



ARTICLE

An Adaptive Trajectory-Assisted Dynamic Indoor Positioning Method Based on RSS Fingerprinting

Jing Liu^{1,2} and Weijie Tan^{1,2,3,*}

¹College of Computer Science and Technology, Guizhou Provincial Laboratory of Big Data, Guizhou University, Guiyang, China

²Key Laboratory of Interior Layout Optimization and Security, Institutions of Higher Education of Sichuan Province, Chengdu Normal University, Chengdu, China

³School of Software, Tsinghua University, Beijing, China

*Corresponding Author: Weijie Tan. Email: wjtan@gzu.edu.cn

Received: 03 April 2026; Accepted: 27 May 2026; Published: 23 July 2026

ABSTRACT: Due to its low hardware cost and ease of deployment, WiFi fingerprinting has become a prominent research direction in indoor positioning. However, traditional methods based on Received Signal Strength (RSS) still face three critical challenges: susceptibility to noise interference, low retrieval efficiency as fingerprint databases scale up, and trajectory instability in dynamic environments. These challenges are inherently rooted in the stochastic fluctuation of RSS signals, the high-dimensional and non-Euclidean nature of fingerprint space, and the unpredictability of user movement patterns. To address these limitations, an adaptive trajectory-assisted dynamic indoor positioning algorithm based on RSS fingerprinting, termed AT-WKNN (Adaptive Trajectory-assisted Weighted K-Nearest Neighbor), is proposed. Specifically, a hybrid distance metric incorporating adaptive distance constraints is first designed to identify high-quality neighboring fingerprints while filtering out noisy samples, thereby improving positioning accuracy. Subsequently, a Hierarchical Navigable Small-World (HNSW) structure is employed to enable efficient fingerprint retrieval. In addition, a Kalman filter is utilized to smooth trajectory estimation and suppress dynamic noise. Experimental results demonstrate that the proposed AT-WKNN algorithm achieves a 43.0% improvement in positioning accuracy and a 6.90× increase in retrieval efficiency compared with the baseline WKNN method. Furthermore, validation on the large-scale UJIIndoorLoc benchmark dataset confirms the scalability and generalization capability of the proposed method.

KEYWORDS: Indoor localization; WiFi fingerprinting; adaptive localization algorithm; trajectory tracking

1 Introduction

In recent years, the rapid development of the mobile Internet has generated a strong demand for high-precision location-based services. Mainstream outdoor positioning systems, such as GPS, BeiDou, and Galileo, typically rely on trilateration using signals from multiple satellites [1]. However, in complex indoor environments, satellite signals are often obstructed or severely attenuated by buildings, resulting in significant ranging errors. Consequently, these systems struggle to provide reliable positioning results indoors, highlighting the urgent need for alternative indoor localization technologies [2].

Indoor localization aims to determine the real-time position and movement trajectory of individuals or objects in environments where satellite signals are weak or unavailable, by leveraging wireless communication and sensor fusion techniques. Existing indoor localization technologies include infrared [3], ultrasonic [4], Machine Learning [5], Bluetooth [6], ultra-wideband (UWB), and WiFi fingerprinting [7].

Although each technique has its own advantages and limitations [8], WiFi fingerprinting has attracted significant attention due to its reuse of existing network infrastructure, zero additional hardware cost, wide coverage, and strong potential for adaptive optimization across diverse scenarios.

WiFi fingerprinting is constructed by collecting Received Signal Strength (RSS) values from multiple Access Points (APs) at specific locations. Since each location exhibits a unique combination of RSS values, these measurements can be used to form distinctive fingerprints for localization. However, in practical applications, RSS measurements are affected by factors such as hardware heterogeneity, environmental dynamics, and human activity, resulting in random fluctuations. These disturbances degrade both the discriminability between locations and the stability of measurements, ultimately reducing localization accuracy. Although existing studies have made significant progress in improving distance metrics, neighbor selection strategies, and trajectory smoothing techniques, most approaches focus on optimizing individual modules independently, lacking a unified framework that jointly considers accuracy, efficiency, and trajectory stability. Recent studies have further confirmed the potential of clustering-based signal processing methods for improving robustness and feature representation in dynamic wireless environments [9,10].

To address the aforementioned challenges, this paper proposes an adaptive indoor localization framework for smart city Internet of Things (IoT) systems in smart cities (AT-WKNN). Unlike conventional approaches that treat each module separately, the proposed framework emphasizes system-level coordinated optimization. The framework jointly optimizes three key aspects: neighbor selection, similarity modeling, and trajectory refinement, thereby improving system efficiency and stability while maintaining high localization accuracy.

The proposed method fully leverages existing WiFi infrastructure and reduces computational complexity, making it well-suited for large-scale and energy-efficient IoT deployment scenarios. By integrating hybrid distance metrics, adaptive range constraints, efficient HNSW-based retrieval, and trajectory-aware Kalman filtering into a unified pipeline, the proposed method achieves a balanced improvement in localization accuracy, computational efficiency, and trajectory stability. The main contributions are summarized as follows:

1. An adaptive trajectory-assisted dynamic indoor localization algorithm, named AT-WKNN (Adaptive Trajectory-assisted WKNN), is proposed. The algorithm leverages RSS fingerprinting to mitigate RSS fluctuations, enhance neighbor retrieval quality, and ensure stable and accurate indoor localization.
2. AT-WKNN adopts a hybrid distance metric that integrates Manhattan distance and cosine similarity, combined with the Hierarchical Navigable Small-World (HNSW) algorithm for efficient approximate nearest neighbor search. In addition, an adaptive range constraint mechanism is introduced to further refine candidate selection, improving both robustness and retrieval efficiency. This approach reduces the sensitivity of a single metric to noise and alleviates the influence of outliers, thereby improving both localization accuracy and efficiency.
3. A Kalman filter is tightly integrated into the localization pipeline to smooth the trajectory and dynamically adjust the weights between predicted and observed data, effectively suppressing trajectory jitter caused by RSS fluctuations and significantly improving the reliability of dynamic localization.
4. Experiments conducted on real-world RSS datasets demonstrate that, compared with the baseline WKNN method, AT-WKNN improves localization accuracy by 43.0% and enhances efficiency by 6.90 $\times$, while significantly improving trajectory smoothness. Furthermore, validation on the large-scale UJIIndoorLoc benchmark dataset confirms the scalability and generalization capability of the proposed method in complex large-scale environments.

2 Related Work

RSS fingerprint-based localization was first introduced by Paramvir Bahl and Venkata N. Padmanabhan through the RADAR system, which achieved meter-level accuracy in office environments and laid the foundation for subsequent research on WiFi fingerprinting [11]. This pioneering work achieved meter-level accuracy in office environments and laid the foundation for subsequent research on WiFi fingerprinting.

A variety of RSS fingerprinting-based localization methods have since been developed, including KNN [12], K-means [13], SVM [14], Random Forest (RF) [15], and neural networks [16]. Among these, the KNN algorithm is widely adopted due to its simplicity, training-free nature, and relatively high accuracy. Its extension, the weighted KNN (WKNN), introduces a weighting mechanism to further improve localization performance. However, traditional WKNN relies on a single Euclidean distance metric to evaluate fingerprint similarity, which may fail to fully capture the spatial and signal characteristics of fingerprints [17]. Recent studies have further improved WKNN-based indoor positioning by incorporating adaptive weighting strategies, enhancing localization accuracy and robustness in complex environments [18].

To enhance similarity measurement, numerous studies have proposed alternative distance metrics and weighting schemes, such as feature scaling [19], frequency-weighted RSS [20], improved Manhattan distance [21], AP variance-based weighting [22], combined Euclidean and physical distance [23], and signal strength-based AP weighting [24]. Although these methods improve distance computation, most still rely on a single metric and may not fully reflect fingerprint similarity.

Beyond distance refinement, range constraint techniques have been proposed to exploit the limited movement between consecutive user positions. These techniques can be categorized into hard range constraints and soft range constraints. Hard range constraints filter out neighbors located outside a predefined region based on previous estimates and user mobility, whereas soft range constraints assign weights according to the distance from the previous position [25,26]. However, both approaches neglect localization errors in previous estimates, which may lead to suboptimal neighbor selection.

Other improvements focus on weight computation and search strategies, including historical RSS trend fusion [27], gradient descent-based exhaustive search [28], and motion state-adaptive window sizing [29]. However, these methods either incur high computational costs or introduce latency due to motion state detection.

Finally, trajectory smoothing methods have been widely applied in indoor localization to mitigate observation noise. Techniques such as particle filtering [30,31], Bayesian estimators [32], Kalman filtering [33], and Long Short-Term Memory (LSTM)-based filtering [34] have been explored. While these approaches can suppress noise and improve localization accuracy, they often require complex parameter tuning or suffer from delayed adaptation to user movement. Recent studies have further explored deep learning-based indoor localization, including Transformer-based models, which improve robustness and adaptability in complex environments [35].

Overall, although existing methods have achieved significant progress in distance metrics through weighted distance modeling [24], cosine similarity-based matching [17], and feature-scaling-based optimization [19], as well as improvements in spatial constraint strategies and trajectory smoothing techniques, they still face limitations in handling dynamic RSS fluctuations and efficiently generating stable trajectories. As indoor localization evolves toward smart city IoT systems, methods are required not only to ensure accuracy but also to consider computational efficiency and energy consumption. Therefore, designing lightweight and efficient localization approaches has become an important research direction.

3 Preliminaries

3.1 RSS Fingerprinting

RSS is a key parameter in wireless communication systems that can be obtained in a simple and direct manner. Due to its independence from signal bandwidth and wide availability across existing infrastructure, RSS has become an attractive feature for indoor localization applications.

Assuming a fixed signal transmitter, the average RSS attenuation at different distances follows a logarithmic relationship with respect to distance. In the simplest case, the RSS can be expressed as:

$$RSS = P_t - K - 10\alpha \log_{10}(d), \quad (1)$$

where α is the path loss exponent, P_t is the transmit power, d denotes the transmitter–receiver distance, and K is a constant dependent on the propagation environment and carrier frequency.

In indoor environments, multiple Access Points (APs) are typically deployed. When a user device performs localization, it can detect WiFi signals transmitted from multiple APs and measure the corresponding RSS values. These RSS measurements from different APs form a vector, referred to as the RSS fingerprint.

For example, in an indoor scenario containing four APs (i.e., AP_1 , AP_2 , AP_3 , and AP_4), where each AP is equipped with a single transmitting antenna, the RSS fingerprint collected by a user device can be represented as:

$$(RSS_1, RSS_2, RSS_3, RSS_4). \quad (2)$$

3.2 Hierarchical Navigable Small World Graphs

Hierarchical Navigable Small World (HNSW) graphs, proposed by Malkov and Yashunin [36] in 2018, represent a state-of-the-art approximate nearest neighbor (ANN) search algorithm designed to address performance bottlenecks in large-scale high-dimensional data retrieval. The core idea is based on the small-world network property, where a hierarchical graph structure is constructed with sparse upper layers and dense lower layers to enable efficient search. This design overcomes the limitations of traditional brute-force matching, tree-based structures, and clustering-based methods in high-dimensional scenarios.

During the index construction phase, each data vector is randomly assigned a level, with most vectors residing in the lower layers. In each layer, nodes are connected only to their nearest neighbors, and the connectivity and construction quality are controlled by parameters M and $ef_{\text{construction}}$. During the search phase, a top-down greedy strategy is adopted: upper layers are used to quickly narrow down the search space, while lower layers refine the candidate set. This process reduces the search complexity to approximately $\mathcal{O}(\log N)$, achieving high recall performance (typically above 95%) with millisecond-level query latency, while maintaining strong adaptability to high-dimensional data and dynamic updates.

In our implementation, the HNSW parameters are set as follows: $M = 16$, $ef_{\text{construction}} = 100$, and $ef_{\text{search}} = 50$, which provide a good trade-off between retrieval accuracy and computational efficiency.

To evaluate the effectiveness of the HNSW-based retrieval, we measure the recall defined as the proportion of true nearest neighbors successfully retrieved compared to exact search. Under these settings, the HNSW algorithm achieves an average recall of approximately 96%, indicating high-quality approximate nearest neighbor performance.

The conventional WKNN algorithm relies on brute-force vector matching, resulting in a time complexity of $\mathcal{O}(N)$. As the fingerprint database scales due to large-area coverage and massive sample collection, the retrieval time increases significantly, leading to poor real-time performance and becoming a major bottleneck for large-scale deployment.

The characteristics of HNSW align well with these challenges. Its adaptability to high-dimensional data handles dense AP deployments effectively, its efficient search capability maintains millisecond-level matching even with millions of fingerprints, and its dynamic update capability allows local index updates without full reconstruction. By replacing the brute-force search in traditional WKNN with HNSW-based indexing, the proposed framework significantly improves localization efficiency without substantial loss of accuracy.

3.3 Kalman Filtering

The Kalman filter is a recursive algorithm that integrates a system model with observational data to iteratively update the state estimate and provide an optimal estimation.

In indoor localization, user movement can be modeled as a dynamic system, where the observed positions correspond to measurement values, and localization errors are treated as observation noise. The Kalman filter dynamically adjusts the Kalman gain to balance the predicted state and the observed measurements, thereby effectively reducing observation noise, smoothing the trajectory, and improving localization accuracy. Fig. 1 illustrates the smoothing process.

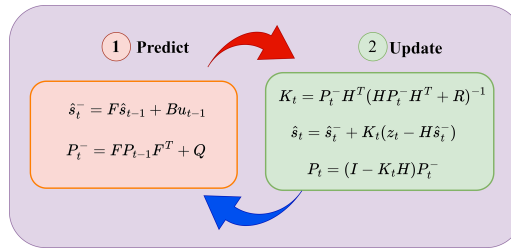


Figure 1: Kalman filter smoothing process.

In this work, the user state at time step t consists of the position coordinates along the x -axis and y -axis, as well as the velocity. The initial velocity is set to zero. The state evolves according to a linear motion model, where process noise accounts for variations in velocity and direction. The observed position is modeled as a measurement with additive noise.

The Kalman filter operates in two stages: prediction and update. In the prediction stage, the prior state estimate is computed based on the previous state and the linear motion model. The covariance of the prior estimate is also updated to account for process noise, which can be expressed as:

$$\hat{s}_t^- = F\hat{s}_{t-1} + Bu_{t-1}, \quad (3)$$

$$P_t^- = FP_{t-1}F^T + Q, \quad (4)$$

where F is the state transition matrix, B is the control matrix, u_{t-1} denotes the control input, and Q represents the process noise covariance.

In this work, the process noise covariance matrix is set as $Q = \text{diag}(0.01, 0.01, 0.1, 0.1)$ and the measurement noise covariance matrix is set as $R = \text{diag}(0.5, 0.5)$. These values are determined through empirical tuning to balance prediction and observation, ensuring stable convergence and smooth trajectory estimation.

In the update stage, the Kalman gain is computed as:

$$K_t = P_t^- H^T (HP_t^- H^T + R)^{-1}, \quad (5)$$

where H is the observation matrix and R denotes the measurement noise covariance.

The posterior state estimate and covariance are updated as follows:

$$\hat{\mathbf{s}}_t = \hat{\mathbf{s}}_t^- + \mathbf{K}_t (\mathbf{z}_t - \mathbf{H}\hat{\mathbf{s}}_t^-), \quad (6)$$

$$\mathbf{P}_t = (\mathbf{I} - \mathbf{K}_t\mathbf{H}) \mathbf{P}_t^-, \quad (7)$$

where $\hat{\mathbf{s}}_t$ is the posterior state estimate, $\mathbf{P}_t$ is the posterior error covariance, $\mathbf{K}_t$ denotes the Kalman gain, $\mathbf{z}_t$ is the measurement vector, $\mathbf{H}$ is the observation matrix, and $\mathbf{I}$ is the identity matrix.

Through this recursive prediction–update cycle, the Kalman filter smooths the trajectory, suppresses measurement noise, and provides a reliable estimate of user motion for subsequent localization.

It is worth noting that the Kalman filter requires a short initialization period to reach convergence. During the initial stage, the estimation may be affected by higher uncertainty due to the initial state assumptions. However, as sequential observations are incorporated, the filter rapidly converges to a stable state. In our experiments, the convergence is typically achieved within a few time steps, and its impact on the overall localization accuracy is negligible, as most positioning results are obtained after convergence, while significantly improving trajectory smoothness and stability.

In addition, the constant velocity model is adopted for its simplicity and computational efficiency. Although indoor pedestrian motion may involve changes in speed and direction, the primary role of the Kalman filter in this work is to smooth short-term localization noise rather than to model complex motion dynamics. Experimental results demonstrate that this assumption is sufficient to achieve stable and accurate localization performance, while maintaining low computational overhead. More advanced motion models may be explored in future work.

4 Proposed Method

In this section, the AT-WKNN algorithm is proposed, which is a high-accuracy and high-efficiency trajectory-assisted adaptive RSS fingerprinting localization method. First, in the online localization stage, the collected RSS fingerprint data are preprocessed through outlier removal and noise suppression. Meanwhile, a portion of the data is separately reserved as a test dataset to support subsequent range constraint operations. Subsequently, a localization strategy integrating distance metrics and range constraints is developed. Specifically, a hybrid similarity measure is constructed by combining Manhattan distance and cosine similarity in a weighted manner. In addition, a candidate filtering mechanism based on range constraints is introduced to enable precise selection of candidate reference points. Next, the HNSW algorithm is employed to perform efficient nearest neighbor retrieval, replacing the brute-force search in the traditional WKNN algorithm. This step produces an initial estimated trajectory, which inevitably contains a certain amount of noise and errors. Finally, a Kalman filter is applied to smooth the noisy trajectory, effectively suppressing localization fluctuations.

Through the above procedures, high-accuracy indoor localization with low computational cost is achieved. The overall workflow of the proposed method is illustrated in [Fig. 2](#).

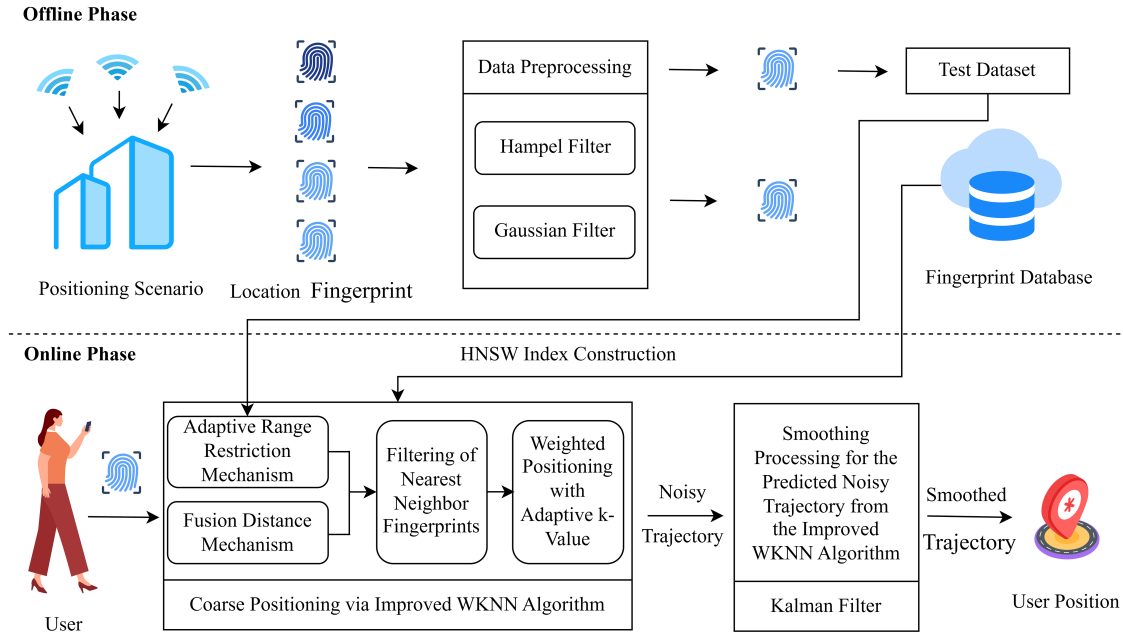


Figure 2: Algorithm flowchart.

The main notations used in this paper are summarized in Table 1.

Table 1: Main notations used in this paper.

Notation	Description
$\mathbf{x} \in \mathbb{R}^2$	Spatial coordinate
$\mathcal{D} = \{\mathbf{x}_i, \mathbf{RSS}_i\}_{i=1}^N$	Fingerprint database
$\mathbf{RSS} = [r_1, r_2, \dots, r_M]$	RSS vector
K	Number of nearest neighbors
$d(\cdot, \cdot)$	Distance metric
$\alpha \in [0, 1]$	Weight coefficient
$\hat{\mathbf{x}}$	Estimated location
$\mathcal{N}_K$	Neighbor set
$\mathbf{x}_k$	Kalman state at time step k
$\mathbf{z}_k$	Observation at time step k

4.1 RSS Fingerprint Database Preprocessing

RSS fingerprint data collected in indoor environments are susceptible to various factors, such as multipath propagation, device hardware imperfections, and external electromagnetic interference. These factors introduce outliers and random noise, which degrade the reliability of fingerprint data and ultimately affect localization accuracy. Therefore, a two-step preprocessing strategy is adopted in this work, where Hampel filtering is used for outlier removal and Gaussian filtering is applied for noise suppression, in order to enhance the quality of the collected RSS fingerprint data.

Hampel filtering is a nonlinear filtering method based on statistical characteristics, which is particularly effective for removing isolated outliers in data sequences. Its core idea is to identify and replace anomalous

data points by measuring their deviation from the local neighborhood within a sliding window. This method demonstrates strong performance in detecting outliers in normally distributed data while maintaining low computational complexity, making it suitable for real-time preprocessing of RSS fingerprint data.

Assume that the collected RSS fingerprint sequence is denoted as $\mathbf{X} = [x_1, x_2, \dots, x_n]$, where n represents the length of the data sequence. The Hampel filtering process can be described as follows:

Step 1: Define a sliding window of size k , where each side of the window contains $m = \frac{k-1}{2}$ data points. Traverse each data point x_i in the sequence ($m+1 \leq i \leq n-m$).

Step 2: Compute the median value within the window:

$$\text{Med} = \text{median}(x_{i-m}, x_{i-m+1}, \dots, x_{i+m}), \quad (8)$$

Step 3: Compute the median absolute deviation (MAD) within the window:

$$\text{MAD} = \text{median}(|x_{i-m} - \text{Med}|, |x_{i-m+1} - \text{Med}|, \dots, |x_{i+m} - \text{Med}|), \quad (9)$$

Step 4: Compute the threshold for outlier detection:

$$T = 3 \times 1.4826 \times \text{MAD}, \quad (10)$$

where 1.4826 is the scale factor that converts the median absolute deviation (MAD) to the standard deviation under the assumption of a normal distribution, and 3 is an empirical threshold coefficient.

Step 5: Determine whether each data point is an outlier:

$$|x_i - \text{Med}| > T. \quad (11)$$

If the above condition is satisfied, x_i is identified as an outlier and is replaced by the median value Med; otherwise, the original data point is retained.

The core formulation of the Hampel filter can be expressed as:

$$x'_i = \begin{cases} \text{Med}, & |x_i - \text{Med}| > 3 \times 1.4826 \times \text{MAD} \\ x_i, & \text{otherwise} \end{cases}. \quad (12)$$

After outliers are removed by Hampel filtering, a small amount of random Gaussian noise may still remain in the RSS data. Therefore, Gaussian filtering is further applied to smooth the data.

Gaussian filtering is based on the principle of weighted averaging using a Gaussian function. By performing a weighted summation over the neighboring data points of each sample, it effectively suppresses high-frequency noise while preserving the underlying trend of the original data, thereby avoiding the loss of fingerprint information caused by over-smoothing.

Assume that the window size of the Gaussian filter is $2s+1$. The Gaussian weights corresponding to each data point are computed using a two-dimensional Gaussian function, whose core formulation is given by:

$$G(u, v) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{u^2 + v^2}{2\sigma^2}\right), \quad (13)$$

where (u, v) denotes the coordinate offset of the data points in the window relative to the central data point, and σ is the standard deviation of the Gaussian function, which determines the smoothing degree of the filtering. A larger σ results in a stronger smoothing effect but may lead to the loss of data details; conversely,

an excessively small σ leads to poor noise suppression performance. In this paper, $\sigma = 1.5$ is determined through experiments to balance noise suppression and data fidelity.

The RSS sequence $X = [x_1, x_2, \dots, x_n]$ after Hampel filtering is processed by Gaussian filtering, with the specific steps as follows: traverse each data point in the sequence, construct a sliding window centered at this point, calculate the Gaussian weight of each data point in the window according to the above formula, and perform weighted summation on the data in the window to obtain the final filtered data point x_i'' , namely:

$$x_i'' = \frac{\sum_{j=-s}^s G(0, j) \cdot x'_{i+j}}{\sum_{j=-s}^s G(0, j)}. \quad (14)$$

To validate the effectiveness of the proposed preprocessing strategy, a comparative analysis is conducted among the original RSS fingerprint data, the data after Hampel filtering, and the data after Gaussian filtering. The preprocessing results are illustrated in Fig. 3.

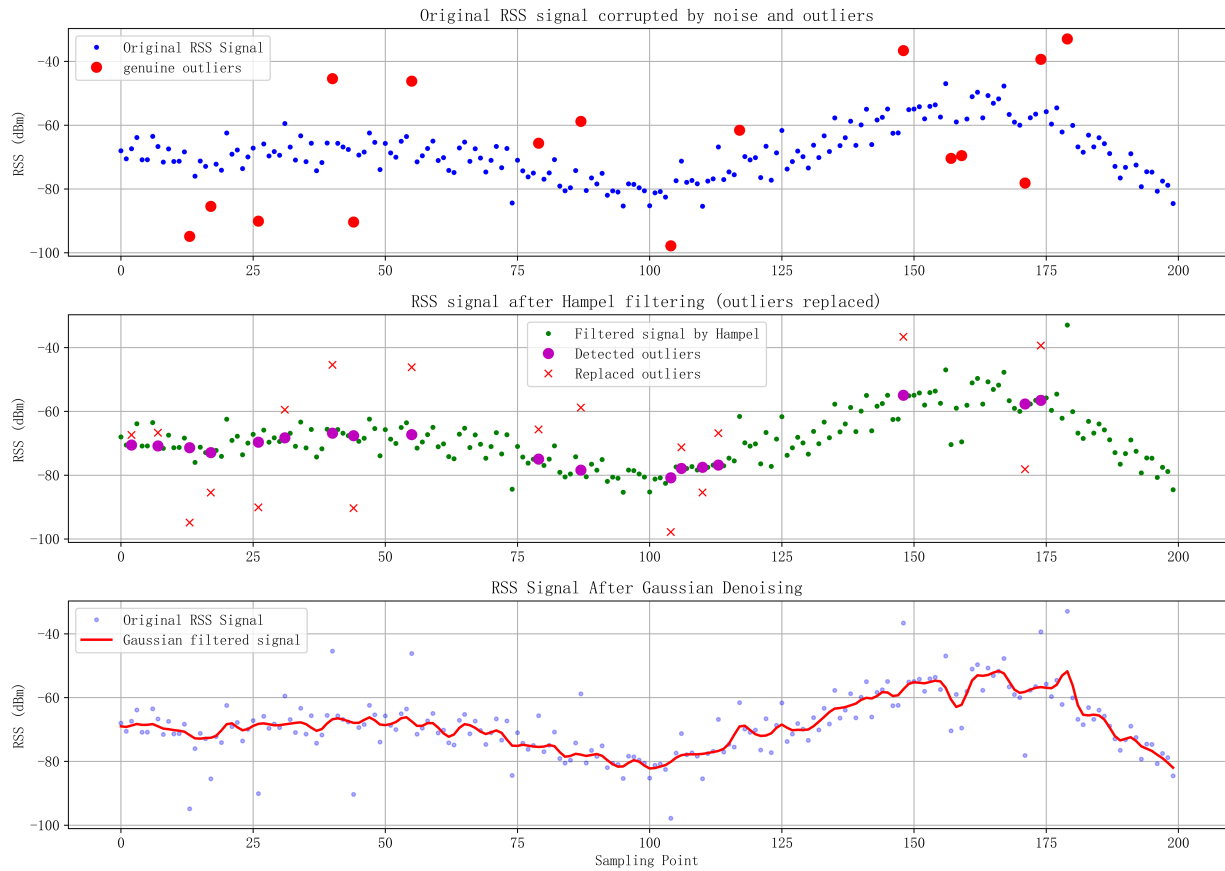


Figure 3: Noise and outlier removal from RSS sequences.

4.2 Localization Strategy with Hybrid Distance Metric and Adaptive Range Constraint

The traditional WKNN algorithm relies on a single distance metric to evaluate fingerprint similarity during candidate selection. However, it is highly susceptible to RSS fluctuations caused by dynamic indoor environments, which may lead to mismatched fingerprints and degraded localization accuracy. Moreover, a single distance metric is insufficient for filtering out invalid samples, resulting in an excessive number of

candidate points and significantly increasing the computational cost of subsequent nearest neighbor search, thereby reducing real-time performance.

To address these issues, this section proposes a dual-optimization localization strategy that integrates a hybrid distance metric with adaptive range constraints. By leveraging multi-dimensional similarity evaluation, the proposed method improves matching accuracy, while range constraints are employed to filter out irrelevant samples. This enables precise candidate selection and enhances computational efficiency, laying a solid foundation for subsequent nearest neighbor retrieval.

Specifically, a hybrid distance metric is constructed by combining Manhattan distance and cosine similarity in a weighted manner, taking into account both the absolute differences and the trend consistency of RSS fingerprints, thereby improving robustness and matching accuracy. The Manhattan distance is used to capture the absolute numerical differences between RSS fingerprint vectors. It is computationally simple and less sensitive to local outliers, effectively reflecting quantitative differences in RSS values across locations. In contrast, cosine similarity measures the trend consistency between fingerprint vectors, focusing on the variation patterns rather than absolute magnitudes, which makes it robust against RSS shifts caused by environmental dynamics. By integrating these two metrics, their complementary advantages are exploited to construct a comprehensive similarity measure, which is defined as follows:

$$D_{ij} = \alpha \cdot \text{Manhattan}(x_i'', x_j) + (1 - \alpha) \cdot (1 - \text{Cos}(x_i', x_j)), \quad (15)$$

where D_{ij} denotes the composite distance between the preprocessed optimized fingerprint x_i'' and the training sample x_j in the fingerprint database. A smaller D_{ij} indicates a higher similarity between the two fingerprints, corresponding to a better position matching performance.

α is the weight coefficient used to balance the contributions of the Manhattan distance and cosine similarity. In this work, α is set to 0.6 based on empirical evaluation. To further evaluate the impact of the weight coefficient α , a sensitivity analysis is conducted by varying α within the range [0.3, 0.9]. The results are shown in Fig. 4. As α increases from 0.3 to 0.6, the localization error decreases and reaches the minimum at $\alpha = 0.6$. When α continues to increase beyond 0.6, the localization error gradually increases. In addition, the performance remains relatively stable within the range of 0.4 to 0.7, indicating that the proposed method is not highly sensitive to the exact value of α . These results demonstrate that $\alpha = 0.6$ provides a good balance between the Manhattan distance and cosine similarity components.

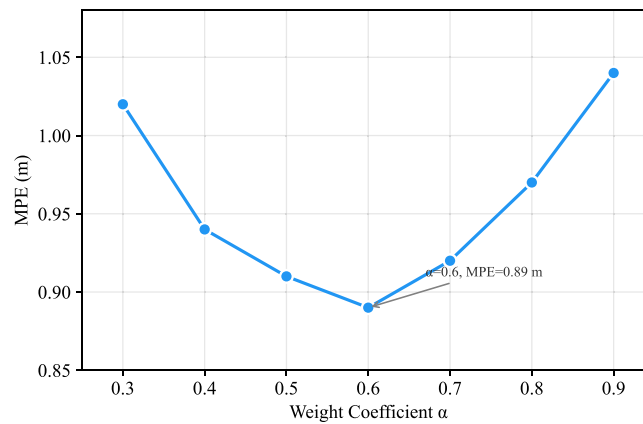


Figure 4: Sensitivity analysis of the weight coefficient α on localization accuracy.

This value ensures the dominant role of absolute numerical differences while taking into account the auxiliary optimization effect of trend consistency. $\text{Manhattan}(\cdot)$ is the Manhattan distance calculation function, and $\text{Cos}(\cdot)$ is the cosine similarity calculation function. Since the value range of cosine similarity is $[0, 1]$, $1 - \text{Cos}(\cdot)$ is used to convert it into a distance metric, unifying the dimensions of the two indicators and ensuring the rationality and effectiveness of the fusion formula.

4.3 Range Restriction Mechanism for Candidate Optimization

Secondly, a candidate selection filtering mechanism based on range constraints is introduced to further optimize the candidate set and reduce invalid calculations. This mechanism takes the positioning range model constructed from the test dataset as the core basis, combining the physical boundaries in the indoor space and the distribution characteristics of RSS fingerprints. Two dual constraint conditions are established:

1. Comprehensive distance threshold D_{th} : By statistically analyzing the comprehensive distance distribution of normal samples in the test dataset, D_{th} is set to the mean value plus one standard deviation to ensure that only samples with high similarity are retained.
2. Spatial range constraint: Based on the position information of samples in the fingerprint database, a reasonable spatial range is defined as a circular region centered at the previously estimated position, with an adaptive radius given by

$$r = v\Delta t + \text{upperquartile}(e), \quad (16)$$

where v denotes the average walking speed (set to 1.4 m/s, corresponding to a typical pedestrian walking speed in indoor environments), Δt is the time interval between consecutive localization steps, and $\text{upperquartile}(e)$ represents the upper quartile of the localization error. This formulation jointly considers motion continuity and localization uncertainty, enabling robust candidate filtering.

Through the dual constraints, only fingerprint samples with a comprehensive distance $D_{ij} \leq D_{th}$ and within the reasonable spatial range are retained as candidate nearest neighbors. This mechanism not only effectively eliminates invalid samples and abnormal matching samples outside the actual positioning area, but also adaptively constrains the candidate search region based on motion and uncertainty, significantly reducing the computational overhead of nearest neighbor search while improving the overall quality of candidates. The schematic diagram of the range restriction mechanism is shown in Fig. 5.

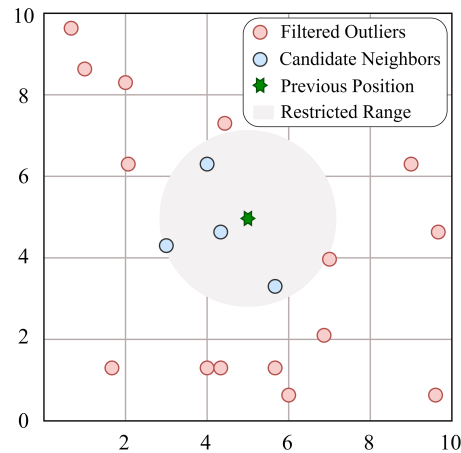


Figure 5: Schematic diagram of range limiting mechanism.

4.4 Efficient Nearest Neighbor Retrieval Based on HNSW

After candidate point selection, nearest neighbor retrieval is required to identify the samples that best match the current RSS fingerprint, thereby enabling the estimation of an initial localization trajectory.

In the traditional WKNN algorithm, a brute-force search is performed by traversing all candidate points, computing distances individually, and selecting the nearest neighbors. When the number of candidate points is large, the time complexity increases significantly, reaching $\mathcal{O}(N)$, which makes it difficult to meet the real-time requirements of indoor localization.

To address this issue, the HNSW algorithm is employed to replace the conventional brute-force search. By leveraging an efficient hierarchical graph-based retrieval mechanism, HNSW enables fast nearest neighbor matching among candidate points. This approach significantly improves computational efficiency while maintaining high retrieval accuracy, thereby allowing rapid acquisition of the initial estimated trajectory.

The HNSW algorithm is an ANN search method based on a hierarchical graph structure. Its core advantage lies in achieving a balance between retrieval accuracy and efficiency through the construction of multi-layer graph structures.

Specifically, the algorithm builds multiple levels of graphs. The bottom layer is a fully populated graph containing all candidate points, where each node is connected to several neighboring nodes with high similarity. As the hierarchy ascends, the graph becomes increasingly sparse. The top layer contains only a small subset of nodes, preserving only the most significant connections between highly similar nodes.

The search process follows a top-down strategy. It starts from the top layer, where candidate nodes that are most similar to the target fingerprint are quickly identified. Guided by these candidates, the search progressively descends to lower layers, refining the candidate set at each level. Finally, precise nearest neighbors that best match the target fingerprint are identified in the bottom layer.

This hierarchical search strategy eliminates the need to traverse all candidate points and instead localizes the search region through successive filtering. As a result, the time complexity can be reduced to approximately $\mathcal{O}(\log N)$, which is significantly lower than the $\mathcal{O}(N)$ complexity of brute-force search. Meanwhile, by optimizing the graph connectivity, HNSW achieves retrieval accuracy close to that of brute-force methods, thereby satisfying the accuracy requirements of indoor localization while maintaining high efficiency.

4.5 Trajectory Smoothing Based on Kalman Filtering

To effectively suppress localization jitter, correct trajectory deviations, and improve the stability and accuracy of localization results, a Kalman filter is employed to smooth the initially estimated trajectory, thereby eliminating random noise interference.

The Kalman filter is a recursive optimal estimation algorithm based on linear state-space models. Its key advantage lies in its ability to provide optimal estimates without requiring prior knowledge of noise statistics. Through an iterative prediction–update process, it can effectively handle noisy data with relatively low computational complexity and strong real-time capability, making it well-suited for trajectory smoothing in dynamic indoor localization scenarios. Compared with traditional filtering methods, the Kalman filter preserves the true motion trend of the trajectory while effectively suppressing random noise, avoiding the lag effects caused by over-smoothing, and thus better adapting to dynamic movements of indoor users or devices.

In the proposed trajectory smoothing framework, the initially estimated trajectory points are treated as observations. By incorporating the motion characteristics of the target, the state equation and observation

equation are constructed, and trajectory optimization is achieved through iterative computation. The core steps are summarized as follows:

- (1) State Prediction: Based on the optimal trajectory point obtained at the previous time step, the current state is predicted using the motion model. Meanwhile, the error covariance matrix is updated to reflect the uncertainty of the prediction.
- (2) State Update: The predicted trajectory point at the current time step is treated as the observation. The Kalman gain is computed according to the observation model, and the predicted state is then corrected using the Kalman gain to obtain the optimal trajectory estimate. The error covariance matrix is subsequently updated for the next iteration.

Through the smoothing effect of the Kalman filter, random noise and localization jitter in the initial trajectory can be effectively suppressed, while preserving the true motion trend. This further improves the localization accuracy and stability of the proposed algorithm.

5 Experiments and Results

5.1 Experimental Setup and Dataset Description

To further evaluate the adaptability and stability of the proposed localization scheme under different indoor environments, four experimental scenarios are designed for comparative experiments, ensuring the generality of the experimental results. Scenarios 1 and 2 are self-collected small-scale indoor environments, while Scenarios 3 and 4 are constructed based on the publicly available UJIIndoorLoc benchmark dataset to validate the scalability of the proposed algorithm in large-scale environments.

Scenario 1 represents a typical office environment, with an experimental area of approximately 10×10 m. The area contains common indoor furniture such as desks, chairs, storage cabinets, and air conditioners. In addition, interference from unrelated access points (APs) and a moderate level of human activity are present. This setup effectively simulates practical indoor localization conditions in daily office environments, ensuring the applicability of the experimental results.

Scenario 2 represents a complex indoor environment with severe obstructions. The experimental area remains the same as in Scenario 1. However, additional large obstacles such as bookshelves and filing cabinets are introduced. Furthermore, the interference from unrelated APs is intensified, and the frequency of human activity is increased. This scenario is designed to simulate highly obstructed and dynamic indoor environments, thereby evaluating the robustness and stability of the localization scheme under complex interference conditions.

The experimental deployment is kept consistent across both scenarios to ensure the comparability of results. Prior to the experiments, reference point calibration and AP deployment are conducted. The spacing between reference points directly affects both experimental efficiency and localization accuracy: a smaller spacing increases the workload of constructing the RSS fingerprint database in the offline phase, whereas a larger spacing reduces localization accuracy. Considering the 10×10 m experimental area, the spacing between reference points is set to 1 m in both scenarios.

In RSS fingerprint-based localization, at least three APs are required for effective positioning. Considering data collection accuracy, device compatibility, and cost efficiency, three APs are deployed in each scenario. The ESP8266 NodeMCU development board is used as both the AP device and RSS data acquisition unit. These devices are placed on different tables within the experimental area to ensure comprehensive and uniform signal coverage.

To construct the test dataset and evaluate localization performance, multiple test points are arranged at intervals of 0.5 m within the environment. Some test points coincide with reference points, enabling comparative analysis of localization error and fingerprint matching accuracy under different scenarios.

To further evaluate the scalability of the proposed algorithm in large-scale environments, Scenarios 3 and 4 are constructed based on the publicly available UJIIndoorLoc benchmark dataset [37]. The UJIIndoorLoc database covers three buildings of Universitat Jaume I, Spain, with 4 or more floors and a total area of approximately 110,000 m². It comprises 19,937 training records and 1111 validation records with 520 detected Wireless Access Points (WAPs), collected by more than 20 users using 25 Android devices, and is one of the most widely adopted benchmark datasets in the indoor localization community.

In this work, each floor is treated as an independent 2D localization scenario, consistent with the positioning framework adopted in Scenarios 1 and 2. Scenario 3 corresponds to Building 2, Floor 1 of the UJIIndoorLoc dataset, representing a large-scale open corridor and office environment. Scenario 4 corresponds to Building 2, Floor 2, featuring a more heterogeneous spatial layout with higher structural complexity. Building 2 is selected because its per-floor sample density is the highest among all three buildings in the dataset. For both scenarios, the RSS value of +100 dBm used in UJIIndoorLoc to indicate an undetected WAP is replaced with −105 dBm prior to preprocessing, following common practice in the literature. Only WAPs with at least one valid detection across all samples are retained to reduce input dimensionality. The retained training records are split into training and test subsets using a 70:30 ratio.

It should be noted that although the training sample counts of Scenarios 3 and 4 are slightly lower than those of Scenarios 1 and 2, the test sample counts are substantially larger (649 and 474 vs. 100), providing greater statistical power for performance evaluation. Furthermore, the significantly higher AP dimensionality (161 and 149 vs. 3) and the much larger spatial coverage represent a considerably more challenging localization environment, further demonstrating the generalization capability of the proposed method.

The main parameters of all four experimental scenarios are summarized in Table 2.

Table 2: Parameter settings of the four experimental scenarios.

Parameter Name	Scenario 1	Scenario 2	Scenario 3	Scenario 4
Data source	Self-collected	Self-collected	UJIIndoorLoc (Bld.2, Fl.1)	UJIIndoorLoc (Bld.2, Fl.2)
Area size	10 m × 10 m	10 m × 10 m	104.8 m × 112.4 m	102.1 m × 105.5 m
Indoor obstacles	Desks, chairs, storage cabinets, air conditioners	Conventional low furniture, 4 sets of 1.8 m-high bookshelves, 6 sets of 1.5 m-high file cabinets	Corridors, offices, labs	Corridors, offices, labs
Unrelated AP interference intensity	−85~−75 dBm	−75~−65 dBm	Mixed (multi-device)	Mixed (multi-device)
Human activity frequency	2~3 people/10 min	5~6 people/10 min	Crowdsourced (18 users)	Crowdsourced (18 users)
AP deployment height	1.2 m	1.2 m	—	—

(Continued)

Table 2 (continued)

Parameter Name	Scenario 1	Scenario 2	Scenario 3	Scenario 4
AP signal transmission power	15 dBm	15 dBm	—	—
AP count	3	3	161	149
Training samples	1920	1920	1513	1103
Test samples	100	100	649	474

The ESP8266 NodeMCU development board (IEEE 802.11b/g/n) is used as both the AP device and RSS data acquisition unit, featuring low power consumption and ease of deployment, making it well-suited for constructing the RSS fingerprint database in this study.

The experiment requires the collection of three types of data: (1) data for constructing the RSS fingerprint database; (2) a test dataset for simulating localization error estimation; and (3) a test dataset for evaluating localization accuracy.

During the construction of the RSS fingerprint database, the operator carries a data acquisition device and traverses all reference points. At each point, 30 RSS fingerprints are collected at a sampling rate of 1 Hz. These fingerprints, along with their corresponding coordinates, are stored in the fingerprint database. After preprocessing, the database is utilized in the online localization phase.

During the test dataset construction phase, the operator moves randomly among predefined coordinate points while carrying the acquisition device, simultaneously collecting RSS fingerprints and their corresponding ground-truth positions. In this study, two independent test datasets are constructed, each containing 100 RSS fingerprint samples.

The first dataset is used to analyze the distribution characteristics of localization errors based on the WKNN algorithm, and to determine a reference threshold for the localization error constraint radius in subsequent models; this dataset is referred to as the localization error simulation dataset. The second dataset is specifically used to quantitatively evaluate the average localization error of the proposed method and is referred to as the localization error evaluation dataset.

In this experiment, the size of the RSS fingerprint database is 1920×5 , while each of the two test datasets has a size of 100×5 . Each fingerprint sample consists of three RSS values, and each data record contains one RSS fingerprint vector and two coordinate values.

5.2 Evaluation Metrics

To comprehensively evaluate the performance of the proposed positioning algorithm, several quantitative metrics are adopted in this paper, including the mean position error (MPE), mean running time (MRT), median absolute error (MedAE), root mean squared error (RMSE), cumulative distribution function (CDF), and box plot analysis. Comparative evaluations are conducted from three perspectives: average error level, probability distribution characteristics, and error dispersion features.

The mean position error reflects the overall positioning accuracy of the algorithm, which is defined as the average positioning error over all test samples. The corresponding calculation formula is given in Eq. (17):

$$\text{MPE} = \frac{1}{n} \sum_{i=1}^n \sqrt{(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2}, \quad (17)$$

where n denotes the number of test points, $(\hat{x}_i, \hat{y}_i)$ represents the i -th predicted coordinate, and (x_i, y_i) is the i -th ground-truth coordinate.

The CDF is used to describe the probability that a random variable is less than or equal to a specific value. In indoor positioning, the CDF is commonly adopted to represent the proportion of test samples with positioning error less than a given error threshold, relative to the total number of test samples. Its mathematical expression is defined as:

$$F_E(e) = P(E \leq e), \quad (18)$$

where E denotes the random variable of the positioning error distance for all test samples, and $P(E \leq e)$ represents the probability that the positioning error is less than e .

As an intuitive statistical visualization tool, the box plot can clearly display the central tendency, dispersion degree, and outlier distribution of positioning errors. In this paper, the upper and lower boundaries of the box in the box plot correspond to the upper quartile and lower quartile of the positioning errors, respectively. The height of the box is the interquartile range (IQR), which reflects the concentration degree of the errors. The horizontal line inside the box is the median of the positioning errors, representing that approximately 50% of the test samples have positioning errors less than this value. The whiskers extending from the box cover the data points within the interval $[Q1 - 1.5 \times IQR, Q3 + 1.5 \times IQR]$, and the endpoints of the whiskers are the minimum and maximum error values within this interval. The scattered points outside the whiskers are extreme positioning errors beyond the normal fluctuation range, which can reflect the deviation of the algorithm in a small number of scenarios.

5.3 Ablation Experiment

To ensure the rigor and comparability of the experiments, this study designs 5 groups of comparative models to systematically verify the contribution of each core module in the proposed AT-WKNN algorithm. Among them, Model M1 is the classic WKNN algorithm, which is adopted as the baseline method; Model M5 is the complete AT-WKNN algorithm, serving as the final benchmark scheme; Models M2 to M4 are simplified versions obtained by sequentially removing the core modules, and the performance gain of each module to the overall system is quantified through a “step-by-step decomposition” approach. All models adopt a unified data preprocessing pipeline, including WiFi fingerprint database construction, signal strength normalization, and outlier filtering, to eliminate the interference of preprocessing differences on the experimental results. The specific design is shown in Table 3.

Table 3: Ablation experiment setup.

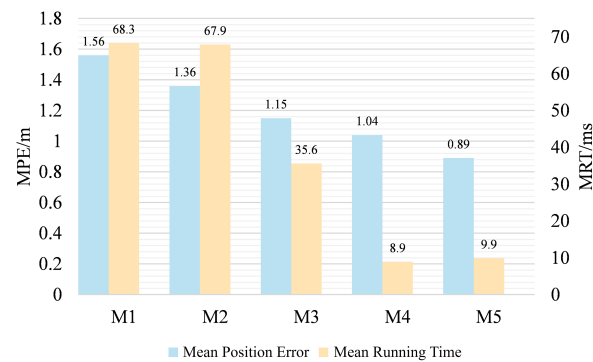
No.	Model Name	Description
M1	Baseline Model (Control Group)	Retains the traditional single Euclidean distance and only uses the brute-force search algorithm
M2	Fusion Distance Model	Replaces the Euclidean distance in the WKNN algorithm with the proposed fusion distance
M3	RLWKNN	Adds the range restriction mechanism based on Model M2

(Continued)

Table 3 (continued)

No.	Model Name	Description
M4	HLWKNN	Replaces the brute-force search algorithm with the HNSW algorithm based on Model M3
M5	AT-WKNN (Complete Algorithm)	Smooths the positioning trajectory using the Kalman filter based on Model M4

The experimental results are shown in Fig. 6. In this experiment, a bar chart is used to visually compare the core performance metrics of the five models M1 to M5. The blue bars represent the MPE in meters (m), which reflects the positioning accuracy of the model; the orange bars represent the MRT in milliseconds (ms), which reflects the computational efficiency and real-time performance of the model. All experimental data are the average values after 10 repeated tests, with a unified experimental environment and parameter configuration, ensuring the objectivity and comparability of the results.

**Figure 6:** Comparison chart of positioning error and effect for M1–M5.

From the visualization results, two significant trends can be clearly observed. First, from the baseline model M1 to the complete AT-WKNN model M5, the MPE exhibits a consistent and substantial decreasing trend, demonstrating the incremental contribution of each optimization module to localization accuracy. Second, the MRT shows a staged variation pattern of “initially stable, then sharply decreasing, and finally slightly increasing,” eventually stabilizing within the threshold required for real-time localization. Overall, the AT-WKNN algorithm achieves a dual optimization in both accuracy and efficiency, indicating strong potential for practical deployment.

The mean positioning error (MPE) is defined as the average Euclidean distance between the estimated positions and the corresponding ground-truth locations. The experimental results show a clear decreasing trend in MPE across the models. Specifically, M1 (baseline WKNN) achieves an MPE of 1.56 m, which is progressively reduced to 1.36 m (M2), 1.15 m (M3), 1.04 m (M4), and finally 0.89 m (M5). Compared with the baseline WKNN method, this corresponds to an improvement of approximately 43.0% in positioning accuracy.

The performance gain from M1 to M2 is mainly attributed to the introduction of the hybrid distance metric, which improves fingerprint matching accuracy by integrating Manhattan distance and cosine similarity, thereby reducing the initial localization bias. The improvement from M2 to M3 demonstrates the effectiveness of the range constraint module, which filters out invalid samples and mitigates the impact

of noisy data. The small performance difference between M3 and M4 indicates that the HNSW-based nearest neighbor retrieval module primarily enhances computational efficiency rather than localization accuracy. The most significant improvement is observed from M4 to M5, owing to the Kalman filter-based trajectory smoothing module, which effectively suppresses trajectory jitter and minor deviations, enabling the proposed method to achieve high-precision indoor localization.

The MRT of each model exhibits a notable staged decreasing trend: M1 requires 68.3 ms, M2 remains nearly unchanged at 67.9 ms, M3 decreases to 35.6 ms, M4 sharply drops to 8.9 ms, and M5 slightly increases to 9.9 ms. This corresponds to an overall reduction of 85% compared with M1, ultimately satisfying real-time localization requirements. Specifically, the similar runtime of M1 and M2 indicates that the hybrid distance metric introduces negligible computational overhead. The reduction from M2 to M3 (47.6%) results from the range constraint module, which significantly reduces the number of candidate points. The transition from M3 to M4 represents the core efficiency improvement, where the HNSW module reduces the search complexity from $\mathcal{O}(N)$ to $\mathcal{O}(\log N)$, enabling real-time performance. The slight increase of approximately 1 ms from M4 to M5 indicates that the Kalman filtering process incurs minimal computational overhead while improving accuracy.

Based on the quantitative analysis of localization results, it can be concluded that the AT-WKNN algorithm achieves a significant breakthrough in both localization accuracy and computational efficiency through the synergistic integration of four key modules: hybrid distance metric, range constraint, HNSW-based efficient nearest neighbor retrieval, and Kalman filter-based trajectory smoothing.

To further validate the effectiveness of the Kalman filter-based trajectory smoothing module, a straight-line indoor movement trajectory is selected as the test case. The target moves uniformly from coordinates (1.0, 1.0) to (10.0, 1.0), during which 100 RSS fingerprint samples are continuously collected. The trajectories generated by M4 (without Kalman filtering) and M5 (with Kalman filtering) are compared with the ground-truth trajectory in terms of accuracy and smoothness.

Furthermore, trajectory visualization is conducted in two scenarios. Scenario 1 consists of a regular rectangular path, reflecting the continuity of localization under stable motion conditions. Scenario 2 involves a complex path with multiple turns, used to evaluate the smoothing capability and tracking stability under dynamic motion conditions.

Fig. 7 further shows the comparison between the positioning trajectories and the ground-truth trajectories of the two models. The original positioning results present an obvious scattered distribution near the real trajectory, and are prone to large positioning deviations especially at the turning positions. After Kalman filtering, the estimated trajectory can fit the real motion path more smoothly, significantly reducing the random jitter while ensuring the continuity of the trajectory. In particular, in the complex path scenario of Scenario 2, the Kalman filter can effectively alleviate the jump problem of positioning points, making the final trajectory more consistent with the real movement trend.

The experimental results show that introducing the Kalman filter trajectory smoothing module can effectively improve the stability and continuity of positioning results without changing the structure of the original positioning algorithm, thus providing more reliable position estimation for subsequent dynamic positioning and trajectory analysis.

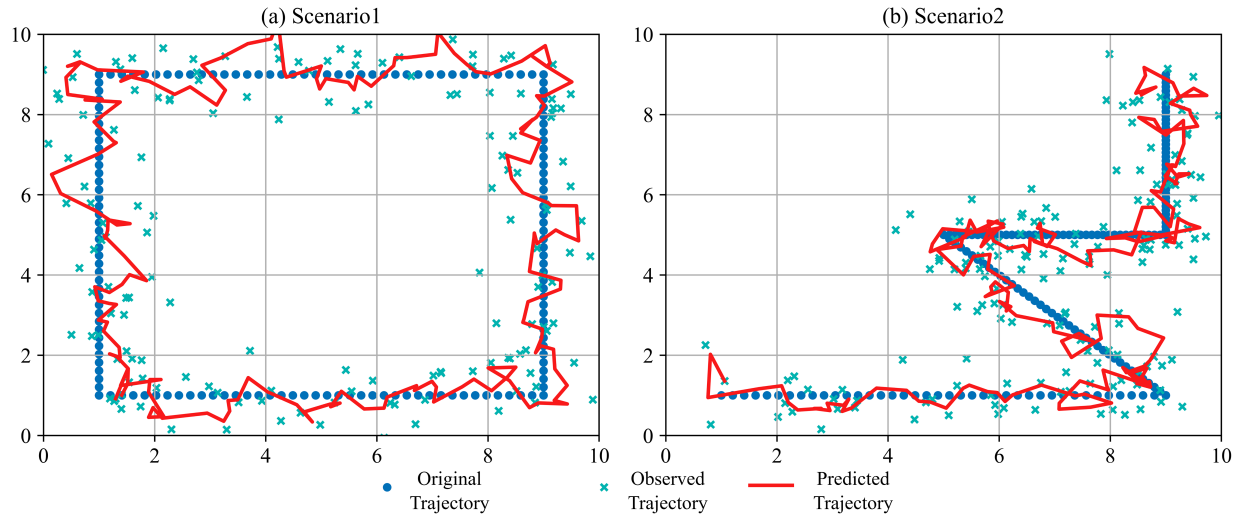


Figure 7: Comparison of trajectories before and after Kalman filtering.

5.4 Indoor Localization Performance Evaluation

To evaluate the localization performance of the proposed AT-WKNN algorithm across both small-scale and large-scale environments, different baseline methods are selected for the two groups of scenarios. For Scenarios 1 and 2, five methods are compared: KNN, WKNN, Random Forest (RF), Deep Learning (DL), and the proposed AT-WKNN, consistent with the original experimental setup. For Scenarios 3 and 4, four methods are compared: WKNN, Convolutional Neural Network-based localization (CNN), LSTM-based localization (LSTM), and the proposed AT-WKNN. The CNN-based method employs a fully-connected residual network that extracts spatial features from RSS fingerprint vectors for position estimation. The LSTM-based method combines Multi-Layer Perceptron (MLP)-based feature extraction with sequence modeling to capture the structural characteristics of RSS fingerprints. Both deep learning models are trained with the Adam optimizer using a learning rate of 1×10^{-3} and a batch size of 32, with early stopping applied to prevent overfitting.

The localization errors of different algorithms are presented in Tables 4 and 5. The statistical results indicate that the traditional KNN algorithm exhibits relatively low positioning accuracy, with overall large errors. The WKNN algorithm, which incorporates a weighting mechanism, and the RF algorithm based on ensemble learning, reduce the localization error to some extent and outperform KNN; however, there remains considerable room for improvement.

Table 4: Localization errors of five methods in Scenarios 1 and 2.

Scenario	Metric	AT-WKNN	DL	RF	WKNN	KNN
Scenario 1	MPE	0.89 m	1.296 m	1.333 m	1.187 m	1.706 m
	MPE	1.25 m	1.48 m	1.56 m	1.580 m	1.900 m
Scenario 2	MedAE	1.226 m	1.439 m	1.559 m	1.618 m	1.902 m
	RMSE	1.307 m	1.555 m	1.629 m	1.635 m	1.964 m

Table 5: Localization errors of four methods in Scenarios 3 and 4.

Scenario	Metric	AT-WKNN	CNN	LSTM	WKNN
Scenario 3	MPE	2.234 m	3.526 m	4.116 m	2.763 m
	MedAE	1.112 m	2.938 m	3.445 m	1.575 m
	RMSE	4.095 m	4.406 m	5.537 m	4.743 m
Scenario 4	MPE	1.941 m	2.848 m	3.067 m	2.767 m
	MedAE	0.891 m	2.324 m	2.546 m	1.289 m
	RMSE	4.648 m	3.588 m	3.843 m	6.698 m

Benefiting from its strong feature extraction and fitting capabilities, the DL approach further improves localization accuracy, achieving significantly lower errors compared with traditional machine learning methods.

To further validate the scalability and generalization capability of the proposed algorithm, experiments are conducted on Scenarios 3 and 4 derived from the large-scale UJIIndoorLoc benchmark dataset. In both large-scale scenarios, AT-WKNN achieves the lowest MPE and MedAE among all compared methods. Specifically, in Scenario 3, AT-WKNN achieves an MPE of 2.234 m and a MedAE of 1.112 m, outperforming WKNN (2.763 m), CNN (3.526 m), and LSTM (4.116 m). In Scenario 4, AT-WKNN achieves an MPE of 1.941 m and a MedAE of 0.891 m, representing improvements of 29.8% and 30.9% over WKNN in MPE and MedAE, respectively.

It is worth noting that while AT-WKNN achieves the lowest MPE and MedAE in Scenario 4, its RMSE (4.648 m) is slightly higher than that of CNN (3.588 m) and LSTM (3.843 m). This is attributed to a small number of test samples with relatively large localization errors that disproportionately elevate the RMSE. The median error of AT-WKNN (0.891 m) is substantially lower than that of CNN (2.324 m) and LSTM (2.546 m), confirming that the error distribution of AT-WKNN is more concentrated and the majority of its localization results are more accurate.

Regarding the deep learning baselines, the CNN-based method achieves reasonable accuracy across Scenarios 3 and 4 through effective residual feature extraction. The LSTM-based method achieves competitive performance with MPE values of 4.116 and 3.067 m in Scenarios 3 and 4, respectively. However, both deep learning methods require substantially more computational resources and training time compared to AT-WKNN. In contrast, AT-WKNN achieves superior localization accuracy without requiring GPU resources, maintaining a lightweight and efficient computational profile that is well-suited for energy-constrained IoT deployments.

To provide a more intuitive comparison of the performance differences among these algorithms, CDF curves and box plots are employed for visualization and analysis in the following section.

As shown in Fig. 8, the CDF of the proposed algorithm rises more rapidly than those of the other methods, indicating that its positioning errors are more concentrated within a smaller range, thereby achieving higher localization accuracy.

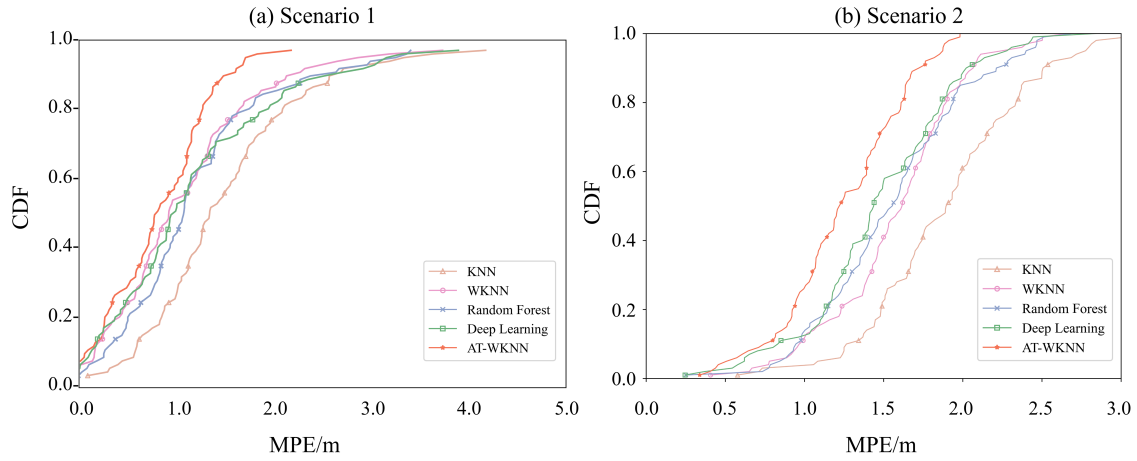


Figure 8: CDF curves of five localization algorithms in Scenarios 1 and 2.

To further evaluate the stability and consistency of the localization results, [Fig. 9](#) presents the boxplots of positioning errors for five algorithms in Scenarios 1 and 2. It can be observed that AT-WKNN outperforms the other methods across several key statistical metrics. Specifically, its whiskers are significantly shorter, indicating a narrower error range and reduced variability. Moreover, the upper quartile (Q3), median, and lower quartile (Q1) are all at lower levels, suggesting that most localization errors are smaller and more tightly distributed. In addition, AT-WKNN exhibits fewer and less extreme outliers, further demonstrating its superior stability and robustness in dynamic environments.

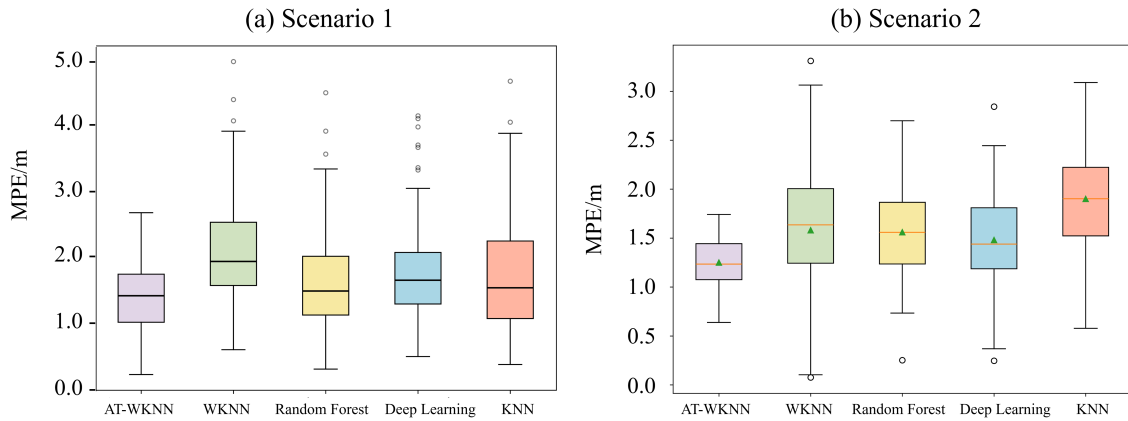


Figure 9: Boxplot comparison of five positioning algorithms in Scenarios 1 and 2.

The CDF curves and boxplots for Scenarios 3 and 4 are presented in [Figs. 10](#) and [11](#), respectively. As shown in [Fig. 10](#), the CDF curve of AT-WKNN rises most steeply in both scenarios, confirming that its positioning errors are more concentrated within a smaller range compared to all other methods. The boxplot results in [Fig. 11](#) further illustrate that AT-WKNN achieves the lowest median error and the narrowest interquartile range across both large-scale scenarios, demonstrating its robustness and consistency under complex, high-dimensional indoor environments.

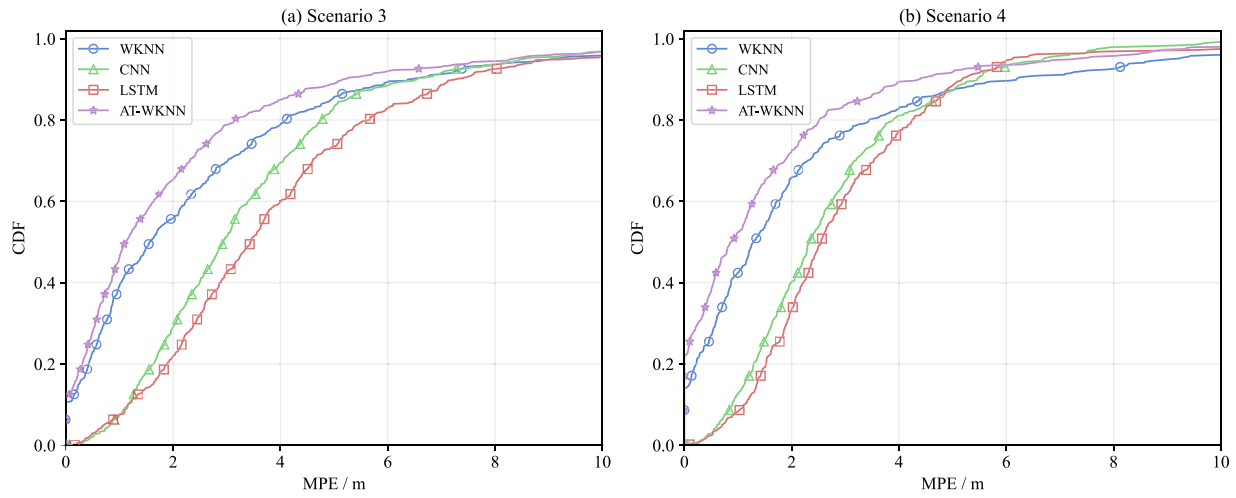


Figure 10: CDF curves of four localization algorithms in Scenarios 3 and 4.

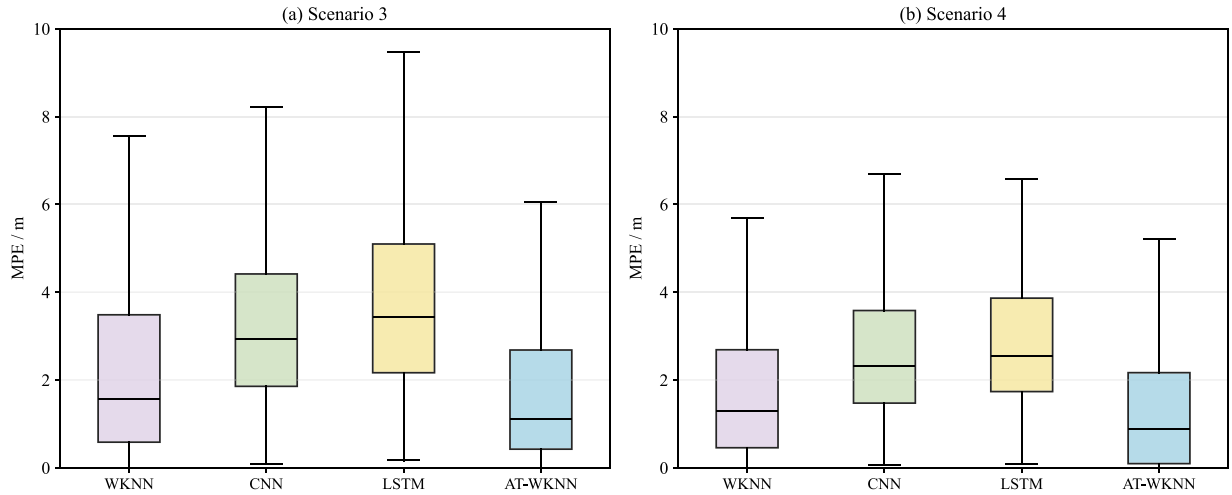


Figure 11: Boxplot comparison of four positioning algorithms in Scenarios 3 and 4.

Based on the comprehensive analysis results of the mean position error, cumulative distribution of errors, and dispersion of error distribution, the proposed AT-WKNN algorithm demonstrates significant performance advantages in complex and dynamic indoor positioning scenarios, as well as in large-scale benchmark environments. The consistent superiority of AT-WKNN across all four scenarios fully verifies its generalization capability and practical applicability for large-scale indoor localization deployments.

6 Conclusion

This paper presents AT-WKNN, an adaptive trajectory-aided dynamic indoor positioning algorithm based on RSS fingerprints, which enables high-precision and high-efficiency positioning in dynamic indoor environments. By integrating adaptive neighbor selection, efficient retrieval, and trajectory optimization mechanisms, the proposed approach achieves a coordinated improvement in both localization accuracy and computational efficiency. Experimental results demonstrate that the algorithm outperforms baseline methods in both positioning accuracy and runtime efficiency, while maintaining strong robustness against

device imperfections, environmental variations, and human interference. The method requires no additional hardware, featuring low cost and low computational complexity, making it well-suited for large-scale and energy-sensitive IoT deployments.

Furthermore, experiments conducted on the large-scale UJIIndoorLoc benchmark dataset demonstrate that AT-WKNN maintains competitive performance against CNN-based and LSTM-based deep learning methods, validating its scalability and practical applicability in large-scale environments.

It should be acknowledged that the constant velocity assumption of the Kalman filter and the limited scale of self-collected scenarios are existing shortcomings of this work. Future work will explore more advanced motion models, larger deployment environments, and multi-modal sensing technologies to further enhance system performance.

Acknowledgement: We thank the State Key Laboratory of Public Big Data and the College of Computer Science and Technology at Guizhou University, the Key Laboratory of Interior Layout Optimization and Security at Chengdu Normal University, as well as the School of Software at Tsinghua University for providing the experimental environment, technical support and constructive discussions throughout this research.

Funding Statement: This work was supported by the National Natural Science Foundation of China (Grant No. 62361010), the Key Laboratory of Interior Layout Optimization and Security, Institutions of Higher Education of Sichuan Province, Chengdu Normal University, China (Research on Intelligent Pedestrian Positioning Methods for Multi-Storey Buildings, Grant No. SNKJ202504), the General Project of Guizhou Provincial Basic Research Program (Grant No. Qian-ke-he Foundation MS[2026]159), the Scientific and Technological Innovation Platform Research Project of Guizhou Province (Grant No. CXPTXM[2025] No. 024), and the Scientific and Technological Innovation Talent Team of Guizhou Province for Cybersecurity Protection of Computing Power Network (Grant No. Qiankehe Talent CXTD[2025]029).

Author Contributions: Jing Liu and Weijie Tan contributed equally to this research. Jing Liu was responsible for conceptualization, methodology design, software implementation, experimental analysis and drafting the original manuscript, while Weijie Tan provided research supervision, revised and polished the paper, and took charge of project administration and funding support. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The datasets used in this study include (1) a subset of the UJIIndoorLoc dataset, which is publicly available through the UCI Machine Learning Repository and can be accessed at <https://archive.ics.uci.edu/dataset/310/ujiiindoorloc> (see Ref. [37]), and (2) a self-collected RSS fingerprint dataset acquired in the indoor experimental environment described in this paper. The self-collected dataset is available from the corresponding author upon reasonable request.

Ethics Approval: Ethics approval and consent to participate are not applicable as this study involves only algorithm optimization and analysis of publicly available data, with no human or animal subjects involved.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Naciri N, Yi D, Bisnath S, de Blas FJ, Capua R. Assessment of Galileo High Accuracy Service (HAS) test signals and preliminary positioning performance. *GPS Solut.* 2023;27(2):73. doi:10.1007/s10291-023-01410-y.
2. Faragher R, Harle R. Location fingerprinting with bluetooth low energy beacons. *IEEE J Select Areas Commun.* 2015;33(11):2418–28. doi:10.1109/jsac.2015.2430281.
3. Ngamakeur K, Yongchareon S, Yu J, Sheng QZ. Deep CNN-LSTM network for indoor location estimation using analog signals of passive infrared sensors. *IEEE Internet Things J.* 2022;9(22):22582–94. doi:10.1109/jiot.2022.3183148.

4. Shi Q, Sun Z, Le X, Xie J, Lee C. Soft robotic perception system with ultrasonic auto-positioning and multimodal sensory intelligence. *ACS Nano*. 2023;17(5):4985–98. doi:10.1021/acsnano.2c12592.s001.
5. Waleed Pasha M, Yasir Umair M, Mirza A, Rao F, Wakeel A, Akram S, et al. Enhanced fingerprinting based indoor positioning using machine learning. *Comput Mater Contin*. 2021;69(2):1631–52. doi:10.32604/cmc.2021.018205.
6. Luo Z, Li W, Wu Y, Dong H, Bian L, Wang W. Accurate indoor localization for bluetooth low energy backscatter. *IEEE Internet Things J*. 2025;12(2):1805–16. doi:10.1109/jiot.2024.3468428.
7. Yu D, Li C. An accurate WiFi indoor positioning algorithm for complex pedestrian environments. *IEEE Sens J*. 2021;21(21):24440–52. doi:10.1109/jsen.2021.3113376.
8. J NSC, Wahab NHA, Sunar N, Ariffin SHS, Wong KY, Aun Y. Indoor positioning system: a review. *Int J Adv Comput Sci Appl*. 2022;13(6):477–90. doi:10.14569/ijacsa.2022.0130659.
9. Li Z, Xu P, Gao Q, Li C, Tan W. Reconfigurable intelligent surface-aided secret key generation using auto-encoder and K-means quantization. *Front Inform Technol Electron Eng*. 2025;26(8):1486–500. doi:10.1631/fitee.2400799.
10. Tan W, Gao Q, Li Z, Liu Z, Li C, Xu G. Physical layer key generation based on SOM and K-means clustering quantization for TDD-OFDM systems over unstable channels. *J King Saud Univ Comput Inf Sci*. 2025;37(8):221. doi:10.1007/s44443-025-00245-6.
11. Bahl P, Padmanabhan VN. RADAR: an in-building RF-based user location and tracking system. In: *Proceedings of the IEEE INFOCOM 2000*; 2000 Mar 26–30; Tel Aviv, Israel. p. 775–84.
12. Adiyatma FYM, Cherntanomwong P. Radio map augmentation using DBSCAN and KNN regression for improved indoor positioning. In: *Proceedings of the 2024 19th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP)*; 2024 Nov 11–15; Chonburi, Thailand. p. 1–5.
13. Matharia AS, Kaity S, Das L, Bharti J. Improvement of position measurement by TDOA technique using K-means clustering. In: *Proceedings of the 2023 3rd International Conference on Range Technology (ICORT)*; 2023 Feb 23–25; Balasore, India. p. 1–6.
14. Abbas HA, Boskany NW, Ghafoor KZ, Rawat DB. Wi-Fi based accurate indoor localization system using SVM and LSTM algorithms. In: *Proceedings of the 2021 IEEE 22nd International Conference on Information Reuse and Integration for Data Science (IRI)*; 2021 Aug 10–12; Las Vegas, NV, USA. p. 416–22.
15. Maung NAM, Lwi BY, Thida S. An enhanced RSS fingerprinting-based wireless indoor positioning using random forest classifier. In: *Proceedings of the 2020 International Conference on Advanced Information Technologies (ICAIT)*; 2020 Nov 4–5; Yangon, Myanmar. p. 59–63.
16. Jang JW, Hong SN. Indoor localization with WiFi fingerprinting using convolutional neural network. In: *Proceedings of the 2018 Tenth International Conference on Ubiquitous and Future Networks (ICUFN)*; 2018 Jul 3–6; Prague, Czech Republic. p. 753–8.
17. Han S, Zhao C, Meng W, Li C. Cosine similarity-based fingerprinting algorithm in WLAN indoor positioning against device diversity. In: *Proceedings of the 2015 IEEE International Conference on Communications (ICC)*; 2015 Jun 8–12; London, UK. p. 2710–4.
18. Kaushal H, Jiang A, Prodanoff Z, Shaw W, Costner J, Hayes C. Improving accuracy of beacon-based indoor positioning system using weighted adaptive KNN algorithm. In: *Smart cities and sustainable manufacturing*. Amsterdam, The Netherlands: Elsevier; 2025. p. 77–95.
19. Li D, Zhang B, Li C. A feature-scaling-based k-nearest neighbor algorithm for indoor positioning systems. *IEEE Internet Things J*. 2016;3(4):590–7. doi:10.1109/glocom.2014.7036847.
20. Yang L, Chen H, Cui Q, Fu X, Zhang Y. Probabilistic-KNN: a novel algorithm for passive indoor-localization scenario. In: *Proceedings of the 2015 IEEE 81st Vehicular Technology Conference (VTC Spring)*; 2015 May 11–14; Glasgow, UK. p. 1–5.
21. Li C, Qiu Z, Liu C. An improved weighted K-nearest neighbor algorithm for indoor positioning. *Wirel Pers Commun*. 2017;96(2):2239–51. doi:10.1007/s11277-017-4295-z.
22. Liu Z, Luo X, He T. Indoor positioning system based on the improved W-KNN algorithm. In: *Proceedings of the 2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*; 2017 Mar 25–26; Chongqing, China. p. 1355–9.

23. Peng X, Chen R, Yu K, Ye F, Xue W. An improved weighted K-nearest neighbor algorithm for indoor localization. *Electronics*. 2020;9(12):2117. doi:10.3390/electronics9122117.
24. Wang B, Gan X, Liu X, Yu B, Jia R, Huang L, et al. A novel weighted KNN algorithm based on RSS similarity and position distance for Wi-Fi fingerprint positioning. *IEEE Access*. 2020;8:30591–602. doi:10.1109/access.2020.2973212.
25. Mirowski P, Steck H, Whiting P, Palaniappan R, MacDonald M, Ho TK. KL-divergence kernel regression for non-Gaussian fingerprint-based localization. In: *Proceedings of the 2011 International Conference on Indoor Positioning and Indoor Navigation*; 2011 Sep 21–23; Guimaraes, Portugal. p. 1–10.
26. Niu J, Wang B, Shu L, Duong TQ, Chen Y. ZIL: an energy-efficient indoor localization system using ZigBee radio to detect WiFi fingerprints. *IEEE J Select Areas Commun*. 2015;33(7):1431–42. doi:10.1109/jsac.2015.2430171.
27. Zhang H, Wang Z, Ni Y, Xia W, Zhao H. Improved KNN algorithm with historical information fusion for indoor positioning. In: *Proceedings of the 2021 IEEE 4th International Conference on Electronics and Communication Engineering (ICECE)*; 2021 Dec 17–19; Xi'an, China. p. 184–8.
28. Cheng C, Guan J, Wang C, Wang P, Shi F. An improved indoor positioning method based on received signal strengths. In: *Proceedings of the 2022 14th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*; 2022 Jan 15–16; Changsha, China. p. 317–21.
29. Leng Y, Huang F, Tan W. A Wi-Fi fingerprint positioning method based on RLWKNN. *IEEE Sens J*. 2025;25(1):1706–15. doi:10.1109/jsen.2024.3476335.
30. Shi S, Sigg S, Chen L, Ji Y. Accurate location tracking from CSI-based passive device-free probabilistic fingerprinting. *IEEE Trans Veh Technol*. 2018;67(6):5217–30. doi:10.1109/tvt.2018.2810307.
31. Zhou R, Tang M, Gong Z, Hao M. FreeTrack: device-free human tracking with deep neural networks and particle filtering. *IEEE Syst J*. 2020;14(2):2990–3000. doi:10.1109/jsyst.2019.2921554.
32. Choi J. Sensor-aided learning for Wi-Fi positioning with beacon channel state information. *IEEE Trans Wireless Commun*. 2022;21(7):5251–64.
33. Sun Y, Zhong Y, Hu C, Xiong A, Zhao H. WiFi indoor positioning and tracking algorithm based on compressive sensing and sage-husa adaptive Kalman filter. *Open J Appl Sci*. 2024;14(2):379–90. doi:10.4236/ojapps.2024.142026.
34. Zhang Z, Lee M, Choi S. Deep-learning-based Wi-Fi indoor positioning system using continuous CSI of trajectories. *Sensors*. 2021;21(17):5776. doi:10.3390/s21175776.
35. Hazzazi MM, Shukla PK, Shukla PK, Bassfar Z, Aljaedi A, Shah MA. A secure and high-accuracy indoor positioning system using deep temporal fusion transformer and Galois field encryption. *J Big Data*. 2026;13(1):35. doi:10.1186/s40537-025-01356-7.
36. Malkov YA, Yashunin DA. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE Trans Pattern Anal Mach Intell*. 2020;42(4):824–36. doi:10.1109/tpami.2018.2889473.
37. Torres-Sospedra J, Montoliu R, Martinez-Uso A, Avariento JP, Arnau TJ, Benedito-Bordonau M, et al. UJIIndoor-Loc: a new multi-building and multi-floor database for WLAN fingerprint-based indoor localization problems. In: *Proceedings of the 2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*; 2014 Oct 27–30; Busan, Republic of Korea. p. 261–70.