PAPER • OPEN ACCESS

A new Wi-Fi dynamic selection of nearest neighbor localization algorithm based on RSS characteristic value extraction by hybrid filtering

To cite this article: Xuesheng Peng *et al* 2021 *Meas. Sci. Technol.* **32** 034003

View the article online for updates and enhancements.

A new Wi-Fi dynamic selection of nearest neighbor localization algorithm based on RSS characteristic value extraction by hybrid filtering

Xuesheng Peng[1] , **Ruizhi Chen**[1,*], **Kegen Yu**[2], **Guangyi Guo**[1], **Feng Ye**[1] **and Weixing Xue**[3]

[1] State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing (LIESMARS), Wuhan University, Wuhan, People's Republic of China
[2] School of Environmental Science and Spatial Informatics, China University of Mining and Technology, Xuzhou, People's Republic of China
[3] Department of Shenzhen Key Laboratory of Spatial Smart Sensing and Services, Shenzhen University, Shenzhen, People's Republic of China

E-mail: ruizhi.chen@whu.edu.cn

CrossMark

Abstract

Fingerprinting localization based on Wi-Fi received signal strength (RSS) is the most widely used indoor localization method. It typically includes offline training and online matching phases. The selection of the RSS characteristic value is a key step. The weighted K nearest neighbor (WKNN) algorithm is the most commonly used position-determination algorithm. The mean value of the RSS data collected over a time interval is usually taken as its characteristic value. However, the RSS measurements contain Gaussian and non-Gaussian noise, which cannot be filtered out effectively by the mean value method. The traditional WKNN algorithm adopts a fixed K. However, reference points far away from the test point (TP) may be selected as the nearest neighbors to participate in the position calculation, which may result in accuracy degradation. This paper proposes the weighted dynamic K nearest neighbor algorithm (WDKNN-HF), which utilizes a hybrid of particle filtering and Kalman filtering to extract the RSS characteristic value. In the online matching phase, a dynamic K matching algorithm based on Euclidean distances is developed to determine the coordinates of TPs. Two experiments are conducted in two different indoor scenes. Experimental results demonstrate that the proposed algorithm can obtain better positioning accuracy than existing algorithms, such as KNN, WKNN, enhanced-WKNN (EWKNN) and self-adaptive weighted K nearest neighbor (SAWKNN).

---

* Author to whom any correspondence should be addressed.

(Some figures may appear in color only in the online journal)

1. Introduction

Indoor positioning systems (IPSs) have become a research hot-spot in recent years and many IPS technologies have been proposed. These technologies are divided into infrastructure-based and infrastructure-free technologies [1]. Infrastructure-based technologies mainly include UWB (ultra-wideband) [2], Bluetooth [3] and RFID (radio frequency identifier) [4] techniques, whereas infrastructure-free technologies typically include Wi-Fi [5, 6], magnetic field [7] and PDR (pedestrian dead reckoning) [8, 9] techniques. Infrastructure-based technologies require pre-installation facilities and specialized hardware, whereas infrastructure-free technologies require no additional hardware or facilities. Wi-Fi positioning technology based on received signal strength (RSS) [10] is the most commonly used indoor positioning technology because of the wide deployment of Wi-Fi access points (APs) and the popularity of mobile devices such as mobile phones.

Wi-Fi positioning methods mainly include Wi-Fi round trip time [11], channel state information [12] and Wi-Fi RSS. Wi-Fi RSS-based positioning methods can be divided into two categories: the triangulation method and the fingerprinting method. The triangulation positioning method converts RSS observations into distances between APs and the test point (TP) using a signal propagation model; therefore, it needs to know the locations of at least three APs. The fingerprinting method based on Wi-Fi RSS is the most frequently used method [13] because it does not need the locations of APs or a signal propagation model. The fingerprinting method typically includes two phases: offline training and online matching [14]. In the offline phase, the task is to collect RSS values of all reference points (RPs) in the location area and then extract RSS characteristic values to establish the fingerprint database; the database contains RSS characteristic values of all RPs and their corresponding physical information, such as serial number, coordinates, etc. In the online phase, the RSS characteristic value of the TP is measured and then the location of the TP can be calculated via a probability algorithm and deterministic algorithm. The probability algorithm requires knowledge of the signal propagation model and repeated observations of each physical point to establish an empirical distribution. Traditional deterministic algorithms include the NN (nearest neighbor), KNN (*K* nearest neighbor) [15, 16] and WKNN (weighted *K* nearest neighbor) [17] algorithms. WKNN is the most frequently used algorithm, for which *K* is the number of nearest RPs selected to calculate the coordinates of the TP. The selection of the *K* value in the positioning phase can affect the positioning accuracy.

The selection of the RSS characteristic value in fingerprint positioning is a necessary procedure for both the offline phase and online phase. Different RSS characteristic values may result in diverse positioning results. In general, traditional RSS characteristic values include the mean value, median value, mode value, maximum value, and so on. The mean value is the most frequently used one. As indoor signal propagation often has a multipath effect [18] and NLOS (non-line-of-sight) propagation [19–21], there are both Gaussian and non-Gaussian interference signals in the indoor RSS signal. The mean value algorithm does not deal with interference signals effectively. The Kalman filtering (KF) algorithm [22, 23] has better performance in processing a signal containing Gaussian noise, whereas the particle filtering (PF) algorithm [24, 25] can filter out non-Gaussian noise. A single filtering algorithm cannot completely eliminate the interference signals. In this paper, a hybrid filtering algorithm combining PF and KF is proposed to extract the RSS characteristic values for both RPs and TPs.

In the traditional WKNN algorithm, a fixed number of RPs that have the minimum Manhattan distances or Euclidean distances from the TP are usually selected to calculate the coordinates of the TP. However, not all TPs can obtain the best positioning accuracy with the same *K* value; for example, in [26], the best positioning accuracy is obtained when *K* is 4, but in [27], *K* is 3. The major cause for this phenomenon is that there is a certain defect in using the fixed *K* value: at one TP, the RPs with larger actual distances from the TP may be included for position calculation, whereas at another TP, the RPs with smaller actual distances from the TP are excluded. Thus, the selection of RPs in indoor areas should be carefully made for both good accuracy and simple implementation. Different numbers of RPs can be selected for position calculation at different TPs, meaning that the value of *K* is dynamically adjusted. In this paper, a new dynamic *K* selection algorithm WDKNN (weighted dynamic-*K*-nearest neighbor) based on Euclidean distance is proposed in the online matching phase. Different from the traditional KNN or WKNN algorithms, the proposed WDKNN algorithm can dynamically adjust the number of RPs selected for position calculation and it does not require any prior knowledge in the offline phase or additional hardware from the positioning system.

The rest of this article is structured as follows. The related works are introduced in section 2. Section 3 provides the details of our proposed WDKNN-HF algorithm, and experiments and the result analyses are conducted in section 4. Section 5 presents the conclusion of the entire work and suggestions for future research.

2. Related works

Indoor environments are complex and refraction, reflection, occlusion and other factors may cause multipath and NLOS propagation in the RSS signal propagation process. In [28], different RSS characteristic values, including the mean value, median value, mode value and maximum value, are selected,

and their positioning performances are compared and analyzed. Kaemarungsi and Krishnamurthy [29] analyze the influence of environment and equipment on the RSS characteristic value and point out that the RSS signal characteristic value distributions vary in different environments. The authors suggest that in addition to the mean value, the standard deviation (STD) and the distribution of RSS should be included. Sun and Zhu [30] put forward a hybrid filter algorithm based on the mean filter, median filter and Gaussian filter and RSS data are optimized by using this algorithm. Zhang *et al* [31] first analyze the influence of the environment on RSS and then the performances of three experimental data processing methods are compared; the Gaussian filter algorithm is considered to be the best. In [32], to optimize RSS observations, the author uses a filter algorithm mixed by a mean filter, a Gaussian filter, a KF and linear interpolation to process RSS observations. Song *et al* [33] develop an RSS-based indoor positioning algorithm using a PF; the RSS measured at the TP is used as a direct observation in the correction stage of the filtering algorithm and the PF is applied to the nonlinear tracking mode. Zhuang *et al* [34] propose a two-filter integration algorithm for motion sensor-based positioning and RSS-based fingerprinting. A KF is first adopted to obtain a smoothed constrained Wi-Fi fingerprinting and then an extended KF is used to integrate smoothed constrained Wi-Fi fingerprinting and sensor-based positioning. Huang *et al* [4] utilize a KF to process RSS observations. They compare the results of RSS raw data, RSS data through traditional statistics (the mean RSS value) and RSS data through the KF, finding that the KF algorithm can significantly remove the drift effect from RSS raw data, but its result is similar to the mean value.

Traditional online matching algorithms include the NN algorithm, KNN algorithm and WKNN algorithm. First, Manhattan distances or Euclidean distances D_{iT} from the TP to all RPs are calculated via:

$$D_{iT} = \left(\sum_{j=1}^{m} (|RSS_T^j - RSS_i^j|)^r \right)^{1/r},$$
$$i = 1, 2, 3, \ldots, n, j = 1, 2, 3, \ldots, m. \quad (1)$$

In (1), RSS_T^j represents the online RSS observations of the *j*th AP at the TP, RSS_i^j represents the offline RSS observations of the *j*th AP at the *i*th RP and it is assumed that there are *m* APs and *n* RPs; when *r* is one, D_{iT} represents Manhattan distance, and when *r* is two, D_{iT} represents Euclidean distance. Then, sort all Manhattan distances or Euclidean distances in ascending order.

In the NN algorithm, the RP that has the smallest Manhattan distance or Euclidean distance from the TP is selected and its coordinates are taken as the TP's estimated coordinates. However, in complex and dynamic indoor environments, the positioning result of the NN algorithm has a great contingency; that is, the RP far away from the TP may has the smallest Manhattan distance or Euclidean distance rather than the real nearest RP and it may produce significant position error. The KNN algorithm selects the *K*-nearest RPs and takes the

average of their coordinates as the estimated coordinates of the TP, reducing the contingency of taking a single nearest point. The WKNN algorithm selects *K*-nearest RPs to calculate the TP's coordinates, as well. Considering that the Manhattan distances or Euclidean distances between the TP and these selected RPs are different, WKNN gives different weights to RPs considering their different contributions to the positioning result. The WKNN algorithm can obtain a better positioning accuracy than those of the KNN and NN algorithms. In the WKNN algorithm, the weights of the selected RPs are calculated as inversely proportional to the Manhattan distance or Euclidean distance; that is, an RP with a small RSS distance is assigned a larger weight, while an RP with a large RSS distance is assigned a smaller weight. The weight of the RPs is calculated by:

$$w_{iT} = \frac{1/D_{iT}}{\sum\limits_{i=1}^{K} 1/D_{iT}}. \quad (2)$$

The estimated coordinates of the TP are the weighted average of the coordinates of these selected RPs and are calculated by:

$$(x_T, y_T) = \sum_{i=1}^{K} w_{iT}(x_i, y_i). \quad (3)$$

In both the KNN algorithm and the WKNN algorithm, the value of *K* affects the positioning accuracy. Different *K* values may lead to different positioning results. In [26], the best positioning accuracy is obtained when *K* is 4; in [27], the best positioning accuracy is obtained when *K* is 3; however, in [35], *K* is 13. This phenomenon is mainly caused by the difference of environments and the distributions of RPs. Some researchers have studied dynamic *K*-value selection methods. Shin *et al* [27] propose the EWKNN (enhanced weighted *K*-nearest neighbor) algorithm. The EWKNN utilizes two comparisons to select RPs for position calculation, compared with the traditional WKNN algorithm, the mean position error (ME) is reduced from 2.87 m to 2.11 m. Hu [36] propose the SAWKNN (self-adaptive weighted *K*-nearest neighbor) algorithm, which can achieve better positioning precision than the traditional WKNN. The experimental results show that the SAWKNN outperforms traditional WKNN by 21% in positioning accuracy. The EWKNN and SAWKNN algorithms both adopt Manhattan distance. Positioning performances of the KNN, Euclidian–WKNN, Manhattan–WKNN and EWKNN algorithms are compared and analyzed in [37].

3. The weighted dynamic-*K* nearest neighbor algorithm based on hybrid filtering RSS characteristic value extraction

This section presents the WDKNN-HF algorithm (weighted dynamic-*K* nearest neighbor algorithm that uses hybrid filtering to extract the RSS characteristic value).

The architecture of the proposed WDKNN-HF algorithm is illustrated in figure 1. The WDKNN-HF algorithm includes
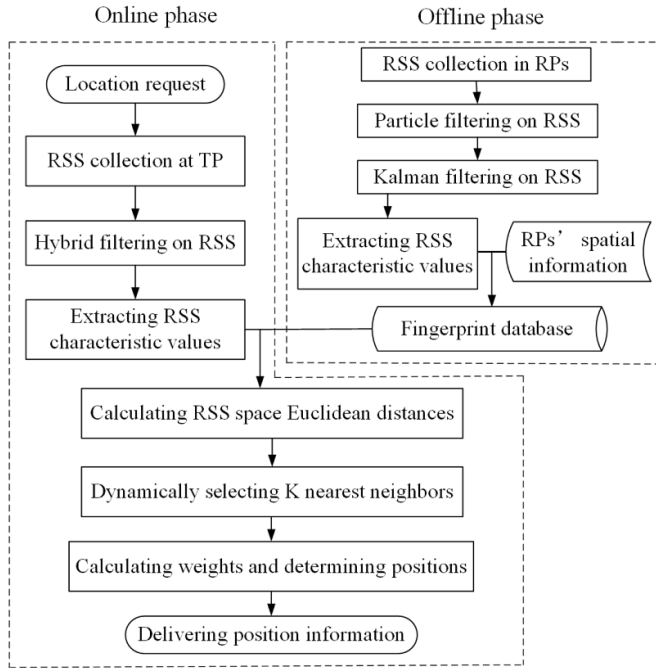
Figure 1. Framework of the proposed WDKNN-HF algorithm.

an offline training phase and an online matching phase. In the offline phase, RSS observations are collected at all RPs by a mobile device. To extract RSS characteristic values, the hybrid filtering algorithm is adopted to process RSS observations. Then, all of the RPs' RSS characteristic values and their corresponding spatial information are stored to establish the fingerprint database. In the online phase, the RSS characteristic values of the TP are measured and instead of using a fixed K value, a dynamic K-value selection algorithm based on Euclidean distance is adopted to calculate the TP's coordinates.

3.1. Hybrid filtering algorithm based on particle filtering and Kalman filtering

The framework of the WDKNN-HF algorithm indicates that the RSS characteristic value extraction is a key step in both offline and online phases. In an indoor environment, the RSS observations contain Gaussian and non-Gaussian noise. To deal with both of these noise signals, a hybrid filtering algorithm that combines PF and KF is adopted for the extraction of the RSS characteristic value.

3.1.1. Hybrid filtering algorithm. Step 1: The PF is applied to RSS observation values. PF is a process of approximating the probability density function by finding a group of random samples propagating in the state space, and replacing the integral operation with the sample mean value to obtain the minimum variance distribution of the state [24, 25].

The state space model of the nonlinear dynamic system is expressed as follows:

$$\begin{aligned} x_i &= f(x_{i-1}, v_{i-1}), \\ z_i &= h(x_i, u_i). \end{aligned} \tag{4}$$

x_i is the state of the system at the time i, z_i is the observation vector at time i, f is the state transfer function of the system, h is the system measurement function and v_i and u_i represent the process noise and the observation noise, respectively.

$$RSS = [rss_1, rss_2, \ldots, rss_i, \ldots, rss_N] \tag{5}$$

Equation (5) represents a list of RSS observation values from an AP at the sampling point.

First, M particles $\{p_k, w_k\}$, normally distributed, are generated; the RSS observation rss_i is taken as the expected value; V is the variance of particles $\{p_k\}$:

$$p_k \sim N(rss_i, V), k = 1, 2, 3, \ldots, M, i = 1, 2, 3, \ldots, N-1. \tag{6}$$

Then, the differences between the $(i+1)$th RSS observation and each particle are calculated, and equation (7) is used to determine the weight w_k of each particle:

$$w_k = \frac{1}{\sqrt{2\pi V}} e^{\frac{-(rss_{i+1}-p_k)^2}{2V}}. \tag{7}$$

The weight is then normalized as:

$$\overline{w_k} = \frac{w_k}{\sum\limits_{k=1}^{M} w_k}. \tag{8}$$

Particle degradation is a common defect in PF, to reduce this phenomenon, the resampling of particles is conducted. The resampling of particles may result in the reduction of the particles' diversity. To deal with this problem, a criterion N_{eff} is used to calculate the number of valid particles to determine whether to perform resampling.

$$N_{eff} = \frac{1}{\sum\limits_{k=1}^{M} (\overline{w_k})^2}. \tag{9}$$

If N_{eff} is less than a certain threshold $N_{threshold}$, the system resampling [38] is adopted to select the particles, which are denoted as $\overline{p_k}$, and the particles with small weight are replaced by particles with large weight. $N_{threshold}$ can be given as an empirical value. In this article, $N_{threshold}$ is set to $\frac{2}{3}M$.

Finally, the weighted average of these particles is calculated as the estimated value of the ith RSS after PF, which is denoted as $\overrightarrow{rss_i}$:

$$\overrightarrow{rss_i} = \sum_{k=1}^{M} \overline{p_k w_k}. \tag{10}$$

All RSS observations of (5) are processed by PF, and the processed observations are denoted as $\overrightarrow{RSS}$:

$$\overrightarrow{RSS} = [\overrightarrow{rss_1}, \overrightarrow{rss_2}, \ldots \overrightarrow{rss_i}, \ldots, \overrightarrow{rss_N}]. \tag{11}$$

Step 2: KF is applied to the processed RSS values $\overrightarrow{RSS}$ by the PF. The KF [22, 23] takes the minimum mean square error as the optimal estimation criterion, adopts the state space

model of signal and noise, updates the estimation of state variables with the estimated value of the previous moment and the observed value of the current moment, and calculates the estimated value of the current moment.

The system process and system measurement equations of the KF are described as follows:

$$
\begin{aligned}
x_i &= A\overrightarrow{x_{i-1}} + w_i, \\
z_i &= H\overrightarrow{rss_i} + v_i, i = 1, 2, 3, \ldots N.
\end{aligned}
\tag{12}
$$

The discrete process system is one-dimensional without control input, $A = 1$, $H = 1$. w_i and v_i represent the noise of process and measurement, respectively; they follow the normal distributions:

$$
\begin{aligned}
p(w) &\sim N(0, Q), \\
p(v) &\sim N(0, R).
\end{aligned}
\tag{13}
$$

In equation (13), Q is the variance of process noise and R is the variance of measurement noise. All $\overrightarrow{RSS}$ components are taken as system measurements and the variance R is calculated:

$$
R = \mathrm{var}(\overrightarrow{rss_i} - \overrightarrow{\overline{rss}}), i = 1, 2, 3 \ldots N.
\tag{14}
$$

In equation (14), $\overrightarrow{\overline{rss}}$ is the average value of $\overrightarrow{RSS}$. Q can be given as an empirical value, in this process, we set $Q = 2R$.

$\overrightarrow{rss_1}$ is taken as the initial value of the system state x_0, and the predicted value of the state at time i is calculated:

$$
x_i^- = x_{i-1}.
\tag{15}
$$

The initial covariance P_0 is set to 1; the covariance P_i^- corresponding to the state prediction value is:

$$
P_i^- = P_{i-1} + Q.
\tag{16}
$$

Then, the optimal estimated RSS value x_i at time i is calculated by:

$$
x_i = x_i^- + Kg(\overrightarrow{rss_i} - x_i^-).
\tag{17}
$$

Kalman gain Kg can be calculated via equation (18), and the updated P_i can be calculated via equation (19).

$$
Kg = P_i^- (P_i^- + R)^{-1}
\tag{18}
$$

$$
P_i = P_i^- - KgP_i^-.
\tag{19}
$$

Finally, RSS characteristic value $\overline{RSS}$ is obtained via:

$$
\overline{RSS} = \frac{\sum_{i=1}^{N} x_i}{N}.
\tag{20}
$$

Step 3: The RSS values collected from all APs at all sampling points are processed by steps 1 and 2 to obtain all of the RSS characteristic values.
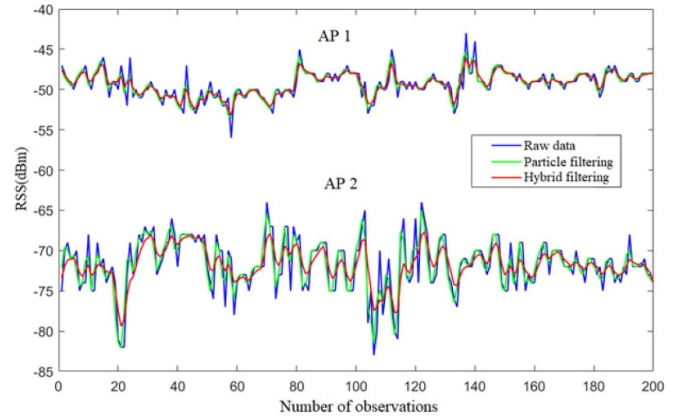

Figure 2. RSS results by different processing methods.

3.1.2. Test of the hybrid filtering algorithm on RSS data process. To test the performance of the hybrid filtering algorithm on RSS data, an experiment was conducted. At the sampling point, RSS data from two different APs (AP1 and AP2) are collected by the same smartphone. AP1 is close to the sampling point and AP2 is far away from the sampling point, each AP is measured 200 times.

Figure 2 shows the results of different processing methods on RSS observations from these two APs. It can be intuitively seen that the fluctuation amplitude of RSS data by hybrid filtering is smaller than raw RSS data and RSS data by PF. Take AP2 for instance, the minimum and maximum RSS data of the hybrid filtering algorithm are −79.4dBm and −67.7dBm, which are better than the −83dBm and −64dBm of raw RSS data, and the −82dBm and −65dBm of the PF algorithm. It can be concluded that the hybrid filtering method can deal with the signal noise better.

The STD reflects the degree of dispersion of a set of data; the smaller the STD is, the smaller the dispersion of the data is.

Christoph [39] proposed the smoothness index S to quantify the smoothness of the curve. The smaller S is, the smoother the curve is and the better the signal quality is.

S is calculated by:

$$
\begin{aligned}
S &= STDEV\left(RSS_i - \frac{RSS_{i-1} + RSS_i + RSS_{i+1}}{3}\right) \\
&i = 2, 3, 4, \ldots, N-1.
\end{aligned}
\tag{21}
$$

In (21), N is the number of observations.

Table 1 shows the comparison of STD and S of RSS observations by different processing methods. Both STD and of the hybrid filtering method for AP 1 and AP 2 are the smallest, quantitatively illustrating the effect of the hybrid filtering method on mitigating signal noise.

3.2. Weighted dynamic-K nearest neighbor (WDKNN) algorithm

As the distance between the sampling point and the AP increases, the signal strength value received by the sampling

Table 1. STD and S of different processing methods.

AP	Method	STD	S
1	Raw data	1.71	0.72
	Particle filtering	1.47	0.29
	Hybrid filtering	1.34	0.17
2	Raw data	3.37	1.30
	Particle filtering	2.86	0.62
	Hybrid filtering	2.08	0.23

point gradually weakens [40–43]. Equation (22) is the wireless signal attenuation model [44–46], in which $P(d)$ and $P(d_0)$ respectively represent the RSS values measured at distances d and d_0 from an AP, η is the path loss exponent and d_0 is a benchmark, usually with a value of 1 m, and ω is the environment noise.

$$P(d) = P(d_0) + 10\eta \log_{10} \frac{d}{d_0} + \omega. \qquad (22)$$

Equation(22) indicates that the $P(d)$ of the sampling points in the same space may depend on the value of d and that the closer TP and RP have similar $P(d)$ values. Moreover, the Euclidean distance reflects the similarity degree of $P(d)$ values between the two points.

From equation (22), we can ascertain that if the actual distance of two sampling points is small, then they have a small RSS value gap, and thus their Euclidean distance (23) is also small. However, it is not absolutely true in the real indoor environment. The experimental result in [36] indicated that the chance of having $d_{1T} < d_{2T}$ when $D_{1T} < D_{2T}$ is about 97%. We can infer that if $d_{1T} < d_{2T}$, then $RSS_{1T} < RSS_{2T}$, and then $D_{1T} < D_{2T}$ in general. In this article, Euclidean distance is used to select RPs for position calculation.

Now, the WDKNN algorithm is described as follows:

Step 1 Equation (23) is used to calculate the Euclidean distances from the TP to all RPs, denoted as d_i, where n is the number of RPs and m is the number of APs, signals of which can be received at both the TP and the RP. Then all Euclidean distances are arranged in ascending order such that d_1 is the minimum Euclidean distance, denoted as $d_{\min}$.

$$d_i = \sqrt{\sum_{j=1}^{m} \left(RSS^j - RSS_i^j\right)^2}, i = 1, 2, 3, \ldots, n. \qquad (23)$$

Step 2: Equation (24) is used to calculate the ratio γ_j of the remaining $d_j (j = 2, 3, 4 \ldots n)$ to $d_{\min}$:

$$\gamma_j = d_j / d_{\min}. \qquad (24)$$

Equation (25) is used to select RPs to participate in the position calculation; the threshold value $\gamma_{\text{threshold}}$ is used as the metric for whether the RP is selected or discarded. The number of selected RPs is K and its value varies for different TPs.

$$\begin{cases} \gamma_j \leq \gamma_{\text{threshold}}, \text{ corresponding RP is selected for matching} \\ \gamma_j > \gamma_{\text{threshold}}, \text{ corresponding RP is discarded.} \end{cases} \qquad (25)$$

The $\gamma_{\text{threshold}}$ varies with different equipment and environment, and it can be given by the empirical value.

Step 3: The RPs adopted for position calculation are selected by step 2, and their weights are calculated via:

$$w_i = \frac{1/d_i}{\sum\limits_{i=1}^{K} 1/d_i}. \qquad (26)$$

Then, the weighted average values of the selected RPs' coordinates are calculated by equation (27) as the estimated coordinates of the TP:

$$(x_{\text{estimation}}, y_{\text{estimation}}) = \sum_{i=1}^{K} w_i(x_i, y_i). \qquad (27)$$

In (27), (x_i, y_i) represents the coordinates of the ith RP.

Finally, repeat steps 1, 2, and 3 and all TP estimated coordinates are calculated.

3.3. Evaluation indicators of positioning performance

The proposed WDKNN-HF algorithm is compared with traditional algorithms in terms of the ME, STD, root-mean-square error (RMSE), 90th percentile error and cumulative distribution function (CDF). The ME, STD and RMSE are calculated as follows:

$$\delta = \sqrt{\left(x_{\text{real}} - x_{\text{estimation}}\right)^2 + \left(y_{\text{real}} - y_{\text{estimation}}\right)^2}, \qquad (28)$$

$$ME = \frac{1}{n} \sum_{i=1}^{n} \delta_i, \qquad (29)$$

$$STD = \sqrt{\frac{1}{n} \sum_{i=1}^{n} \left(\delta_i - ME\right)^2}, \qquad (30)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} \delta_i^2}. \qquad (31)$$

In (28)–(31), δ is the position error of the TP, $(x_{\text{estimation}}, y_{\text{estimation}})$ stands for the TP's estimated coordinates, $(x_{\text{real}}, y_{\text{real}})$ are the TP's's real coordinates and n is the total number of TPs.

4. Experiments and result analyses

4.1. Experimental setup

Figure 3(a) shows the floor plan of the third floor (3F) scene, including corridors, a lobby and student rooms. The RP spacing is approximately 2.5 m; 133 RPs and 136 TPs are selected. Figure 3(b) is the floor plan of an office, 30 regularly distributed RPs are collected, where the distance between the
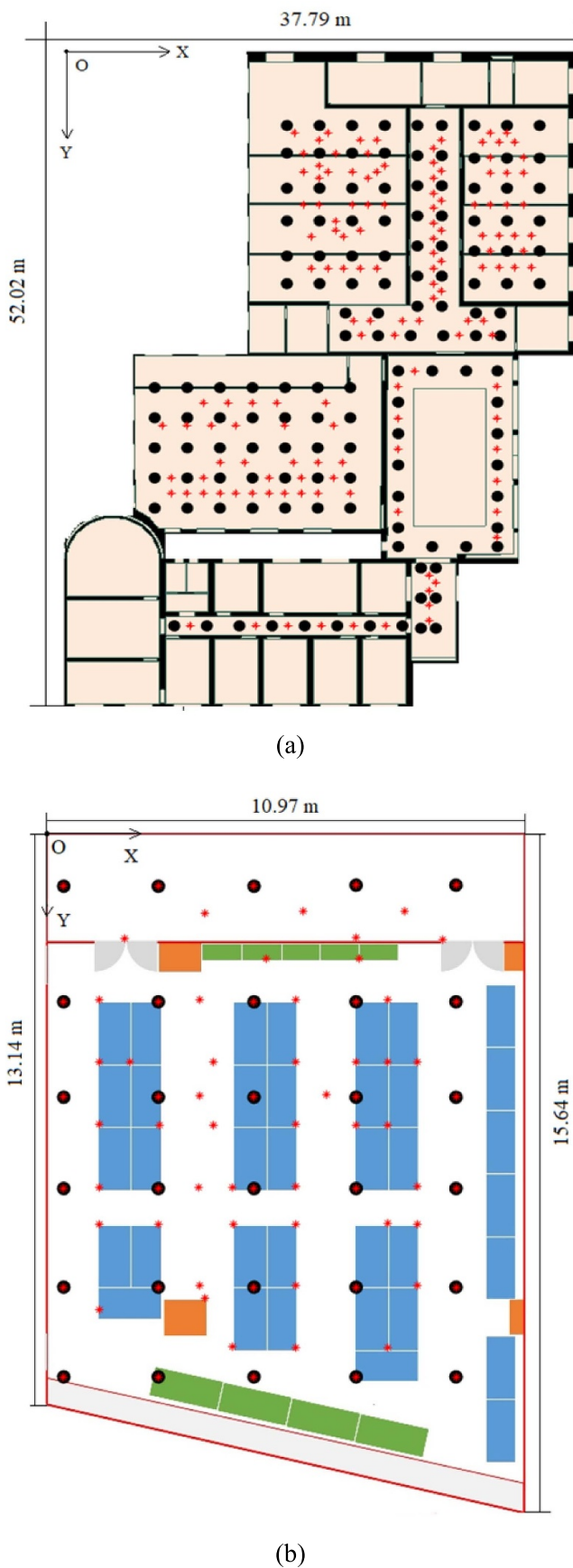
(a)

(b)

Figure 3. Schematic diagrams of the experimental point distribution of the 3F (a) and the office (b). The black dots represent reference points and the red dots represent test points.

adjacent points is approximately 2.2 m, and 75 TPs are selected. In both the 3F and the office, each RP is measured 60 times and each TP is measured 40 times.

To avoid the variability of heterogeneous devices, all experimental data were collected by the same smartphone, a Huawei P9 smartphone with the Android operating system. For convenience, an independent coordinate system in each scene was also established. In the schematic diagram of the experimental environments as shown in figures 3(a) and (b), a plane Cartesian coordinate system is respectively established with O as the origin, the horizontal axis as the *X*-axis and the vertical axis as the *Y*-axis. The coordinates of sampling points were measured by using the Leica TS60 total station, which can achieve 3 millimeter precision.

4.2. Experimental result analysis

4.2.1. Selection of $\gamma_{\text{threshold}}$.　　Figure 4 shows the trend of the MEs of the WDKNN algorithm with the change of $\gamma_{\text{threshold}}$ in the 3F (a) and the office (b), respectively. Figure 4 indicate that with the increase of $\gamma_{\text{threshold}}$, the ME tends to decrease first and then increase on the whole. At a certain $\gamma_{\text{threshold}}$, the ME reaches the minimum, that is, at this $\gamma_{\text{threshold}}$, the best position accuracy is obtained.

Figure 4 demonstrates that the best position accuracy is achieved when $\gamma_{\text{threshold}}$ is 1.3 in the 3F and 1.13 in the office. The reason for this difference is that in the separate scenes, the indoor environments are different and the spacing of RPs is also diverse. In this paper, $\gamma_{\text{threshold}}$ is set to 1.3 for the 3F and 1.13 for the office.

4.2.2. Positioning performance of the proposed hybrid filtering algorithm.　　In this section, the position results of the hybrid filtering algorithm with the traditional KNN and WKNN algorithms are compared. WKNN-HF represents the RSS characteristic values that are extracted by hybrid filtering algorithm and the online phase adopts the traditional WKNN algorithm.

Figure 5 show the position accuracy of the KNN algorithm, the WKNN algorithm and the WKNN-HF algorithm with *K* ranging from 2 to 10 for the 3F (a) and the office (b), respectively. As the figures show, the ME tends to decrease first and then increase, and the position error of WKNN-HF is smaller than that of KNN and WKNN on the whole. In the office, when *K* is small, such as 2, 3, 4 and 5, the position accuracy of the proposed WKNN-HF algorithm improves obviously. With the increase of *K*, the gaps of the position results between the WKNN and WKNN-HF algorithms are reduced, and some position errors of WKNN-HF are even slightly larger than that of WKNN, such as when *K* is 7. The reason for this phenomenon is that, when *K* is large, the RPs selected for the position calculation will inevitably include the RPs that have large actual distances from the TP. The WDKNN-HF algorithm proposed in this paper can improve this phenomenon, and specific experimental analysis is presented below.
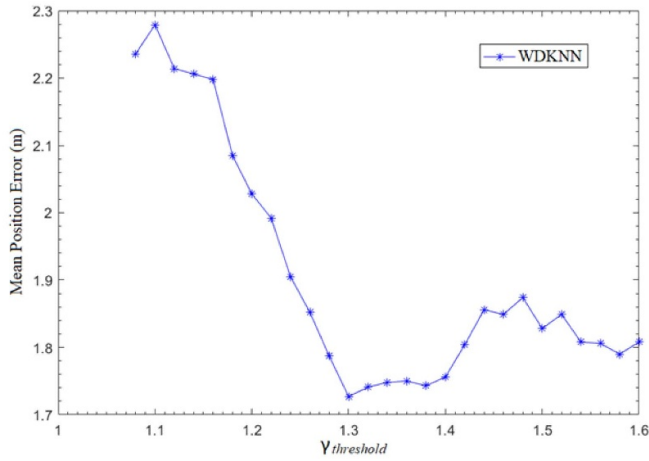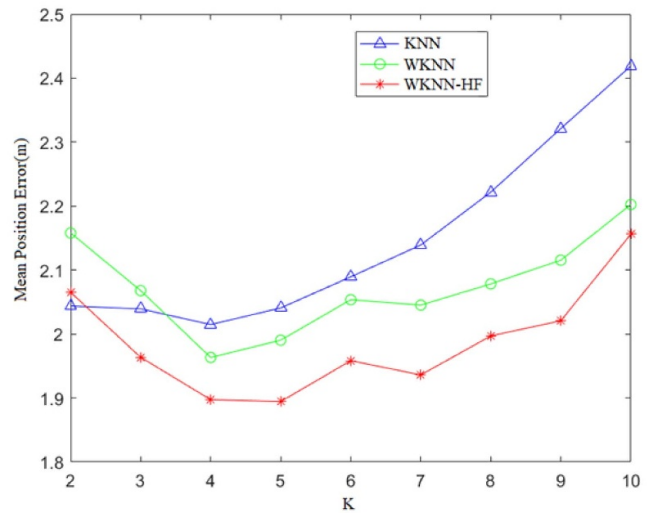
(a)
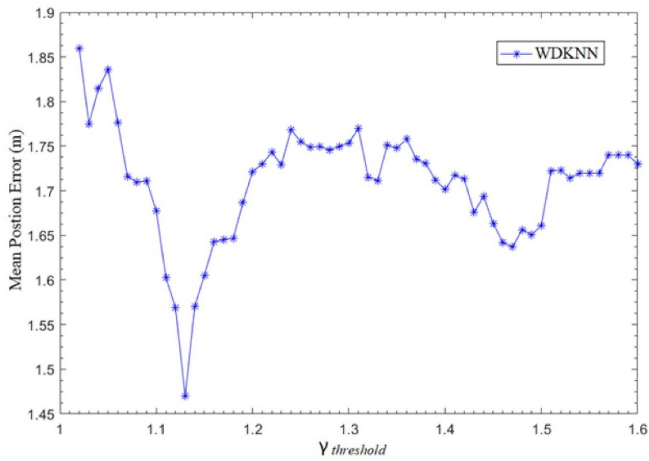
(b)

Figure 4. Mean position error with different $\gamma_{\text{threshold}}$ of the 3F (a) and the office (b).

(a)

(b)

Figure 5. Comparison of location accuracy of three algorithms in terms of mean position error of the 3F (a) and the office (b).
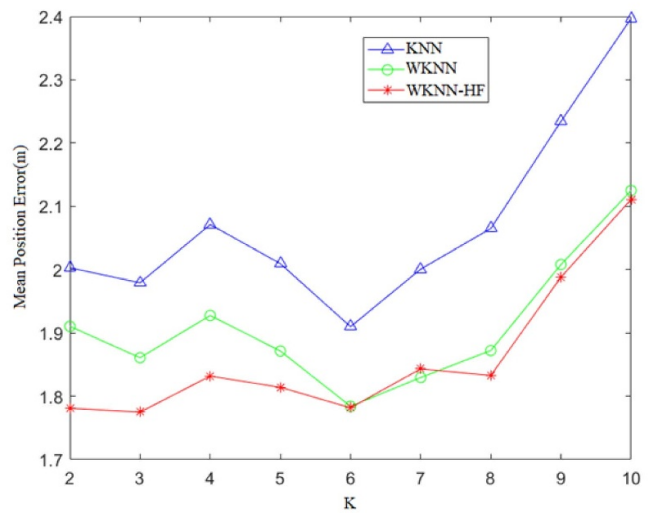
4.2.3. Positioning performance of the WDKNN algorithm. In this section, the positioning performance of WDKNN (taking the mean RSS as the characteristic value) and the traditional KNN and WKNN are compared. The last section indicates that in the 3F, the KNN and WKNN algorithms obtain the best position accuracy when K is 4, whereas in the office, K is 6. In this paper, the KNN and WKNN algorithms' K are set to 4 and 6 for the 3F and the office, respectively.

Figure 6 shows boxplots of these three algorithms, the boxplots display some positioning error statistics including the max, 90th, 75th, 25th and 10th percentile errors, median, min and mean error (the red line).

Figure 6 demonstrate that most of the indicators in the 3F and all indicators in the office of WDKNN are better than those of the KNN and WKNN algorithms. Take the office scene for instance: the ME of WDKNN is 1.63 m, which is better than the 1.91 m of the KNN and 1.78 m of WKNN. The 90th, 75th and 25th percentile errors of the WDKNN are 2.67 m, 2.27 m and 0.94 m, which are better than the 2.96 m, 2.57 m and 1.19 m of the KNN, and the 2.90 m, 2.42 m and 1.21 m of the WKNN, respectively.

4.2.4. Positioning performance of the proposed WDKNN-HF algorithm. In this section, the positioning performances of the WDKNN-HF algorithm and other algorithms, including KNN, WKNN, WKNN-HF and WDKNN, are compared. Table 2 depicts some positioning error statistics, including the ME, STD, RMSE and 90th percentile error.

From the results displayed in table 2, we can observe that the proposed WDKNN-HF algorithm obtains better position performance than the other four algorithms in terms of the ME, STD, RMSE and 90th percentile error. In 3F, the ME of WDKNN-HF is 0.32 m better than that of KNN, 0.23 m better than that of WKNN, 0.17 m better than that of WKNN-HF and 0.1 m better than that of WDKNN. The RMSE of WDKNN-HF is 2.10 m, which is better than the 2.48 m, 2.36 m, 2.28 m and 2.17 m of the KNN, WKNN, WKNN-HF and WDKNN, respectively. In the office, the ME of WDKNN-HF is 0.44 m better than that of KNN, 0.29 m better than that of WKNN
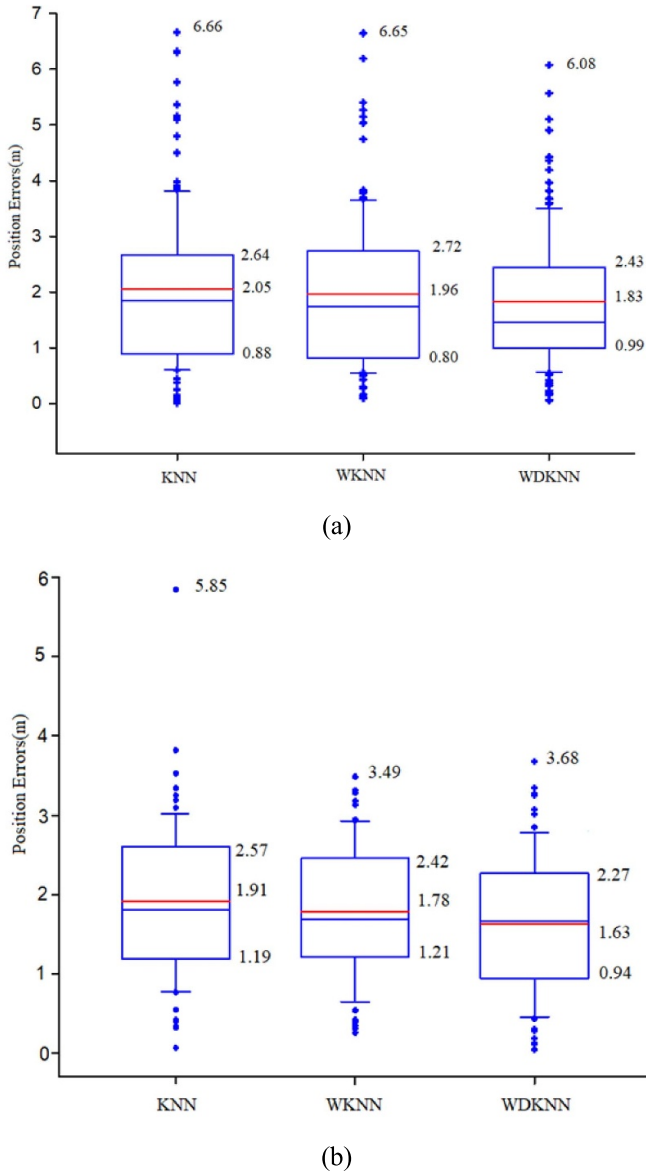
(a)

(b)

Figure 6. Comparison of the location accuracy of three algorithms in terms of the boxplot of the 3F (a) and the office (b).

Table 2. Positioning error statistics.

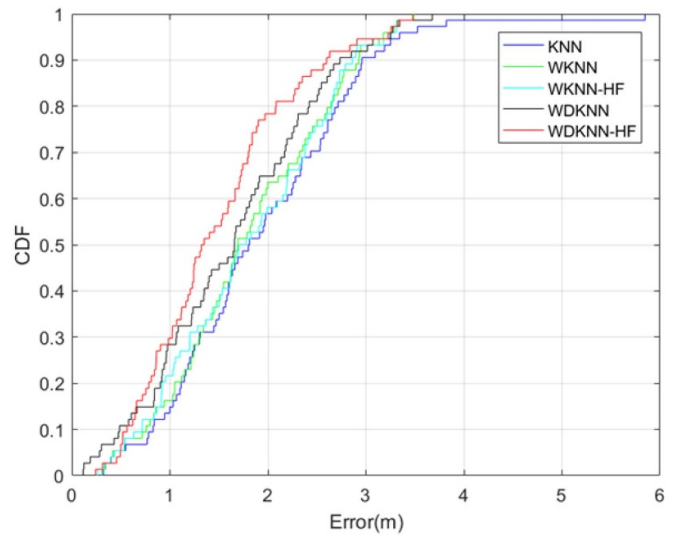| Scene	Algorithm	ME (m)	STD (m)	RMSE (m)	90th (m)
3F	KNN	2.05	1.40	2.48	3.71
	WKNN	1.96	1.31	2.36	3.58
	WKNN-HF	1.90	1.27	2.28	3.55
	WDKNN	1.83	1.17	2.17	3.45
	WDKNN-HF	1.73	1.19	2.10	3.12
Office	KNN	1.91	0.97	2.15	2.96
	WKNN	1.78	0.82	1.96	2.93
	WKNN-HF	1.78	0.85	1.97	2.86
	WDKNN	1.63	0.87	1.84	2.73
	WDKNN-HF	1.47	0.78	1.67	2.60

Figure 7. Comparison of the position accuracy of five algorithms in terms of CDF in the office.

Table 3. Positioning error statistics.

Scene	Algorithm	ME (m)	STD (m)	RMSE (m)	90th (m)
3F	EWKNN	1.95	1.60	2.52	3.71
	SAWKNN	1.86	1.21	2.22	3.54
	WDKNN-HF	1.73	1.19	2.10	3.12
Office	EWKNN	1.63	0.85	1.83	2.90
	SAWKNN	1.59	1.19	1.99	3.15
	WDKNN-HF	1.47	0.78	1.67	2.60

and WKNN-HF, and 0.16 m better than that of WDKNN. The RMSE of WDKNN-HF is 1.67 m, which is better than the 2.15 m, 1.96 m, 1.97 m and 1.84 m of the KNN, WKNN, WKNN-HF and WDKNN, respectively.

The 90th percentile error is a frequently used indicator to evaluate position performance. From table 2, we can observe that in the 3F, the proposed WDKNN-HF algorithm produces a 90th percentile error of 3.12 m, which is significantly better than the 3.71 m, 3.58 m, 3.55 m and 3.45 m of KNN, WKNN, WKNN-HF and WDKNN, respectively. In the office, the proposed WDKNN-HF algorithm produces a 90th percentile error of 2.60 m, which is significantly better than the 2.96 m, 2.93 m, 2.86 m and 2.73 m of KNN, WKNN, WKNN-HF and WDKNN, respectively.

Figure 7 shows the CDF of the position errors of these five algorithms of the office. When the error thresholds are 1 m, 2 m and 3 m, the CDF values of WDKNN-HF are 30.67%, 78.67% and 97.67%, respectively, which are significantly higher than the 16%, 57% and 90.67% of the KNN, the 17.33%, 64% and 93.33% of the WKNN, the 22.67%, 58.67% and 90.67% of the WKNN-HF, and the 29.33%, 65.33% and 92% of the WDKNN, respectively. The comparison of error statistics and the CDF indicates that the proposed WDKNN-HF algorithm significantly outperforms the other four algorithms.

4.2.5. Positioning performance of the proposed WDKNN-HF algorithm. Some existing dynamic *K* selection algorithms (EWKNN and SAWKNN) and the proposed WDKNN-HF
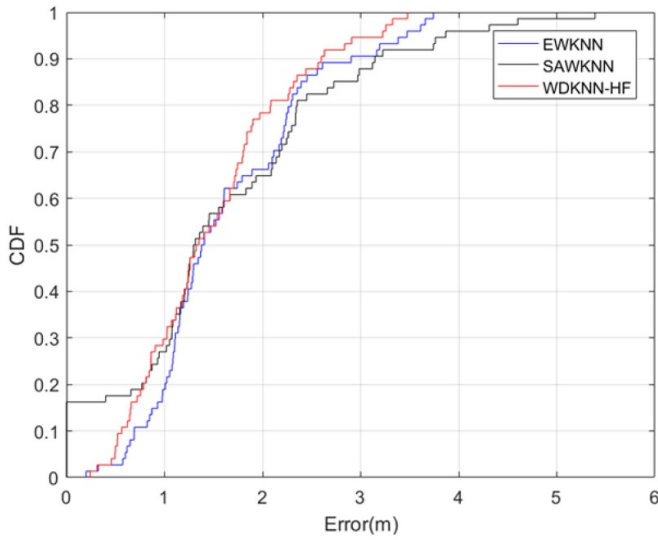
Figure 8. Comparison of the position accuracy of three algorithms in terms of CDF in the office.

algorithm are compared in this section. The evaluation indicators are the ME, STD, RMSE, 90th percentile error and CDF of the position errors.

The results displays in table 3 indicate that the proposed WDKNN-HF algorithm obtains better position performance than the other three algorithms. In 3F, the ME of the proposed WDKNN-HF is 1.73 m, which is better than the 1.95 m and 1.86 m of the EWKNN and SAWKNN, respectively. The RMSE of WDKNN-HF is 2.1 m, which is better than the 2.52 m and 2.22 m of the EWKNN and SAWKNN, respectively. WDKNN-HF produces a 90th percentile error of 3.12 m, which is better than the 3.71 m and 3.54 m of the EWKNN and SAWKNN, respectively. In the office, the WDKNN-HF algorithm produces an ME of 1.47 m, which is better than the 1.63 m and 1.59 m of the EWKNN and SAWKNN, respectively. The RMSE of WDKNN-HF is 1.67 m, which is better than the 1.83 m and 1.99 m of the EWKNN and SAWKNN, respectively. WDKNN-HF produces a 90th percentile error of 2.60 m, which is better than the 2.90 m and 3.15 m of the EWKNN and SAWKNN, respectively.

Figure 8 shows the CDF of the position error of these three algorithms of the office, and we can observe that all position errors of the WDKNN-HF and EWKNN algorithms are less than 4 m, whereas the max position error of the SAWKNN algorithm is 5.39 m. When the error thresholds are 1 m, 2 m and 3 m, the CDF values of WDKNN-HF are respectively 30.67%, 78.67% and 97.67%, which are higher than the 20%, 66.67% and 90.67% of the EKNN and the 28%, 65.33% and 88% of the SAWKNN. Therefore, compared to the existing dynamic K selection algorithms, the proposed WDKNN-HF algorithm can achieve a performance gain.

Figure 9 displays the location performance of the seven algorithms in terms of the error vector at each TP of the office. The error vector is represented by an arrow pointing from the real position to the estimated position. Figure 9
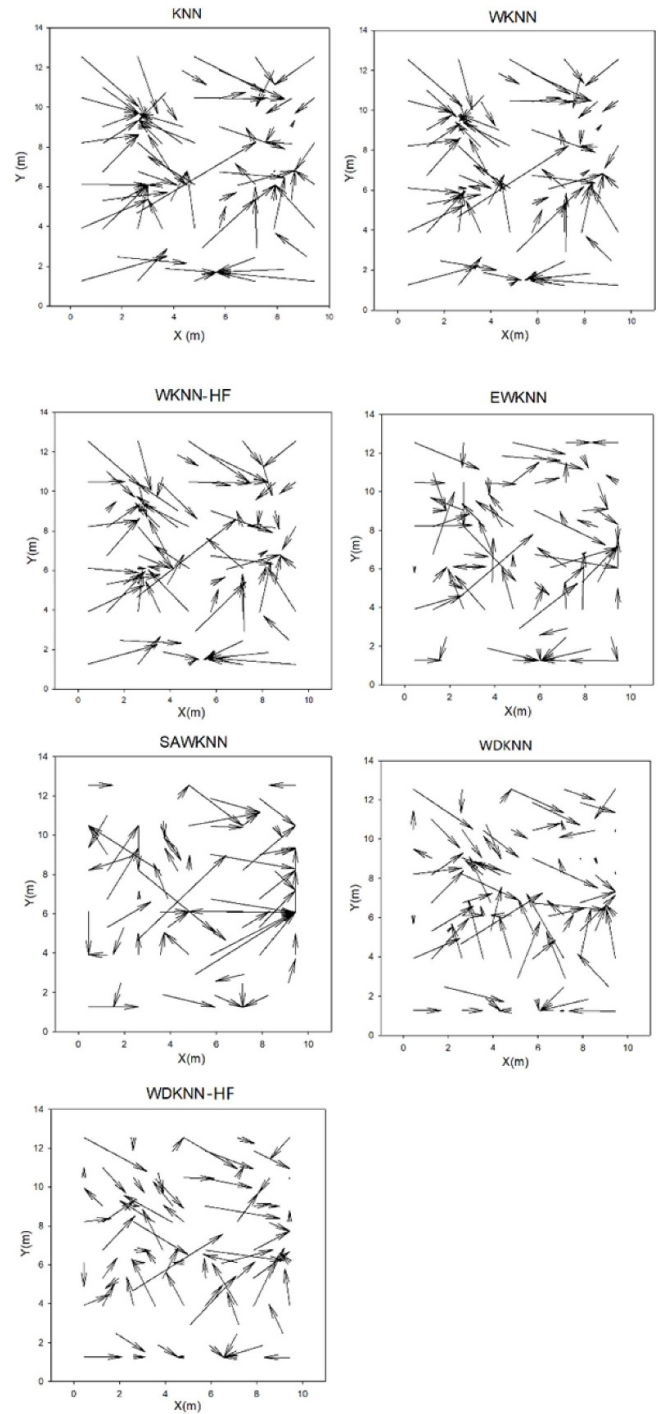

Figure 9. Comparison of the position accuracy of seven algorithms in terms of error vectors in the office.

demonstrates that the proposed WDKNN-HF outperforms the other algorithms. The position results of the traditional KNN, WKNN and WKNN-HF algorithms exhibit the centralization phenomenon, whereas the results of SAWKNN exhibit polarization; the WDKNN-HF algorithm can improve these phenomena. Relatively, EWKNN and WDKNN perform better, close to WDKNN-HF, but their MEs are larger than that of WDKNN-HF.

4. Conclusion

This paper has presented a novel WDKNN-HF algorithm, which adopts a hybrid filtering algorithm to extract RSS characteristic values and a better dynamic *K* value selection matching algorithm in the online phase. Field tests were carried out to verify the proposed WDKNN-HF algorithm. Experimental results demonstrated that compared with the traditional mean RSS value, the hybrid filtering algorithm can effectively improve the quality of the RSS signals, which considerably improves the positioning accuracy. The proposed dynamic *K* value selection algorithm based on Euclidean distance is intended to eliminate RPs that have large actual distances from the TP in the online positioning phase. According to the experimental results of two scenes, the improvement of the proposed WDKNN-HF algorithm in localization performance is diverse in different environments, and the performance of the office is better than that of the 3F. Generally, the position performance of the proposed WDKNN-HF algorithm is considerably better than that of traditional algorithms, such as the KNN and WKNN algorithms, and it also outperforms existing dynamic *K* selection algorithms such as EWKNN and SAWKNN. Possible directions for future work include the selection of APs to further improve position accuracy, the rapid construction of a fingerprint database and an efficient position calculation algorithm to improve the efficiency of the positioning system.

Acknowledgments

ORCID iD

Xuesheng Peng https://orcid.org/0000-0003-4342-6514

References

[1] Ali M, Hur S, Park S and Park Y 2019 Harvesting indoor positioning accuracy by exploring multiple features from received signal strength vector *IEEE Access* **7** 52110–21

[2] Mahfouz M, Fathy A, Kuhn M and Wang Y 2009 Recent trends and advances in UWB positioning *2009 IEEE MTT-S International Microwave Workshop on Wireless Sensing, Local Positioning, and RFID* (*Cavtat, Croatia*) pp 1–4

[3] Wang Y, Yang X, Zhao Y, Liu Y and Cuthbert L 2013 Bluetooth positioning using RSSI and triangulation methods *2013 IEEE 10th Consumer Communications and Networking Conf. (CCNC)* pp 837–42

[4] Huang C, Lee L, Ho C, Wu L and Lai Z 2015 Real-time RFID indoor positioning system based on Kalman-filter drift removal and heron-bilateration location estimation *IEEE Trans. Instrum. Meas.* **64** 728–39

[5] Sadowski S and Spachos P 2018 RSSI-based indoor localization with the internet of things *IEEE Access* **6** 30149–61

[6] Li Z, Liu J, Yang Y, Niu X, Li L, Wang Z and Chen R 2018 A Bayesian density model based radio signal fingerprinting positioning method for enhanced usability *Sensors* **18** 4063

[7] Jung J, Lee S and Myung H 2014 Indoor mobile robot localization and mapping based on ambient magnetic fields and aiding radio sources *IEEE Trans. Instrum. Meas.* **64** 1922–34

[8] Chen R, Chen W, Chen X, Zhang X and Chen Y 2010 Sensing strides using EMG signal for pedestrian navigation *GPS Solut.* **15** 161–70

[9] Chen W, Chen R, Chen X, Zhang X, Chen Y, Wang J and Fu Z 2011 Comparison of EMG-based and accelerometer-based speed estimation methods in pedestrian dead reckoning 2011 *J. Navig.* **64** 265–80

[10] Tian X, Shen R, Liu D, Wen Y and Wang X 2017 Performance analysis of RSS fingerprinting based indoor localization *IEEE Trans. Mob. Comput.* **16** 2847–61

[11] Guo G, Chen R, Ye F, Peng X, Liu Z and Pan Y 2020 Indoor smartphone localization: a hybrid WiFi RTT-RSS ranging approach *IEEE Access* **7** 176767–81

[12] Chen R and Feng Y 2018 An overview of indoor positioning technology based on Wi-Fi channel state information *Geo. Inf. Sci. Wuhan Univ.* **43** 2064–70

[13] Chen R, Chu T, Liu K, Liu J and Chen Y 2015 Inferring human activity in mobile devices by computing multiple contexts *Sensors* **15** 21219–38

[14] Liu H, Darabi H, Banerjee P and Liu J 2007 Survey of wireless indoor positioning techniques and systems *IEEE Trans. Syst. Man Cybern. C Appl. Rev.* **37** 1067–80

[15] Bahl P and Padmanabhan V 2000 Radar: an in-building RF-based user location, and tracking system *Conf. Comput. Commun. Nineteenth Annual Joint Conf. IEEE Comput. Commun. Soc. Tel Aviv, Israel, Israel* pp 775–84

[16] Xu Y, Zhou M, Meng W and Ma L 2010 Optimal KNN positioning algorithm via theoretical accuracy criterion in WLAN indoor environment *2010 IEEE Global Telecommunications Conf. GLOBECOM 2010* (*Miami, FL*) pp 1–5

[17] Brunato M and Battiti R 2005 Statistical learning theory for location fingerprinting in wireless LANs *Comput. Netw.* **47** 825–45

[18] Zhou M, Wang Y, Tian Z, Lian Y, Wang Y and Wang B 2019 Calibrated data simplification for energy-efficient location sensing in internet of things *IEEE Internet Things J.* **6** 6125–33

[19] Liu H, Lo W and Tseng C 2014 A Wi-Fi-based weighted screening method for indoor positioning systems *Wireless Pers. Commun.* **79** 611–27

[20] Xue W, Qiu W, Hua X and Yu K 2017 Improved Wi-Fi RSSI measurement for indoor localization *IEEE Sensors J.* **17** 2224–30

[21] Yu K and Guo Y 2008 Improved positioning algorithms for nonline-of-sight environments *IEEE Trans. Veh. Technol.* **57** 2342–53

[22] Diversi R, Guidorzi R and Soverini U 2005 Kalman filtering in extended noise environments *IEEE Trans. Autom. Control* **50** 1396–402

[23] Chen Z, Zou H, Jiang H, Zhu Q, Soh Y and Xie L 2015 Fusion of Wi-Fi, smartphone sensors and landmarks using the Kalman filter for indoor localization 2015 *Sensors* **15** 715–32

[24] Arulampalam M, Maskell S, Gordon N and Clapp T 2002 A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking *IEEE Trans. Signal Process.* **50** 174–88

[25] Wang F, Zhang J, Lin B and Li X 2018 Two stage particle filter for nonlinear Bayesian estimation *IEEE Access* **6** 13803–9

[26] Ma R, Guo Q, Hu C and Xue J 2015 An improved Wi-Fi indoor positioning algorithm by weighted fusion *Sensors* **15** 21824–43

[27] Shin B, Lee J, Lee T and Kim H 2012 Enhanced weighted k-nearest neighbor algorithm for indoor Wi-Fi positioning systems *2012 8th Proc. IEEE Int. Conf. Comput. Technol. Inf. Manage. (NCM and ICNIT) Seoul, South Korea* 574–7

[28] Xue W, Qiu W, Hua X, Wu B and Cai J 2016 Influence analysis of RSS signal characteristic value on Wi-Fi indoor positioning accuracy *J. Geomatics* **41** 23–26

[29] Kaemarungsi K and Krishnamurthy P 2012 Analysis of WLAN's received signal strength indication for indoor location fingerprinting *Pervasive Mob. Comput.* **8** 292–316

[30] Sun T and Zhu M 2015 A hybrid filter algorithm based on RSSI *Ind. Control Comput.* **28** 64–65

[31] Zhang J, Zhang L, Ying Y and Gao F 2009 Research on distance measurement based on RSSI of ZigBee *Chin. J. Sensors Actuators* **22** 285–8

[32] Dong S 2016 Research of positioning system in wireless sensors network based on mixed filter algorithm M.S. thesis Dept. Electron. Eng., Inner Mongolia Univ.

[33] Song Y and Yu H 2009 A RSS based indoor tracking algorithm using particle filters *2009 Global Mobile Congress* (*Shanghai, China*) pp 1–4

[34] Zhuang Y, Li Y, Qi L, Lan H, Yang J and El-Sheimy N 2016 A two-filter integration of MEMS sensors and Wi-Fi fingerprinting for indoor positioning *IEEE Sensors J.* **16** 5125–6

[35] Torres-Sospedra J, Montoliu R, Trilles S, Belmonte O and Huerta J 2015 Comprehensive analysis of distance and similarity measures for Wi-Fi fingerprinting indoor positioning systems *Expert Syst. Appl.* **42** 9263–78

[36] Hu J, Liu D, Yan Z and Liu H 2018 Experimental analysis on weight k-nearest neighbor indoor fingerprint positioning *IEEE Internet Things J.* **6** 891–7

[37] Xue W, Hua X, Li Q and Yu K 2018 A new weighted algorithm based on the uneven spatial resolution of RSSI for indoor localization *IEEE Access* **6** 26588–95

[38] Gordon N, Salmond D and Smith A 1993 Novel approach to nonlinear/non-Gaussian Bayesian state estimation *Proc. Inst. Elect. Eng. F* **140** 107–13

[39] Christoph B 1998 Surface emissivity and temperature retrieval for a hyperspectral sensor *IGARSS '98. Sensing and Managing the Environment. 1998 IEEE Int. Geosci. Remote Sens* (*Seattle, WA*) pp 546–9

[40] Joshi P, Colombi D, Thors B, Larsson L and Törnevik C 2017 Output power levels of 4G user equipment and implications on realistic RF EMF exposure assessments *IEEE Access* **5** 4545–50

[41] Guo G, Chen R, Ye F, Chen L, Pan Y, Liu M and Cao Z 2018 A pose awareness solution for estimating pedestrian walking speed *Remote Sens.* **11** 1–18

[42] Liu J, Chen R, Pei L, Guinness R and Kuusniemi H 2012 A hybrid smartphone indoor positioning solution for mobile LBS *Sensors* **12** 17208–33

[43] Sharp I and Yu K 2018 *Wireless Positioning: Principles and Practice* (Berlin: Springer) pp 225–39

[44] Chen R and Chen L 2017 Development state and challenges of indoor positioning technology based on smart phones *Acta Geodaetica Et Cartographica Sinica* **46** 1316–26

[45] Zhou M, Wang Y, Liu Y and Tian Z 2019 An information-theoretic view of WLAN localization error bound in GPS-denied environment *IEEE Trans. Veh. Technol.* **68** 4089–93

[46] Nguyen C, Georgiou O, Yonezawa Y and Doi Y 2017 The wireless localization matching problem *IEEE Internet Things J.* **4** 1312–132