2005. Retrieved on August 31, 2010 from http://www.cs.unibo.it/~roffilli/pub/Backpropagation.pdf.

[13] K. Kaemarungsi, "Distribution of WLAN received signal strength indication for indoor location determination," National Electronics and Computer Technology Center, Thailand, Technical Report. 2006.