IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 9 March 2023, accepted 28 March 2023, date of publication 31 March 2023, date of current version 5 April 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3263583

APPLIED RESEARCH

A WiFi Indoor Location Tracking Algorithm Based on Improved Weighted K Nearest Neighbors and Kalman Filter

JIUSONG HU AND CONGWEI HU
College of Railway Transportation, Hunan University of Technology, Zhuzhou 412007, China
Corresponding author: Jiusong Hu (hujiusong2008@163.com)

This work was supported in part by the Natural Science Youth Foundation of Hunan Province under Grant 2020JJ5144, and in part by the Scientific Research Project of Hunan Education Department under Grant 19C0563.

ABSTRACT The weighted K-nearest neighbors (WKNN) algorithm is a widely adopted lightweight methodology for indoor WiFi positioning based on location fingerprinting. Nonetheless, it suffers from the disadvantage of a fixed K value and susceptibility to incorrect reference point matching. To address this issue, we present a novel algorithm in this paper, referred to as static continuous statistical characteristics-soft range limited-self-adaptive WKNN (SCSC-SRL-SAWKNN). Our algorithm not only takes into account location tracking in the motion state but also exploits the continuous statistical features of extended periods of inactivity to enhance localization. In the motion state, we initially employ the self-adaptive WKNN (SAWKNN) algorithm to determine the optimal K value, followed by the employment of the soft range limited KNN (SRL-KNN) algorithm to identify the correct reference point and ultimately estimate the position. When a prolonged stationary state is detected, we first utilize the moving window method to obtain a more stable position fingerprint, and then proceed with the positioning process in the same motion state. Ultimately, we use Kalman filter to generate the location trajectory. Our experimental findings demonstrate that the proposed SCSC-SRL-SAWKNN algorithm outperforms traditional WKNN, SAWKNN, and SRL-KNN techniques in terms of localization accuracy and location trajectory. Specifically, the localization accuracy of our algorithm is 56.7% and 36.6% higher than that of traditional WKNN in the static state and overall situation, respectively.

INDEX TERMS WiFi, indoor location, WKNN, fingerprint.

I. INTRODUCTION

With the widespread usage and advancement of mobile devices, the demand for location-based services (LBS) has skyrocketed. Applications related to LBS have now become an essential element of both industrial production and everyday life. In outdoor environments, GPS and Beidou navigation and positioning systems have reached a state of maturity and perfection, providing highly accurate results at a lower cost. Notable examples include commonly used map navigation, real-time public transport location broadcasting, and shared vehicle rental and return points. All of these applications are developed and implemented based on GPS or Beidou navigation and positioning systems. Conversely, GPS has significant limitations in indoor environments, where satellite signals are often obstructed by building walls. Consequently, indoor positioning techniques have been extensively researched, including Ultra Wide Band (UWB) [1], Bluetooth [2], WiFi [3], electromyography (EMG) signals [4], and others [5]. These positioning technologies each possess their own unique characteristics, with WiFi indoor positioning technology being one of the most widely studied in recent years. Its advantages include the extensive deployment of wireless networks in indoor environments and the fact that most mobile devices are equipped with built-in WiFi modules, thus obviating the need for additional hardware. WiFi indoor positioning is the most universal and inexpensive option when compared to other positioning technologies.

The associate editor coordinating the review of this manuscript and approving it for publication was Stefano Scanzio.

Terminal equipment can be located using indoor wireless network signals. As an illustration, the location fingerprint-based method can be considered. Indoor wireless networks typically consist of several access points (APs), each emitting a wireless signal covering a particular area. The transmission power of an access point (AP) is theoretically fixed, so the signal strength in a particular area is theoretically consistent. The mobile device can determine its location by evaluating the signal strength received from the AP. This fingerprint-based localization method exhibits great potential [6]. Euclidean distance [7], [8] and Manhattan distance [9], [10] are commonly employed in localization methods based on location fingerprints. Time of arrival-based methods [11] and angle of arrival-based methods [12] are also available for WiFi indoor positioning, but these require additional hardware, which is absent in most mobile terminals, and as a result, have not gained significant attention.

Enhancing the accuracy of WiFi positioning, which remains relatively low, poses the greatest challenge. The location-based fingerprint method in WiFi positioning is categorized into three groups: deterministic methods [13], probabilistic methods [14], and deep learning methods [15]. Despite their unparalleled accuracy, deep learning methods are inherently complex, requiring significant computational resources. Conversely, deterministic methods are comparable to probabilistic methods in terms of performance but offer the added benefit of reduced complexity, rendering them a favored choice among researchers.

Deterministic methods achieve localization by calculating the similarity of fingerprints between the online and offline phases. In the offline phase, the task involves constructing a fingerprint database comprising the received signal strength index (RSSI) gathered by the mobile terminal at each reference points (RPs). During the online phase, the collected location fingerprints are matched with those in the offline phase. The positioning algorithm is then utilized to determine the positioning. However, in the actual environment, RSSI is influenced by various factors, such as object movement, changes in electromagnetic waves, variations in antenna directivity, and radio frequency interference. Therefore, it is highly probable that the offline database may not contain the location fingerprint corresponding to the online stage. In this scenario, the RPs with the highest location fingerprint similarity between the online phase and the offline database is used as a candidate point. The mobile terminal is subsequently positioned at the center of the K candidate points. This approach is known as the K-nearest neighbors (KNN) positioning method. On this basis, weights corresponding to the candidate points are added, which further improves the positioning accuracy [16]. This is a prevalent algorithm known as the weighted K-nearest neighbors (WKNN).

The WKNN algorithm has been the subject of numerous discussions in the academic literature. However, it continues to confront a series of daunting challenges, including the following:

A. THE LONG-TERM STATIC CASE IS NOT SPECIFICALLY CONSIDERED

In practical situations, the existence of multipath effects and channel fading leads to variations in RSSIs collected by terminal devices, even during successive time intervals while situated in the same location. Regrettably, the long-term static state that is vital for fingerprint stability is often ignored in the present-day WiFi-based indoor positioning methodology. Therefore, improving the stability of the fingerprint in the static state is often overlooked.

B. SETTING OF K VALUE: DIFFERENT K VALUE MEANS MATCHING DIFFERENT NUMBERS OF PRS

This will result in different positioning accuracy. Furthermore, the optimal value of K for achieving the highest degree of accuracy varies according to differences in the indoor and network environmental conditions.

C. PHYSICAL DISTANCE DOES NOT MATCH FINGERPRINT SIMILARITY

Some location fingerprints that are physically distant may bear resemblance to those which are physically closer. The WKNN algorithm may therefore be misled, leading to a relatively significant error.

The challenge of setting the K value has been addressed by the proposed self-adaptive weight K-nearest neighbors (SAWKNN) algorithm [17]. The issue of physical distance not matching fingerprint similarity is tackled by the soft range limited K-nearest neighbors (SRL-KNN) algorithm [18]. To overcome these challenges, this paper proposes a localization algorithm based on the static continuous statistical characteristics of fingerprint signals. The proposed algorithm, named static continuous statistical characteristics-soft range limited-self-adaptive WKNN (SCSC-SRL-SAWKNN), retains the recent location information of the user and combines SAWKNN and SRL-KNN algorithms to achieve the best positioning accuracy in different localization states. This algorithm sets itself apart from its counterparts through its distinctive approach. It discerns between the stationary and moving states in the user positioning process, and handles them separately. In the static state, the algorithm employs the concept of a moving window to first process the fingerprint signal. Then, the cumulative mean of the signal is used to enhance the stability of RSSI. Finally, in conjunction with SAWKNN, the positioning is completed. On the other hand, in the moving state, the soft range limit is used to resolve the issue of physical distance not matching fingerprint similarity. The SAWKNN algorithm is combined with the soft range limit to complete the positioning. The proposed SCSC-SRL-SAWKNN algorithm in this paper retains the qualities of different localization states, and processes them separately based on their respective characteristics. The algorithm is capable of achieving optimal positioning accuracy in different positioning states, and it doesn't require prior knowledge. As a result, there is no need to deploy any additional hardware. Numerous experimental results attest that the accuracy of the proposed algorithm surpasses that of traditional WKNN and the optimized SAWKNN and SRL-KNN algorithms under both static and dynamic conditions. As compared to the traditional WKNN, the overall positioning accuracy has been improved by 36.6%. Moreover, being a lightweight algorithm, it can enable real-time positioning while consuming minimal system resources.

The remainder of this paper shall be arranged in the following manner: Section II will introduce related research. Section III shall expound upon the indoor positioning system and discuss pertinent optimization issues. In Section IV, the SCSC-SRL-SAWKNN algorithm shall be presented. Section V will furnish a comprehensive assessment via an ample array of real-world experiments. Finally, in Section VI, we will conclude the paper and outline potential areas of future research.

II. RELATE WORK

Over the past decade, there has been a staggering surge in the proliferation of mobile terminals. Consequently, various indoor positioning systems have been extensively studied. These systems can broadly be classified into two categories based on hardware deployment. The first category includes infrared [19], ultrasonic [20], Bluetooth [2], Ultra Wide Band (UWB) [21], Radio Frequency Identification (RFID) [22], which can provide high positioning accuracy but require additional specialized hardware deployment. The second category includes positioning technologies such as WiFi [23] and geomagnetic field [24], where users can rely on existing wireless networks to enable location functionality with the added advantage of lower costs. The research presented in this paper pertains to the second category of methods. Among these, fingerprint-based localization methods are the most common.

In the realm of WiFi positioning, distance-based methods have also been utilized [25], [26]. However, deploying additional hardware and prior knowledge of the location of each AP are required for these methods. As such, their practical application is limited and they are often not prioritized in most situations.

The method most commonly used for location-based fingerprinting is the KNN algorithm proposed by RANDA [27]. An optimization algorithm based on KNN was proposed in [28]. In addition, the weighted KNN (WKNN) algorithm was proposed in [29], which slightly improves the positioning accuracy compared to the KNN algorithm through the addition of weights. However, determining the optimal K value has always been a challenging problem, and a fixed experience value is typically used. The optimal K value changes with the environment, and thus cannot be obtained using traditional WKNN. To address this issue and ensure the best K value is obtained for each positioning, the SAWKNN algorithm was proposed in [17]. It enables adaptive adjustment of the K value through the fingerprint similarity threshold, resulting in improved localization accuracy compared to WKNN. However, the problem of mismatch between physical distance and fingerprint similarity is ignored, which can result in large positioning errors if the wrong candidate RPs is chosen.

Additionally, a method exists [30] that records the user's latest location information to determine their speed and direction, enabling prediction of their next location. However, if the user is moving slowly or in an irregular pattern, the prediction error may be significant. Studies have also explored the mismatch between distance and fingerprint similarity, wherein a location fingerprint at a greater physical distance may exhibit significant similarity to the fingerprint of the current location, leading to a large positioning error for that location. The SRL-KNN method proposed in [18] centers the circle on the user's previous position and sets the radius as a range factor based on the actual movement situation to limit the range. This solution addresses the issue of physical distance not matching fingerprint similarity during positioning, although it does not consider the static positioning scenario. Additionally, the signal trend index (STI) has been used in some studies to compare the vector shape of the RSSI of the mobile terminal device during the online phase with that of the offline database. The signal trend exponential weighted K-nearest neighbor (STI-WKNN) algorithm [31] has been proposed as a solution with better performance targeting heterogeneous devices.

In recent years, the field of deep learning has made remarkable progress, leading to the emergence of a plethora of indoor localization techniques employing deep learning methods [15], [32], [33], [34]. Indoor localization can be accomplished by means of probability estimation, which sets it apart from deterministic techniques such as WKNN. Recent breakthroughs [35], [36], [37] have yielded comparable localization accuracy as deterministic methods but at the expense of higher computational complexity. Deep learning technology, while undoubtedly possessing exceptional accuracy, is unfortunately accompanied by the highest degree of complexity, leading to a significant requirement for hardware resources that pose a challenge to its widespread adoption for real-time positioning.

III. INDOOR POSITIONING SYSTEM

In this section, we shall undertake a systematic analysis of the three primary issues with the WKNN algorithm. These issues include a lack of specific considerations for static situations, problems with setting the K value, and a mismatch between physical distance and fingerprint similarity. Firstly, we shall provide an overview of the system configuration, followed by an explanation of how each of these three problems can significantly affect the accuracy of indoor positioning.

A. SYSTEM CONFIGURATION

Fig. 1 depicts the comprehensive block diagram of the WiFi indoor positioning system. The system is comprised of two stages: the offline stage and the online stage. The prime objective of the offline stage is to accomplish the establishment of the offline location fingerprint database. The online stage is

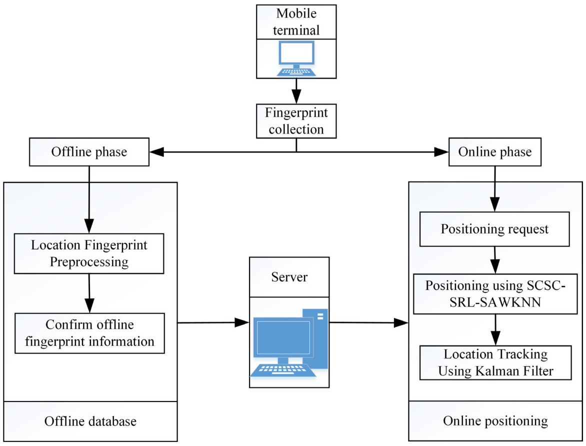

FIGURE 1. Block diagram of WiFi indoor positioning system.

responsible for utilizing the proposed positioning algorithm to match the location fingerprints of the online phase with those of the offline phase to achieve accurate positioning. Lastly, the Kalman filter is employed to accomplish positioning tracking.

We have constructed an indoor positioning system in China Power Valley located in Zhuzhou City, which is illustrated in Fig. 2. In the building, there is no centralized deployment of WiFi devices. Instead, these WiFi networks are distributed across various locations in each room. During our field tests, we identified a total of 339 APs in the building, with a grid spacing of 1 meter.

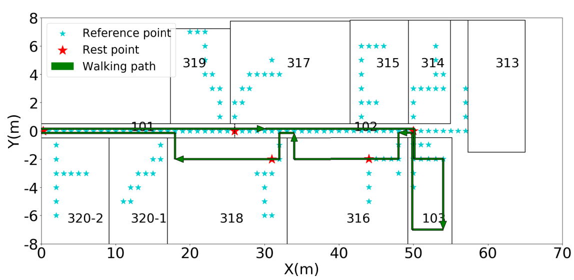

FIGURE 2. Layout of the indoor positioning system. Each blue dot represents a reference point.

We conducted our indoor positioning system experiment using an ASUS-FX50J mobile terminal device equipped with a Realtek 8811CU Wireless LAN 802.11ac USB NIC, as illustrated in Fig. 2. The experiment was conducted in a building containing 10 rooms and a corridor, where WiFi networks were distributed throughout the building with no centralized deployment of APs. A total of 339 APs were found in the building with a grid spacing of 1 meter. In order to facilitate data statistics, the corridor was divided into two parts, 101 and 102, starting from 26 meters. A total of 155 RPs fingerprints were collected and marked with blue stars, with adjacent RPs separated by 1 meter. Due to the presence of walls and desks, the distribution of RPs was non-uniform in the rooms, with each room containing 10 RPs except for 101 and 102, which contained 29 and 36 RPs, respectively.

We performed 100 consecutive signal acquisitions on each RPs at a rate of 1 per second, and experimental data showed that each RPs can receive WiFi signals from 27 to 61 APs.

B. OBSERVATION

We carry out three sets of experiments using the collected fingerprint data with the aim of demonstrating three issues concerning the conventional WKNN positioning algorithm.

1) THE IMPORTANCE OF LONG-TERM STATIC STATES IS OFTEN OVERLOOKED

The RSSI collected by the mobile terminal under static conditions can be easily affected by various environmental factors, such as the movement of people or objects, changes in electromagnetic waves, and variations in antenna directionality. These factors can cause fluctuations in the collected RSSI, making it challenging to accurately match location fingerprints between the online and offline stages.

We consecutively acquired signals 100 times at a specific test points (TPs). From dozens of available APs, we randomly selected the RSSI values of 5 APs. The RSSI values of these 5 selected APs are presented in Fig. 3.

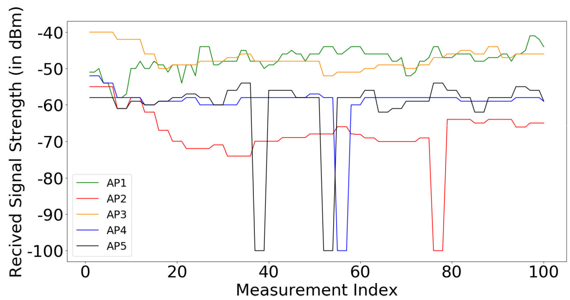

FIGURE 3. The RSS values over time using same devices from the diverse AP and at the fixed location.

The Fig. 3 reveals that there are significant fluctuations in the RSSI collected by the same device even when it is in a static state. Directly using an RSSI with significant fluctuations as the position fingerprint for positioning will result in poor accuracy. However, if the static conditions during the positioning process are taken into account, the stability of the position fingerprint can be considerably enhanced, leading to improved positioning accuracy.

2) SETTING OF K VALUE

We carried out a comprehensive set of 860 online localization experiments utilizing the WKNN algorithm, wherein each test point (TP) was scanned at least once. The value of K was varied from 1 to 20. The resulting positioning errors for the WKNN algorithm, at varying K values, have been graphically represented in Fig. 4.

It is evident that the average error is significantly influenced by the variation of the K value. Within the range of K between 5 and 10, the average error remains relatively stable. However, the average error steadily increases

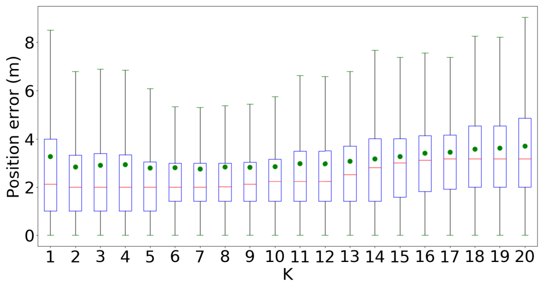

FIGURE 4. Position result of traditional WKNN algorithm with different K value. The error is measured in box charts.Each box chart contains six values from top to bottom: the upper edge (maximum value), the upper quartile (75%), the average (small circle), the median (red line), the lower quartile (25%), and the lower edge (minimum value).

as K values ascend from 11 to 20. In tandem, the upper limit of the error box also increases, indicating a surge in the number of larger errors. The K value of 7 is optimal for minimizing the average error, and its values fluctuate within a narrow error box (representing 75% of the sample). The error box for K values 1-4 is larger compared to K value 7. Here, the upper quantile of the box is rising, while the lower quantile is decreasing, suggesting that the overall positioning error fluctuates greatly. Although many situations result in large positioning errors, WKNN exhibits more precise positioning accuracy than when K is set to 7 at certain times.

3) PHYSICAL DISTANCE DOES NOT CORRESPOND TO FINGERPRINT SIMILARITY

Due to the occurrence of channel fading and the multipath effect in the signal propagation process, in addition to the impact of people's movements, the opening and closing of doors and windows, and changes in the location of objects in the actual environment. The phenomenon in question may result in the fingerprint at a distant physical location appearing similar to that of the current location at a particular instance. This could lead to erroneous outcomes in WKNN, consequently amplifying the positioning error. The situation is aptly demonstrated in Fig. 5.

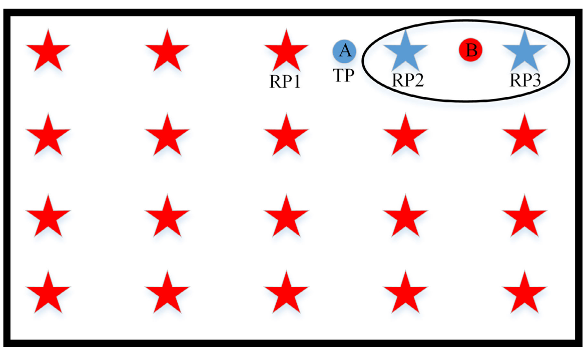

FIGURE 5. The problem of mismatch between physical distance and fingerprint similarity.

In Fig. 5, point A represents the location of TP, while RP1, RP2, and RP3 are three reference points respectively. TP lies between RP1 and RP2, but the position fingerprint of TP matches the position fingerprint of RP2 and RP3 most. Consequently, the position of TP is erroneously estimated at point B. This demonstrates that the accuracy of localization is heavily influenced by the disparity between physical distance and fingerprint resemblance. Hence, a certain level of physical constraints is necessary.

Next we will focus on how to solve these three problems.

IV. SCSC-SRL-SAWKNN ALGORITHM

In this section, we present the SCSC-SRL-SAWKNN algorithm for indoor localization. Initially, we will provide a brief overview of the relevant knowledge of WKNN. Subsequently, we will elaborate on how to leverage the static properties in the localization process. Next, the analysis focuses on addressing the challenges related to setting the K value and resolving the mismatch between physical distance and fingerprint similarity. Additionally, we will review the SAWKNN and SRL-KNN algorithms separately. Ultimately, we will amalgamate the aforementioned three characteristics and present the SCSC-SRL-SAWKNN algorithm.

A. WKNN BASICS

In this section, we first provide an overview of the relevant concepts related to WKNN. Let $rss_{i,j}$ denote the average number of RSSIs collected from the i_{th} AP at the j_{th} reference point (RP), where the total number of APs is denoted by N. In situations where a mobile terminal can detect multiple APs at one RP, the fingerprint vector collected by the j_{th} RP can be represented by $\vec{r}ss_j$:

$$\vec{r}ss_j = \left[rss_{1,j}, rss_{2,j}, \ldots, rss_{i,j}, \ldots, rss_{N,j}\right]. \tag{1}$$

During the online testing phase, the RSSI fingerprint vector $\vec{r}ss_r$ received at the r_{th} TP can be expressed as:

$$\vec{r}ss_r = \left[rss_{1,r}, rss_{2,r}, \ldots, rss_{i,r}, \ldots, rss_{N,r}\right]. \tag{2}$$

The location fingerprint similarity difference between the j_{th} RP and r_{th} TP can be defined as:

$$dif_{r,j} = \frac{\left\|\vec{r}ss_r \cdot P^T - \vec{r}ss_j \cdot P^T\right\|^2}{M} \tag{3}$$

where || || represents the Manhattan distance. Equation (3) defines the Manhattan distance D_j^r between the j_{th} PR and the r_{th} TP. M in Equation (3) represents the number of APs that can be collected in the online phase, and P is represented by:

$$P = \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_i \\ \vdots \\ p_M \end{bmatrix}. \tag{4}$$

The vector p_i is a row vector of dimension $[1 * N]$ influenced by N APs. In our approach, p_i is defined as a sparse vector where only one element is 1 and all other elements are set to 0. This indicates that only one corresponding AP is chosen in p_i. Due to the variability in the number of APs that can be detected at different RPs, it is advisable to set the K value to be lower than the total number of available APs.

After computing the $dif_{r,j}$ for the r_{th} TP and all RPs using formula (3), WKNN selects the K RPs with the smallest difference in fingerprint similarity for position estimation:

$$LOC'(x_r, y_r) = \frac{\sum_{l=1}^{K} \omega_l LOC(x_l, y_l)}{\sum_{l=1}^{K} \omega_l} \tag{5}$$

In the formula, $LOC'(x_r, y_r)$ represents the estimated position after performing the positioning algorithm. The position of the l_{th} RP among the K candidate points is denoted by $LOC(x_l, y_l)$. The weight of the l_{th} position is represented by ω_{th}, which is determined by $1/dif_{r,j}$. A weighted strategy was suggested in [29], which yielded a slight improvement in accuracy compared to the KNN algorithm.

B. STATIC SITUATION OF ONLINE POSITIONING

In this section, we present a method to enhance the stability of WiFi location fingerprints in stationary states during the online localization phase. First, we distinguish between stationary and moving states in the positioning process and process them separately. Then, we apply specific techniques to handle the RSSI values in the stationary state.

In the second section, we analyzed the fluctuations in signal strength when the device is in a stationary state. To improve the stability of the RSSI in static conditions, we first determine the state of static based on the condition that the number of consecutive single-point acquisitions is greater than L. Then we analyze the statistical properties of the RSSI in the static state to improve positioning accuracy by addressing the issue of signal fluctuation and obtaining a stable fingerprint signal. To achieve this, we preprocess the RSSI using a moving window approach when the condition in equation (6) is met.

$$\frac{\sum_{n=k}^{k+L-1} Num(n)_i}{L} > \sigma \tag{6}$$

In this case, $Num(n)i$ denotes the validity of the ith AP signal when the n_{th} signal is collected. Any signal that is recorded as -100 dB during the signal collection process is considered invalid. We define σ be the threshold for signal validity, where signals exceeding the threshold are considered valid.

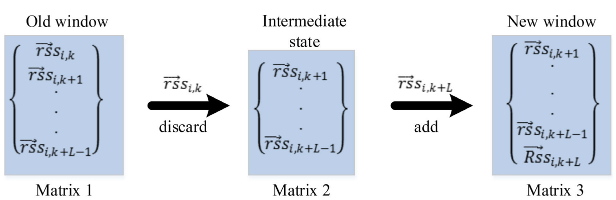

FIGURE 6. Signal preprocessing based on moving window in static state.

As illustrated in Fig. 6, the signal processing method based on moving window in static state is divided into two steps. In the first step, during the $k + L - 1_{th}$ signal acquisition in the online stage, the position fingerprint of the i_{th} AP collected between the k_{th} time and the $k + L - 1_{th}$ time is used as the old window. The earliest received test signal in this window $\vec{rss}_{i,k}$ is discarded. In the second step, during the $k + L_{th}$ signal acquisition in the online phase, we process the newly obtained $\vec{rss}_{i,k+L}$ signal through equation (7) to obtain $\vec{Rss}_{i,k+L}$. Then add it to the excessive state and get the new window. The $\vec{Rss}_{i,k+L}$ is the average of the sum of $\vec{rss}$ of the i_{th} AP from $\vec{rss}_{i,k}$ to $\vec{rss}_{i,k+L}$:

$$\vec{Rss}_{i,k+L} = \frac{\sum_{n=k+1}^{k+L} rss_{i,n}}{\sum_{n=k+1}^{k+L} Num(n)_i} \tag{7}$$

Next, we proceed with a further update of $\vec{Rss}_{i,k+L}$ using formula (8) to obtain a more refined and smoother signal. The formula is defined as follows:

$$\vec{RSS}_{i,k+L} = \frac{(\sum_{n=k+1}^{k+L} rss_{i,n}) + \vec{Rss}_{i,k+L}}{L} \tag{8}$$

The $\vec{RSS}_{i,k+L}$ is the RSSI information of i_{th} access point after the $k + L_{th}$ signal acquisition is refined using the moving window. This refined information is then used to replace the original signal $\vec{rss}_{i,k+L}$, signifying the completion of the signal processing task. To further solidify the stability of RSSI, the cumulative average method is employed based on the moving window signal processing approach. The steps involved in this method can be summarized as follows:

$$\vec{RSS}^{1}_{i,k+L} = \frac{(\sum_{s=1}^{n} \vec{rss}_{i,s})}{cnt} \tag{9}$$

The $\vec{rss}_{i,s}$ represents the location fingerprint of the i_{th} AP's RSSI that has undergone static processing and was collected at the s_{th} online stage. The variable cnt represents the number of signal acquisitions in the current static state. In this particular scenario, we focus on the AP corresponding to Fig. 3. To obtain the RSSI data, 100 static signal acquisitions were performed, and the parameter L was set to 10. The resulting RSSI data after undergoing SCSC processing is depicted in Fig. 7.

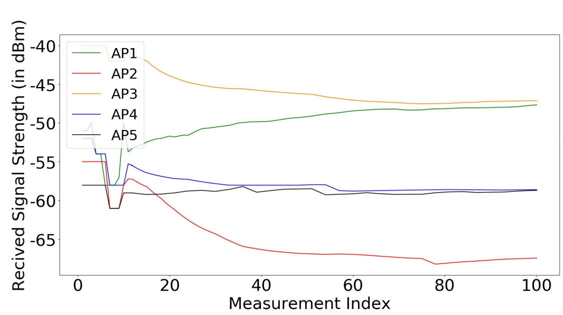

FIGURE 7. RSSI processed by SCSC in static state.

Upon comparing Fig. 3 and Fig. 7, it is evident that the stability of RSSI has been substantially enhanced through the implementation of SCSC processing. As the number of acquisitions increases, the stability of the location fingerprint will also improve. Thus, by strategically utilizing the static conditions in the positioning process, we can obtain more stability location fingerprints, leading to enhanced positioning accuracy.

C. K VALUE SETTING AND PHYSICAL DISTANCE DO NOT CORRESPOND TO FINGERPRINT SIMILARITY

In this subsection, we will briefly discuss the solutions to two problems: setting the K value and addressing the mismatch between physical distance and location fingerprint similarity. The setting of the K value is described in detail in SAWKNN [17]. Interested readers can refer to the original source for more information. Here is a brief overview of how the K value is dynamically obtained. Firstly, the classical path loss model for radio signals is used:

$$P(d)[dBm] = P(d_0)[dBm] - 10n\log[\frac{d}{d_0}] + \omega \tag{10}$$

The path loss coefficient is denoted by n, and the RSSI measured at distances d and d_0 from the AP are represented by $P(d)$ and $P(d_0)$, respectively. Here, d_0 serves as the reference distance and is based on 1 m. In addition, ω represents the ambient noise. Typically, the value of $P(d)$ in the same area is largely dependent on the distance d. The location fingerprint similarity difference $dif_{(r,j)}$ between the RP and the TP in formula (3) is likely to be consistent with $d_{r,j}$ to a large extent.

1) IF $dif_{A,O} < dif_{D,O}$ THEN $d_{AO} < d_{DO}$,
2) IF $dif_{A,O} \approx dif_{D,O}$ THEN $d_{AO} \approx d_{DO}$,
3) IF $dif_{A_1,O} \approx dif_{A_2,O} \approx dif_{A_3,O} \cdots < dif_{D,O}$ THEN $d_{A_1O} \approx d_{A_2O} \approx d_{A_3O} \cdots < d_{DO}$.

According to the three properties mentioned above, the RPs with the smallest $dif_{r,j}$ between the location fingerprint of the offline stage and the location fingerprint of the online stage is more likely to be close to the TP. Therefore, the positioning problem can be transformed into finding the RPs fingerprints that are most similar to the TP fingerprints. WKNN selects K RPs with the highest similarity to the TP fingerprints as candidate points. However, the number of RPs with high similarity location fingerprints varies under different TPs, especially in cases where K is 1 or 2, which are more easily ignored. Hence, the approach of dynamically acquiring the optimal K value can be implemented to enhance the accuracy of the positioning system. The issue of setting the K value can be transformed into determining the number of RPS fingerprints that exhibit high similarity to TP fingerprints. Suppose that $dif_{r,j1}$ represents the smallest dif, then the correlation between $dif_{r,j1}$ and the dissimilarity in fingerprint similarity values of other locations is established as follows:

$$\begin{cases} \gamma = dif_{r,other} / dif_{r,j_1} - 1 \\ dif_{r,j_1} < dif_{r,other}, \gamma > \gamma_{th} \\ dif_{r,j_1} \approx dif_{r,other}, \gamma \leqslant \gamma_{th} \end{cases} \tag{11}$$

The similarity between two pairs of dif is represented by γ. The threshold of fingerprint similarity, $\gamma_t h$, can be determined through experimentation. Using this threshold, we can filter out the number of RPs that meet the required conditions, thus establishing the value of K. The positioning can then be achieved using formula (5). In Algorithm 1, the maximum K value that can be obtained through experimentation is denoted by K_{max}.

Algorithm 1 The Pseudo-Code for Obtaining the K Value

```
for j = 2 : K_max + 1 do
    if dif_{r,other} / dif_{r,j1} − 1 > γ_th then
        break;
    end
end
K = j − 1;
return K;
```

Clearly, the K value is subject to dynamic changes based on the fingerprint similarity between the online and offline stages. Therefore, the K value can be adjusted adaptively and utilized flexibly within the WKNN algorithm. This leads to the automatic determination of the optimal K value.

Aiming to address the issue of discrepancy between physical distance and fingerprint similarity in the positioning process, we have referenced the soft range limiting concept of SRL-KNN [18]. Here, we provide a brief overview. The fundamental concept is that there is a limit to the velocity at which a person or object can move during localization. It is implausible for objects to suddenly emerge beyond a particular range in the ongoing online positioning process. A circle is drawn with the location of the previous moment as the center and $\vec{d}$ as the radius. The positioning space is restricted within this circle. $\vec{d}$ is determined by the individual's or object's movement velocity and positioning frequency during localization. With soft range restriction, the position near the previous moment is more likely to be chosen as the candidate position for localization results. However, unlike [18], where the Euclidean distance expression is transformed, we transform the expression for the Manhattan distance.

$$\vec{D}_r^j = \frac{W_r^j + D_r^j}{\sum_{i=1}^{s} W_r^j} \tag{12}$$

$$W_r^j = exp\frac{(x_r - x_{pre})^2 + (y_r - y_{pre})^2}{4\vec{d}^2} \tag{13}$$

The function W_r^j represents the penalty imposed on position r. The variable s denotes the number of RPs in the offline database, and (x_{pre}, y_{pre}) represents the user's previous position. The maximum distance the user can move between two consecutive positioning events is defined as $\vec{d}$. Typically, the walking speed of a person is within 2 m/s, and the maximum movement speed can be set according to the actual situation. It is unnecessary to know the specific movement speed and direction of the user. With this approach, the problem of mismatch between physical distance and fingerprint similarity can be effectively addressed.

D. PROPOSED SCSC-SRL-SAWKNN METHOD

In this subsection, we shall consolidate the information from sections IV-A, IV-B, and IV-C to provide a comprehensive overview of the SCSC-SRL-SAWKNN algorithm.

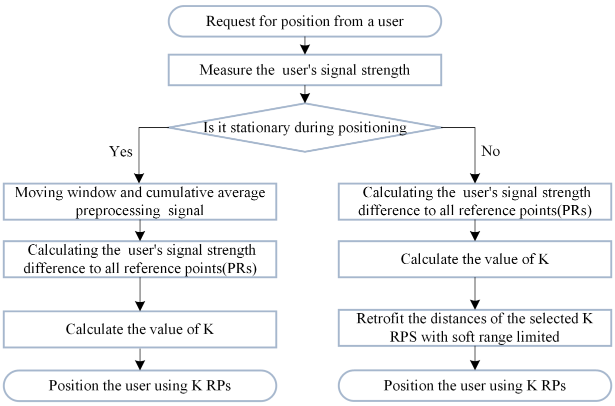

FIGURE 8. Outline of the SCSC-SRL-SAWKNN method.

Fig. 8 illustrates the flow of the SCSC-SRL-SAWKNN algorithm. The localization problem is to find offline fingerprints that closely match the fingerprints collected during the online phase. When a user requests their location, the mobile terminal collects location fingerprints and processes different motion states separately. In the case of a static state, a moving window signal processing method is applied to improve the stability of the location fingerprint in the online phase, followed by the method of mean accumulation to obtain robust online fingerprints. The calculation of the location fingerprint similarity difference between the online and offline stages is then completed, and an adaptive algorithm is used to calculate the K value and estimate the position. If the user is in a moving state, the signal processing steps are omitted, and the fingerprint similarity difference is directly computed to obtain the K value through Algorithm 1. The Manhattan distance in the fingerprint similarity expression is then transformed using the penalty function of the range factor, and the final modified K RPs is used for position estimation.

V. EXPERIMENTS

In this section, we will assess the performance of the proposed SCSC-SRL-SAWKNN algorithm through actual experiments. Firstly, we will analyze the influence of several crucial parameters, including L, σ, γ_{rh} and K_{max}, under various scenarios. Secondly, we will compare the performance of the WKNN algorithm before and after applying the moving window signal processing method to stabilize the position fingerprints during the online positioning process in static states. Then, we will compare the localization accuracies of WKNN, SAWKNN, SRL-KNN, and SCSC-SRL-SAWKNN algorithms in different states. Finally, we will delve into the application development of location tracking. For the traditional WKNN and SRL-KNN algorithms, we have set K to an empirical value of 3. The computer utilized to provide location services was equipped with an Intel(R) Core(TM) i5-4200H CPU and 12 GB of RAM.

A. THE EFFECT OF MOVING WINDOW SIGNAL PROCESSING

Firstly, we will analyze the impact of key parameters under static conditions on the performance of the moving window processing method. As discussed in section IV, the crucial parameters of this method are denoted by L and σ. Our aim is to identify the optimal parameter values that would result in the highest positioning accuracy. Subsequently, we will evaluate the impact of the moving window signal processing method on the localization accuracy in the real environment using these values.

Fig. 2 depicts the real experimental environment that has been constructed, wherein the blue pentagrams denote RPs with a total of 155 RPs. The trajectory and direction of motion in the online phase are represented by the green lines, with a movement speed of approximately 0.5 m/s. The signal acquisition frequency has been set to 1 Hz. At each static continuous collection point represented by the red pentagram, we collected 100 fingerprint signals continuously, with an acquisition frequency of 1 Hz. The entire experiment was conducted during a bustling time period with many teachers and students present in the experimental environment. Additionally, the state of indoor doors and windows opening and closing was continuously changing. In order to maintain the authenticity of the experiment, all algorithms have utilized the real-time collected RSSI as the original fingerprint.

1) SETTING OF L

Fig. 9 depicts the error pattern of static continuous statistical characteristics weight K-nearest neighbor (SCSC-WKNN) algorithm in a static state when employing different values of parameter L. It is discernible that the error curve of the positioning accuracy fluctuates considerably when L is within the range of [2, 30]. Nonetheless, it is evident that the best positioning accuracy is attained when L is equal to 4. As such, the value of L is set to 4.

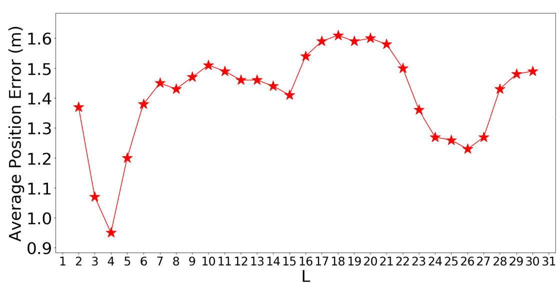

FIGURE 9. Position results of SCSC-WKNN under different L.

2) SETTING OF σ

Fig. 10 illustrates the average localization error of the SCSC-WKNN algorithm at varying values of σ. It can be observed that the positioning errors are relatively small and closely clustered when σ falls within the range of [0.1, 0.7]. To maximize the preservation of fingerprint data, the value of σ is set to 0.2.

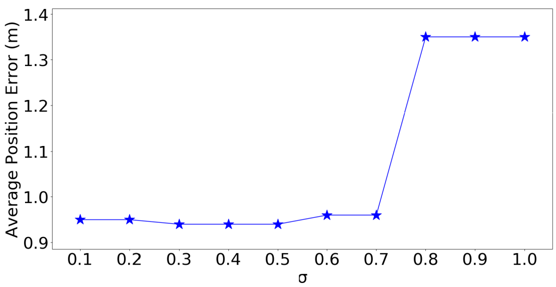

FIGURE 10. **Position results of SCSC-WKNN under different σ.**

3) STATIC LOCATION TEST

The positioning accuracy of the WKNN algorithm before and after the moving window processing in the static state is depicted in Fig. 11. The figure illustrates the green dots which represent the average error and the red line indicating the median error. Under static conditions, the traditional WKNN and SCSC-WKNN algorithms have average positioning errors of 2.14 and 0.95 meters, respectively. This suggests that the localization accuracy of WKNN is significantly improved after the moving window signal processing, demonstrating an excellent positioning effect. Furthermore, the localization accuracy of SCSC-WKNN is enhanced by 55.6% compared to traditional WKNN. Simultaneously, the overall boxplot of SCSC-WKNN is considerably lower than that of traditional WKNN, indicating the superiority of the WKNN algorithm after moving window signal processing in achieving higher positioning accuracy. Additionally, the SCSC-WKNN box has a lower upper quantile, implying that there are fewer large error cases compared to WKNN.

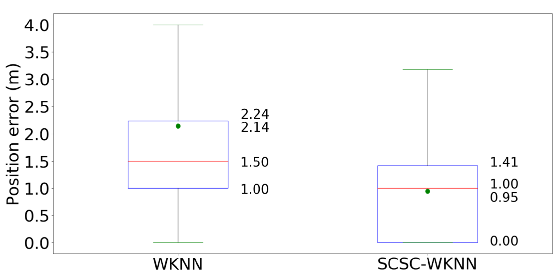

FIGURE 11. **Positioning accuracy of WKNN before and after moving window processing.**

B. LOCALIZATION EFFECT OF SCSC-SRL-SAWKNN

In this section, we shall examine the impact of critical parameters of SCSC-SRL-SAWKNN on localization accuracy. As highlighted in Section IV, the algorithm is characterized by four essential parameters. The values of L and σ yielding the utmost localization accuracy were determined in section V-A. Our objective is to determine the most favorable values of γ_{rh} and K_{max}.

1) SETTING OF γ_{rh}

The localization errors of SCSC-SRL-SAWKNN for various γ_{rh} values are presented in Fig. 12. It is evident that the most accurate positioning results can be achieved when the γ_{rh} value falls between the range of 0.4 and 1. This is because the formula (11) is satisfied by a greater number of location fingerprints as the value of γ_{rh} increases, leading to increased complexity of the algorithm. Therefore, a value of 0.4 is ideal for achieving the highest computational efficiency while maintaining the best positioning accuracy.

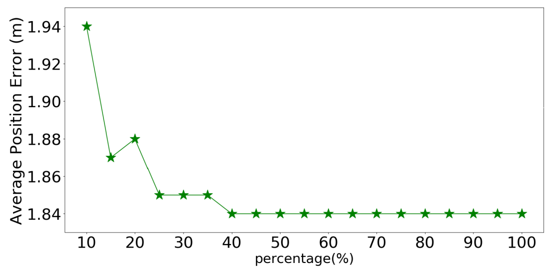

FIGURE 12. **Position results of SCSC-SRL-SAWKNN under different γ_{rh}.**

2) SETTING OF K_{max}

The mean localization error of SCSC-SRL-SAWKNN at different K_{max} values is illustrated in Fig. 13. It is observed that the positioning accuracy diminishes as K_{max} increases. The optimal positioning accuracy is achieved when K_{max} is set to 3. Additionally, the smaller the value of K_{max}, the fewer cycles of the algorithm will be executed, thereby enhancing its computational efficiency.

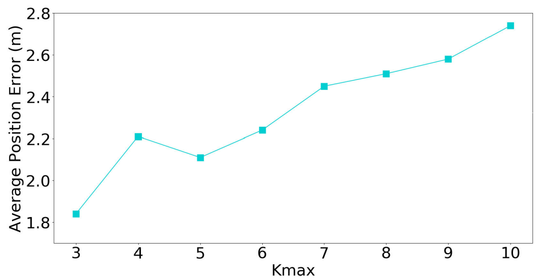

FIGURE 13. **Position results of SCSC-SRL-SAWKNN under different K_{max}.**

3) STATIC LOCATION TEST

The positioning error distributions of WKNN, SAWKNN, SRL-KNN algorithms and our proposed SCSC-SRL-SAWKNN algorithm in the static state are illustrated in Fig. 14. The traditional WKNN and the improved SAWKNN and SRL-KNN algorithms exhibit average localization errors of 2.14, 2.06, and 1.40 meters, respectively, under stationary conditions. However, the mean positioning error of the SCSC-SRL-SAWKNN algorithm under static conditions is

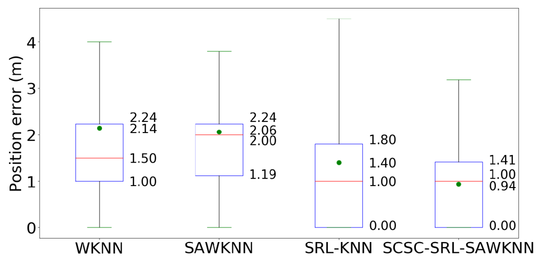

FIGURE 14. Comparing with traditional WKNN, SAWKNN, SRL-KNN. The figure shows the distribution of positioning errors of WKNN, SAWKNN, SRL-KNN and SCSC-SRL-SAWKNN in static state.

a mere 0.94 meters, indicative of its significantly improved localization accuracy in stationary conditions. In fact, its positioning accuracy is outstanding. Compared with the traditional WKNN algorithm, SCSC-SRL-SAWKNN has improved the localization accuracy by 56.7%. Additionally, SCSC-SRL-SAWKNN showed the lowest box plot compared to the other three algorithms, indicating an overall better positioning accuracy. Moreover, the upper quantile of this box plot is lower than that of the other three algorithms, indicating that SCSC-SRL-SAWKNN has the least instances of large positioning errors.

4) MOVING LOCATION TEST

Subsequently, we will compare the positioning performance of WKNN, SAWKNN, SRL-KNN, and SCSC-SRL-SAWKNN under mobile conditions.

The illustration presented in Fig. 15 depicts the performance of four distinct algorithms, namely WKNN, SAWKNN, SRL-KNN, and SCSC-SRL-SAWKNN, in terms of positioning accuracy under mobile conditions. The average positioning errors of these algorithms are 4.65, 4.29, 4.32 and 3.94 meters, respectively. It is observed that SCSC-SRL-SAWKNN outperforms other algorithms in terms of positioning accuracy in the mobile state. The boxplots of SCSC-SRL-SAWKNN exhibit smaller upper and lower quantiles in comparison to the other algorithms. This signifies that the proposed algorithm outperforms the others in terms of high-precision localization during motion, while also exhibiting the least number of large-error localizations. However, a comparison of Fig. 15 reveals that the positioning accuracy in motion state is significantly lower than that in stationary state as a whole. Undoubtedly, the diminished precision in the localization of the mobile state compared to that of the stationary state can be attributed to the fact that a sole signal collection is conducted for every location in the mobile state. This implies that the number of signal acquisition frames is extremely limited, resulting in the fingerprint signal's instability.

5) OVERALL LOCATION TEST

The static and kinematic scenarios will be integrated next, and the overall performance of the SCSC-SRL-SAWKNN algorithm will be assessed comprehensively.

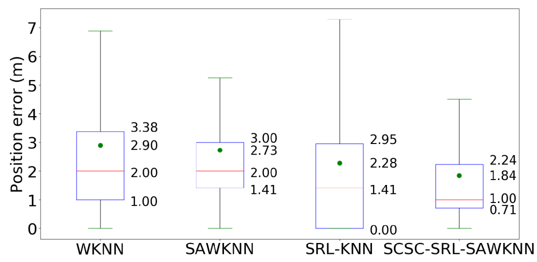

FIGURE 16. Comparing with traditional WKNN, SAWKNN, SRL-KNN. The figure shows the distribution of positioning errors of WKNN, SAWKNN, SRL-KNN and SCSC-SRL-SAWKNN in Overall state.

The positioning error results of the comprehensive positioning experiment of WKNN, SAWKNN, SRL-KNN, and SCSC-SRL-SAWKNN are depicted in Fig. 16. The mean positioning errors of these algorithms are 2.90, 2.73, 2.28, and 1.84 meters, respectively. Notably, SCSC-SRL-SAWKNN outperforms the other algorithms in terms of positioning accuracy. Compared with the traditional WKNN algorithm, our proposed algorithm demonstrates a 36.6% improvement in accuracy. The lower section of the box plot of SCSC-SRL-SAWKNN is significantly lower than that of traditional WKNN, indicating that it performs much better in terms of high-precision positioning. Additionally, the quantile on the box plot is much lower than that of other algorithms, indicating that the occurrence of large errors in SCSC-SRL-SAWKNN is much less frequent than in other algorithms. Overall, the algorithm exhibits excellent positioning accuracy.

Fig. 17 presents a comparison of the cumulative distribution function (CDF) of the overall localization error between

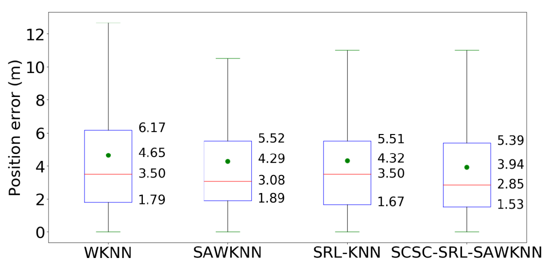

FIGURE 15. Comparing with traditional WKNN, SAWKNN, SRL-KNN. The figure shows the distribution of positioning errors of WKNN, SAWKNN, SRL-KNN and SCSC-SRL-SAWKNN in Moving state.

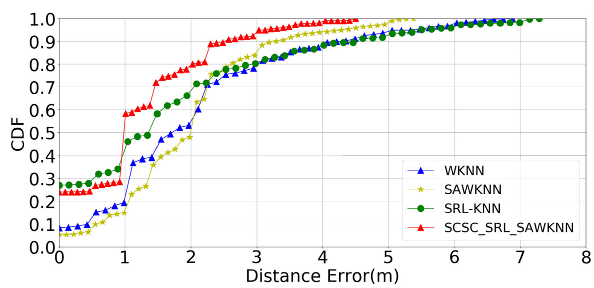

FIGURE 17. CDF of overall positioning error for WKNN, SAWKNN, SRL-KNN, and SCSC-SRL-SAWKNN.

SCSC-SRL-SAWKNN and the other three KNN methods. It is evident that SCSC-SRL-SAWKNN (depicted by the red line) outperforms the other methods overall in terms of localization accuracy. Furthermore, in terms of high-precision performance within 1 meter, SRL-KNN and SCSC-SRL-SAWKNN outperform the other two algorithms. Remarkably, the maximum positioning error of SCSC-SRL-SAWKNN is only 4.54 meters, whereas the maximum positioning errors of WKNN, SAWKNN, and SRL-KNN are 6.95 meters, 5.39 meters and 7.38 meters, respectively. Thus, it can be deduced that SCSC-SRL-SAWKNN excels in both high precision and stability.

C. LOCATION TRACKING

Upon observing Section V-B, it is evident that the positioning error in the mobile state is significantly higher than that in the stationary state. In order to enhance the positioning accuracy in the mobile state, a high-precision tracking function is implemented by combining the Kalman filter [38] with the localization algorithm. The positioning trajectories of each localization algorithm after Kalman filtering are displayed in Fig. 18.

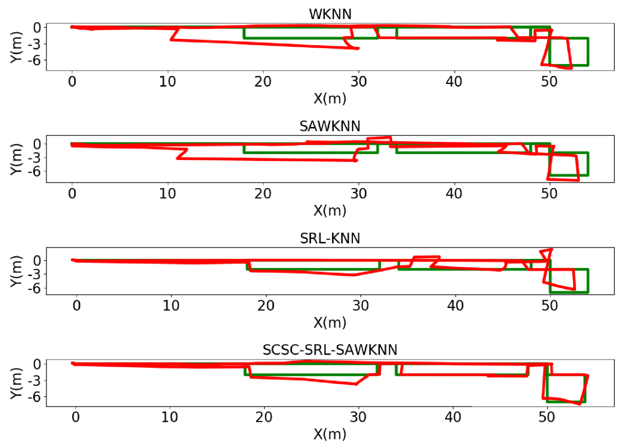

FIGURE 18. Ground truth and estimated trajectories. Green line represents the trajectory ground truth. Red lines are estimated trajectories.

The green line in Fig. 18 represents the actual trajectory, while the red line shows the positioning tracking effect. SCSC-SRL-SAWKNN algorithm exhibits the most precise location tracking effect, with its positioning trajectory fitting the real path better than other algorithms. The positioning errors of the four algorithms combined with Kalman filtering are ultimately presented in the ensuing figure.

After applying Kalman filtering, the mean errors of WKNN, SAWKNN, SRL-KNN, and SCSC-SRL-SAWKNN decrease to 2.44, 2.40, 1.82, and 1.27 meters, respectively. Notably, the SCSC-SRL-SAWKNN algorithm demonstrates significantly smaller errors than the other methods. Additionally, the lower and upper bounds of the box plot for this algorithm were considerably smaller than those of the other algorithms, indicating its superiority in all directions.

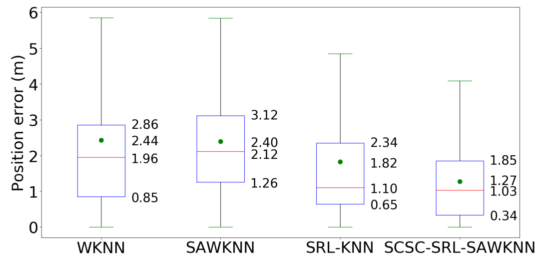

FIGURE 19. Comparing with traditional WKNN, SAWKNN, SRL-KNN. This figure shows the error distribution of WKNN, SAWKNN, SRL-KNN and SCSC-SRL-SAWKNN after Kalman filtering.

It is evident that SCSC-SRL-SAWKNN outperforms the other three algorithms in terms of positioning accuracy, whether in static or dynamic states, overall performance, or combined with Kalman filtering. Particularly, in static and overall conditions, its localization accuracy is significantly better than that of WKNN, SAWKNN, and SRL-KNN. The average time of SCSC-SRL-SAWKNN localization is only 6.8ms. This algorithm not only enhances positioning accuracy, but also operates with minimal resource consumption, thereby offering an advantageous high real-time performance in practical applications.

VI. CONCLUSION

The current WiFi-based indoor positioning methods typically focus only on a single situation, either in stationary or moving states, neglecting the fact that actual positioning involves a combination of both static and dynamic situations. This paper proposes the SCSC-SRL-SAWKNN algorithm, which intelligently leverages the characteristics of both states. By processing signals based on the features of different positioning states, our proposed algorithm updates the fingerprint via the mean accumulation method after processing the signal with a moving window under static conditions, which significantly improves the stability of RSSI. Then combined with the SAWKNN algorithm, the positioning accuracy in static state can be greatly improved. To address the mismatch between physical distance and fingerprint similarity in the moving state, the algorithm transforms the Manhattan distance by incorporating the soft range limitation concept, and then combines it with SAWKNN to improve positioning accuracy. Particularly for overall positioning cases that involve both static and dynamic states, the state-by-state processing approach of SCSC-SRL-SAWKNN algorithm effectively utilizes the characteristics of each state, leading to a remarkable improvement in positioning accuracy. Our future research aims to develop a more robust fingerprint combination and a signal preprocessing method for variable-step moving windows to further enhance the positioning accuracy and resolve the issue of equipment heterogeneity.

REFERENCES

[1] A. Poulose and D. S. Han, "UWB indoor localization using deep learning LSTM networks," *Appl. Sci.*, vol. 10, no. 18, p. 6290, Sep. 2020.

[2] L. Bai, F. Ciravegna, R. Bond, and M. Mulvenna, "A low cost indoor positioning system using Bluetooth low energy," *IEEE Access*, vol. 8, pp. 136858–136871, 2020.

[3] A. Poulose, J. Kim, and D. S. Han, "A sensor fusion framework for indoor localization using smartphone sensors and Wi-Fi RSSI measurements," *Appl. Sci.*, vol. 9, no. 20, p. 4379, Oct. 2019.

[4] X. Hou and J. Bergmann, "Pedestrian dead reckoning with wearable sensors: A systematic review," *IEEE Sensors J.*, vol. 21, no. 1, pp. 143–152, Jan. 2021.

[5] Y. Yu, R. Chen, L. Chen, S. Xu, W. Li, Y. Wu, and H. Zhou, "Precise 3-D indoor localization based on Wi-Fi FTM and built-in sensors," *IEEE Internet Things J.*, vol. 7, no. 12, pp. 11753–11765, Dec. 2020.

[6] J. Xue, J. Liu, M. Sheng, Y. Shi, and J. Li, "A WiFi fingerprint based high-adaptability indoor localization via machine learning," *China Commun.*, vol. 17, no. 7, pp. 247–259, Jul. 2020.

[7] B. Huang, R. Yang, B. Jia, W. Li, and G. Mao, "A theoretical analysis on sampling size in WiFi fingerprint-based localization," *IEEE Trans. Veh. Technol.*, vol. 70, no. 4, pp. 3599–3608, Apr. 2021.

[8] W. Xue, K. Yu, X. Hua, Q. Li, W. Qiu, and B. Zhou, "APs' virtual positions-based reference point clustering and physical distance-based weighting for indoor Wi-Fi positioning," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 3031–3042, Aug. 2018.

[9] B. Wang, X. Gan, X. Liu, B. Yu, R. Jia, L. Huang, and H. Jia, "A novel weighted KNN algorithm based on RSS similarity and position distance for Wi-Fi fingerprint positioning," *IEEE Access*, vol. 8, pp. 30591–30602, 2020.

[10] J. Niu, B. Wang, L. Shu, T. Q. Duong, and Y. Chen, "ZIL: An energy-efficient indoor localization system using ZigBee radio to detect WiFi fingerprints," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 7, pp. 1431–1442, Jul. 2015.

[11] S. Li, M. Hedley, K. Bengston, D. Humphrey, M. Johnson, and W. Ni, "Passive localization of standard WiFi devices," *IEEE Syst. J.*, vol. 13, no. 4, pp. 3929–3932, Dec. 2019.

[12] Y. Zheng, M. Sheng, J. Liu, and J. Li, "Exploiting AoA estimation accuracy for indoor localization: A weighted AoA-based approach," *IEEE Wireless Commun. Lett.*, vol. 8, no. 1, pp. 65–68, Feb. 2019.

[13] M. R. Nowicki and P. Skrzypczynski, "Leveraging visual place recognition to improve indoor positioning with limited availability of WiFi scans," *Sensors*, vol. 19, no. 17, p. 3657, Aug. 2019.

[14] M. T. Hoang, B. Yuen, X. Dong, T. Lu, R. Westendorp, and K. R. Tarimala, "Semi-sequential probabilistic model for indoor localization enhancement," *IEEE Sensors J.*, vol. 20, no. 11, pp. 6160–6169, Jun. 2020.

[15] Z. Chen, H. Zou, J. Yang, H. Jiang, and L. Xie, "WiFi fingerprinting indoor localization using local feature-based deep LSTM," *IEEE Syst. J.*, vol. 14, no. 2, pp. 3001–3010, Jun. 2020.

[16] A. Poulose and D. S. Han, "Performance analysis of fingerprint matching algorithms for indoor localization," in *Proc. Int. Conf. Artif. Intell. Inf. Commun. (ICAIIC)*, Feb. 2020, pp. 661–665.

[17] 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.

[18] M. T. Hoang, Y. Zhu, B. Yuen, T. Reese, X. Dong, T. Lu, R. Westendorp, and M. Xie, "A soft range limited K-nearest neighbors algorithm for indoor localization enhancement," *IEEE Sensors J.*, vol. 18, no. 24, pp. 10208–10216, Dec. 2018.

[19] T. Raharijaona, R. Mawonou, T. Nguyen, F. Colonnier, M. Boyron, J. Diperi, and S. Viollet, "Local positioning system using flickering infrared LEDs," *Sensors*, vol. 17, no. 11, p. 2518, Nov. 2017.

[20] M. O. Khyam, M. N.-A. Rahim, X. Li, A. Jayasuriya, M. A. Mahmud, A. M. T. Oo, and S. S. Ge, "Simultaneous excitation systems for ultrasonic indoor positioning," *IEEE Sensors J.*, vol. 20, no. 22, pp. 13716–13725, Nov. 2020.

[21] X. Zhu, J. Yi, J. Cheng, and L. He, "Adapted error map based mobile robot UWB indoor positioning," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 9, pp. 6336–6350, Sep. 2020.

[22] A. A. N. Shirehjini and S. Shirmohammadi, "Improving accuracy and robustness in HF-RFID-based indoor positioning with Kalman filtering and Tukey smoothing," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 11, pp. 9190–9202, Nov. 2020.

[23] W. Zhang, K. Yu, W. Wang, and X. Li, "A self-adaptive AP selection algorithm based on multiobjective optimization for indoor WiFi positioning," *IEEE Internet Things J.*, vol. 8, no. 3, pp. 1406–1416, Feb. 2021.

[24] M. Sun, Y. Wang, S. Xu, H. Yang, and K. Zhang, "Indoor geomagnetic positioning using the enhanced genetic algorithm-based extreme learning machine," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–11, 2021.

[25] J. Koo and H. Cha, "Localizing WiFi access points using signal strength," *IEEE Commun. Lett.*, vol. 15, no. 2, pp. 187–189, Feb. 2011.

[26] C. Feng, S. Arshad, S. Zhou, D. Cao, and Y. Liu, "Wi-multi: A three-phase system for multiple human activity recognition with commercial WiFi devices," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 7293–7304, Aug. 2019.

[27] P. Bahl and V. N. Padmanabhan, "Radar: An in-building RF-based user location and tracking system," in *Proc. IEEE INFOCOM Conf. Comput. Commun., 19th Annu. Joint Conf. IEEE Comput. Commun. Societies*, vol. 2, Mar. 2000, pp. 775–784.

[28] S. Zhang, X. Li, M. Zong, X. Zhu, and R. Wang, "Efficient kNN classification with different numbers of nearest neighbors," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 5, pp. 1774–1785, May 2018.

[29] M. Brunato and R. Battiti, "Statistical learning theory for location fingerprinting in wireless LANs," *Comput. Netw.*, vol. 47, no. 6, pp. 825–845, Apr. 2005.

[30] M. Sun, Y. Wang, S. Xu, H. Qi, and X. Hu, "Indoor positioning tightly coupled Wi-Fi FTM ranging and PDR based on the extended Kalman filter for smartphones," *IEEE Access*, vol. 8, pp. 49671–49684, 2020.

[31] H. Zou, M. Jin, H. Jiang, L. Xie, and C. J. Spanos, "WinIPS: WiFi-based non-intrusive indoor positioning system with online radio map construction and adaptation," *IEEE Trans. Wireless Commun.*, vol. 16, no. 12, pp. 8118–8130, Dec. 2017.

[32] T. Koike-Akino, P. Wang, M. Pajovic, H. Sun, and P. V. Orlik, "Fingerprinting-based indoor localization with commercial mmWave WiFi: A deep learning approach," *IEEE Access* vol. 8, pp. 84879–84892, 2020.

[33] X. Song, X. Fan, C. Xiang, Q. Ye, L. Liu, Z. Wang, X. He, N. Yang, and G. Fang, "A novel convolutional neural network based indoor localization framework with WiFi fingerprinting," *IEEE Access*, vol. 7, pp. 110698–110709, 2019.

[34] C. Liu, C. Wang, and J. Luo, "Large-scale deep learning framework on FPGA for fingerprint-based indoor localization," *IEEE Access*, vol. 8, pp. 65609–65617, 2020.

[35] X. Guo, L. Li, F. Xu, and N. Ansari, "Expectation maximization indoor localization utilizing supporting set for Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2573–2582, Apr. 2019.

[36] Y. Li, S. Williams, B. Moran, and A. Kealy, "A probabilistic indoor localization system for heterogeneous devices," *IEEE Sensors J.*, vol. 19, no. 16, pp. 6822–6832, Aug. 2019.

[37] M. Zhou, X. Li, Y. Wang, S. Li, Y. Ding, and W. Nie, "6G multisource-information-fusion-based indoor positioning via Gaussian kernel density estimation," *IEEE Internet Things J.*, vol. 8, no. 20, pp. 15117–15125, Oct. 2021.

[38] Z. Chen, H. Zou, H. Jiang, Q. Zhu, Y. C. Soh, and L. Xie, "Fusion of WiFi, smartphone sensors and landmarks using the Kalman filter for indoor localization," *Sensors*, vol. 15, no. 1, pp. 715–732, Jan. 2015.

JIUSONG HU received the M.Sc. and Ph.D. degrees from Hunan University, Changsha, China, in 2014 and 2019, respectively. He is currently a Lecturer with the Department of Information Engineering, Hunan University of Technology, Zhuzhou, China. His research interests include intelligent perception, information processing, and indoor positioning.

CONGWEI HU received the B.S. degree in electronic information engineering from the Hunan University of Technology, in 2020, where he is currently pursuing the master's degree in electronic information. His current research interests include WSN, intelligent information processing, fault detection, and indoor location.