Postprint: WLAN Heterogeneous Terminal Localization Method Based on DBSCAN-GRNN-LSSVR Algorithm

Authors: Zhang Yong, Heng Fan, Wang Yujie

Date: 2018-05-02T00:00:00+00:00

Abstract

To address the problem of excessive positioning deviation caused by heterogeneous terminals (fingerprint database terminals and test terminals) in WLAN indoor positioning systems, a solution based on the DBSCAN-GRNN-LSSVR algorithm is proposed. This paper employs Least Squares Support Vector Regression (LSSVR) to construct a mapping model between the Received Signal Strength (RSS) of fingerprint database terminals and their physical coordinates; lists RSS values collected by heterogeneous terminals at calibration points to generate a scatter plot; utilizes a density-based clustering method to eliminate boundary points and noise points; constructs a mapping function for heterogeneous terminal RSS using a General Regression Neural Network; and locates test points through the LSSVR model. Experimental results demonstrate that, compared with using the LSSVR algorithm alone, the positioning accuracy of test terminals improves by 18-40%, effectively resolving the issue of excessive positioning deviation.

Full Text

Preamble

Title: WLAN Heterogeneous Terminal Location Method Based on DBSCAN-GRNN-LSSVR Algorithm

Authors: Zhang Yong, Fan Heng, Wang Yujie (School of Computer & Information, Hefei University of Technology, Hefei 230009, China)

Abstract: Aiming at the excessive location error caused by heterogeneous terminals (fingerprint database terminal and test terminal) in WLAN indoor localization systems, this paper proposes a solution based on the DBSCAN-GRNN-LSSVR algorithm. The least squares support vector regression (LSSVR) algorithm is employed to build the mapping relationship model between the received signal strength (RSS) of the fingerprint database terminal and physical coordinate locations. The scatter plot is obtained by listing the RSS values collected by the fingerprint database terminal and the test terminal at calibration points. Boundary points and noise points are eliminated using density-based spatial clustering. The generalized regression neural network (GRNN) is used to construct the heterogeneous terminal mapping function of the RSS. The LSSVR model is then used to determine the location of the test point. Experimental results demonstrate that compared with using the LSSVR algorithm alone, the proposed DBSCAN-GRNN-LSSVR algorithm improves test terminal positioning accuracy by 18-40%, effectively solving the problem of excessive localization deviation caused by heterogeneous terminals.

Keywords: WLAN indoor localization; heterogeneous terminal; LSSVR; DBSCAN; GRNN

0 Introduction

Location fingerprinting has become a popular research direction due to its algorithmic flexibility and relatively accurate positioning. In recent years, the increasing demand for location-based services in airports, warehouses, exhibition halls, shopping malls, mines, hospitals, and other indoor environments has promoted the development of localization technologies. However, GPS cannot meet people' s requirements for indoor positioning accuracy due to signal obstruction by buildings. Current indoor positioning technologies include infrared indoor positioning, RFID indoor positioning, ZigBee indoor positioning, ultrasonic indoor positioning, ultra-wideband indoor positioning, Bluetooth indoor positioning, and WLAN indoor positioning. Among these, WLAN indoor positioning has received widespread attention due to its low cost, fast positioning speed, lack of requirement for additional hardware devices, wide applicability, and high overall accuracy.

The main methods for WLAN indoor positioning include proximity sensing, geometric measurement, and scene analysis. Scene analysis includes location fingerprinting and signal propagation modeling. However, in practical applications of WLAN-based indoor positioning, heterogeneous terminals cause significant problems. Different terminals at the same location receive different received signal strength (RSS) values from various access points (APs), leading to excessive positioning deviation. Consequently, extensive research has been conducted to address the problem of large positioning errors caused by heterogeneous terminals in WLAN indoor positioning.

Qin Siming [2] proposed an RSSI calibration method based on BP neural networks, which establishes a BP neural network model to convert the RSS values of test terminals into those of fingerprint database terminals. Dong [3] proposed a method of pairwise subtraction of fingerprint vector components, which can improve the positioning accuracy based on nearest neighbor (NN) by 20% and that based on Bayesian inference (BI) by 12%, as demonstrated by experimental results. Kim et al. [4] observed the differences in RSS between fingerprint database terminals and test terminals and proposed a method using the location of RSS peaks for positioning. This method assumes that the locations with the strongest RSS for both fingerprint database terminals and test terminals are the same, thereby addressing the impact of terminal differences. Kjærgaard et al. [5] proposed a hyperbolic calibration method that performs pairwise division of RSS vector components to eliminate the influence of slope terms in linear transformations, with experiments showing that this method can reduce the impact of heterogeneous terminals. Zhao Cong [6] proposed a cosine similarity method that calculates the similarity between test terminal RSS and fingerprint database terminal RSS using the cosine similarity algorithm. The greater the similarity, the better the match between two reference points, which reduces the large positioning deviation caused by heterogeneous terminals to a certain extent. Hossain et al. [7] proposed an RSS difference (SSD) method. Assuming the original RSS vector is N-dimensional, this method subtracts the anchor point component from other RSS components, resulting in a new RSS vector containing only N-1 RSS difference values. Gu Yang et al. [8] proposed a device-adaptive wireless signal feature extraction and positioning method (SSDR) that processes RSS vectors to eliminate the influence of slope and intercept before positioning, improving positioning accuracy by 10-20% compared to direct signal strength and Euclidean distance-based methods. However, the positioning accuracy improvements of these heterogeneous terminal positioning methods are not significant.

To ensure the accuracy of WLAN indoor positioning and solve the problem of excessive positioning deviation caused by heterogeneous terminals, this paper proposes a solution based on the DBSCAN-GRNN-LSSVR algorithm. At fingerprint points, the fingerprint database terminal collects RSS values, and least squares support vector regression [9-11] (LSSVR) is used to establish the nonlinear relationship between the RSS collected by the fingerprint database terminal and its corresponding position coordinates, creating an LSSVR mapping model. The test terminal collects RSS values at calibration points to obtain a scatter plot of RSS values from both fingerprint database and test terminals. The density-based spatial clustering method [12-14] (DBSCAN) is used to cluster data points, marking core points, boundary points, and noise points, and then removing boundary and noise points. The generalized regression neural network [15-16] (GRNN) is employed to fit the scatter plot processed by DBSCAN, obtaining a mapping function between fingerprint database terminal RSS and test terminal RSS to convert test terminal RSS values to fingerprint database terminal RSS values. Finally, the LSSVR model is used to locate test points. Experimental results show that compared with using only the LSSVR algorithm, the proposed DBSCAN-GRNN-LSSVR algorithm improves the positioning performance for OPPO phones, SmartQ tablet 1, and SmartQ tablet 2 by 40%, 20%, and 18% respectively, effectively solving the problem of excessive positioning deviation caused by heterogeneous terminals.

---

1 DBSCAN-GRNN-LSSVR Algorithm

Location fingerprinting consists of two phases: an offline phase and an online phase. The offline phase primarily involves the fingerprint database terminal collecting RSS at fingerprint points and establishing a database mapping RSS to fingerprint point locations. During the online phase, the test terminal receives AP signal strengths in the scene, processes them using algorithms, and estimates the location coordinates of the point to complete online positioning. The DBSCAN-GRNN-LSSVR heterogeneous terminal positioning algorithm proposed in this paper is based on location fingerprinting.

The block diagram of the DBSCAN-GRNN-LSSVR positioning system is shown in Figure 1 [Figure 1: see original paper]. In the offline phase, fingerprint points are selected in the scene, and the fingerprint database terminal collects RSS values at each fingerprint point. The positioning features and corresponding fingerprint point locations are used as input sample pairs to complete the LSSVR learning and training process, establishing an LSSVR mapping relationship model between positioning features and physical locations. In the online phase, the test terminal collects RSS values at calibration points to obtain a scatter plot of RSS values from both fingerprint database and test terminals. DBSCAN is used to eliminate boundary and noise points, and GRNN is employed to fit the scatter plot of fingerprint database terminal RSS and test terminal RSS processed by DBSCAN. This yields a mapping function between fingerprint database terminal RSS and test terminal RSS. After selecting test points, the test terminal RSS values are converted to fingerprint database terminal RSS values and input into the constructed LSSVR model to estimate the actual location of the test points.

1.1 Density-Based Spatial Clustering of Applications with Noise

DBSCAN is a typical density-based clustering method whose core idea is to measure the density of the space where a point is located by the number of neighboring points within its neighborhood. It can find clusters of irregular shapes and does not require prior knowledge of the number of clusters.

The main definitions of DBSCAN are as follows: Assume the data collection is $\{x_1, x_2, \dots, x_n\} = X$, and denote Eps (radius) and MinPts (given integer) as ε and M respectively.

Definition 1: Eps Neighborhood. Let $N_\varepsilon(x) = \{y \in X \mid d(y, x) \le \varepsilon\}$ be the Eps neighborhood of x, where $d(y, x)$ represents the distance between two objects y and x in X.

Definition 2: Core Point, Boundary Point, and Noise Point. Let $x \in X$. If the number of objects in the Eps neighborhood of x satisfies $|N_\varepsilon(x)| \ge M$, then x is called a core point of X. Denote the set of all core points in X as X_c. Objects that are not core points but fall within the Eps neighborhood of some core point are called boundary points. Starting from any core point object, all objects density-reachable from that object constitute a cluster. Objects that do not belong to any cluster are noise points.

Definition 3: Directly Density-Reachable. If $y \in N_\varepsilon(x)$ and $|N_\varepsilon(x)| \geq M$, then y is said to be directly density-reachable from x.

Definition 4: Density-Reachable. Let $P_1, P_2, \ldots, P_n \in X$. If they satisfy: P_{i+1} is directly density-reachable from P_i for $i = 1, 2, \ldots, m-1$ and $m \geq 2$, then P_m is said to be density-reachable from P_1.

Definition 5: Density-Connected. Let $x, y, z \in X$. If both x and y are density-reachable from z, then x and y are said to be density-connected. Density connectivity is symmetric.

When ε and M are given, the DBSCAN clustering process is as follows: a) Select any unprocessed data point and determine whether it is a core point; b) If it is, find all data points density-reachable from that point to form a cluster; if not, determine whether it is a noise point or boundary point and mark it accordingly; c) Repeat the above steps until all data points are processed, obtaining core points, noise points, and boundary points. Remove boundary points and noise points, and form a new sample set from the core points.

1.2 Generalized Regression Neural Network

The generalized regression neural network is a type of radial basis neural network. GRNN possesses strong nonlinear mapping capabilities, a flexible network structure, and high fault tolerance and robustness. GRNN is structurally similar to RBF networks, consisting of four layers: input layer, pattern layer, summation layer, and output layer, as shown in Figure 2 [Figure 2: see original paper].

The network input is $X = [x_1, x_2, \ldots, x_n]^T$. The GRNN model prediction value $\hat{Y}(X)$ is given by:

$$\hat{Y}(X) = \frac{\sum_{i=1}^{n} Y_i \exp\left(-\frac{(X-X_i)^T(X-X_i)}{2\sigma^2}\right)}{\sum_{i=1}^{n} \exp\left(-\frac{(X-X_i)^T(X-X_i)}{2\sigma^2}\right)}$$

where X_i and Y_i are the training sample values of X and Y respectively, n is the number of training samples, and σ is the width coefficient of the Gaussian function, here called the smoothing factor. The prediction value $\hat{Y}(X)$ is the weighted average of all training sample values Y_i, where the weight factor for each training sample value Y_i is the exponential of the squared Euclidean distance between the corresponding X_i and the test sample value X.

GRNN has few manually adjustable parameters, only one spread value (the spread speed of the radial basis function). Therefore, when building GRNN, only an appropriate spread value needs to be selected. The spread value has a significant impact on the network approximation results. A smaller spread value leads to better approximation of sample data by the neural network, while a larger spread value makes the approximation process smoother, but simultaneously increases network output error.

When constructing the GRNN fitting function, the input is the sample set collected by the test terminal at calibration points, and the output is the sample set of the fingerprint database terminal. Let the sample set of the fingerprint database terminal at calibration points be $\{(S_f^1, L_1), (S_f^2, L_2), \dots, (S_f^m, L_m)\}$, where $S_f^i = \{S_{f1}^i, S_{f2}^i, \dots, S_{fn}^i\}$ indicates that the fingerprint database terminal at calibration point L_i receives RSS signal strength values from the n-th AP, n is the number of APs, and $L_i = (x_i, y_i)$ is the physical location coordinate of calibration point i. The sample set of the test terminal at calibration points is $\{(S_c^1, L_1), (S_c^2, L_2), \dots, (S_c^m, L_m)\}$, where $S_c^i = \{S_{c1}^i, S_{c2}^i, \dots, S_{cn}^i\}$ indicates that the test terminal at calibration point L_i receives signal strength values from the n-th AP.

Based on S_f^i and S_c^i, a scatter plot of RSS values from fingerprint database terminals and test terminals is obtained. After processing the data points with the DBSCAN algorithm and removing boundary and noise points, the sample set of the fingerprint database terminal becomes $\{(S_f^{1'}, L_1'), (S_f^{2'}, L_2'), \dots, (S_f^{m'}, L_m')\}$, and the sample set of the test terminal becomes $\{(S_c^{1'}, L_1'), (S_c^{2'}, L_2'), \dots, (S_c^{m'}, L_m')\}$, where $m' \leq m$.

1.3 LSSVR Algorithm

Let the sample set of the fingerprint database terminal at fingerprint points be $\{(S_t^1, L_1), (S_t^2, L_2), \dots, (S_t^t, L_t)\}$, where $S_t^i = \{S_{t1}^i, S_{t2}^i, \dots, S_{tn}^i\}$ indicates the RSS signal strength value received by the fingerprint database terminal at fingerprint point i from the n-th AP, n is the number of APs, and $L_i = (x_i, y_i)$ is the two-dimensional position coordinate of fingerprint point i.

LSSVR maps the input S to a high-dimensional feature space R^m through a nonlinear mapping $\Phi : S \to \Phi(S)$, and then constructs the optimal regression function between position coordinates y and RSSI values in R^m:

$$y = \omega^T \Phi(S) + b$$

where ω is the weight vector and b is the bias term.

The LSSVR optimization problem is formulated as:

$$\min_{\omega, b, \varepsilon} \frac{1}{2}\omega^T \omega + \frac{C}{2}\sum_{i=1}^{t} \varepsilon_i^2$$

subject to:

$$y_i = \omega^T \Phi(S_i) + b + \varepsilon_i, \quad i = 1, 2, \dots, t$$

where C is the penalty parameter and ε_i is the error.

The Lagrangian function can be used to solve equations (4) and (5):

$$L(\omega, b, \varepsilon, \alpha) = \frac{1}{2}\omega^T \omega + \frac{C}{2}\sum_{i=1}^{t} \varepsilon_i^2 - \sum_{i=1}^{t} \alpha_i \left[\omega^T \Phi(S_i) + b + \varepsilon_i - y_i\right]$$

where $\alpha = [\alpha_1, \alpha_2, \dots, \alpha_t]^T$ are the Lagrange multipliers.

This paper uses the LS-SVM toolbox to solve the optimization problem. This toolbox employs a least squares algorithm to solve the optimization problem in the SVR model. LSSVR uses a Gaussian kernel function with relatively few parameters and strong adaptability:

$$K(S_m, S_n) = \exp\left(-\frac{\|S_m - S_n\|^2}{2\delta^2}\right)$$

Therefore, the regression functions for position coordinates and RSS values are obtained as:

$$x = \sum_{i=1}^{t} \alpha_i K(S, S_i) + b$$

$$y = \sum_{i=1}^{t} \alpha_i' K(S, S_i) + b'$$

where α_i, α_i', b, and b' are the corresponding Lagrange multipliers and biases obtained.

---

2 Experimental Results and Analysis

2.1 Experimental Conditions

To verify the effectiveness of the proposed DBSCAN-GRNN-LSSVR algorithm, experiments were conducted on the 6th floor of the Yifu Building at Hefei University of Technology. The partial floor plan of the 6th floor of Yifu Building is shown in Figure 3 [Figure 3: see original paper], including offices and corridors across three areas. The offices contain desks, bookshelves, and other office supplies. Six DWL-2414N APs were deployed throughout the positioning area, all mounted on brackets 2.4 m above the ground.

Fingerprint points were uniformly distributed in a 3×3 meter grid in areas one, two, and three, while test point locations were randomly selected. During signal collection, handheld terminals scanned each AP 50 times at 0.1-second intervals at sampling points in the experimental area. Each test point measured 250 sets of data, from which the maximum and minimum values were removed and the average was taken.

The positioning error err_i and average positioning error $AverErr$ are defined as:

$$err_i = \sqrt{(x_i - x_i')^2 + (y_i - y_i')^2}$$

$$AverErr = \frac{1}{h}\sum_{i=1}^{h} err_i$$

where (x_i, y_i) is the actual coordinate of the i-th test point, (x_i', y_i') is the estimated coordinate of the i-th test point, and h is the total number of test points.

The four terminals used in this paper are listed in Table 1 .

2.2 Results Analysis

Twenty-five fingerprint points (also serving as calibration points) and 22 test points were selected for this experiment, distributed in areas one, two, and three shown in the figure above. The Meizu M3 note3 was used as the fingerprint database terminal, while OPPO, SmartQ tablet 1, and SmartQ tablet 2 served as test terminals. The DBSCAN algorithm parameters were set to commonly used values: MinPts = 5 and Eps = 7. For the GRNN algorithm, the spread value was selected from the range [0, 2], and the optimal spread value was identified by comparing the root mean square error of approximation.

Figures 4(a)-(c) [Figure 4: see original paper] show the scatter plots of RSS values collected by the OPPO phone, SmartQ tablet 1, SmartQ tablet 2, and Meizu M3 note3 at all calibration points. Figures 4(d)-(f) show the scatter plots after processing with the DBSCAN algorithm.

The data after removing boundary and noise points from Figures 4(d)-(f) were fitted using GRNN to obtain the RSS relationship between heterogeneous terminals. Using SmartQ tablet 1 as the test terminal, GRNN was employed to fit the relationship between fingerprint database terminal RSS and test terminal RSS. The relationship between spread value and root mean square approximation error when the spread parameter varies is shown in Figure 5 [Figure 5: see original paper]. The minimum root mean square approximation error occurs when the spread value is 0.9. Therefore, when fitting the relationship between SmartQ tablet 1 and Meizu M3 note3 RSS values, the spread value was selected as 0.9, and the fitting curve is shown in Figure 6 [Figure 6: see original paper]. The same method was used to determine the optimal spread values and fitting curves for OPPO and SmartQ tablet 2.

Using SmartQ tablet 1 as the test terminal, Figure 7 [Figure 7: see original paper] shows the relationship between the number of calibration points and average positioning error. As can be seen from Figure 7, the positioning error decreases significantly when the number of calibration points reaches 20.

Table 2 compares the average positioning errors of test terminals using different algorithms. For the DBSCAN-GRNN-LSSVR algorithm, the DBSCAN algorithm first removes noise and boundary points, after which the GRNN algorithm fits more accurately. Compared with the LSSVR algorithm, this algorithm achieves a 40% improvement in positioning performance for OPPO phones, 20% for SmartQ tablet 1, and 18% for SmartQ tablet 2. Compared with the SSD-LSSVR calibration algorithm, the positioning performance improves by 10%, 9%, and 9% respectively. Compared with the GRNN-LSSVR calibration algorithm, the positioning performance improves by 21%, 11%, and 11% respectively. The experimental results demonstrate that the DBSCAN-GRNN-LSSVR algorithm provides better calibration performance.

Figure 8 [Figure 8: see original paper] shows the cumulative probability distribution of positioning errors for several algorithms when SmartQ tablet 1 is used as the test terminal. When the positioning error is below 3m, the cumulative probability of the DBSCAN-GRNN-LSSVR algorithm is 63.63%, while those of SSD-LSSVR, GRNN-LSSVR, and LSSVR algorithms are 59.09%, 50%, and 50% respectively. At a cumulative probability of 50%, the positioning error values for DBSCAN-GRNN-LSSVR, SSD-LSSVR, GRNN-LSSVR, and LSSVR algorithms are 2.43m, 2.62m, 2.77m, and 2.89m respectively. Figure 8 demonstrates that the DBSCAN-GRNN-LSSVR algorithm outperforms the other algorithms.

3 Conclusion

To address the problem of excessive positioning deviation in WLAN indoor localization caused by heterogeneous terminals, this paper proposes a solution based on the DBSCAN-GRNN-LSSVR algorithm. The RSS values of test terminals and fingerprint database terminals are listed to obtain scatter plots. The DBSCAN density clustering algorithm is used to eliminate noise and boundary points, and the GRNN algorithm is employed to fit the processed scatter plots to obtain the mapping relationship between fingerprint database terminal RSS and test terminal RSS. Test terminal RSS values are converted to fingerprint database terminal RSS values, and test points are located through the LSSVR model. Comparative experiments demonstrate that the proposed DBSCAN-

GRNN-LSSVR algorithm effectively solves the problem of excessive positioning deviation caused by heterogeneous terminals, outperforming the LSSVR algorithm, GRNN-LSSVR calibration algorithm, and SSD-LSSVR calibration algorithm. However, since the algorithm requires a relatively large number of calibration points, reducing the number of calibration points will be the focus of future research.

References

[1] Zhang Minghua, Zhang Shensheng, Cao Jian. Indoor localization based on signal strength in wireless LAN [J]. Computer Science, 2007, 34(6): 68-75.

[2] Qin Siming. Research on Wi-Fi indoor positioning technology based on location fingerprint [D]. Chengdu: University of Electronic Science and Technology of China, 2013.

[3] Dong F, Chen Y, Liu J, et al. A calibration-free localization solution for handling signal strength variance [C]// Proc of International Conference on Mobile Entity Localization and Tracking in Gps-Less Environments. [S. l.]: Springer-Verlag, 2009: 79-90.

[4] Kim Y, Shin H, Chon Y, et al. Smartphone-based Wi-Fi tracking system exploiting the RSS peak to overcome the RSS variance problem [J]. Pervasive & Mobile Computing, 2013, 9(3): 406-420.

[5] Kjaergaard M B, Munk C V. Hyperbolic location fingerprinting: A calibration-free solution for handling differences in signal strength [C]// Proc of the IEEE Int' l Conf. on Pervasive Computing. [S. l.]: IEEE Press, 2008: 110-116.

[6] Zhao Cong. Research on WLAN indoor positioning algorithm based on location fingerprint [D]. Harbin: Harbin Institute of Technology, 2014.

[7] Hossain A K M M, Jin Y, Soh W S, et al. SSD: a robust RF location fingerprint addressing mobile devices' heterogeneity [J]. IEEE Trans on Mobile Computing, 2013, 12(1): 65-77.

[8] Gu Yang, Jiang Xinlong, Liu Junfa, et al. Device-adaptive wireless signal feature extraction and positioning method [J]. Journal of Software, 2014, 25(S2): 12−20.

[9] Luo Qingwang, Shi Yibing, Wang Zhigang, et al. Research on inversion method of physical parameters of ferromagnetic pipeline [J]. Chinese Journal of Scientific Instrument, 2016, 37(1): 9-14.

[10] Dai Liang, Xu Hongke, Chen Ting, et al. MapReduce-based least squares support vector machine regression model [J]. Application Research of Computers, 2015, 32(4): 1060-1064.

[11] Suykens J, Van G T, De B P, et al. Least Square Support Vector Machines [M]. Singapore: World Scientific Publishing, 2002: 71-111.

[12] Tian Zengshan, Wang Xiangyong, Zhou Mu, et al. Cellular network indoor fingerprint location algorithm based on DBSCAN subspace matching [J]. Journal of Electronics & Information Technology, 2017, 39(5): 1157-1163.

[13] Feng Zhenhua, Qian Xuezhong, Zhao Nana. Greedy DBSCAN: An improved DBSCAN algorithm for multi-density clustering [J]. Application Research of Computers, 2016, 33(9): 2693-2700.

[14] Guo Baoqing, Zhu Liqiang, Shi Hongmei. Railway foreign object intrusion detection algorithm based on fast DBSCAN clustering [J]. Chinese Journal of Scientific Instrument, 2012, 33(2): 241-247.

[15] Wu Changcheng, Song Aiguo, Zeng Hong, et al. Hand output force estimation based on sEMG and GRNN [J]. Chinese Journal of Scientific Instrument, 2017, 38(1): 97-104.

[16] Ge Liufei, Li Keqing, Dai Huan. Wireless indoor positioning algorithm based on adaptive GRNN [J]. Computer Engineering, 2016, 42(6): 81-85, 90.

Note: Figure translations are in progress. See original paper for figures.