

Location Estimation for Personalization based on Received Signal Strength from AP causing the GPS Signal Attenuation in Building

Seong-Jin Cho^{1,1} and Ho-Kyun Park^{1*}

¹ School of IT Convergence Engineering, Shinhan University, Republic of Korea
chosjpaul@gmail.com, hkpark@shinhan.ac.kr

Abstract

Location estimation (LE) is important in Ubiquitous Computing Environments (UCE). For positioning application, Global Positioning System (GPS) and Received Signal Strength (RSS) from Access Pointer (AP) using LE is increasingly popular choice especially after pervasive adoption of IEEE 802.11 Wireless Sensor Networks (WSN). Fundamental requirement of such LE is to estimate location from RSS at a particular location. Multi-Path Fading Effects (MPFE) make RSS to fluctuate in unpredictable manner, introducing uncertainty in LE. Moreover, in practical situations, RSS values RF Signal (RFS) from AP are not available at some locations all the time making the problem more difficult. To deal with this problem, machine learning techniques have been applied so that the carried along devices can learn and make decision where they are in the building, especially. Recent machine learning techniques remain many unsolved problem such as high cost of computation, high complexity of model structures and scalability. In this paper, we will introduce a few methods which give high accuracy and overcome other methods' disadvantages, such as Support Vector Machine (SVM), Push Pull Estimation (PPE), and Modular Multi-Layer Perceptron (MMLP) with Neural Networks (NN) using RFS from AP.

Keywords: LE, RSS, GPS, SVM, MMLP, PPE, WSN, MPFE, NN, UCE, AP, RFS

1. Introduction

1.1. Location Estimation(LE)

In the era of information, people try to gather more and more information in any way which is as fast as possible. The more information we can get, the more efficiency are our decision in terms of time, energy, profit, etc. This is the reason for the births of Mobile Devices, Internet, and lately, the birth of WSN. What we desire is to respond with what happens around in the way we want. However, if the location information is not available, all the collected information become meaningless since no interaction can be made when something happens. To name some, the applications can be detecting and recognizing moving objects, triggering missiles to detected devices, giving alert to others if there is a fire some where, monitoring and support forgetful patients, etc.

1.2. Interest for LE

The best accuracy that GPS can give is at 3 meters for military purpose and 15 meters for common living purposes for the positioning using GPS. For the positioning application in GPS signal attenuation, that accuracy is not acceptable while the requirement for accurately

* Corresponding author

positioning a person is necessary because a room may be very small with several meters in sizes.

In addition, GPS signal attenuation is seriously because of the construction materials, thus GPS results in even poorer accuracy when the devices we want to localize are in door. The attenuation is not the only reason, multi-path fading also make the signal even weak and unstable. Therefore, other methods for the LE in GPS signal attenuation are necessary.

For the development and the requirement of higher quality of life, many Healthcare systems which are based on WSN have been developed. Then the locations of patients and related people such as their doctors, physicians, nurses and even relatives are needed in both emergency cases and daily monitoring purposes. The requirement of accuracy is critical because several meters of error can cause serious error if a person is detected to be in a living room instead of to be actually in the bedroom. This kind of serious error can't be tolerated for supporting living especially for healthcare system.

1.3. Received Signal Strength (RSS)

There may be some other approaches that can provide information of a device's location. The input data can be image, video stream, ultra sound signal, radio frequency signals, *etc.*, However, RFS is the most convenient scheme to be used for personal monitoring because of following reasons:

Current recognition techniques based on image and video stream are still limited in terms of accuracy. Identifying someone requires lots of face image data and complicated learning face. That is difficult for the scenarios when a person entering a building without looking at a camera closely, not to mention the low speed and low quality of digital cameras.

Meanwhile, a mobile device is usually determined with identification, which is unique and related too much information of the holder. When trying to localize many people in a crowded area, other choices of input channels like images, videos, sounds become limited. Images and videos have the blockages caused by people and it is difficult to separate to recognize if the object image is of a human or not, let alone the problem of recognize who is who. LE methods based on ultra-sound signal usually utilize the information of Time of Arrival (TOA) or Time of Different Arrivals (TODA). Moreover, RFS has an important property, which is the ability of going through most of objects.

For wireless devices, RFS is now available with the pervasive adoption of IEEE 802.11 (a/b/g) WSN (LAN, Wi-Fi). Since the measurements of RSS must be reported by the wireless network interface card as the part of standard compliance, LE based on Personalization using the RSS of WSN is feasible. WiFi RSS based on location awareness applications include, but are not limited to, a wide range of services to the end user like automatic call forwarding to user's location, robotic global LE, exploration and navigation tasks, Finder, guiding and escorting systems, first hop communication partners, liaison applications, location based on the advertisement and positioning of entities in large warehouses.

1.4. Machine Learning Techniques

RFS has many advantages to be chosen for the scheme of collecting input data. However, it has the properties of electromagnetic waves: although RFS has the ability to go through most of objects, but suffer from different attenuation rates depending on objects material. Moreover, RFS also reflects when hitting a smooth surface, causing multi-path propagations besides the line-of-sight to the receiver as illustrated in Figure 1-(a).

One more thing, the RFS emitters never radiates the signal with the same energy at every direction as illustrated in Figure 1-(b). The fluctuation does not only happen in the change of

the angle view to the emitters but also in the every direction over time. Those phenomena make the strength of RFS fluctuate in high range. That means the measurements are not stable especially for the settings in GPS signal attenuation where there are walls and objects.

Even when the position does not change, measurements of the strength of RFS are unstable and fluctuate in the wide range. Not to mention when the mobile devices are rotated in some way, or change the positions a little bit. Since the measurements depend much on the building structures and the errors suffer from uncertainty properties, the best way to recognize a location within the setting in GPS signal attenuation is to use machine learning techniques to learn relations between the locations and the ranges of RFS strength values. After training phase, the knowledge can be stored in the mobile devices and the location can be recognized by the knowledge of those relations.

Those are the reasons of the development of many machine learning techniques which can be named here: GPS like triangulation LE, K-nearest neighbors (KNN) LE, Statistical learning theory LE, Bayesian classification and filtering LE, Fuzzy Logic LE, NN's LE, Kalman Filtering LE, *etc.*

2. Methods of Research

2.1. LE based on MMLP

Let's take a look at the Artificial Neural Network (ANN) structure. For an overview of understanding on NN's nature and behaviors, it should be emphasized that a typical NN has three layers: input layer, hidden layer and output layer. Input data is driven by the weights w_{ij} , and then they are summed up before being fed into each cell of the hidden layer, then again the output data of the hidden layer is driven by the weights w_{jk} to reveal the output data. Hidden layer has a set of "neurons" which do transformations or mappings from the input to the output of the "neurons". The function for mappings is usually a continuous and derivable function. The common choice for this mapping function is:

$$y = \sigma(x) = \frac{1}{1 + \exp(-x)}$$

If we just set aside all the procedure of calibrating, filtering (filter noise), standardizing (translate input data into the range [-1, 1] before put it to the NN) and translating output (output data is usually coded from labels into the ranges of value and decoded in the contrariwise way), the key idea can be expressed like this. Given input data and coded output data (all is call training data set), the training phase tries to update the weight set w_{ij} and the weight set w_{jk} so that the errors between calculated output and given output are as small as possible. Given a trained network with assigned weights, an input will be translated into coded output. This is called the decision phase.

The idea of putting ANN in to LE can be seen as a labeling task. Input APs' readings and a set of related positions (reference points) is used to train the NN. Later, input APs' readings and NN are used to infer which reference points is the best assignment. The idea is that it is not efficient to take all the APs' RSS readings into one NN for learning and inferring because at a specific reference points, only several APs' RSS readings are available or significant.

Therefore the ANN should be modified into another form, so called MMLP. The criterion of selection a set of input APs' readings for a particular position is the visibility (a.k.a. accessibility). Figure 1 displays the architecture of MMLP where each ANN module processes learning and inferring tasks, depending on the set of APs' RSS readings. Note that the input readings for each ANN module are different set of APs' readings. In order to build

up the selection, it is necessary to construct a visibility matrix for deciding how to group the APs' RSS readings.

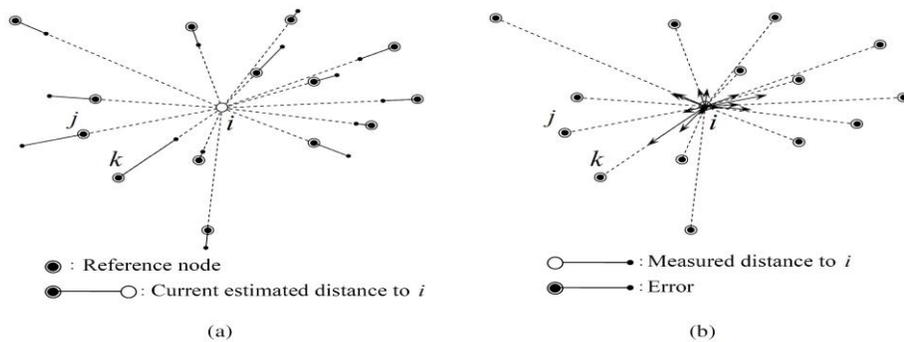
2.2. The First Phase of PPE in NN based on LE

PPE is a method we originally developed for minimizing the virtual force-vectors modeled from differences of the current LE and the distance measurement. Our proposed algorithm is based on geometry in which the errors of measurements are modeled into pushing and pulling forces. The influence of these forces leads a node to the point where all of the forces are balanced or, in other words, where the errors are minimized and canceled out through the averaging mechanism.

The algorithm is named the PPE because of this original concept. PPE is designed to meet the basic requirement that if the range measurement is shorter than the calculated distance between two nodes, these two nodes pull each other, and vice versa. In addition, the force magnitude should be a monotone function of the discrepancy between the measured and the calculated distances in order for PPE to work and converge.

The bigger is the discrepancy, the higher is the influence it gives to reduce itself. This idea is demonstrated in Figure 1 where node i needs to update its coordinates to the balanced position? Node j causes a pushing force on i because the range measurement between i and j is longer than the current calculated distance between them, and contrarily, node k exerts a pulling force on i . It should be noted that in the concept of this model, we give normal nodes the moving ability and consider location updating as a process in which normal nodes move under the influence of forces caused by their neighbors. When a normal node attempts to locate its balanced position, it considers the other nodes to be still.

Assuming that the total node number of a 2-D deployed WSN is N , $N = m + n$ where n is the number of normal nodes and m is the number of anchors, $m \ll n$. Let $\mathbf{x} = \{x_i : i = 1..N\}$, $x_i \in \mathbf{R}^2$, be the true coordinate vector of normal sensors $\{x_i : i = 1..n\}$ and anchors $\{x_i : i = (n+1)..N\}$. The true distance d_{ij} is the Euclidean distance between i and j ,



**Figure 1. (a) Node i and Measurement Errors from i to its Reference Nodes
 (b) The Errors are Modeled as PPE**

$$d_{ij} = \|x_j - x_i\|; i, j = 1..N,$$

We also define the current estimated coordinate vector to be $\mathbf{x} = \{x_i : i = 1..N\}$, $x_i \in \mathbf{R}^2$, certainly we always have $x_j = x_j, j = (n+1)..N$. The LE problem can be succinctly stated: the input data of the algorithm are measurements δ_{ij} and anchors' locations $\mathbf{x} = \{x_i : i = 1..N\}$; the

algorithm will give an initial \mathbf{x} and will update each x_i of \mathbf{x} so that the difference between \mathbf{x} and \mathbf{x} is as small as possible, \mathbf{x} then is the solution to the problem. Let the current calculated pair-wise ranges be d_{ij} then

$$d_{ij} = \|x_j - x_i\|, i, j = 1..N.$$

Now the distributed PPE algorithm for a normal node i is presented with three phases. In each phase node i uses iterations to produce a balanced position by updating position x_i with the two following equations:

$$\begin{aligned} \overline{F}_i^{(p)} &= \frac{1}{M_i^{(p)}} \sum \overline{f}_{ij}^{(p)} \\ \tilde{x}_i &\leftarrow \tilde{x}_i + \alpha^{(p)} \overline{F}_i^{(p)}. \end{aligned}$$

$\alpha^{(p)}$ is the rate of moving and $\overline{F}_i^{(p)}$ is the mean force caused by a set of $M_i^{(p)}$ related nodes. These parameters vary in different PPE's phase p .

Phase 1: Raw estimation: Node i needs at least 3 different beacons' positions and measurements from itself to them. And then node i gets its raw estimated location where the sum of forces caused by the related beacons is balanced. At each iteration, these forces are

$$\overline{f}_{ij}^{(1)} = (\tilde{d}_{ij} - \delta_{ij}) \overline{e}_{ij}$$

where j is the index of beacons related to i and \overline{e}_{ij} is the unit vector pointing from \tilde{x}_i to \tilde{x}_j to indicate the direction of \overline{f}_{ij} . Initial position of \tilde{x}_i is chosen as the mean position of these beacons. $M_i^{(1)}$ is the number of beacons related to i .

Phase 2: Pre-refinement: After normal nodes finish raw estimation phase, node i takes measurements to all of its related nodes or $M_i^{(2)}$, consisting of related neighbors and beacons, and their current updated positions for finding its new balanced position. For normal neighbor j , \tilde{x}_j is the current estimated position and for related beacon j , it is the actual position. At each round in this case, j is the index of all nodes related to i , the individual forces are

$$\overline{f}_{ij}^{(2)} = (\tilde{d}_{ij} - \delta_{ij}) \overline{e}_{ij}$$

For both phase 2 and phase 3, after some certain number of iterations, node i must send out its current updated position to normal neighbors and get the new updated positions of neighbors for better location computing. The noise model we are dealing with do not have to be a specific one or can be represented by mathematical formula. The only constrain of the noise model is to have a zero-mean distribution or to have unbiased statistical measurements. Suppose we have a function $g(\cdot)$ that changes the distribution of variable δ_{ij} into an unbiased distribution of variable $g(\delta_{ij})$, $g(\cdot)$ is a monotone increasing function so that PPE can define the direction of the force correctly, \tilde{d}_{ij} and δ_{ij} can be replaced by $g(\tilde{d}_{ij})$ and $g(\delta_{ij})$ respectively to calculate $\overline{f}_{ij}^{(1)}$ and $\overline{f}_{ij}^{(2)}$.

Below are the summary of the method:

<p>Algorithm 1: Phase 1: Raw estimation</p> <ul style="list-style-type: none"> + Send out requests for nearest anchors' coordinates x_j (at least 3) and measured distances to them. + Set initial position $\tilde{x}_i = \frac{1}{M_i^{(1)}} \sum_j x_j$. + Set initial moving step $\alpha^{(1)}$. + $k = 0$, k is iteration index repeat <ul style="list-style-type: none"> $k = k + 1$ $\vec{F}_i^{(1)} = \frac{1}{M_i^{(1)}} \sum_j \vec{f}_{ij}^{(1)}$; ($j$ is the index of the nearest beacons). if $F_i^{(1)}$ still decreases over iterations then <ul style="list-style-type: none"> └ Double $\alpha^{(1)}$ else <ul style="list-style-type: none"> └ Halve $\alpha^{(1)}$; └ Stop updating $\alpha^{(1)}$ $\tilde{x}_i \leftarrow \tilde{x}_i + \alpha^{(1)} \vec{F}_i^{(1)}$ until ($k > k_{max}^{(1)}$) OR ($\ \vec{F}_i^{(1)}\ < \epsilon$); 	<p>Algorithm 2: Phase 2: Pre-refinement</p> <ul style="list-style-type: none"> + $\alpha^{(2)} = \alpha^{(1)}$. + $k = 0$, k is iteration index repeat <ul style="list-style-type: none"> if $\text{mod}(k, R) == 0$ then <ul style="list-style-type: none"> Request current updated coordinates of neighbors. Send out current self updated position \tilde{x}_i to neighbors $k = k + 1$ $\vec{F}_i^{(2)} = \frac{1}{M_i^{(2)}} \sum_j \vec{f}_{ij}^{(2)}$; ($j$ is the index of all referenced nodes). $\tilde{x}_i \leftarrow \tilde{x}_i + \alpha^{(2)} \vec{F}_i^{(2)}$ until ($k > k_{max}^{(2)}$) OR ($\ \vec{F}_i^{(2)}\ < \epsilon$);
---	---

Figure 2. PPE's Phase 1 and NN based on LE

This method fulfill three missions

- RSS strength values from the RFS (radio frequency) sources, or the APs.
- Calculating the distances to these RFS sources.
- Estimate the current position of the object in the setting.

There are 2 stages for this proposed method, training stage and localizing stage as illustrated in Figure 3 in which RSS is the input vector whose entries are the measured RSS values, it has the form: $[RSS_1, RSS_2, \dots, RSS_k]^T$.

Training stage takes this vector and the prior information about the real distance from the APs to the mobile device. In this stage, block "NN Training" is used to learn the knowledge of the relation between the RSS vector and the real distances. This knowledge is stored under the form of Matrices (the block "NN (Matrices)" in Figure 3, the upper one) and an offset vector (not displayed in the Figure 3).

Estimation phase takes the input data (after being noise filtering process is done) and uses the learnt knowledge ("NN (Matrices)", the lower one) to estimate the distances from the mobile device to the APs. Then these distances are input into PPE component where PPE's first phase is performed to give the location of the tracked mobile device, or in other words, the device holder's position.

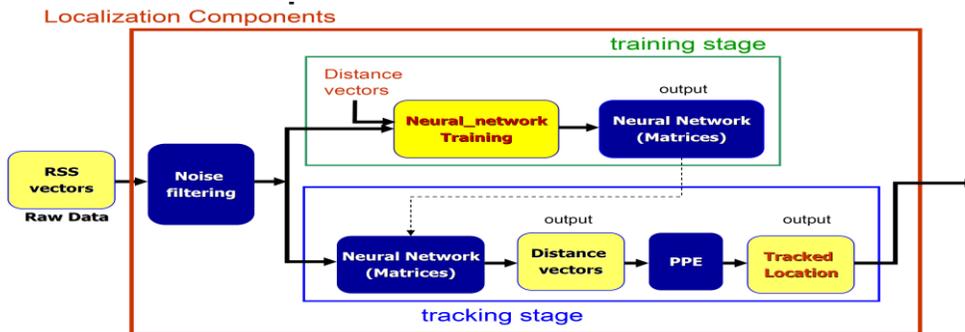


Figure 3. Details of the Design for "PPE's Phase 1 and NN based on LE" Method

- RSS: a vector whose entries are the measured RSS values.
- NN: including matrices which play the role of transforming input vectors to higher dimensions, linear multiplying and compensate the offset.
- PPE is gives the LE based on the input distance vector.

Figure 3 is the work flow that is consistent with the design in Figure 2. The “training request” triggers a training phase which uses the filtered input RSS vector and the prior knowledge of the device position at reference points to train the NN. The learnt knowledge is store under the form of NN Matrices. The estimating phase is triggered by a “tracking request” and the NN Matrices are loaded for ready. The input data RSS is also noise filtered, then the knowledge presented by NN Matrices translates the input RSS into distance vector. Distance vector is used by the PPE’s first phase to reveal the device’s position

3. Experiment and Discussions

3.1. Data Collecting Methodology

As in most of the schemes that use RSS values for LE, for collecting RSS of the RFS emitted by the APs, we have intervened and modify a program which can access to the output data of the WiFi driver of the smart phone. The modified program connects through Bluetooth protocol to a laptop, which plays the role of the base station, getting the RSS data of all APs in the range and sends the data to the laptop. It can update the RSS status at a rate defined by users, or after every Δt .

Figure 4 is the result of collecting data for training phase, 5 strongest readings from APs are plotted in different colors: 1000 samples in 4 minutes while the smart phone is kept still. The fluctuation is significant even when it is not moved or not rotated, the RSS values change even tens of times.

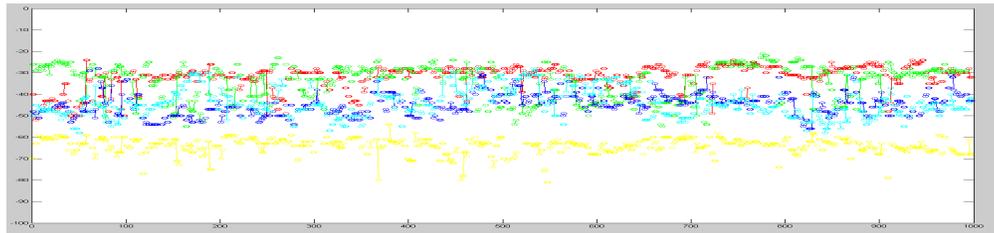


Figure 4. High Noise Input RSS Readings of 5 Strongest (Nearest) APs, Each RSS Value from an AP is displayed in One Color at a Reference AP. Note that the Unit of Measurement is in [dBm] and the Values Vary in High Ranges

3.2. The NN and First Phase of PPE based on LE

3.2.1. Experiment Results

This set of experiments aims at improving the work in above Section, in which the positions used in the experiments are just sparse reference points, both for training and estimating phase.

Using the data collecting methodology described in Section 3.1, we collect the data on the flight and write to a file whose content can be seen in Figure 5 below.

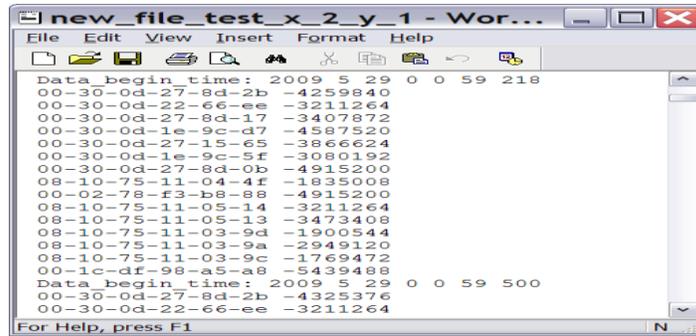


Figure 5. Collected RSS Data is Written into a File

For training phase, each file is the result of measured RSS of RFS collected in 4 minutes with 1200 samples for one reference AP. For inferring phase, the new file is used on the flight (nearly real-time) by the inferring MATLAB software. The program actually does not only read the data from the file, but also updates or renewed the file when it is too long.

For the purpose of increase accuracy, we deploy some more APs in the settings, the area for testing is also smaller: $45m^2$. This is just because we can utilize the frame work of the previous MMLP method to break down the big region into smaller ones. What we are aiming at here is improve the accuracy of a small scale of the previous problem. The monitoring area is plotted in Figure 6. The resolution for the learning phase is one square meter.

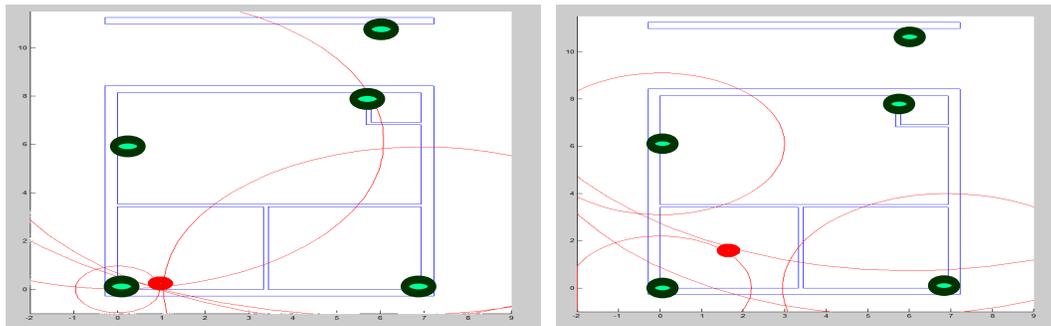


Figure 6. Ideal Estimated Position and the Real Estimated Position 5 Nearest APs are Plotted in the Monitoring Area

The result of the online testing can be seen in Table 1 in which the time of testing is 5 minutes for each run, the location calculation is perform on the flight at Laptop. Movement speed of the person carrying the smart phone is slow, through a given path with given stop positions. At each step on the path, the person stands still for 20 seconds. The results in Table 1 are the average values in 20 runs of recording and estimating

Table 1. The Result of the Online Testing

Time for testing	RMSE (m)
Morning	3.10
Afternoon	3.24
Night	3.02

Training data is collect in all ‘morning’, ‘afternoon’ and ‘night’. However, for tracking, the result at night gives the lowest error of estimation. The average RMSE for the system is therefore 3.12m.

Note that previous MMLP method decides if the device is at a reference point. Therefore the decision making for a point is actually an area around that point to compare with a random path (any AP). The area around each point is approximately 4.5m x 4.5m.

3.2.2. Advantages

- Does the semi-tracking task to the mobile device in the monitoring area.
- Avoid discrete results of ‘yes/no in cell’ method in learning classification method. Increase accuracy because of the average mechanism of continuous results.
- This methodology can reduce the reference points, or in other words, reduce the load of work for training phase.
- The relation between RSS readings and the distance is study therefore the distance vector can be considered to be unbiased without much calibration. The NN engine actually does the calibration.

3.2.3. Disadvantages

Since this method also requires the learning phase with reference points, therefore like in other methods. Although using the idea of visibility matrix and multi-smaller NN, we still face the scaling problem in training phase when the area is much larger. This method still could not deal well with the highly unpredictable input signal. Even though the smart phone is held still in 2 minutes (1000 time steps of updating), the fluctuation is too significant (in [dBm] scale), not to mention when the smart phone tilts or changes its position a little bit.

4. Proposed Idea and Analysis for Collaborative Scheme

A new scenario and the proposed method: LE in an area with high density of tracked objects. An example is to localize people in a crowded setting as in Figure 7 in which video and audio are not suitable as mentioned above. However, if the tracked mobile devices communicate with each other and the distances between mobile devices to the nearby neighbors can be estimated, then the accuracy of above proposed method can be improved depending on the number of neighbor mobile devices and the accuracy of the measurements to them. How the improvement of the estimation can be achieved is related to three statements that are presented as followings

4.1. Statement 1

"Estimation of a normal node in phase 1 is unbiased"

Proof of statement 1: In this phase, referenced nodes of a normal node i are the anchors. Since $E\{\delta_{ij}\} = d_{ij}$, the expectation of the mean force is

$$E\{\overline{F_i^{(1)}}\} = \sum_j (\tilde{d}_{ij} - d_{ij}) \overline{e_{ij}}$$

where j is the index of known-location nodes. This equation implies that the expectation of the mean force is built based on correct measurements from known location nodes. Each individual force $\overline{f_{ij}}$ tries to move i to the circle C_{ij} having the radius d_{ij} , to minimize itself's magnitude (see Figure 7). As the result, i 's position converges to the intersection point of these circles C_{ij} , or $E\{\tilde{x}_i\} = x_i$

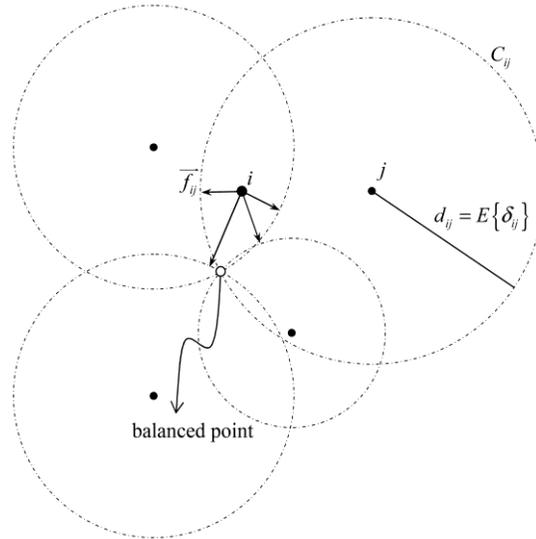


Figure 7. The Measurements have no Error and Referenced Nodes' Locations are known

4.2. Statement 2

"Estimations of a normal node in phase 2 and phase 3 are unbiased"

Proof of statement 2: In these two phases, every normal node has its raw LE and it now uses all of its referenced nodes to get a better estimation. Since the algorithm mainly deals with probability issues, we do not only consider the effect of forces to be specifically caused by nodes but also consider it as the effect caused by sub-areas. We assume that the whole deployed area is divided into small sub-areas in each of which the node number is large and the node distribution is uniform. Figure 8 illustrates the idea: the actual positions are the black round nodes, e.g., k_0 , while the current estimated positions are the black square nodes, e.g., k . The solid links between them are the current errors of estimations. What we want to do here is prove that even with these errors of estimations, the balanced position of node i under PPE mechanism is its actual position i_0 .

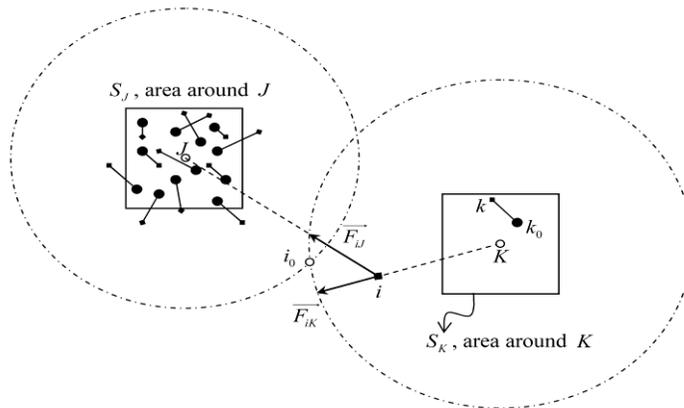


Figure 8. The Force Effect of a Sub-area to a Node is Equivalently the Force Effect

Let the mean force on i caused by the area S_K .

$$\overline{F_{iK}} = \frac{1}{n_k} \sum_k \left(\frac{\tilde{d}_{ik} - \delta_{ik}}{L_{ik}} \right) \frac{\overline{ik}}{\tilde{d}_{ik}}$$

$$E\{\overline{F_{iK}}\} = E\left\{ \left(\frac{\tilde{d}_{ik} - d_{ik} - d_{ik} \cdot noise_{ik}}{L_{ik}} \right) \frac{\overline{iK} + \overline{Kk}}{\tilde{d}_{ik}} \right\}$$

In equation K is the center point of S_K , n_k is the number of nodes in S_K . And $noise_{ik}$ and \overline{Kk} are independent of each other and of d_{ij} , \tilde{d}_{ij} . Beside, $E\{\overline{Kk}\} = 0$ since estimation of phase 1 is unbiased. Therefore it is rewritten as

$$E\{\overline{F_{iK}}\} = E\left\{ \left(\frac{ik - i_0 k_0}{\tilde{d}_{ik} L_{ik}} \right) \right\} \overline{iK}$$

Assume that the estimation errors of nodes in S_k is small so that k still has uniformed distribution in S_k , or $E\{ik\} = E\{i_0 k_0\}$. If $ik = i_0 K$, $E\{ik_0 - i_0 k_0\} = 0$. Since the value $\tilde{d}_{ik} L_{ik}$ is a positive and bounded value, then $E\{\overline{F_{iK}}\} = 0$. If $ik > i_0 K$, then $E\{ik_0 - i_0 k_0\} > 0$ and $E\{\overline{F_{iK}}\}$ becomes a pull force with direction of \overline{iK} . Contrariwise, if $ik < i_0 K$, $E\{\overline{F_{iK}}\}$ becomes a push force. Hence, the balanced point of i belongs to the circle $C(K, i_0 K)$. Combining the effect of other subareas on i , we find that i_0 becomes the unique balanced position of i . Besides, with the previous proof that PPE leads i to a balanced point, we can conclude that the estimations of i in phase 2 and phase 3 are unbiased.

In these two phases, a normal node acts as if it is affected by forces caused by groups of referenced nodes where each group plays the role of a beacon. With more referenced beacons comparing with phase 1, the accuracy of estimation in phase 2 is better than in phase 1. In addition, it should be noted that the assumption that k has uniformed distribution in S_k is not a tight one. However, over iterations, each node updates and gets better and better estimation, the assumption becomes tighter and tighter. As a result, the expectation of balanced point of a normal node i is the very i_0 and PPE drags i towards i_0 even when the current estimations of referenced nodes are suffering the errors. In other words, one node's updating is quite independence from others'. Therefore, PPE is a convergent and robust algorithm.

4.3. Statement 3

“Phase 2 can give the higher accuracy than phase 1, if the neighbors around a node are present.”

Proof of statement 3: In both of two phases, if the referenced node number is infinitely large, the node's balanced position will exactly be at the actual position. Unfortunately, the referenced node number is limited to $M_i^{(p)}$, and the errors of measurements make the mean-force non-zero at the correct position i_0 , drive i out of i_0 , giving birth to PPE's error. Obviously, the bigger the variance of the mean-force's magnitude, the bigger error the final result gets.

Therefore, we examine the variance of $\|\overline{F_i^{(p)}}\|$ at position i_0 when all the referenced nodes are at correct positions to compare the performances of three phases. For ease of analysis, we change the notation $\overline{f_j}$ to $\overline{f_{ij}}$ in this Subsection and let $\|\overline{F_i^{(p)}}\| = F_i^{(p)}$, $\|\overline{f_j}\| = f_j$.

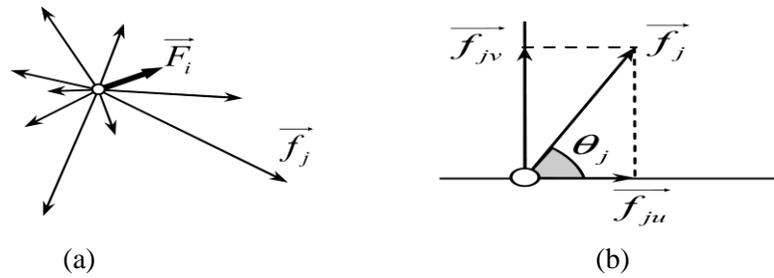


Figure 9. (a) Sum Force is not a Zero Vector at the Correct Position. (b) A Force into Horizontal and Vertical Components

$$\overline{F_i^{(p)}} = \frac{1}{M_i^{(p)}} \sum_j \overline{f_j}$$

Decomposing each individual force \vec{f}_j into components as in Figure 9, $\vec{f}_j = (f_{ju}, f_{jv}) = (f_j \cos \theta_j, f_j \sin \theta_j)$ and remarking that $E\{f_j\} = 0$, $E\{F_i^{(p)}\} = 0$, (unbiased measurement error and unbiased estimation), f_j and θ_j are two independent variables, we obtain

$$\begin{aligned} \overline{F_i^{(p)}} &= \left(\frac{1}{M_i^{(p)}} \sum_j f_j \cos \theta_j, \frac{1}{M_i^{(p)}} \sum_j f_j \sin \theta_j \right) \\ \text{var}(F_i^{(p)}) &= E\left\{ \left(F_i^{(p)} \right)^2 \right\} - E^2\{F_i^{(p)}\} \\ \text{var}(F_i^{(p)}) &= E\left\{ \left(\frac{1}{M_i^{(p)}} \sum_j f_j \cos \theta_j \right)^2 + \left(\frac{1}{M_i^{(p)}} \sum_j f_j \sin \theta_j \right)^2 \right\} \\ \text{var}(F_i^{(p)}) &= \frac{\text{var}(f_j)}{M_i^{(p)}}. \end{aligned}$$

The equation implies that the variance of $F_i^{(p)}$ does not depend on the directions of forces caused by neighbors but only on the referenced node number and the variance of individual forces. It also guarantees that phase 2 is necessary to improve phase 1's result which is obtained by only anchors ($M_i^{(1)} < M_i^{(2)}$).

The proposed method now is stated: Utilize the frame work of ‘‘PPE’s phase 1 and NN based on LE’’ to avoid calibration for the RSS values received by the mobile device from the APs. This frame work also gives the first raw estimations for most of the devices in the monitoring area. The raw estimations will be the data to be exchanged between normal nodes (mobile devices) and used as input data for PPE’s phase 2. Applying PPE’s phase 2, the accuracy will be improved.

4.4. Advantages

Collaborative PPE method actually has many advantages whose analyses are not presented in this report. This method increases the accuracy because more neighbors’ distance estimations help improve the accuracy due to the average mechanism of continuous results.

One important point should be remarked is that the lack of minor information can be accepted by PPE. Drop-out or add-in nodes will not affect much a node's position updating as long as the number of these nodes is small comparing to the number of referenced neighbors.

Hence, it is unnecessary to make a node send out requests to its neighbors for referenced updated positions after every iteration. In stead, we can cut down the communication cost by just letting a normal node send the request task once after every R internal iterations on it, or in other words, we skip the task $R-1$ times. Therefore, updating position of a node does not

need a tight synchronization. For comparison in terms of computation cost, PPE also outperformed with the simulated data as can be seen in the Table.

Table 2. Computation Cost Table

	Multiplications per iteration	Iteration number
wdMDS	$3(M_i^{(3)})^2 + 13M_i^{(3)}$	80
MLE	$7M_i^{(3)} + 3M_i^{(3)} * Iter_0$	52
PPE	$5M_i^{(3)}$	40

4.5. Disadvantages

Implementation requires some scheme so that the normal nodes can communicate with each other and estimate the range between them. Even though the method with Neural Network and PPE's first phase does not need serious calibration, the range estimation between normal nodes (mobile devices) needs the work of calibration. And then that the range estimation is not biased. (Although the biased result of position estimation is acceptable).

5. Conclusion

This paper goes through most of the techniques for the LE based on the RSS value of RFS causing the GPS signal attenuation in building. These techniques can be categorized to Signal Propagation Modeling and Fingerprinting.

Also we are researching that our work on "Collecting data methodology", "ANN based on MMLP" and "NN and PPE's first phase based on LE". The key contribution of MMLP is using visibility matrix to avoid using big NN (which may lead to big but sparse matrices of knowledge). The training and estimating phases do not use a big NN but multi-smaller NN instead. This contribution solves the problem of scaling, or in other words, when the area is at large scale, MMLP can solve the problem easily in both training and inferring phases. Updating any small changes is also convenient with this method.

The key contribution of "NN and PPE's first phase based LE" is trying to positioning the actual continuous coordinates of the mobile device instead of several target/reference points or several cells in the monitoring area.

In the final part, we present the necessity of utilizing collaborative methods to improve the accuracy of LE when the mobile device is close to a group of other mobiles. Although the necessity and the efficiency of collaborative methods are fully discussed, particularly the PPE's phase 2, the work has not been done with hardware in reality experiments. This should be our next future work.

Acknowledgements

This article is a revised and expanded version of a paper entitled "Location Estimation based on Received Signal Strength from Access Pointer and Machine Learning Techniques" presented at the 3rd International Conference on Next Generation Computer and Information Technology (NGCIT 2014) held on October 24-26, 2014 at Liberty Central Saigon Hotel, Hochiminh, Vietnam.

References

- [1]. P. Bahl, *et al.*, "A software system for locating mobile users: Design, evaluation, and lessons", Microsoft Research, MSR-TR-2000-12, (2000) April.
- [2]. P. Bahl and V. N. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," in INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, vol. 2, (2000), pp. 775–784.
- [3]. Cynthia, *et al.*, "Challenges in Location-Aware Computing" Published by IEEE ComSoc 1536-1268/03/ 2003 IEEE.
- [4]. Andrew, *et al.*, "Using Wireless Ethernet for Localization", IEEE/RSJ International Conference on Intelligent Robots and Systems, (2002).
- [5]. M. Hazas, *et al.*, "Location-Aware Computing Comes of Age," Computer, vol. 37, no. 2, (2004) February, pp. 95-97.
- [6]. R. Battiti, *et al.*, "Statistical Learning Theory for Location Fingerprinting In Wireless Lans", Technical Report # DIT-02-0086 University of Trento Italy, (2002) October.
- [7]. R. Want and B. Schilit, "Expanding the horizons of location-aware computing", IEEE Computer, vol. 34, no. 8, (2001) August, pp. 31–34.
- [8]. K. Pehlavan, *et al.*, "Indoor Geolocation Science and Technology", IEEE Communications Magazine, (2002).
- [9]. Y. Sasaki and S. Kagami, "Multiple sound source mapping for a mobile robot by self-motions triangulation", In: International Conference on Intelligent Robots and Systems, IEEE, (2006), pp. 380-385.
- [10]. Guvenç, *et al.*, "Enhancements to RSS Based Indoor Tracking Systems Using Kalman Filters," GSPx & International Signal Processing Conference, Dallas, TX, (2003) March 31-April 3.
- [11]. R. Battiti, *et al.*, "Neural Network Model for intelligent networks: deriving the Location from signal patterns", The First Annual Symposium on Autonomous Intelligent Networks and Systems, (2002).
- [12]. U. Nasir, *et al.*, "On Building a Reflective Middleware Service for Location-Awareness", The 11 IEEE International Conference on Embedded and Real-Time Computing Systems and Application (RTCSA), Hong Kong, (2005) August 17-19.
- [13]. M. Iqbal, *et al.*, "Reflective Middleware for Location-Aware Application Adaptation", ICCSA, no. 2, (2005), pp. 1045-1054.

Authors



Seong-Jin Cho, He has been the President of Computer Center at the Pigeon Corporation, from 1997 to 2001. He joined the research professor in the Department of Computer Engineering at Hongik University from 2002 to 2007. He received a Ph.D. for ubiquitous computing parts in Department of Computer Engineering from Kyunghee University, Korea, in 2013. His major research interests include ubiquitous healthcare, home-network and ubiquitous computing system.



Ho-Kyun Park, He received his B.S. and M.S. degrees in computer science from Kwangwoon University, Seoul Korea, in 1987 and 1989, respectively. He obtained his Ph.D. in multimedia network from same university, 1998. He is currently a Professor with the School of IT Convergence Engineering at the Shinhan University, Korea, since March 1992. His major research interests include network security, smart home-network system, and ubiquitous computing system.