Postprint: Research on CMAES-SVR-Based WLAN Indoor Positioning Algorithm

Authors: Rao Hua, Wang Zhong, Li Xin

Date: 2018-10-11T00:00:00+00:00

Abstract

To address the issues of low positioning accuracy, poor stability, and insufficient real-time performance in traditional WLAN fingerprint positioning algorithms, this paper proposes a WLAN indoor positioning algorithm based on CMAES-SVR. The algorithm first performs statistical analysis on the received signal strength (RSS) from access points (APs), employs Gaussian filtering for signal preprocessing, and then utilizes the K-means clustering algorithm to partition the positioning area in the original fingerprint database into clusters. Subsequently, it adopts the Covariance Matrix Adaptation Evolution Strategy (CMAES) to optimize the parameters of the Support Vector Regression (SVR) machine, thereby establishing a CMAES-SVR indoor positioning learning model. Through this model, the nonlinear mapping relationship between RSS signals and physical locations is constructed for each positioning sub-region. Finally, the cluster to which the test point belongs is determined, and regression prediction is performed using the trained CMAES-SVR model for that cluster. Experimental results demonstrate that, compared with the WKNN, traditional SVR, and PSO-SVR algorithms, the proposed algorithm exhibits improvements in positioning accuracy, stability, and real-time performance.

Full Text

Preamble

Research on WLAN Indoor Positioning Algorithm Based on CMAES-SVR

Rao Hua, Wang Zhong, Li Xin
(School of Electrical & Information Engineering, Sichuan University, Chengdu 610065, China)

Abstract: Aiming at the problems of low precision, poor stability, and inadequate real-time performance in traditional WLAN fingerprint positioning algorithms, this paper proposes a WLAN indoor positioning algorithm based on CMAES-SVR. The algorithm first performs statistical analysis on the received signal strength (RSS) from access points (APs) and employs Gaussian filtering for signal preprocessing. It then utilizes the K-means clustering algorithm to partition the positioning area in the original fingerprint database. Subsequently, the covariance matrix adaptation evolution strategy (CMAES) is adopted to optimize the support vector regression (SVR) parameters, thereby establishing a CMAES-SVR indoor positioning learning model. This model constructs nonlinear mapping relationships between RSS signals and physical locations within each positioning sub-region. Finally, the algorithm determines the cluster to which a test point belongs and performs regression prediction using the trained CMAES-SVR model for that cluster. Experimental results demonstrate that compared with WKNN, traditional SVR, and PSO-SVR algorithms, the proposed algorithm achieves improvements in positioning accuracy, stability, and real-time performance.

Keywords: indoor positioning; location fingerprint; cluster analysis; covariance matrix adaptation evolution strategy; support vector regression

---

0 Introduction

The widespread adoption of mobile intelligent terminals and the rapid development and proliferation of wireless networks have significantly promoted the growth of location-based service applications. These services demonstrate tremendous development potential and application prospects in navigation, medical rescue, logistics, and other production and living domains. Determining user location information represents the core problem of location-based services. Due to the complex and variable indoor environment, Global Positioning System (GPS) cannot satisfy indoor positioning accuracy requirements. Current indoor positioning technologies can be categorized based on different media into methods using specific devices, WLAN signal-based methods, and mobile sensor-based methods. Among these, WLAN-based indoor positioning has become a research hotspot due to its low cost, easy deployment, high accuracy, and strong universality.

WLAN indoor positioning methods primarily include range-based positioning and fingerprint-based positioning, with the location fingerprint method offering algorithmic flexibility and high positioning accuracy. Typical WLAN fingerprint positioning algorithms include nearest neighborhood (NN), maximum likelihood estimation, kernel methods, artificial neural networks, and support vector regression (SVR). SVR comprehensively considers the VC dimension of the learning function and training error, seeking a learning function that minimizes actual risk and improves generalization capability. Through kernel function mapping, SVR establishes a reliable nonlinear mapping relationship between RSS signals and position coordinates, significantly enhancing positioning accuracy.

Numerous scholars have conducted in-depth research on this topic. The literature proposes a PCA-LSSVR positioning algorithm that employs principal component analysis for data dimensionality reduction and decorrelation, using least squares support vector regression (LS-SVR) to establish a nonlinear relationship between fingerprint feature data and positions. Another study proposes a positioning algorithm fusing Kalman filtering with SVR, which utilizes the relationship between consecutive positions to correct movement trajectories and improve positioning results. A different approach proposes an SVR-based positioning error correction algorithm that establishes a nonlinear relationship between RSS positioning results and positioning errors, using the calculated error to correct the final position. Yet another work proposes an improved support vector regression algorithm that introduces correction coordinates to reduce errors from independently constructing x and y coordinates, thereby enhancing the correlation between two-dimensional position information and RSS. While these algorithms can reduce positioning errors to some extent, they fail to consider the impact of SVR hyperparameters on generalization performance, thus affecting positioning accuracy in complex indoor environments.

Currently, particle swarm, ant colony, and genetic algorithms have demonstrated advantages in SVR parameter optimization, such as insensitivity to objective function types and powerful stochastic optimization capabilities. However, influenced by population size, these methods suffer from high time complexity, low efficiency, and premature convergence issues. To address these problems, this paper proposes a CMAES-SVR-based indoor positioning algorithm. During the offline phase, the algorithm first performs statistical analysis on AP received signal strength, uses a Gaussian filtering model for data preprocessing to enhance data stability, and employs the K-means clustering algorithm to partition the original fingerprint database into several sub-regions. It then adopts covariance matrix adaptation evolution strategy (CMAES) to optimize SVR parameters, establishing a CMAES-SVR indoor positioning learning model. This model trains reference points in each sub-region to establish an offline fingerprint database containing nonlinear mapping relationships between RSS and physical locations. During the online phase, the algorithm collects RSS signals at test points in real time, preprocesses them through Gaussian filtering, determines their belonging cluster, and uses the trained CMAES-SVR model to estimate the test point' s location information. Experimental simulation analysis demonstrates that CMAES exhibits excellent global search capability and high optimization efficiency in the SVR parameter optimization process, and the CMAES-SVR positioning algorithm achieves high positioning accuracy.

1.1 Fingerprint Database Clustering and Partitioning

For complex and variable indoor environments, fingerprint database-based indoor positioning methods require a relatively large fingerprint database, leading to substantial computational load during real-time positioning and susceptibility to over-learning. Therefore, this paper proposes clustering the fingerprint database to reduce online positioning computation time. Before clustering the original fingerprint database, statistical analysis of RSS signals collected from a single reference point (RP) and a single AP reveals significant fluctuations due to environmental factors. To remove redundant data and establish a stable and reliable positioning model, signal preprocessing is necessary.

In the positioning area, L RPs are selected at regular intervals, with RP_i' s location coordinates denoted as (x_i, y_i). M reliable APs are deployed. For the RSS signal from the j-th AP at reference point i, N samples are collected, yielding RSS signal $rss_{i,j,k}$ where $k = 1, 2, ..., N$. The signal follows a Gaussian distribution, i.e., $rss_{i,j,k} \sim N(\mu_{i,j}, \sigma_{i,j}^2)$. The mean and variance are calculated as:

$$\mu_{i,j} = \frac{1}{N}\sum_{k=1}^{N} rss_{i,j,k}$$

$$\sigma_{i,j}^2 = \frac{1}{N}\sum_{k=1}^{N}(rss_{i,j,k} - \mu_{i,j})^2$$

The interval $[\mu_{i,j} - 1.65\sigma_{i,j}, \mu_{i,j} + 1.65\sigma_{i,j}]$ is used as the high-probability retention region, filtering out low-probability redundant data outside this range. Mean filtering is then applied to this region to obtain the preprocessed RSS feature vector $R_i = [rss_{i,1}, rss_{i,2}, ..., rss_{i,M}]$. The fingerprint feature composed of the preprocessed RSS feature vector and position coordinates is denoted as $[R_i, (x_i, y_i)]$.

The K-means clustering algorithm is then applied to partition the original fingerprint database with fingerprint features, grouping RPs with high similarity into the same cluster. K-means clustering utilizes the distances between data points and cluster centers. The algorithm randomly selects k data points from the set as initial cluster centers, calculates the distance from each data point to the k centers, and assigns each point to the nearest cluster:

$$C_j = \{rss_i \mid \arg\min_j \|rss_i - c_j\|\}$$

where C_j represents the j-th cluster and $\|rss_i - c_j\|$ denotes the Euclidean distance. The cluster centers are then recalculated as:

$$c_j^{\text{new}} = \frac{1}{N_j}\sum_{rss_i \in C_j} rss_i$$

where c_j^{new} becomes the new center of the cluster. This process repeats until the cluster centers no longer change or fall below a given threshold.

Through clustering, the original fingerprint database is divided into k positioning sub-regions. The specific steps are as follows:

a) Initialize cluster centers. Select k fingerprint features from the fingerprint database as initial cluster centers (to avoid clustering dispersion or concentration caused by random initial centers, k initial cluster centers are uniformly selected from positioning sub-regions to maintain consistency with physical space).

b) Calculate the Euclidean distance between the RSS values corresponding to each fingerprint feature and the k cluster centers, and assign each feature to the nearest cluster.

c) After processing all fingerprints, obtain new clusters and update the cluster centers.

d) Repeat steps b) and c) until the k cluster centers no longer change or are smaller than a given threshold, terminate iteration, and output the k cluster centers and corresponding fingerprint feature sets.

Through K-means clustering of fingerprint features, k positioning sub-regions are formed. Model training is performed separately for each sub-region to obtain the mapping relationship between RSS signals and physical locations, forming a new fingerprint database. During the online phase, for RSS signals collected in real time, after preprocessing, the algorithm compares their Euclidean distance to each sub-region’ s cluster center, finds the nearest cluster center, and localizes the test point to the positioning sub-region where this center resides. The fingerprint database learning algorithm of the indoor positioning system is completed within the i-th cluster.

1.2 SVR Positioning Model

After partitioning the original fingerprint database into k positioning sub-regions, the SVR algorithm is used to construct the nonlinear relationship between received signal strength and position coordinates.

Standard SVR is single-output, while reference point coordinates are two-dimensional (x_i, y_i). A multi-output approach is adopted to implement multi-output SVR regression. The input fingerprint data is represented as:

$$R_i = [rss_{i,1}, rss_{i,2}, ..., rss_{i,j}, ..., rss_{i,M}]$$

where $i = 1, 2, ..., N$ and $j = 1, 2, ..., M$.

First, through a nonlinear mapping function $\Phi(R)$, the input R is mapped to a high-dimensional kernel feature space F. Then, in the high-dimensional space, linear regression functions for the x and y coordinates are constructed separately.

Taking the x-coordinate regression function as an example, the optimal linear regression estimation function between position coordinates and received signal strength in space F is constructed as:

$$f(x) = w^T \cdot \Phi(R) + b$$

where w is the weight coefficient and b is the bias constant. $\Phi(R)$ is the nonlinear mapping function that maps the input fingerprint data RSS signals to the high-dimensional space.

Based on the principle of structural risk minimization, the SVR algorithm solves the convex quadratic optimization problem:

$$\min_{w,b,\xi,\xi^*} J(w,\xi,\xi^*) = \frac{1}{2}w^T w + c\sum_{i=1}^{N}(\xi_i + \xi_i^*)$$

subject to:

$$\begin{cases} y_i - w^T\Phi(R_i) - b \leq \varepsilon + \xi_i \\ w^T\Phi(R_i) + b - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0, \quad i = 1, 2, ..., N \end{cases}$$

where ξ_i and ξ_i^* are slack variables, ε is the insensitive loss function controlling regression error, and c is the penalty factor satisfying $c > 0$. The empirical risk size is determined by $\sum_{i=1}^{N}(\xi_i + \xi_i^*)$, which affects the VC dimension and consequently the confidence range.

By introducing Lagrange multipliers into the convex quadratic optimization problem, we obtain:

$$L(w,b,\xi,\xi^*,\alpha,\alpha^*,\eta,\eta^*) = \frac{1}{2}w^T w + c\sum_{i=1}^{N}(\xi_i + \xi_i^*) - \sum_{i=1}^{N}\alpha_i(\varepsilon + \xi_i + y_i - w^T\Phi(R_i) - b) - \sum_{i=1}^{N}\alpha_i^*(\varepsilon + \xi_i^* - y_i + w^T\Phi(R_i) + b)$$

where α_i and α_i^* are Lagrange multipliers satisfying $\alpha_i, \alpha_i^* \geq 0$.

To avoid the curse of dimensionality, SVR models employ kernel functions. This paper selects the Gaussian kernel function as the SVR model' s kernel:

$$k(R_i, R_j) = \exp\left(-\frac{\|R_i - R_j\|^2}{2\delta^2}\right)$$

The final SVR positioning function can be obtained as:

$$f(x) = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) k(R_i, R) + b$$

During the offline phase, two independent SVR positioning functions are obtained through corresponding sample training and parameter searching, outputting x and y coordinates separately.

1.3 Covariance Matrix Adaptation Evolution Strategy (CMAES)

The CMAES algorithm is an evolution strategy algorithm for solving complex nonlinear, non-convex optimization problems, demonstrating good optimization performance in global optimization with advantages of fast convergence, strong robustness, and rotational invariance. The basic steps of the CMAES algorithm are as follows:

a) Parameter setting and initialization. Set the number of parent individuals μ and offspring individuals λ in the population, the recombination weights ω_i required for adaptive adjustment, the maximum iteration number G, and the initial population mean $m^{(0)} \in \mathbb{R}^N$, step size $\sigma^{(0)} \in \mathbb{R}^+$, evolution path $p_\sigma^{(0)} = 0$, $p_c^{(0)} = 0$, and covariance matrix $C^{(0)} = I_{N \times N} \in \mathbb{R}^{N \times N}$.

b) Sampling operation. Sample in the optimization problem solution space through Gaussian distribution $N(0, C^{(g)})$ to generate a population distribution composed of λ individuals $x_j^{(g+1)}$, which correspond to the population in the optimization algorithm:

$$x_j^{(g+1)} \sim m^{(g)} + \sigma^{(g)} N(0, C^{(g)}), \quad j = 1, 2, ..., \lambda$$

where $m^{(g)}$ is the distribution mean of the g-th generation population; $C^{(g)}$ is the covariance matrix of the population distribution; $\sigma^{(g)}$ is the global step size of the g-th generation; $x_j^{(g+1)}$ is the j-th individual of the (g+1)-th generation population; $B^{(g)}$ is an orthogonal matrix whose column vectors form an orthonormal basis of eigenvectors of $C^{(g)}$, used for rotating the population distribution hyper-ellipsoid; $D^{(g)}$ is a diagonal matrix with diagonal elements being the square roots of eigenvectors corresponding to each column vector of $B^{(g)}$, used for scaling the population distribution hyper-ellipsoid.

c) Evaluation and selection. Evaluate and rank the fitness of offspring individuals one by one, selecting the μ best individuals to form the current optimal offspring population. Calculate the corresponding fitness function values $f(x_j^{(g+1)})$ for the newly generated solutions, and sort these objective function values:

$$f(x_{1:\lambda}^{(g+1)}) \leq f(x_{2:\lambda}^{(g+1)}) \leq ... \leq f(x_{\lambda:\lambda}^{(g+1)})$$

d) Parameter update

(a) Mean update:

$$m^{(g+1)} = \sum_{i=1}^{\mu} \omega_i x_{i:\lambda}^{(g+1)}$$

where ω_i are the set weights satisfying $\sum_{i=1}^{\mu} \omega_i = 1$ and $\omega_1 \geq \omega_2 \geq ... \geq \omega_\mu \geq 0$.

(b) Covariance matrix adaptation. First update the evolution path of the covariance matrix, then adaptively adjust the covariance matrix C:

$$p_c^{(g+1)} = (1 - c_c)p_c^{(g)} + \sqrt{c_c(2 - c_c)\mu_{eff}} \cdot \frac{m^{(g+1)} - m^{(g)}}{\sigma^{(g)}}$$

$$C^{(g+1)} = (1-c_1-c_\mu)C^{(g)} + c_1 p_c^{(g+1)}(p_c^{(g+1)})^T + c_\mu \sum_{i=1}^{\mu} \omega_i \frac{x_{i:\lambda}^{(g+1)} - m^{(g)}}{\sigma^{(g)}} \left(\frac{x_{i:\lambda}^{(g+1)} - m^{(g)}}{\sigma^{(g)}} \right)^T$$

where c_c is the learning rate for p_c update; c_1 and c_μ are the learning rates for "rank-1" and "rank-μ" updates of C, respectively; $\mu_{eff} = \frac{1}{\sum_{i=1}^{\mu} \omega_i^2}$.

(c) Step size update. First update the step size evolution path p_σ, then adaptively adjust the cumulative step size:

$$p_\sigma^{(g+1)} = (1 - c_\sigma)p_\sigma^{(g)} + \sqrt{c_\sigma(2 - c_\sigma)\mu_{eff}} \cdot C^{(g)-\frac{1}{2}} \cdot \frac{m^{(g+1)} - m^{(g)}}{\sigma^{(g)}}$$

$$\sigma^{(g+1)} = \sigma^{(g)} \exp\left(\frac{c_\sigma}{d_\sigma} \left(\frac{\|p_\sigma^{(g+1)}\|}{E\|N(0, I)\|} - 1 \right) \right)$$

where c_σ is the learning rate for p_σ update; d_σ is the damping parameter.

e) Termination condition check. If the maximum iteration number G is reached, stop and output the optimal solution and optimal value; otherwise, return to step b).

Since CMAES demonstrates excellent optimization performance in solving global optimization problems, this paper employs CMAES to optimize SVR learning algorithm parameters for fingerprint information and position coordinates in indoor positioning. The specific steps are as follows:

a) Parameter setting and initialization. Set CMAES parameters: maximum iteration number G, SVR parameters including penalty parameter c, kernel function parameter δ, and sensitivity factor ε with their upper and lower bounds. Use data from k positioning sub-regions as the training set.

b) Population sampling. Sample the multidimensional normal distribution to generate a population distribution composed of λ individuals $x_j^{(g+1)}$, as well as the individual value ranges.

c) Calculate fitness for each population individual. Use (c, δ, ε) as training parameters, train the SVR model using the training set, and use the average positioning error as the individual' s fitness:

$$\text{AvgErr}(c, \delta, \varepsilon) = \frac{1}{n} \sum_{i=1}^{n} \sqrt{(x_i' - x_i)^2 + (y_i' - y_i)^2}$$

d) Compare individual fitness values, select the μ individuals with the smallest fitness values to form the current optimal subpopulation.

e) Parameter update. Update the mean m, covariance matrix C, and step size σ.

f) Check if the maximum iteration number G is reached. If yes, stop and output the optimal parameters $(c^*, \delta^*, \varepsilon^*)$ and optimal fitness value AvgErr; otherwise, return to step c).

g) Perform CMAES-SVR model training for each of the k positioning sub-regions.

The CMAES-SVR indoor positioning model is shown in Figure 1 [Figure 1: see original paper].

2.1 Experimental Scenario

The experimental scenario in this paper is the second floor of Building B of the Basic Teaching Building at Sichuan University. The local floor plan is shown in Figure 2 [Figure 2: see original paper]. In areas A and B, 30 reference points are uniformly deployed with 3-meter spacing, and 20 test points are randomly selected. Ten relatively stable APs are selected throughout the experimental scene. A Huawei Nova mobile terminal is used for RSS data collection. RSS signals are collected at different times over three days, with a sampling interval of 1s. Each reference point and test point continuously collects 40 samples as one group of data, with 5 groups collected, resulting in 200 RSS signal samples per AP. The initial population mean is $m^{(0)} = (c^{(0)}, \delta^{(0)}, \varepsilon^{(0)})$, with c and ε set to $[0.01, 100]$ and $[10^{-5}, 10^{-2}]$ respectively. The maximum iteration number is set to 100 to enable the fitness function AvgErr to converge quickly with better convergence performance.

2.2 Experimental Results Analysis

This paper conducts extensive statistical analysis on measured data from reference points and test points. RSS signals collected from a single RP and a single AP exhibit approximately Gaussian statistical characteristics. Figure 3 [Figure 3: see original paper] shows the probability distribution histogram of 200 RSS samples from a single AP at a certain RP. Therefore, Gaussian filtering is first applied to remove low-probability and high-interference data, reducing RSS signal variance and enhancing the stability and reliability of fingerprint data, followed by mean filtering. As shown in Figure 4 [Figure 4: see original paper], after Gaussian filtering, low-probability data with large deviations are removed and curve fluctuations decrease.

After preprocessing, RSS signals form the original fingerprint database with physical positions. For fingerprint database clustering and partitioning, the selection of k value significantly impacts positioning accuracy. A small k value results in low information similarity within each cluster, while an excessively large k value increases the number of positioning sub-regions, reduces the range of each sub-region, increases similarity between classes, and decreases positioning accuracy. To effectively partition the fingerprint database positioning sub-regions, this paper compares the system' s average positioning error when k takes values in the interval $[1, 10]$. The traditional SVR positioning algorithm is used for location learning and prediction. Figure 5 [Figure 5: see original paper] shows the curve of average positioning error varying with k value.

Analysis reveals that when $k = 5$, the system' s average positioning error reaches its minimum. When $k > 5$, the average positioning error gradually increases. Therefore, during the offline phase when partitioning the original fingerprint database, k is set to 5, and CMAES-SVR model training is performed separately for each positioning sub-region. To verify the effectiveness of the CMAES-SVR algorithm, comparisons are made with WKNN ($K = 6$), traditional SVR algorithm, and PSO-SVR algorithm. For PSO-SVR, the initial inertia weight is set to 0.8, the terminal inertia weight to 0.3, learning factors c_1 and c_2 to 1.7 and 1.5 respectively, and maximum iteration number to 100. All three algorithms use fingerprint data from the same positioning sub-regions. The positioning performance of several algorithms is shown in Figure 6 [Figure 6: see original paper].

As shown in Figure 6, when the number of samples is 80, the average positioning error of CMAES-SVR is 1.68m, while those of WKNN, traditional SVR, and PSO-SVR are 2.53m, 2.38m, and 2.21m respectively. When the number of samples exceeds 90, the average positioning error of CMAES-SVR is smaller than that of the other algorithms even when they use 150 samples.

Figure 7 [Figure 7: see original paper] shows the cumulative probability distribution of average positioning errors for several indoor positioning algorithms when the number of RSS signal samples is 100. When the average positioning error is less than or equal to 2m, the cumulative probabilities of CMAES-SVR, PSO-SVR, traditional SVR, and WKNN are 76.8%, 71.25%, 64.8%, and 62.16% respectively. When the average positioning error is less than or equal to 3m, the cumulative probabilities are 91.2%, 87.36%, 83.1%, and 77.6% respectively. Additionally, by comparing the standard deviations of positioning errors among several algorithms, the proposed CMAES-SVR algorithm has a variance of 1.16, which is smaller than other algorithms. This demonstrates that under the same experimental environment, the CMAES-SVR algorithm achieves higher positioning accuracy and stability.

To analyze the impact of clustering on algorithmic computational complexity, the online computational complexity of several positioning algorithms is compared. For WKNN matching algorithms, computational complexity is proportional to the fingerprint database size. Clustering reduces the position fingerprints to be matched online from 30 reference points in the entire database to an average of 6 reference points, effectively reducing the search space by approximately 78.6%. For CMAES-SVR and PSO-SVR learning algorithms, online positioning computational load is determined by the SVR algorithm. Clustering makes RSS signal patterns more concentrated, simplifies the SVR learning function mapping relationship, and significantly reduces the number of support vectors, lowering SVR positioning function online computational complexity by approximately 65.6%. At this point, WKNN has the lowest computational complexity but poorer positioning accuracy, while the proposed CMAES-SVR positioning algorithm improves positioning accuracy while reducing online computational complexity.

Table 1 Positioning errors of different algorithms

Algorithm	<3m Positioning	70% Positioning	90% Positioning
WKNN			
Traditional SVR			
PSO-SVR			
CMAES-SVR			

3 Conclusion

This paper proposes a WLAN indoor positioning algorithm based on CMAES-SVR to address issues of RSS signal redundancy and noise in complex indoor environments, the low learning efficiency and high computational complexity caused by large fingerprint databases, and the significant impact of SVR hyperparameters on positioning accuracy. The algorithm first analyzes the probability distribution of RSS signals and employs Gaussian filtering for signal preprocessing to remove low-probability, high-interference data, thereby reducing signal variance. The K-means algorithm partitions the fingerprint database into blocks to reduce search space, effectively preventing SVR learning algorithm over-fitting and reducing computational complexity. CMAES is used to optimize SVR parameters, and model training is performed for each positioning sub-region to construct an improved SVR model with nonlinear mapping relationships between RSS signals and physical location information. Experimental results and comparative analysis demonstrate that the CMAES-SVR algorithm achieves higher positioning accuracy and stability compared with WKNN, traditional SVR, and PSO-SVR algorithms.

References

[1] Xi Rui, Li Yujun, Hou Mengshu. Survey on indoor localization [J]. Computer Science, 2016, 43 (4): 1-6, 32.

[2] Belay A A, Yayeh Y, Berie G, et al. Indoor localization using K-nearest neighbor and artificial neural network back propagation algorithms [C]// Proc of the 27th Wireless and Optical Communication Conference. [S. I.]: IEEE Press, 2018: 1-2.

[3] Ilyas M, Bayat O, Ileri O. Indoor location estimation by using MLE based algorithm on smallcell networks [C]// Proc of the 23nd Signal Processing and Communications Applications Conference. [S. I.]: IEEE Press, 2015.

[4] Kong Yongping, Zhong Zhiming, Yang Guanglong, et al. A non-parametric kernel method for CDMA2000 network indoor localization using multiple observations [C]// Proc of International Conference on Information Network. Piscataway, [S. I.]: IEEE Press, 2012: 97-101.

[5] Gharghan S K, Nordin R, Ismail M, et al. Accurate wireless sensor localization technique based on hybrid PSO-ANN algorithm for indoor and outdoor track cycling [J]. IEEE Sensors Journal, 2016, 16 (2): 529-541.

[6] Mao Rui, Xiang Peng, Zhang Dian. Precise transceiver-free localization in complex indoor environment [J]. China Communications, 2016, 13 (5): 28-.

[7] Zhang Yong, Huang Jie, Xu Keyu. Indoor positioning algorithm for WLAN based on principal component analysis and least square support vector regression [J]. Chinese Journal of Scientific Instrument, 2015, 36 (2): 408-414.

[8] Zhang Yu, Dong Lian, Lai Lei, et al. Study of indoor positioning method based on combination of support vector regression and kalman filtering [J]. International Journal of Future Generation Communication and Networking, 2016, 9 (3): 201-214.

[9] Guo Jian, Liu Quanjing, Zhao Zhikai. Positioning error correction algorithm based on SVR [J]. Application Research of Computers, 2017, 33 (8): 1-4.

[10] Yao Yingbiao, Mao Weiyong, Yao Ruili, et al. Indoor positioning algorithm based on improved support vector regression [J]. Chinese Journal of Scientific Instrument, 2017, 38 (9): 2112-2119.

[11] Munlin M, Anantathanavit M. New social-based radius particle swarm optimization [C]// Proc of the 12th IEEE Conference on Industrial Electronics and Applications. [S. I.]: IEEE Press, 2017: 838-843.

[12] Dai Chunni. SVM visual classification based on weighted feature of genetic algorithm [C]// Proc of the 6th International Conference on Intelligent Systems Design and Engineering Applications. [S. I.]: IEEE Press, 2015.

[13] Hansen N, Ros R, Mauny N, et al. Impacts of invariance in search: when CMA-ES and PSO face ill-conditioned and non-separable problems [J]. Applied Soft Computing, 2011, 11 (5): 1667−1679.

[14] Chen Yingying, Yang Jie, Trappe W, et al. Detecting and localizing identity-based attacks in wireless and sensor networks [J]. IEEE Trans on Vehicular Technology, 2010, 59 (5): 2418-2434.

[15] Fang S H, Lin T N. A dynamic system approach for radio location fingerprinting in wireless local area networks [J]. IEEE Trans on Communications, 2010, 58 (4): 1020-1025.

[16] Huang Yafei, Liang Ximing, Chen Yixiong. Constrained evolutionary optimization of CMAES based on adaptive tradeoff model [J]. Journal of Central South University: Science and Technology, 2012, 43 (09): 3478-3484.

[17] Kovitz J M, Yahya. R S. A comparative study between CMA evolution strategies and Particle Swarm Optimization for antenna applications [C]// Proc of US National Committee of URSI National Radio Science Meeting, [S. I.]: IEEE Press, 2013.

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