IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received July 16, 2019, accepted August 18, 2019, date of publication August 26, 2019, date of current version September 11, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2937464

A Novel Clustering Algorithm for Wi-Fi Indoor Positioning

JIN REN[1,2], YUNAN WANG[1], CHANGLIU NIU[1], WEI SONG[2], AND SONGYANG HUANG[1]

[1]School of Information Science and Technology, North China University of Technology, Beijing 100144, China
[2]Beijing Key Laboratory on Integration and Analysis of Large-Scale Stream Data, Beijing 100144, China

Corresponding author: Jin Ren (rj@ncut.edu.cn)

This work was supported in part by 2017 Annual Outstanding Young Teacher Training Program Project of North China University of Technology under Grant XN019009, in part by Scientific Research Project of Beijing Educational Committee under Grant KM201710009004, in part by 2018 Science and Technology Activities Project for College Students of North China University of Technology under Grant 110051360007, in part by Research Project on Teaching Reform and Curriculum Construction of North China University of Technology under Grant 18XN009-011, in part by 2019 Beijing University Student Scientific Research and Entrepreneurship Action Plan Project under Grant 218051360019XN004, in part by 2019 Education and Teaching Reform General Project of North China University of Technology under Grant 108051360019XN141/021, and in part by 2019 Fundamental Research Funds for Beijing Universities under Grant 110052971921/004.

ABSTRACT In recent years, the Wi-Fi-based indoor positioning technology has become a research hotspot. This technology mainly locates the indoor Wi-Fi based on the received signal strength indicator (RSSI) signals. The most popular Wi-Fi positioning algorithm is the k-nearest neighbors (KNN) algorithm. Due to the excessive amount of RSSI data, clustering algorithms are generally adopted to classify the data before KNN positioning. However, the traditional clustering algorithms cannot maintain data integrity after the classification. To solve the problem, this paper puts forward an improved public c-means (IPC) clustering algorithm with high accuracy in indoor environment, and uses the algorithm to optimize the fingerprint database. After being trained in the database, all fingerprint points were divided into several classes. Then, the range of each class was determined by comparing the cluster centers. To optimize the clustering effect, the points in the border area between two classes were allocated to these classes simultaneously, pushing up the positioning accuracy in this area. The experimental results show that the IPC clustering algorithm achieved better accuracy with lighter computing load than FCM clustering and k-means clustering, and could be coupled with KNN or FS-KNN to achieve good positioning effect.

INDEX TERMS Wi-Fi, indoor positioning, improved public c-means (IPC) clustering algorithm, the k-nearest neighbors (KNN) algorithm.

I. INTRODUCTION

Recent years has seen a growing demand for location services, especially indoor positioning. In the indoor environment, it is difficult to provide location services via the global positioning system (GPS) [1]. Currently, indoor positioning is mainly achieved through the following technologies: ultrasonic positioning, radio frequency positioning and ultra-broadband positioning. Despite their good effect, these technologies suffer from high cost, small range and low flexibility.

The Wi-Fi is a cost-effective method for indoor positioning. Since it is almost ubiquitous in indoor environment, there is no need to deploy additional hardware. However, the Wi-Fi is not designed specifically for positioning [2]. It is incompatible with the bases with the traditional location methods, such as angle of arrival (AOA) and time difference of arrival (TDOA) [3]. One of the research hotspots in Wi-Fi location services lies in fingerprint positioning.

The Wi-Fi-based fingerprint positioning has two phases, namely, offline collection and online positioning. During offline collection, a Wi-Fi signal detection device collects the received signal strength indicator (RSSI) signals from all access points (APs) in the positioning range, and generates a fingerprint database based on the collected data. Of course, the collected data are filtered to ensure their stability and reliability. During online positioning, the real-time measured data are compared with the fingerprint database, and the final location is determined by algorithms like Bayesian algorithm and the k-nearest neighbors (KNN) algorithm. The details

The associate editor coordinating the review of this article and approving it for publication was Dharm Singh Jat.

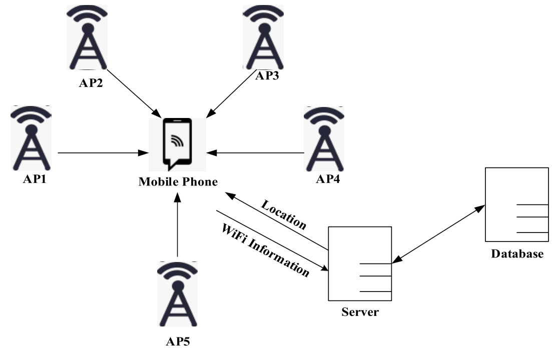

FIGURE 1. Sketch map of Wi-Fi-based fingerprint positioning.

of Wi-Fi-based fingerprint positioning are given in Figure 1 below.

Fingerprint positioning either relies on deterministic algorithms or uses probability algorithms. The former compare the signal features (e.g. vector r) with the calculated statistics in the fingerprint database [4], while the latter compute the probability distribution of the obtained RSSI signals [5]. In this paper, the deterministic algorithms are selected to estimate the target's location, including the nearest neighbor (NN) algorithm [6], the KNN and the weighted KNN (WKNN) [7], [8]. The NN is the basic algorithm for indoor positioning. With a simple structure, the algorithm can achieve a high positioning accuracy. The algorithm can be described as [6]:

$$d_j = (\sum_{i=1}^{n} |rss_i - \overline{rss}_i|^w)^{\frac{1}{w}} \quad j = 1, 2, \ldots, m \tag{1}$$

where, n is the number of APs in the detection range; is the RSSI of the test point received by the APs; is the RSSI of the fingerprint of the reference point in the fingerprint database; is the Euclidean distance between the test point and the j-th reference point. If $w = 1$, the Mahalanobis distance should be calculated; If $w = 2$, the Euclidean distance should be calculated. Then, the distances to all reference points should be sorted, yielding the smallest d. The coordinates of the fingerprint corresponding to this distance should be adopted as the final result. The KNN adds one step to the NN: computing the average of the values of k nearest neighbors. The algorithm can be depicted as [6]:

$$d_j = (\sum_{i=1}^{n} |rss_i - \overline{rss}_i|^2)^{\frac{1}{2}} \tag{2}$$

$$(\overline{x}, \overline{y}) = \frac{1}{k} \sum_{i=1}^{k} (x_i, y_i) \tag{3}$$

where, k is the number of reference points; $(\overline{x}, \overline{y})$ are the coordinates of reference point i; (x_i, y_i) is the fingerprint coordinates of the minimum distance of k nearest neighbors. In the WKNN, the coordinates of each nearest neighbor are multiplied by the weight of the neighbor, i.e. the inverse of the distance.

Comparatively, the WKNN cannot achieve the desirable positioning effect [8]. As for the KNN, this algorithm enjoys immense popularity in fingerprint localization [9], but still has ample room for improving efficiency and accuracy. For example, the Euclidean distance computed by the KNN, using the RSSIs of the test point and the fingerprint point, may be smaller than the actual distance; it is inefficient to pinpoint the location of a target by the KNN, as each positioning needs to use all fingerprint points in the database. To improve the positioning performance, the KNN should be optimized with other algorithms [10].

Clustering algorithms are often introduced to improve the efficiency of positioning algorithms. Typical examples include fuzzy c-means (FCM) clustering [11], k-means clustering [12], spectral clustering, etc. Cho Rong Park [13] adopted the improved k-means clustering to enhance the efficiency and accuracy of the KNN. Hao Jiewang [14] compared the positioning effects of three clustering algorithms, namely, spatial clustering, k-means clustering and affinity propagation clustering. Hatice Arslan [15] combined the FCM clustering and the whale optimization algorithm (WOA) into a highly effective clustering method. Nevertheless, the popular clustering algorithms have a common problem: the declining accuracy after clustering [16].

In light of the above, this paper puts forward an improved public c-means (IPC) clustering algorithm with high accuracy in indoor environment, and uses the algorithm to optimize the fingerprint database. After being trained in the database, all fingerprint points were divided into several classes. Then, the range of each class was determined by comparing the cluster centers. To optimize the clustering effect, the points in the border area between two classes were allocated to these classes simultaneously, pushing up the positioning accuracy in this area.

The remainder of this paper is organized as follows: Section 2 describes the FCM clustering, introduces the KNN based on feature scaling (FS-KNN), and proposes the IPC clustering algorithm; Section 3 verifies the performance of the IPC clustering algorithm through experiments; Section 4 wraps up this paper with conclusions.

II. ALGORITHM DESCRIPTION

A. FCM CLUSTERING

The FCM is a combination of the fuzzy set theory and the c-means algorithm. The fuzzy set theory mainly extends the membership function from the interval {0, 1} to the interval of [0, 1]. In the FCM, n vectors ($i = 1, 2, \ldots, n$) are divided into c fuzzy classes, and the cluster center is determined for each class, such as to minimize the non-similarity index. The value function of this index can be expressed as [17]:

$$J_f = \sum_{j=1}^{c} \sum_{i=1}^{k} \left[u_j(x_i)\right]^b \left\|x_i - m_j\right\|^2 \tag{4}$$

where, $u_j(x_i)$ is the membership function of the i-th sample in class j; $\left\|x_i - m_j\right\|^2$ is the Euclidean distance between the x-th data and the j-th cluster center; $b = 2$ is the weighting coefficient. The minimization of the value function is subjected to the following constraints [17]:

$$u_j(x_i) = \frac{\|x_i - m_j\|^{-2/(b-1)}}{\sum_{s=1}^{k} \|x_i - m_s\|^{-2/(b-1)}} \tag{5}$$

$$m_j = \frac{\sum_{i=1}^{n} [u_j(x_i)]^b x_i}{\sum_{i=1}^{n} [u_j(x_i)]^b} \tag{6}$$

where, $\sum_{i=1}^{n} [u_j(x_i)]^b = 1$; m_j is the cluster center of the j-th class. Under these constraints, the FCM completes clustering in the following steps:

1. Set the number of classes c and the maximum number of iterations, and initialize the cluster centers;
2. Compute the membership function by equation (5) using the current cluster center, and compute the new cluster center by equation (6) using the current membership function.
3. Perform iterations until reaching the maximum number of iterations, and sort the data.
4. Calculate the After sorting the data, calculate the Euclidean distance between the test point and each cluster center, and take the class with the minimal Euclidean distance as the new database, and execute the positioning algorithm.

If applied in fingerprint positioning, the FCM will see a decline in accuracy when the computation is reduced.

B. FS-KNN

The traditional WKNN cannot effectively control the error in RSSI reception, because a fixed distance weight is not beneficial to all fingerprint points. Inspired by feature scaling (FS), this paper divides online RSSI values into n intervals, and assigns a unique weight to each interval. The improved KNN (FS-KNN) can be expressed as [18], [19]:

$$d_l = \sqrt{\sum_{n=1}^{N} (o_{l,n} - v_n)^2 \times w(v_n)} \tag{7}$$

where, $o_{l,n}$ and v_n are the reference point RSSI vector and the online RSSI vector, respectively; is a weight vector representing the weight of the RSSI range received from different APs. The FS-KNN is more accurate than the traditional WKNN [8]. For instance, the RSSI of the signal received by a mobile device is generally between -30 and -80 dBm. If it falls below -80 dBm, then the signal is either too weak or not received. Here, the $[-30, -80]$ range is split evenly to 10 intervals, each of which is assigned with the weight $a_1, a_2, \ldots, a_{10}$. If the RSSI of the signal received from the second AP is -31 dBm, then $w(v_n) = a_1$ (n = 2). Then, a weighted vector can be constructed as $(a_1, a_2, \ldots, a_{10})$. To get the best vector weight, the sum of distance error T can be defined as:

$$T = \sum_{i=1}^{n} \sqrt{(x_i - x_i')^2 + (y_i - y_i')^2} \tag{8}$$

where, x_i, y_i are the real coordinates of the reference points; x_i', y_i' the coordinates computed by the FS-KNN; n is the number of reference points. If the T value is greater than the threshold T', the vector weight should be adjusted and the new T value should be computed; if it is greater than the new T value, the T value should be replaced with the new value and saved. The above steps should be repeated until reaching the maximum number of iterations (1,000) or when the T is smaller than T'. Then, the final result is the best vector weight.

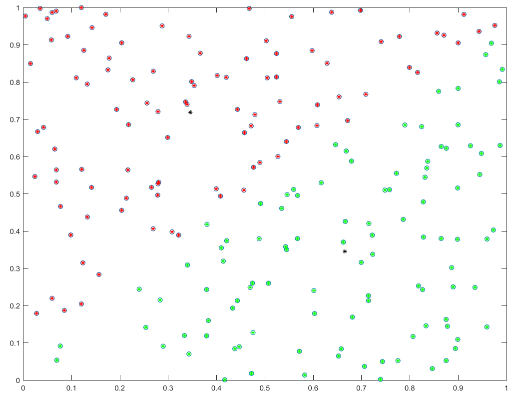

FIGURE 2. Distribution of the fingerprint points after FCM clustering.

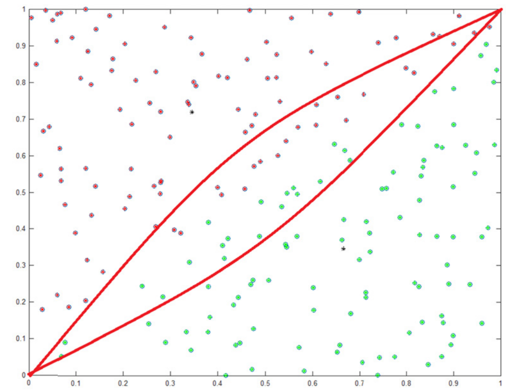

FIGURE 3. The border between the two areas.

C. THE IPC CLUSTERING ALGORITHM

To achieve a high positioning accuracy after clustering, the IPC clustering algorithm was designed to create a common area after clustering. To begin with, the fingerprint points in the database were divided into two classes by FCM clustering.

The classification results in Figure 2 show that the 200 random 2D data points were divided into a red area (Class A) and a green area (Class B). The border between the two areas was circled in red (Figure 3) and subjected to further analysis.

As shown in the Figure 3, the positioning accuracy of the blue points in the border area decreased with their distance to the border.

According to Figure 4, if the blue points are positioned by the KNN as a part of Class A, the final locations will fall into the red area; if these points are positioned by the KNN as a part of Class B, the final locations will fall into the green area. Thus, the computed coordinates differ greatly from the actual coordinates for the points in the border between different classes. If these points are positioned by the FCM

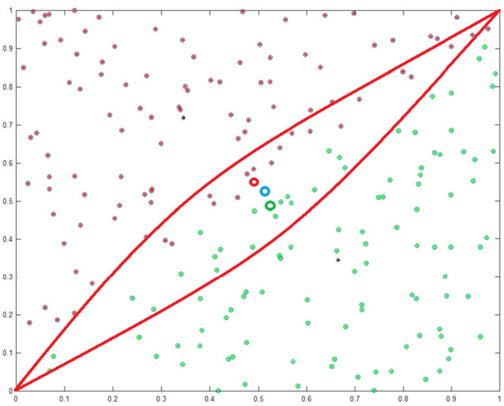

FIGURE 4. Positioning effect of the blue points.

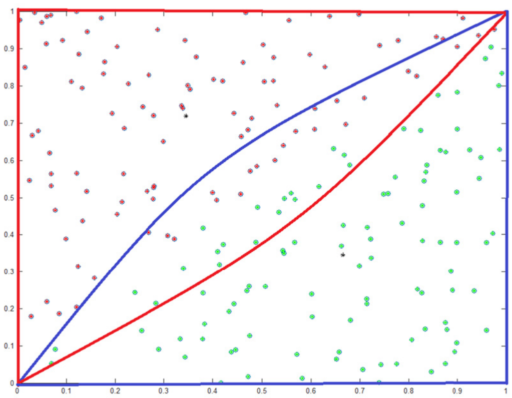

FIGURE 5. Effect of the IPC clustering algorithm.

clustering, the fingerprint points in Class A will be selected for calculation, while the suitable ones in Class B will be ignored.

To solve the above problems, the author proposed the IPC clustering algorithm based on the FCM clustering to acquire sufficient data for accurate clustering of the border points. In this algorithm, the membership of each fingerprint point to the cluster is obtained, creating the membership matrix. If the fingerprint point has similar memberships to both classes, it is added to both classes simultaneously. The clustering effect in Figure 5 show that the points are divided into the blue part and the red part. The intersection between the two parts is known as the public part. In this way, the fingerprint points will no longer have a large distance error after classification.

The IPC clustering algorithm is subjected to the following constraints:

$$|u_1 - u_2| < \alpha \tag{9}$$

$$u_1 + u_2 = 1 \tag{10}$$

where, u_1 and u_2 are the probabilities of each point corresponding to the membership matrices of the two parts. If the difference between u_1 and u_2 is smaller than the preset threshold α, then the point must fall into the public part. To optimize the threshold α, different values of α were tested by the control variable method. In the IPC clustering algorithm, the points in the public part belong to both classes, which improves the positioning accuracy of the border area.

TABLE 1. The specific process of the IPC clustering algorithm.

Algorithm: Ipc clustering algorithm

Input: x=offline RSSI database, n=number of clusters, a=membership probability difference

Output: U= fuzzy membership matrix

1. set n=2;
2. Fuzzy matrix initialization;
3. Calculate the cluster center;

$$m_j = \frac{\sum_{i=1}^{n}[u_j(x_i)]^b x_i}{\sum_{i=1}^{n}[u_j(x_i)]^b}$$

4. update U:

$$u_j(x_i) = \frac{\|x_i - m_j\|^{-2/(b-1)}}{\sum_{s=1}^{k}\|x_i - m_s\|^{-2/(b-1)}}$$

5. Iterate until meet termination condition
6. Get the boundary point:

$$|u_1 - u_2| < \alpha$$
$$u_1 + u_2 = 1$$

7. Get the final two classes

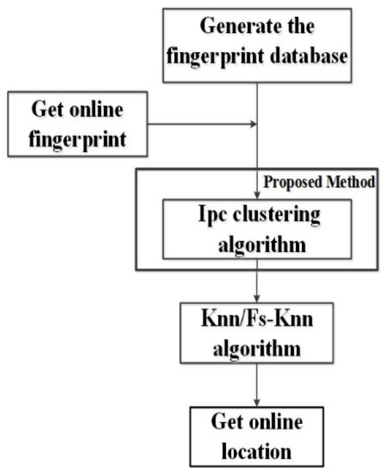

FIGURE 6. Flow chart of indoor positioning by the proposed IPC clustering algorithm.

The specific process the IPC clustering algorithm are illustrated in Table 1, where Steps 6 and 7 are the improvement in our research.

Figure 6 describes the steps of indoor positioning by the proposed IPC clustering algorithm. First, the RSSI of indoor Wi-Fi signals is collected and filtered. Then, the filtered data are saved to the fingerprint database. Next, the fingerprint data are categorized by the IPC clustering algorithm, and the cluster centers are saved. After that, the optimal weight is obtained with the positioning algorithm. Finally, the positioning results are outputted.

III. EXPERIMENTAL VERIFICATION

Our experiments were carried out in a school office. There are 6 APs in the 20m-long, 15m-wide test site (Area I). The area was divided evenly into 300 square grids. In each grid, the RSSI was received in the lower right corner. In total, Area I has 266 (19*14) reference points. After the RSSI of

TABLE 2. Weight selection for each interval in area I.

RSSI	(+∞, -40]	(-40, -45]	(-45, -50]	(-50, -55]
WEIGHT	10	9	5	7
RSSI	(-55, -60]	(-60, -65]	(-65, -70]	(-70, -∞)
WEIGHT	4	2	3	1

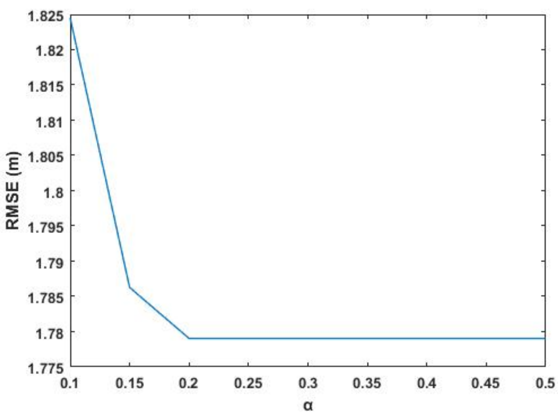

FIGURE 7. The RMSEs at different α values.

each point was loaded to the computer, 20 points were taken as the test points and the others were used as reference points to get the optimal vector. The selected points were filtered to cover the entire area. The positioning effect of the proposed IPC clustering algorithm was compared with the k-means clustering and the FCM clustering, using the root mean square error (RMSE) as the performance indicator.

First, the optimal threshold α for the IPC clustering algorithm was obtained by the control variable method. According to the RMSEs at different α values, the classification accuracy remained the same when α was greater than 0.2. Thus, $\alpha = 0.2$ was selected as the optimal threshold.

Next, the fingerprint database was trained by the FS-KNN to get the optimal weight. The termination condition of the iteration was set as: the weight is less than the threshold $T^{'}$ of 1.5m. The optimal weight of each interval is listed in Table 2 below.

As shown in Table 2, the weight increased with the RSSI in a nonlinear pattern. Through control variable analysis, it is learned that the best situation occurred when the number of classes (k) in the k-means clustering and FCM clustering equaled 2. On this basis, data classification was performed by the IPC clustering, k-means clustering and FCM clustering before KNN positioning. Note that 50 random reference points were taken as test points and the remaining ones as reference points; multiple tests were run to get the mean results (Figure 8).

It can be seen from Figure 8 that the probabilities for k-means clustering error and FCM clustering error to fall within 3m were 76% and 78%, respectively; while the probability for the IPC clustering error to fall within the same

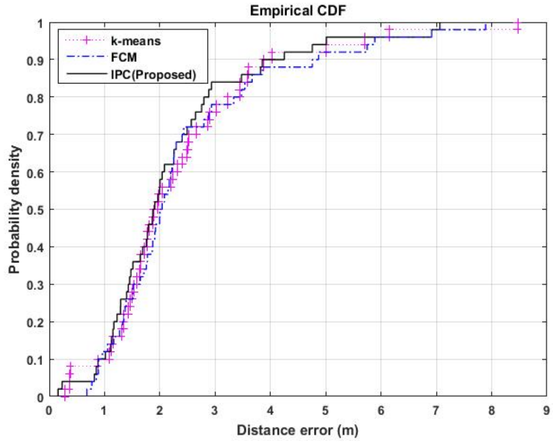

FIGURE 8. Probability density distribution in area I.

TABLE 3. The efficiency and accuracy of the three algorithms in area I.

Algorithm	Number of Point	Within 2m	Within 3m
k-means	146	50 %	76 %
FCM	140	53 %	78 %
IPC	166	57 %	84 %

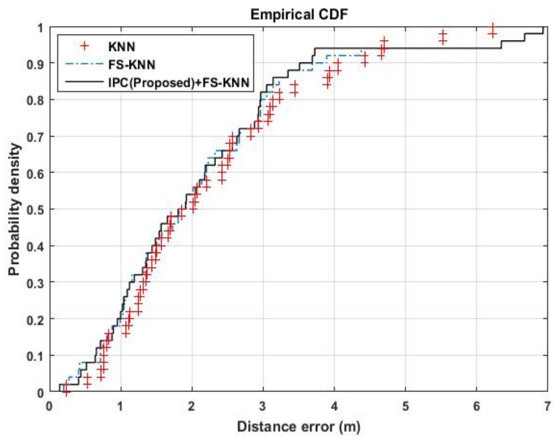

FIGURE 9. Probability density distributions in area I.

distance was 84%. Thus, the IPC clustering algorithm is more accurate than the two contrastive algorithms, with over 60% of the points with a smaller-than-3m error. The number of fingerprint points required for the three algorithms are listed in Table 3 below. Obviously, the three algorithms require similar number of fingerprint point. Among them, the k-means clustering uses the fewest points. In terms of accuracy, the IPC clustering algorithm is better than the other two algorithms. When coupled with the KNN for positioning, the IPC clustering algorithm achieved better effect than the k-means clustering and the FCM clustering.

To verify its adaptability, the IPC clustering algorithm was coupled with the positioning algorithm of FS-KNN, and compared with the other two algorithms under the same condition. The results are presented in Figure 9 and Table 4 below.

TABLE 4. The efficiency and accuracy of the three algorithms in area I.

Algorithm	Number of Point	Within 3m	RMSE (m)
KNN	216	74%	2.61
FS-KNN	216	82%	2.53
IPC+FS-KNN	166	80%	2.46

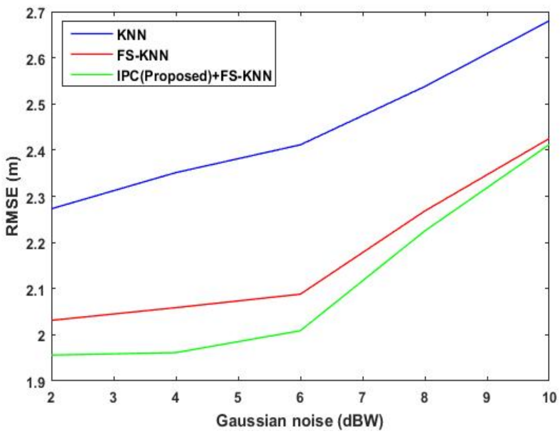

FIGURE 10. The RMSEs of the three algorithms at different levels of gaussian noise in area I.

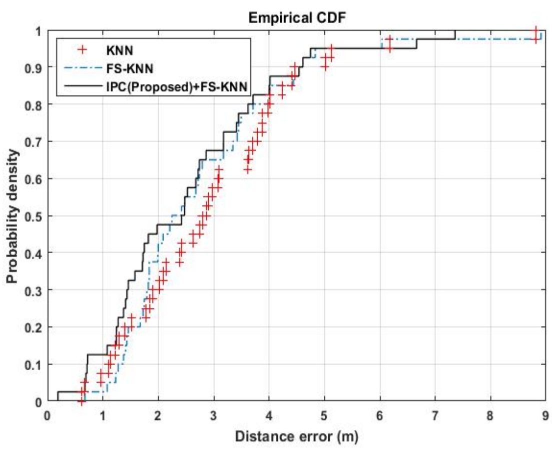

FIGURE 11. Probability density distributions in area II.

It can be seen from the results that the required number of fingerprint points was reduced after using the IPC clustering algorithm before the FS-KNN. Meanwhile, the positioning accuracy was improved because the IPC clustering algorithm rejects the fingerprint points that are close to the test point in RSSI Euclidean distance but far away in actual distance. Next, the three algorithms were compared after introducing Gaussian noise. The positioning results are given in Figure 10 below.

Figure 10 shows that, with the increase in noise, the positioning accuracies of the three algorithms dropped across the board. Comparatively, the IPC clustering algorithm coupling the FS-KNN outperformed the FS-KNN alone. Thus, the proposed algorithm can optimize the clustering effect of the FS-KNN despite noise interference.

TABLE 5. The efficiency and accuracy of the three algorithms in area II.

Algorithm	Number of Point	Within 3m	RMSE (m)
KNN	125	58%	2.96
FS-KNN	125	65%	2.77
IPC+FS-KNN	102	67%	2.54

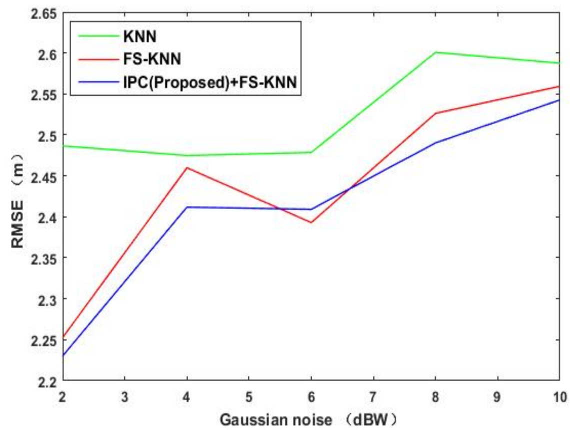

FIGURE 12. The RMSEs of the three algorithms at different levels of gaussian noise in area II.

To verify its integrity, the IPC clustering algorithm was further tested in another test site (Area II). The new test site is 16m long and 12m wide. A total of 165 (15*11) RSSI vectors were collected as reference points, which are evenly distributed in the area. Forty of them were taken as test points to cover the entire area. The threshold and optimal weight were determined the same as above. Then, the same three algorithms were introduced to the experiment. The results of them are given in Figure 11 and Table 5 below.

The results show that the IPC clustering algorithm coupling FS-KNN was still superior to the other algorithms in the new environment. The RMSE of the coupled algorithm was 0.23m smaller than that of the FS-KNN and 0.42m smaller than the traditional KNN. This means the IPC clustering algorithm enjoys good adaptability in different environments.

The three algorithms were further tested after adding Gaussian noises. The new results (Figure 12) show that the IPC clustering algorithm coupling the FS-KNN realized a smaller RMSE than the FS-KNN or KNN alone. Hence, the IPC clustering algorithm has excellent noise resistance in Area II.

IV. CONCLUSION

This paper develops the IPC clustering algorithm to improve the accuracy of indoor positioning. The proposed algorithm optimizes the clustering effect by allocating the points in the border area between two classes to these classes simultaneously, pushing up the positioning accuracy in this area. The experimental results show that the IPC clustering algorithm achieved better accuracy with lighter computing load than FCM clustering and k-means clustering, and could be coupled with KNN or FS-KNN to achieve good positioning effect. Considering its strong adaptability, the proposed algorithm will be extended to other fields in need of clustering.

REFERENCES

[1] Y. Peng, W. Fan, X. Dong, and X. Zhang, "An iterative weighted KNN (IW-KNN) based indoor localization method in Bluetooth low energy (BLE) environment," in *Proc. IEEE Int. Conf. Ubiquitous Intell.*, Jul. 2016, pp. 794–800.

[2] J. Haverinen and A. Kemppainen, "Global indoor self-localization based on the ambient magnetic field," *Robot. Auto. Syst.*, vol. 57, no. 10, pp. 1028–1035, Oct. 2009.

[3] G. Shen, R. Zetik, and R. S. Thoma, "Performance comparison of TOA and TDOA based location estimation algorithms in LOS environment," in *Proc. 5th Workshop Positioning, Navigat. Commun. (WPNC)*, Mar. 2008, pp. 71–78.

[4] B. Cui and X. Zhao, "A hybrid indoor positioning method based on propagation model and lacation fingerprint," *Bull. Surv. Mapping*, vol. 6, pp. 35–38, Jun. 2015.

[5] N. T. Nguyen, R. Zheng, J. Liu, and Z. Han, "GreenLocs: An energy-efficient indoor place identification framework," *ACM Trans. Sensor Netw.*, vol. 11, no. 3, May 2015, Art. no. 43.

[6] J. Yu and J. Liu, "A KNN indoor positioning algorithm that is weighted by the membership of fuzzy set," in *Proc. IEEE Int. Conf. Green Comput. Commun.*, Aug. 2013, pp. 1899–1903.

[7] Z. Wu, C. Rengin, X. Shuyan, W. Xiaosi, and F. Yuli, "Research and improvement of WiFi positioning based on k nearest neighbor method," *Comput. Eng.*, vol. 43, no. 3, pp. 289–293, 2017.

[8] Z. Liu, X. Luo, and T. He, "Indoor positioning system based on the improved W-KNN algorithm," in *Proc. IEEE 2nd Adv. Inf. Technol., Electron. Automat. Control Conf. (IAEAC)*, Mar. 2017, pp. 1355–1359.

[9] X.-L. Zhang, P. Zhao, and G.-Q. Xu, "Research of indoor positioning based on a optimization KNN algorithm," *Electron. Des. Eng.*, vol. 7, pp. 44–46, Jul. 2013.

[10] A. Y. Wang and L. Wang, "Research on indoor localization algorithm based on WiFi signal fingerprinting and INS," in *Proc. Int. Conf. Intell. Transp., Big Data Smart City (ICITBS)*, Jan. 2018, pp. 206–209.

[11] Y. Sun, Y. Xu, L. Ma, and Z. Deng, "KNN-FCM hybrid algorithm for indoor location in WLAN," in *Proc. 2nd Int. Conf. Power Electron. Intell. Transp. Syst. (PEITS)*, Dec. 2009, pp. 251–254.

[12] J. Li, X. He, Y. Cai, and Q. Xu, "Method of WiFi indoor location based on K—Means and random forest," *Control Eng. China*, vol. 4, pp. 787–792, Apr. 2017.

[13] C. Park and S. H. Rhee, "Indoor positioning using Wi-Fi fingerprint with signal clustering," in *Proc. Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, Oct. 2017, pp. 820–822.

[14] H. Wang, X. Zhang, Y. Gu, L. Zhang, and J. Li, "Indoor Wi-Fi RSS-fingerprint location algorithm based on sample points clustering and AP reduction," in *Proc. 6th Int. Conf. Intell. Control Inf. Process. (ICICIP)*, Nov. 2015, pp. 264–267.

[15] H. Arslan and M. Toz, "Hybrid FCM-WOA data clustering algorithm," in *Proc. 26th Signal Process. Commun. Appl. Conf. (SIU)*, May 2018, pp. 1–4.

[16] L. Mengual, O. Marbán, and S. Eibe, "Clustering-based location in wireless networks," *Expert Syst. Appl.*, vol. 37, no. 9, pp. 6165–6175, 2010.

[17] J. F. Luo and Z. H. Suo, "A density based K-means clustering algorithm," *Microelectron. Comput.*, vol. 31, no. 10, pp. 28–31, Oct. 2014.

[18] D. Li, B. Zhang, Z. Yao, and C. Li, "A feature scaling based K-nearest neighbor algorithm for indoor positioning system," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2014, pp. 436–441.

[19] J. Hu, D. Liu, Z. Yan, and H. Liu, "Experimental analysis on weight K-nearest neighbor indoor fingerprint positioning," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 891–897, Feb. 2019.

JIN REN received the Ph.D. degree in information and communications engineering from the School of Electronic and Information Engineering, Dankook University, South Korea, in 2011. He is currently a Lecturer with the School of Information Science and Technology, North China University of Technology. His current research interests include intelligent information processing, wireless communication, and the Internet of Things (IoT) technology.

YUNAN WANG was born in Beijing, China. He is currently pursuing the master's degree in advanced indoor positioning solutions with the North China University of Technology. He is a Laboratory Member of the Cyber-Physical Systems Lab, Dr. Ren's Research Team, North China University of Technology. His research interests include low-cost IoT systems and indoor positioning algorithms.

CHANGLIU NIU was born in 1967. He received the Ph.D. degree in wave electromagnetics from the Beijing University of Posts and Telecommunications, in 2006. He is currently an Associate Professor with the School of Information Science and Technology, North China University of Technology. His research interests cover but not limited to high-speed optical communication, communication network switching, and wireless sensor networks. He taught main courses, including wave electromagnetics, the principle of modern communication, and the theoretical basis of communication networks.

WEI SONG received the Ph.D. degree in traffic information engineering and control from Beijing Jiaotong University, in 2010. He was an Assistant Professor with the Research Institute of Information Technology, Tsinghua University, from 2011 to 2012. He was also a Visiting Scholar with the New Jersey Institute and Technology, from 2017 to 2018. He is currently an Associate Professor and the Vice Dean of the School of Information Engineering, Minzu University of China. He is currently the Director of Media Computing Laboratory. His research interests include multimedia networks, image processing, and natural language processing.

SONGYANG HUANG was born in Beijing, in 1994. He is currently pursuing the master's degree in advanced indoor positioning solutions with the North China University of Technology, where he is a Laboratory Member of the Cyber-Physical Systems Lab, Dr. Ren's Research Team. The research activities of the team include low-cost IoT systems and high-precision positioning solutions for indoor use.