

Article

Not peer-reviewed version

IoT-Dynamic Indoor Localization Leveraging Transfer Learning Techniques

[Feiyu Lin](#) and [Shunqing Zhang](#) *

Posted Date: 15 April 2026

doi: 10.20944/preprints202604.0976.v1

Keywords: indoor localization; bluetooth low energy; transfer learning; adaptive fingerprint database



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a [Creative Commons CC BY 4.0 license](#), which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Article

IoT-Dynamic Indoor Localization Leveraging Transfer Learning Techniques

Feiyu Lin  and Shunqing Zhang * 

School of Communication and Information Engineering, Shanghai University

* Correspondence: shunqing@shu.edu.cn

Abstract

With the rapid growth of location-based services (LBS) in the Internet of Things (IoT), fingerprint-based indoor localization has attracted attention for its high accuracy. However, environmental changes degrade signal stability, and traditional methods require frequent site surveys, leading to high labor and time costs. In response, we propose an adaptive Bluetooth Low Energy (BLE) indoor localization system that relies on an updated fingerprint to address these issues. We integrate the Domain Adaptation Localization (DALoc) method into the system. The framework combines historical data with deep transfer learning to extract features and update fingerprints based on a small amount of labeled data at a new time. To enhance the adaptability of the Received Signal Strength (RSS), we utilize historically collected RSS data to fit a K-order Gaussian mixture model (GMM). Furthermore, we assess the system's performance using the Cramér-Rao lower bound (CRLB) to ensure reliability and robustness. The DALoc approach helps address the challenges posed by mixed and time-varying signals. We conducted multiple sets of experiments related to positioning error in the laboratory corridor, and the results demonstrate that the system's location accuracy exceeds 70% when tested with dynamic signals, with a location error within the meter-level range.

Keywords: indoor localization; bluetooth low energy; transfer learning; adaptive fingerprint database

1. Introduction

Location-based services (LBS) are often associated with the massive deployment of Internet-of-Things (IoT) applications. With highly accurate real-time localization techniques, user equipment (UE) can provide several unmanned functions, including robot collaboration [1] and autonomous driving [2–5]. With Global Navigation Satellite Systems (GNSS) and inertial guidance, meter-level localization accuracy is possible for outdoor IoT applications. However, more precise indoor location is desirable, such as smart healthcare and in-home asset tracking. It is still challenging to provide this type of LBS, where the navigation signals heavily attenuate due to the building blockage effects [6].

The researchers recently exploited various wireless communication systems to construct new navigation signals. For instance, the signal propagation of cellular network and wireless local area network considered in [7,8] and the achievable localization accuracy is around several meters. If sufficient and expensive infrastructure support for Ultra-wideband (UWB) pulse generation is available, UWB signals have been proposed in [9] to reach centimeter-level accuracy. Other types of wireless signals, including radio frequency identification, Zigbee, or magnetic field variations, have been proposed in [10–12], which achieves meter level localization results as well. Many scenes adopt Bluetooth Low Energy (BLE) in modern mobile devices to balance the infrastructure cost and operating energy consumption. Moreover, it also has been regarded as a high-precision localization technique, which is more feasible than other wireless communication systems [13].

Despite different types of wireless navigation signals, as mentioned before, the localization techniques can be mainly summarized into two categories, namely *geometric-based* and *fingerprint-based* schemes. The geometric-based localization approaches exploit the geometric relations between the

localization entity and the reference stations, either through the simple cell proximity[14,15], the triangulation with propagation parameters[16,17], or even a hybrid of them[18]. The fingerprint-based mechanism, however, constructs a *radio map* with location-labeled signal features in the offline stage and compares it with the real-time observations to identify the locations online[19]. Generally speaking, with advanced pattern recognition and deep learning methods, high localization accuracy can be attained using the fingerprint-based scheme, which has attracted a tremendous amount of research attention recently[20].

One of the most critical components behind the highly accurate fingerprint-based localization scheme is the timely and exact signal features to construct the real-time radio map. Due to the time-varying scattering and reflection environments, the localization accuracy is shown to gradually degrade over time for many types of wireless navigation signals[21]. In this paper, we initiate a BLE fingerprint-based localization system to study the temporal correlations of the fingerprint database. We propose a novel approach to construct a self-evolving fingerprint database by applying the transfer learning-based algorithm, which achieves sub-meter level accuracy in the prototype localization systems using off-the-shelf BLE beacons.

2. Related Works

Fingerprint-based localization usually contains two stages, including *offline database construction* and *online feature matching*. Generally speaking, during the first stage, the positioning area is partitioned into multiple grids, and a significant amount of RSS from access points must be gathered at the center of each grid. These collected RSSs are subsequently organized into a fingerprint database or radio map. In the second stage, the localization process for the target device commences with the collection of an observation vector. An identification algorithm is then employed to correlate the observation vector with the fingerprint database, yielding the final positioning outcome. The offline costs associated with gathering the fingerprint database during the offline stage are quite substantial. Therefore, it holds great significance to reconstruct a high-precision fingerprint database.

The existing research efforts focus on reducing the database construction overhead as well as improving the online localization accuracy [22–26]. As an example, [22] lists common fingerprint database construction methods and compares the most popular online localization schemes, including K-nearest neighbor (KNN) algorithms [23], support vector machine (SVM) [24], and other deep learning-based methods [25]. The above methods require a fresh fingerprint database, such as reported in [26], and the localization accuracy usually decreases when the fingerprint database becomes outdated.

In order to compensate for the above issues, the current research efforts focus on improving the robustness of the fingerprint database. For instance, crowdsourcing has been proposed in [27] to periodically update the fingerprint database, which relies on volunteer users to report their real-time fingerprints and locations regularly. Gong et al. [28] proposed an enhanced indoor localization solution that leverages spatial contextual knowledge extracted from sparse dynamic fingerprints. Zhou et al. [29] developed a novel approach that integrates the empirical mode decomposition threshold smoothing method (EMDT) and improved weighted K nearest neighbor (WKNN) to address signal fluctuations. These methods usually guarantee localization accuracy at the expense of reporting efforts.

Concerning another type of scheme, many researchers proved transfer learning methods to be an efficient method to reduce the labeling cost by transferring the knowledge from the source domain to the target domain in the computer vision tasks [30]. As a problem in transfer learning, domain adaptation addresses the issue of having data from two related domains but under different distributions, and the fingerprint data from different periods are viewed as two related domains with other distributions. Recently, some domain adaptation methods, like transfer component analysis (TCA) [31] and joint distribution adaptation (JDA) [32] have been proposed to mitigate the impact of RSS variation. However, most existing methods show poor performance in extreme RSS variations¹

¹ The variations indicate that the range of RSS values under consideration shifts from $[-80dBm, -50dBm]$ to $[-100dBm, -50dBm]$.

caused by changing environments and heterogeneous hardware. Most methods learn new shallow representation features, which slightly reduces the domain discrepancy. The deep neural networks can learn in-depth transferable features to manifest invariant factors underlying different domains, making hidden representations robust to noise [33]. Thus, applying deep neural networks to domain adaptation methods can learn more transferable features.

3. Contributions and Organization

The contributions of this paper contain the following three parts: We formulate a BLE localization problem by considering the correlations² of the fingerprint database, which has not been addressed in the existing literature.

Inspired by the transfer learning framework, we propose a novel approach to convert the original error minimization problem to the maximum mean discrepancy (MMD) minimization problem by exploiting the correlations. Moreover, we also derive a low-complexity localization method thereafter. We analyze the localization accuracy concerning the time-varying parameters to determine the potential application scenarios for the proposed algorithm. Although we cannot cover all types of wireless navigation signals, we hope the proposed scheme can shed some light on the future fingerprint-based localization system design.

The rest of this article is structured as follows. We present the system model for BLE indoor localization in Section 4 and discuss the learning-based problem formulation in Section 5. In Section 7, we propose MMD minimization-oriented scheme and analyze the corresponding performance in Section 8. Numerical examples are demonstrated in Section 9 and we finally conclude this paper in Section 10.

4. System Model

In this section, we briefly describe the topology model, the BLE signal propagation model, and the construction of different fingerprint databases.

4.1. Topology & BLE Channel Models

Consider a BLE localization system with N_i iBeacons uniformly distributed along the corridor as shown in Figure 1, and the corresponding locations are denoted by $\mathcal{L}^1, \dots, \mathcal{L}^{N_i}$. Within the entire space $\mathcal{A}$ defined by this corridor, we uniformly select N_{RP} reference points (RPs) to construct the fingerprint database $\mathcal{DB}$, where their locations are given by $\mathcal{L}_{RP}^1, \dots, \mathcal{L}_{RP}^{N_{RP}}$, respectively. The localization entity, with the target location $\mathcal{L}_m$, collects the received signals from N_i iBeacons and estimates its location by comparing them with the fingerprint database, e.g., $\mathcal{DB}$.

Given a time slot t , the received BLE signal vector at the localization entity, $\mathbf{y}(\mathcal{L}_m, t) \in \mathbb{C}^{N_i \times 1}$, is modeled as,

$$\mathbf{y}(\mathcal{L}_m, t) = \mathbf{h}(\mathcal{L}_m, t)x + \mathbf{n}(\mathcal{L}_m, t), \quad (1)$$

where x and $\mathbf{h}(\mathcal{L}_m, t) = [h^1(\mathcal{L}_m, t), \dots, h^{N_i}(\mathcal{L}_m, t)]^T \in \mathbb{C}^{N_i \times 1}$ denote the transmitted beacon signals with normalized power, and the channel conditions between iBeacons and the localization entity, respectively. $\mathbf{n}(\mathcal{L}_m, t) = [n^1(\mathcal{L}_m, t), \dots, n^{N_i}(\mathcal{L}_m, t)]^T \in \mathbb{C}^{N_i \times 1}$ is additive white Gaussian noise (AWGN) with zero mean and unit variance. The received signal strength (RSS), e.g. $\mathbf{r}(\mathcal{L}_m, t) = [r^1(\mathcal{L}_m, t), \dots, r^{N_i}(\mathcal{L}_m, t)]^T \in \mathbb{C}^{N_i \times 1}$, is thus given by,

$$\mathbf{r}(\mathcal{L}_m, t) = \mathbf{y}(\mathcal{L}_m, t) \odot \mathbf{y}^*(\mathcal{L}_m, t), \quad (2)$$

² Our analysis included examining the wireless signal's impact from both the signal itself and the temporal correlation [34–36] between fingerprint databases. To better represent the stable and instantaneous wireless signal, we used the GMM estimator and analyzed the time domain correlation of RSS-GMM, which can be expressed as an exponential covariance function. We conducted simulations on the positioning results obtained from the constructed fingerprint database and added simulation experiments to verify the impact of long time spans.

where $\odot$ denotes the element-wise multiplication, and $(\cdot)^*$ denotes the complex conjugate operation³.

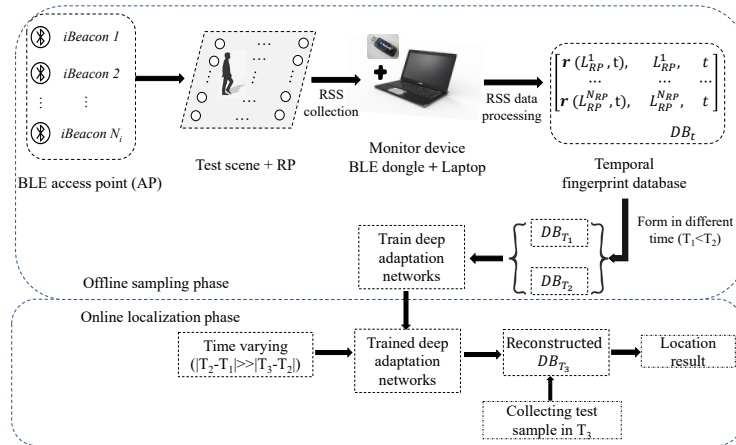


Figure 1. The BLE localization system framework.

4.2. Fingerprint Database

By collecting RSS information of N_{RP} RPs, we can form a time-varying BLE signal fingerprint database, $\mathcal{DB}_t$, for a given time slot t as follows,

$$\mathcal{DB}_t = \begin{bmatrix} \mathbf{r}(\mathcal{L}_{RP}^1, t), & \mathcal{L}_{RP}^1, & t \\ \dots & \dots & \dots \\ \mathbf{r}(\mathcal{L}_{RP}^{N_{RP}}, t), & \mathcal{L}_{RP}^{N_{RP}}, & t \end{bmatrix}. \quad (3)$$

In the practical systems, we can regularly update the fingerprint database by measuring the instantaneous RSS of N_{RP} RPs. Without loss of generality, we choose T_1 and T_2 to be the closest database updating time stamps and T_3 localization time. Therefore, we rely on measured fingerprint databases $\mathcal{DB}_{T_1}$ and $\mathcal{DB}_{T_2}$ to infer the instantaneous database $\mathcal{DB}_{T_3}$ and perform the localization process afterward.

The following assumptions are made throughout the rest of this paper. First, we assume $T_2 - T_1 \gg T_3 - T_2$ ⁴, since the fingerprint database can not be updated very frequently. Secondly, we apply a time-varying K -order Gaussian mixture model (GMM) as explained in [37–39], e.g.,

$$\tilde{\mathbf{r}}(\mathcal{L}, t) = C_{RSS} \sum_{k=1}^K \alpha_k(t) \mathcal{N}(\mu_k(t), \Sigma_k(t)), \quad (4)$$

to fit the collected RSSs in the database $\mathcal{DB}_t$, such that for all $\mathcal{L}_m$ and t , $\|\mathbf{r}(\mathcal{L}_m, t) - \tilde{\mathbf{r}}(\mathcal{L}_m, t)\| \leq \xi$. Last but not least, we assume the GMM consistency holds true for T_3 as discussed in [40–42].

5. Problem Formulation

This section will formulate the mean square error (MSE) minimization problem and then derive the equivalent form for MMD minimization.

³ In the practical systems, we often apply a time averaging operation to obtain more smooth RSS values, e.g., $\mathbf{r}(\mathcal{L}_m, t) = \mathbb{E}_{t' \in [t, t+\Delta t]} [\mathbf{y}(\mathcal{L}_m, t') \odot \mathbf{y}^*(\mathcal{L}_m, t')]$, and we abuse the notation t here for simplicity.

⁴ We analyzed wireless channel propagation, which involves random signal changes and variations in reflective surfaces and reflectors along the path. As a result, the received signal power exhibits both nonlinear and random characteristics. To address this, we approximated the nonlinear wireless signals using the Taylor series. This allowed us to convert the problem into a linear one within the range of T_2 and T_3 .

5.1. MSE Minimization Problem

Denote $\mathcal{F}(\cdot)$ to be the mapping function from the instantaneous (e.g., $t = T_3$) measured RSS, $\mathbf{r}(\mathcal{L}_m, T_3)$, and the estimated location $\hat{\mathcal{L}}_m$. The initial MSE minimization problem can be formulated as follows.

Problem 1 (MSE Minimization). *The MSE minimized strategy, $\mathcal{F}^*(\cdot)$, can be determined by solving the following optimization problem.*

$$\underset{\mathcal{F}(\cdot)}{\text{minimize}} \quad \frac{1}{M} \sum_{m=1}^M \|\mathcal{L}_m - \hat{\mathcal{L}}_m\|_2^2, \quad (5)$$

$$\text{subject to} \quad \hat{\mathcal{L}}_m = \mathcal{F}(\mathbf{r}(\mathcal{L}_m, T_3), \mathcal{DB}_{T_1}, \mathcal{DB}_{T_2}), \quad (6)$$

$$\hat{\mathcal{L}}_m, \mathcal{L}_m \in \mathcal{A}, \forall m \in [1, \dots, M], \quad (7)$$

where $\|\cdot\|_2$ denotes the Euclidean norm as specified in [43] and M is the total number of localization rounds.

The above optimization problem is challenging to solve since the searching space contains all the possible mathematical functions. In order to reduce the implementation complexity of the function $\mathcal{F}(\cdot)$, we decompose the entire MSE minimization into two stages, including the instantaneous database inference and the standard KNN algorithm⁵ based on the inferred database $\widehat{\mathcal{DB}}_{T_3}$. Mathematically, we have the following transferred MSE minimization problem.

Problem 2 (Transferred MSE Minimization). *With the decomposed localization structure, the transferred MSE minimization problem is given by,*

$$\underset{\mathcal{F}^2(\cdot)}{\text{minimize}} \quad \frac{1}{M} \sum_{m=1}^M \|\mathcal{L}_m - \hat{\mathcal{L}}_m\|_2^2, \quad (8)$$

$$\text{subject to} \quad \hat{\mathcal{L}}_m = \mathcal{F}^1(\mathbf{r}(\mathcal{L}_m, T_3), \widehat{\mathcal{DB}}_{T_3}), \quad (9)$$

$$\hat{\mathcal{L}}_m, \mathcal{L}_m \in \mathcal{A}, \forall m \in [1, \dots, M], \quad (10)$$

$$\widehat{\mathcal{DB}}_{T_3} = \mathcal{F}^2(\mathcal{DB}_{T_1}, \mathcal{DB}_{T_2}), \quad (11)$$

where $\mathcal{F}^1(\cdot)$ denotes the standard KNN localization scheme, and $\mathcal{F}^2(\cdot)$ denotes the migration function from the existing databases $\mathcal{DB}_{T_1}$, and $\mathcal{DB}_{T_2}$ to the inferred database $\widehat{\mathcal{DB}}_{T_3}$.

In the above formulation, the optimization framework is still difficult to solve, since there is a limited design guideline for the database migration function $\mathcal{F}^2(\cdot)$. Following the aforementioned time-varying GMM model, we define the ill-conditioned function $\Phi(\cdot)$ to be the mapping relations between the measured RSSs and the fitted RSSs, e.g., $\tilde{\mathbf{r}}(\mathcal{L}_m, t) = \Phi(\mathbf{r}(\mathcal{L}_m, t))$, and the entire migration function $\mathcal{F}^2(\cdot)$ can thus be obtained by learning the time-varying relation of the GMM models and approximating the function $\Phi(\cdot)$ through the following MMD minimization.

6. MMD Minimization Problem

In order to obtain the mapping function $\Phi(\cdot)$, we first define the temporal MMD to describe the differences between two fingerprint databases and propose the corresponding MMD minimization problem as follows.

Definition 1 (Temporal MMD). *Without loss of generality, we define the temporal MMD between two different fingerprint databases to be the discrepancy of two time-varying GMMs in the Reproducing Kernel*

⁵ The KNN algorithm will compare the measured results with the database to determine K nearest RPs and calculate the location thereby to obtain the estimated location $\mathcal{L}_m$, which is shown to be quite effective for practical systems[44].

Hilbert Space (RKHS). Mathematically, the temporal MMD of $\mathcal{DB}_{T_1}$ and $\mathcal{DB}_{T_2}$, e.g., $\mathcal{M}(\mathcal{DB}_{T_1}, \mathcal{DB}_{T_2})$, is given by

$$\mathcal{M}(\mathcal{DB}_{T_1}, \mathcal{DB}_{T_2}) = \left\| \sum_{i=1}^{N_{RP}} \Phi(\mathbf{r}(\mathcal{L}_{RP}^i, T_1)) - \sum_{j=1}^{N_{RP}} \Phi(\mathbf{r}(\mathcal{L}_{RP}^j, T_2)) \right\|_{\mathcal{H}}^2. \quad (12)$$

Problem 3 (MMD minimization). The ill-conditioned function $\Phi(\cdot)$ can be obtained by solving the following MMD minimization problem, which is given by,

$$\underset{\Phi(\cdot)}{\text{minimize}} \quad \mathcal{M}(\mathcal{DB}_{T_1}, \mathcal{DB}_{T_2}), \quad (13)$$

$$\text{subject to} \quad \tilde{\mathbf{r}}(\mathcal{L}_m, T_1) = \Phi(\mathbf{r}(\mathcal{L}_m, T_1)), \quad (14)$$

$$\tilde{\mathbf{r}}(\mathcal{L}_m, T_2) = \Phi(\mathbf{r}(\mathcal{L}_m, T_2)). \quad (15)$$

Lemma 1. Problem 3 and Problem 2 are equivalent.

Proof of Lemma 1. Please refer to Appendix A for the proof. $\square$

The aforementioned MMD minimization problem overcame the challenge of fitting the mapping function $\Phi(\cdot)$, which was not resolved. As a result, we attempt to integrate the underlying feature distributions of the temporal fingerprint databases to explain the solution practicability in section 8.

7. Proposed MMD Minimization Scheme

In this section, we propose a domain adaptation localization (DALoc) method based on solving the MMD minimization problem as defined in Problem 3. To be more specific, we approximate the nonlinear mapping function $\Phi(\cdot)$ via deep domain adaptation networks, and the corresponding framework is depicted in Figure 1.

7.1. Deep Domain Adaptation Networks

In order to shrink the updated cycle of the fingerprint database, the system considers domain adaptation of transfer learning. For the domain adaptation, the probability distributions of the source and target domains are mapped in the same space, and the discrepancy between the probability distributions is minimized. Intuitively, acquiring the mapping function is the primary task of domain adaptation. Therefore, minimizing the discrepancy between the fingerprint databases mapped in the same space is the ultimate goal. Concerning the domain adaptation networks, the input is the two fingerprint databases at time intervals, and the output is the transferable feature between the fingerprint databases.

Our primary objective is to establish a mapping function between location coordinates and fingerprint features in the database. To accomplish this, we use labeled wireless signals as input and produce likelihood probability distributions of position coordinates as output. Initially, we use DCNN for feature extraction and dimensionality reduction. However, since the convolutional layers only learn the general features of the related dataset, the unique features of the data can only emerge when the network is deep enough. Therefore, we add MMD domain adaptation at the end and output the classification results through a fully connected network. In our practical system, we collect RSS at the reference point during T_1 and T_2 to construct the corresponding location fingerprint databases, $\mathcal{DB}_{T_1}$ and $\mathcal{DB}_{T_2}$. We then update the fingerprint database and network parameters by training the neural network. We evaluate the algorithm's ability to build the fingerprint database by using unlabeled T_3 data as test data and passing it through the trained network model.

Inspired by the conventional transfer learning technique as illustrated in [45], we design the neural network architecture as shown in Figure 2, where the deep convolutional neural networks (DCNN) are applied to extract the high dimensional transferable features for the source domain $\mathcal{DB}_{T_1}$ and the target domain $\mathcal{DB}_{T_2}$. Furthermore, with two identical neural network architectures for the

source domain $\mathcal{DB}_{T_1}$ and the target domain $\mathcal{DB}_{T_2}$, we can minimize the temporal MMD by establishing the following discrepancy loss function,

$$\mathbb{L}_m = \mathcal{M}(\mathcal{DB}_{T_1}, \mathcal{DB}_{T_2}), \quad (16)$$

We derive the mapping function $\Phi(\cdot)$ by obtaining transferable features after training with the temporal MMD. We can predict the RSS data based on the relationship between the mapping function and the actual data in problem 3, we can predict the RSS data. Then, we employ the classification layer to train each label to predict the corresponding coordinate task. The reconstruction process employs layers of transferable features extracted by DCNN to construct renewed labels and coordinates. Therefore, for the loss function of the classification layer, we regard the cross-entropy to illustrate the probability distribution between the estimated output probability $\mathbf{g}_{ip}$ and the ground-truth label $\hat{\mathbf{g}}_{ip}$, which can be represented by,

$$\mathbb{L}_c = - \sum_{i=1}^{N_{RP}} \mathbf{g}_{ip} \log \hat{\mathbf{g}}_{ip}. \quad (17)$$

The objective is to train a model that minimizes the classification and the temporal MMD loss function. Therefore, when the general loss function is minimized, the total loss function can be established in the following mathematical form,

$$\begin{aligned} \mathbb{L}_{total} &= \mathbb{L}_c + \lambda \mathbb{L}_m \\ &= - \sum_{i=1}^{N_{RP}} \mathbf{g}_{ip} \log \hat{\mathbf{g}}_{ip} + \lambda \mathcal{M}(\mathcal{DB}_{T_1}, \mathcal{DB}_{T_2}), \end{aligned} \quad (18)$$

where λ is the weight parameter that balances the important proportions between the classification and the discrepancy loss function.

7.2. Data Collection and Processing

Compared with a significant amount of time for establishing the fingerprint database, the influence of the short-term temporal variation caused by collecting the RSS values can be ignored. Since the measured RSS in the practical systems is easily affected by multiple fading paths, and surrounding environments change. Therefore, we removed outliers due to the fluctuations in the RSS caused by environmental changes. Considering large batches of the RSS data will increase the algorithm complexity of the networks. We process the RSS measurements by the average of the quantitative RSS value as follows,

$$r^{n_i}(\mathcal{L}_m, t) = \frac{1}{N} \sum_{o=1}^N \hat{r}_o^{n_i}(\mathcal{L}_m, t). \quad (19)$$

Here, $\hat{r}_o^{n_i}(\mathcal{L}_m, t)$ is the o -th ($o \in \{1, \dots, N\}$) raw collection values at the certain $\mathcal{L}_m$ from the n_i -th ($n_i \in \{1, \dots, N_i\}$) AP and N is a fixed constant that denotes the average number.

7.3. Training Skills

We design the deep transfer learning networks in detail as follows. The proposed DCNN architecture consists of five convolutional, pooling, and FC layers. The aspiration of devising DCNN is to learn generic features that can be prone to transferable and are slightly domain-biased. Therefore, we opt to freeze in *conv1 – conv3* and fine-tune in *conv4 – conv5*. In DCNN, average pooling and fully connected (FC) layers are also utilized to output the transferable features to the one-dimensional feature vector. Since the adaptation layer is often employed to analyze the temporal MMD to minimize the domain discrepancy, we embed a lower-dimensional adaptation layer into the proposed domain adaptation

architecture and fetched the latent mapping function in the temporal domain by minimizing the temporal MMD to accomplish domain adaptation.

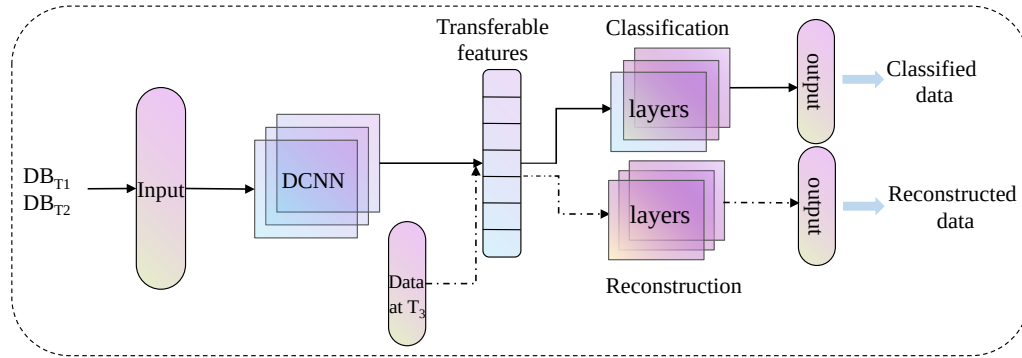


Figure 2. The architecture diagram details the deep transfer learning process combined with the DALoc system.

Furthermore, we introduce the training option parameters of the structure and configuration as depicted in Figure 2. The proposed architecture includes FC hidden layers, the number of cells in each layer, the learning rate, and the gradient update algorithm. We utilize the Rectified Linear Unit (ReLU) as an activation function in the training process. The whole training epoch should be tiny for deep domain adaptation, and optimization proceeds using the Adam optimizer for 10,000 iterations. The batch size is set to 32 for learning at each iteration. We set the number of training epochs to 10. The output layer composes neurons equal to the number of reference points in the fingerprint database. It is finally trained as a multinomial classifier by the softmax function in the output layer.

Moreover, before the neural networks were pre-trained, the global learning rate was arranged to a value of 10^{-4} . For the MMD model, a linear combination of 19 RBF kernels is used. Finally, the source and target domain data are fed to the network to calculate the final estimations, and the performance measures can be obtained. With the above neural network configuration and the loss function setting, the time domain fingerprint evolution knowledge is transferred across different time stamps by minimizing the temporal MMD. The physical interpretation is that the neural network architecture learns domain-invariant features from the discrepancies between two different fingerprint databases, which can be applied to update the fingerprint database for upcoming localization.

7.4. Reconstructed Fingerprint

The temporal RSS data and location are redrafted through the same network to produce the renewed fingerprint database. We obtain the feature vector through the DCNN feature extraction layer for the labeled data. After obtaining the feature vector, labeled or unlabeled data, we reconstruct new labeled data very close to the original data through the reconstruction layer. Similarly, we also accomplish data augmentation through adaptive layers for the labeled data.

DALoc utilizes domain-adaptive transfer learning to construct the fingerprint database. The system learns a function that can transfer localization features, and the function can learn localization knowledge from the initial fingerprint database to the time-varying fingerprint database. Under the condition that the time change is satisfied, DALoc maps the initial DB_{T_1} and the outdated DB_{T_2} fingerprint database in the time domain to the same subspace. Moreover, when the time T_3 meets $T_2 - T_1 \gg T_3 - T_2$, the system generates the split-new fingerprint database by a small number of RPs after the network is trained, thereby reducing the workload in the offline phase overhead. Finally, after acquiring a small number of labels at the new time T_3 , the positioning system automatically revises the fingerprint database to obtain the estimated position coordinates.

During the positioning phase, users can obtain location information by reporting their RSS to the server. The trained network will then update a set of network parameter vectors, which primarily consist of the likelihood probability distribution related to the user's final location. The final position

can be mathematically expressed using a vector that includes the updated probability distribution and its corresponding coordinates, as follows,

$$\hat{\mathcal{L}}_{t+1}^d = \sum_{p=1}^{N_p} \hat{g}_{t+1,p}^d \cdot \mathcal{L}_p^c \quad (20)$$

where $\mathcal{L}_c^p$ represents a set of all possible position coordinates corresponding to the likelihood probabilities $\hat{g}_{t+1,p}^d$ in the d -th target. Besides, N_p is the total number of the location time instants.

8. Performance Analysis

We conduct a draft to derive a theoretical Cramér-Rao Lower Bound (CRLB) analysis of the generalization bound for the MMD minimization problem estimation in this section. Generally, the CRLB [46] is utilized to estimate the performance in the field of localization systems. It is well known that the CRLB is defined as the lower bound for the variance matrix of the unbiased location estimator, which is represented as the inverse of the Fisher Information Matrix (FIM) [47].

8.1. CRLB for Location Error

In the localization system, the coordinate $\mathcal{L}_m = (x_m, y_m)$ and RSS $\mathbf{r}(\mathcal{L}_m, t)$ can be linked together according to the channel attenuation model of BLE in [48]. After the CRLB analysis for the RSS, it is because of the existence of this connection that we can fetch the CRLB of the location error through the transpose matrix.

As the fitted RSS $\mathbf{r}(\mathcal{L}_m, t)$ model, we denote the parameters interrelated $\tilde{\mathbf{r}}(\mathcal{L}_m, t)$ as the set for the GMM unknown parameters $\theta = \{[(\gamma_1^T, \dots, \gamma_K^T)^T]_{k=1}^K \in \mathbb{R}^{K \times 3}$ in the target location coordinate $\mathcal{L}_m = (x_m, y_m) \in \mathbb{R}^2$. Mathematically, we define the k -th GMM component of the vector γ_k as follows,

$$\gamma_k = \{\alpha_k(t), \mu_k(t), \Sigma_k(t)\}. \quad (21)$$

According to [49], the CRLB of the localization error estimation for the target location $\mathcal{L}_m$ is lower bounded by,

$$Cov_{(\hat{\mathcal{L}}_m)}(|\mathcal{L}_m - \hat{\mathcal{L}}_m|^2) = E_{\tilde{\mathbf{r}}(\mathcal{L}_m, t)}\{(\mathcal{L}_m - \hat{\mathcal{L}}_m)(\mathcal{L}_m - \hat{\mathcal{L}}_m)^T\} \geq [\{\mathbb{J}_{(\hat{\mathcal{L}}_m)}\}^{-1}]_{2 \times 2}, \quad (22)$$

where $[\cdot]_{2 \times 2}$ is denoted by the operation to the upper left sub-matrix. In addition, $\mathbb{J}_{(\hat{\mathcal{L}}_m)}$ is the FIM for the estimated parameter θ at the target location.

Corollary 1. When the error q_θ is derived in θ , we can obtain the CRLB of localization error that is defined as,

$$q_\theta = \sqrt{\text{tr}\{[\{\mathbb{J}_{(\hat{\mathcal{L}}_m)}\}^{-1}]_{2 \times 2}\}}, \quad (23)$$

where $\text{tr}\{\cdot\}$ denotes the trace of the matrix. And a lower bound on the estimation variance of the location error is provided in (23).

Proof of Corollary 1. Please refer to Appendix B for the proof. $\square$

Therefore, we can derive the CRLB of the localization error analysis at the estimated location from corollary 1.

⁶ The following K represents the meaning of K components in GMM instead of K in KNN.

8.2. The Temporal RSS Analysis

In the context of obtaining the optimal function $\Phi(\cdot)$, assuming that $\tilde{\theta}$ is the unbiased estimate θ , then its covariance matrix is,

$$\text{Cov}_{\theta}(\tilde{\theta}) = E_{\theta} \{ (\tilde{\theta} - \theta)(\tilde{\theta} - \theta)^T \}, \quad (24)$$

where $E_{\theta}(\cdot)$ is defined as the expectation operator. In the light of the CRLB definition, the covariance matrix also can be expressed as,

$$\text{Cov}_{\theta}(\tilde{\theta}) \geq \{ \mathbb{J}_{\theta} \}^{-1}. \quad (25)$$

Then according to [50], the FIM, $\mathbb{J}_{\theta}$, is depicted as,

$$\mathbb{J}_{\theta} = -E \left\{ \left[\frac{\partial \ln P(\tilde{\mathbf{r}}(\mathcal{L}_m, t); \theta)}{\partial \theta} \right]^2 \right\}. \quad (26)$$

Hence, the formula (26) can be calculated by,

$$\mathbb{J}_{\theta} = \begin{bmatrix} \mathcal{E}(\gamma_1, \gamma_1), & \cdots, & \mathcal{E}(\gamma_1, \gamma_K) \\ \vdots & \ddots & \vdots \\ \mathcal{E}(\gamma_K, \gamma_1), & \cdots, & \mathcal{E}(\gamma_K, \gamma_K) \end{bmatrix}_{K \times K}, \quad (27)$$

where $\mathcal{E}(\gamma_k, \gamma_k)$ denote the item at k -th row and k -th column in the matrix (27). And it is described as the following form,

$$\begin{aligned} \mathcal{E}(\gamma_k, \gamma_k) &= -E \left\{ \frac{[\partial \ln P(\tilde{\mathbf{r}}(\mathcal{L}_m, t); \theta)]^2}{\partial \gamma_k \partial \gamma_k^T} \right\} \\ &= - \begin{bmatrix} \mathcal{G}(\alpha_k(t), \alpha_k(t)), \mathcal{G}(\alpha_k(t), \mu_k(t)), \mathcal{G}(\alpha_k(t), \Sigma_k(t)) \\ \mathcal{G}(\mu_k(t), \alpha_k(t)), \mathcal{G}(\mu_k(t), \mu_k(t)), \mathcal{G}(\mu_k(t), \Sigma_k(t)) \\ \mathcal{G}(\Sigma_k(t), \alpha_k(t)), \mathcal{G}(\Sigma_k(t), \mu_k(t)), \mathcal{G}(\Sigma_k(t), \Sigma_k(t)) \end{bmatrix} \end{aligned} \quad (28)$$

And we denote $P(\tilde{\mathbf{r}}(\mathcal{L}_m, t); \theta)$ as the joint probability density function (PDF) of the estimated $\tilde{\mathbf{r}}(\mathcal{L}_m, t)$ conditioned on θ . Then, because the received RSS of each AP is independent, the joint PDF $P(\tilde{\mathbf{r}}(\mathcal{L}_m, t); \theta)$ can be established by a form as following form multiplied by N_i single Gaussian distribution,

$$\begin{aligned} \mathbf{L} &= P(\tilde{\mathbf{r}}(\mathcal{L}_m, t); \theta) \\ &= \prod_{n_i=1}^{N_i} P(\tilde{\mathbf{r}}^{N_i}(\mathcal{L}_m, t); \theta), \end{aligned} \quad (29)$$

where

$$\ln P(\tilde{\mathbf{r}}(\mathcal{L}_m, t); \theta) = \sum_{n_i=1}^{N_i} \alpha_k(t) \sum_{k=1}^K \left[-\frac{N_i}{2} \ln(2\pi) - \frac{1}{2} |\Sigma_k(t)| - \frac{(\tilde{\mathbf{r}}^{N_i}(\mathcal{L}_m, t) - \mu_k(t))^2}{2\Sigma_k(t)} \right] \quad (30)$$

Next, we can calculate the formula (28), each element of $\mathcal{E}(\gamma_k, \gamma_k)$ can be calculated in Appendix C.

According to the formula (27), the goal is to obtain the solution for the matrix. Combing the above results, we can achieve the covariance matrix for the estimated parameters in RKHS. So far, we can acquire the CRLB for the temporal RSS analysis, which can be expressed by,

$$\begin{aligned} CRLB_1 &= \varrho_{\hat{\theta}} \\ &= \sqrt{\text{tr}\{[\mathbb{J}_{\hat{\theta}}]^{-1}\}_{K \times K}} \\ &= \sqrt{\sum_{k=1}^K \text{tr}\{[\mathbb{J}_{\mathcal{E}(\gamma_k, \gamma_k)}]^{-1}\}} \end{aligned} \quad (31)$$

Corollary 2. Consistent inference of CRLB in the time domain: the $CRLB_2$ derived from the parameters in the time domain is consistent with the parameters $CRLB_1$ estimated by the algorithm, and the following inference is made:

$$\|CRLB_1 - CRLB_2\| \leq \zeta, \quad (32)$$

where ζ denotes a constant with a value nearly approaching zero.

Proof of Corollary 2. Please refer to Appendix D for the proof. $\square$

We mainly report on the evaluation of the proposed algorithm based on the real experimental test-bed in this part. Then, we conduct the following experiments under simulation on the performance of the reconstructed fingerprint database in the time domain. This section primarily encompasses two aspects: test-bed description and the updated fingerprint database evaluation results.

9. Results

9.1. Test-Bed Description

Test Environment and Configurations: The whole experiment scenarios are conducted in the laboratory corridor of the university. Moreover, the scene diagram is shown in the grey area of Figure 3. The grey area is divided into several squares. The blue triangle represents the symbol where to arrange APs, and the red dot shows the location of RPs.

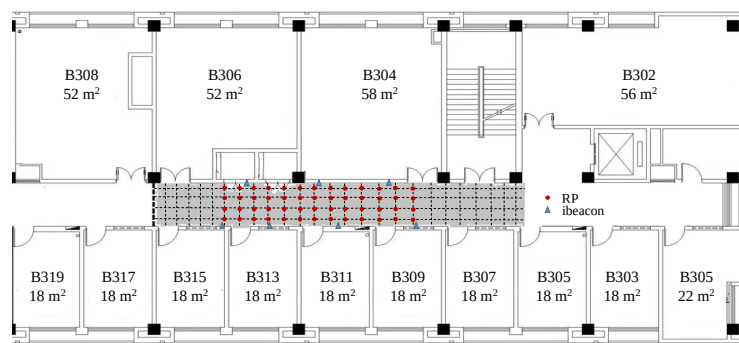


Figure 3. The gray area represents the testing scenarios in the corridor.

The system is mainly relied on manual collection to build the fingerprint database for a different time during the offline phase. We arranged the Brightstone beacons with different minors at each AP. Furthermore, the NRF52832 Dongle is connected to the computer to receive the BLE signals at RPs and test points. The collected BLE signal data are visualized in Wireshark software. The computer tests the experiments with the Intel CORE i7 8th Gen and operator system of Windows 10. We process the RSS data on MATLAB, and the training model is generated on *Pycharm*.

Time Interval Setting: We can observe the disparity between the extreme variations in RSS and the conventional range of RSS fluctuations by examining Figure 4. Some wireless signals with relatively

large fluctuations cannot be monitored, so we set the RSS to the default value of $-100dBm$ in the fingerprint database. This setting reduces the hazard of missing data due to signal fluctuations. We collected the RSS of all RP points at T_1 , T_2 , and T_3 , where $T_2 - T_1 \ll T_3 - T_2$.

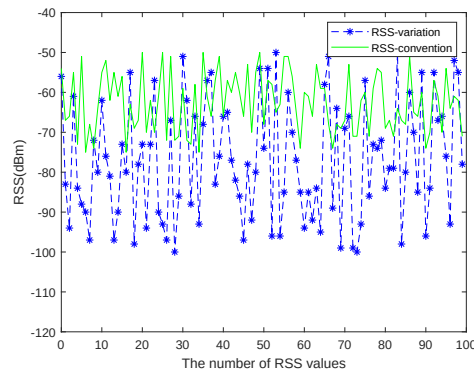


Figure 4. Comparison figure of extreme variations in RSS and conventional RSS fluctuation range.

To fully reflect the feasibility of the proposed algorithm, we randomly select the different time data at four RPs in the fingerprint database for Pearson correlation analysis, as shown in Figure 5. The correlation is highest when the time interval is one day from the Figure 5. Furthermore, we analyzed three different groups of time intervals between T_1 and T_2 : 1-day, 6-days, and 7-days, respectively.

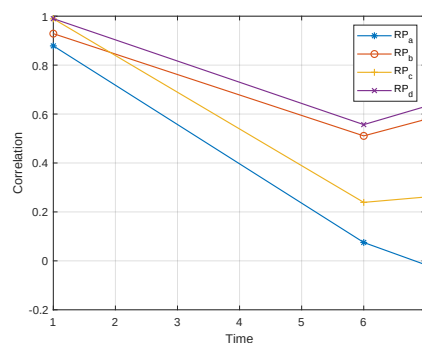


Figure 5. The figure of the temporal correlation.

Fingerprint database: The fingerprint database corresponding to time T is represented by $\mathcal{DB}_T$. We take $\mathcal{DB}_{T_1}$ and $\mathcal{DB}_{T_2}$ as the initial dataset to train the model parameters. A limited amount of RP data at time T_3 is used as the test set, and the trained network is utilized to generate the reconstructed fingerprint database $\mathcal{DB}_{T_3}$. To effectively assess the performance of the DAloc algorithm, we retain the source domain used for transfer training, and the validation set remains associated with the target-domain data of data at T_3 to update the new fingerprint database.

9.2. The Updated Fingerprint Database Evaluation Results

9.2.1. Effect of Reconstructed Fingerprint Database

To study the performance of the reconstructed database, we conducted experiments that verified the location error in T_3 . To establish multiple baselines, we have selected four different fingerprint databases. These include the original fingerprint databases collected at three different time points, namely T_1 , T_2 , and T_3 , as well as the fingerprint database generated using the traditional transfer learning JDA algorithm.

Specifically, we choose T_3 time for the test set of all updated fingerprint databases. We can observe four lines below the solid red curves: blue, purple, green, and cyan. From the results, it can be shown in Figure 6 that the database collected manually in real-time is still the most reliable and accurate. Furthermore, we have plotted the positioning error of the fingerprint database reconstructed by the

DALoc algorithm on a CDF diagram for comparative analysis. From the results in Figure 6, the DALoc algorithm in the paper is greatly improved.

The fingerprint database and network parameters were updated using marked data at T_1 , T_2 , and data at T_3 were collected to test the algorithm after networks were trained. The reconstructed fingerprint database resulted in higher positioning accuracy than JDA, $\mathcal{DB}_{T_2}$, $\mathcal{DB}_{T_1}$, with a significant difference between the blue and purple curves at 1-2 meters. After analyzing several fingerprint databases, our experiments have yielded varying levels of location error of 0.5152 m, 0.8041 m, 1.1830 m, 1.2871 m, and 1.4684 m for the five different databases ($\mathcal{DB}_{T_3}$, $\mathcal{DB}_{DALoc}$, $\mathcal{DB}_{JDA}$, $\mathcal{DB}_{T_2}$, $\mathcal{DB}_{T_1}$). The fingerprint database $\mathcal{DB}_{T_3}$ exhibited the highest positioning accuracy, followed by the DALoc algorithm. However, the JDA algorithm resulted in even lower location errors than the fingerprint database collected in the other two time periods.

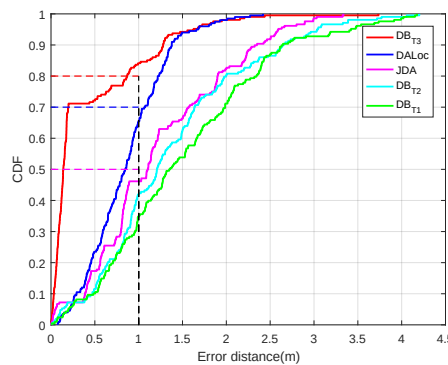


Figure 6. Comparison of DALoc against the traditional method (JDA and KNN) .

Besides, the database updated by DALoc has a probability of within one-meter location error, which is more than 70%. What is more noteworthy is that the probability of the location error within 1.5m in the fingerprint database updated by the DALoc method is more than 90%, and a satisfactory result can be obtained for the positioning error. It can also be observed from the figure that the probability of positioning results below the meter level for other fingerprint databases is less than 50%. In general, although the location accuracy of the fingerprint database updated with JDA is higher than that of the fingerprint database at the original time T_1 and T_2 , the DALoc method obtains better accuracy, which is higher than that of baseline-JDA.

Figure 7 presents the CDF of the localization error for five databases: the real-time manually collected database $\mathcal{DB}_{T_3}$, the proposed DALoc reconstruction, the methods of Hsiao et al.[19] and Huang et al.[20], and the traditional JDA algorithm. The corresponding average errors are 0.5152 m, 0.8041 m, 1.4684 m, 1.2871 m, and 1.1830 m, respectively. Among them, $\mathcal{DB}_{T_3}$ (red solid line) achieves the best performance, serving as the upper bound. The proposed DALoc (blue line) closely follows $\mathcal{DB}_{T_3}$ and significantly outperforms all other baselines across the entire error range.

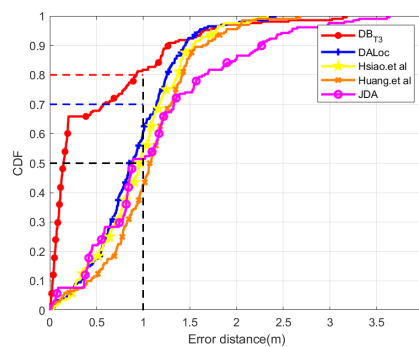


Figure 7. Comparison of DALoc against other methods (Hsiao. et al and Huang. et al) in terms of cumulative distribution of localization error.

Quantitatively, for an error threshold of 1 m, the success probabilities are: $\mathcal{DB}_{T_3}$: 95% (since at 1.0 m, CDF ≈ 0.95), DALoc: 90%, Hsiao et al.: 64%, Huang et al.: 60%, and JDA: 40%. At the 1.5 m threshold, the DALoc method reaches a probability exceeding 90% (from the data, at 1.5 m, DALoc CDF ≈ 0.94), whereas the other baselines (Hsiao et al., Huang et al., JDA) remain below 70%. For sub-meter accuracy, only $\mathcal{DB}_{T_3}$ and DALoc exceed 90% and 70%, respectively; all other methods have probabilities below 50% (e.g., at 0.5 m, DALoc: 80%, JDA: 20%, Huang et al.: 31%, Hsiao et al.: 34%). These results demonstrate that the DALoc reconstruction substantially improves localization reliability compared to conventional transfer learning and other published methods.

From an engineering perspective, considering the high labor cost of maintaining a real-time manually collected database, the DALoc-based fingerprint reconstruction offers a practical trade-off. It achieves a satisfactory average error of 0.8041 m and a 1.5-m accuracy of over 90%, making it a feasible solution for large-scale indoor localization systems.

9.2.2. Effect of Different Time Intervals

In this section, the experiment assesses the location error of the fingerprint database following the construction of various RP intervals. Three RP intervals were chosen for the experiment, namely 0.6 m, 1.2 m, and 1.8 m. Due to the limited range of the experimental site, we chose RPs that were equally spaced and evenly distributed over the location area. The localization system's performance in terms of localization error gradually deteriorates with an increasing label distance, as shown in Figure 8. As can be seen from Figure 8, the location errors are 0.8041m, 0.8381m, and 0.8921m in sequence. The results show that the localization accuracy is inversely proportional to the size of the RP interval.

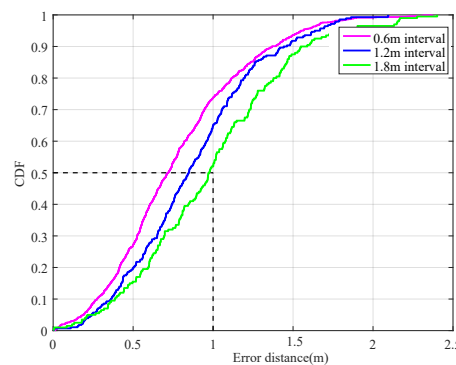


Figure 8. The influence of the regular and random RPs on location error.

9.2.3. Effect of Outliers

In Figure 9, after removing outliers from the statistical analysis, we draw the following conclusions based on the box plot. Firstly, when examining the skewness measurement of the location error distribution in the figure, we observe that the average location error from the small to the large as follows: $\mathcal{DB}_{T_3}$, $\mathcal{DB}_{DALoc}$, $\mathcal{DB}_{JDA}$, $\mathcal{DB}_{T_2}$, $\mathcal{DB}_{T_1}$. Secondly, the analysis of the number of outliers reveals that the DALoc method exhibits the most reasonable robustness in Figure 9.

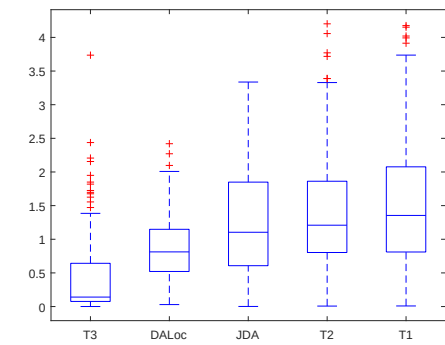


Figure 9. Boxplot of location error distribution for outlier removal.

9.2.4. Effect of Eliminating Non-Line of Sight (NLOS) Situation

This experiment section evaluates the algorithm’s feasibility in Non-Line-of-Sight (NLOS) scenarios. To mitigate the impact of NLOS and achieve higher accuracy, we have considered the utilization of a location fingerprint database approach and incorporated the use of Mean Excess Delay (MED) to detect signal instabilities in the NLOS test environment. Identifying and mitigating NLOS paths are critical factors to enhance localization accuracy in NLOS testing environments. Our approach to NLOS path recognition and filtering underscores the importance of incorporating NLOS recognition in test data to achieve superior indoor localization outcomes in NLOS scenarios. Additionally, we conducted simulation experiments to further illustrate the practicality and versatility of our proposed algorithm in NLOS situations.

In Figure 10, the dotted line represents the curve obtained using the MED method, while the solid line shows the CDF curve without this method. The results indicate that identifying and eliminating NLOS paths leads to better positioning accuracy in 2D scenarios. To evaluate our approach, we collected time intervals of 1-day, 6-day, and 7-day for migration. For ease of comparison, we calculated the average positioning errors using MATLAB and presented them in Table 1.

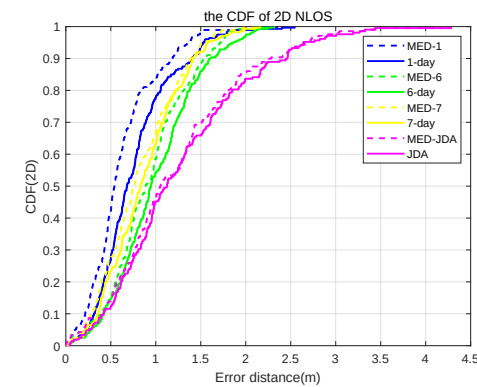


Figure 10. Comparison of MED method and Non-MED method in 2D NLOS positioning.

Table 1. The Location Error of NLOS Situations

Algorithm	Location error(m)
MED – 1	0.8041
1 – day	0.9251
MED – 6	0.9762
6 – day	1.1175
MED – 7	0.9864
7 – day	1.1403
MED – JDA	1.6490
JDA	1.6963

In conclusion, the experimental analysis yields the following conclusions: the fingerprint database is influenced in the time domain, and the DALoc algorithm demonstrates the expected efficacy for the fingerprint database.

10. Conclusions

In this paper, we proposed an adaptive indoor localization framework based on Bluetooth Low Energy (BLE) signals, which addresses the temporal degradation of fingerprint databases using deep transfer learning. Specifically, we formulated the database update problem as a maximum mean discrepancy (MMD) minimization task and developed a domain adaptation localization (DALoc) method that learns a mapping function between historical and current signal distributions. By leveraging a Gaussian mixture model (GMM) to characterize the time-varying received signal strength (RSS), the proposed system reconstructs the fingerprint database with only a small number of labeled samples at the new time instant. Experimental results in a laboratory corridor environment demonstrate that the DALoc algorithm achieves a one-meter localization accuracy with over 70% probability and an average positioning error of approximately 0.8m, significantly outperforming traditional methods such as JDA and KNN. Moreover, the system reduces the labor-intensive overhead of frequent site surveys while maintaining robust performance under dynamic signal conditions and non-line-of-sight (NLOS) scenarios. These results confirm the effectiveness of transfer learning for adaptive indoor localization in IoT environments.

Future work will integrate large language models (LLMs) and generative AI to enhance adaptability, interpretability, and generalization. Specifically, LLMs will enable semantic-aware localization through natural language priors and zero-shot alignment with visual landmarks, while generative models (e.g., diffusion models) will synthesize high-fidelity fingerprints and predict temporal RSS evolution, drastically reducing offline site survey overhead. Furthermore, LLMs will serve as intelligent orchestrators for dynamic multi-modal fusion and explainable positioning, and foundation models with prompt-based adaptation will facilitate rapid cross-scene transfer without full retraining. These advancements collectively address key challenges including costly database updates, poor robustness to environmental changes, lack of interpretability, and limited cross-domain generalization.

Author Contributions: Conceptualization, F. L. and S. Z.; methodology, F. L. and S. Z.; software, F. L. and S. Z.; validation, F. L. and S. Z.; formal analysis, F. L. and S. Z.; investigation, F. L. and S. Z.; resources, F. L. and S. Z.; data curation, F. L. and S. Z.; writing—original draft preparation, F. L. and S. Z.; writing—review and editing, F. L. and S. Z.; visualization, F. L. and S. Z.; supervision, F. L. and S. Z.; project administration, F. L. and S. Z.; funding acquisition, F. L. and S. Z.. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (NSFC) under Grants 62071284, 61871262, 61901251, and 61904101, the National Key Research and Development Program of China under Grants 2017YEF0121400 and 2019YFE0196600, the Innovation Program of Shanghai Municipal Science and Technology Commission under Grant 20JC1416400, Pudong New Area Science & Technology Development Fund, and research funds from Shanghai Institute for Advanced Communication and Data Science (SICS).

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset used for this study are available upon request to the corresponding author.

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

Appendix A. Proof of Lemma 1

We want to show that the absolute difference between the estimated $\hat{\mathbf{r}}(\mathcal{L}_m, t)$ and the real $\mathbf{r}(\mathcal{L}_m, t)$ is close to its expected value, which is subject to the restriction about $T_2 - T_1 \gg T_3 - T_2$. To this end, we prove the result based on the GMM consistency and adopt the result to the certificate that problem 3 and problem 2 are equivalent.

The result we need is an upper bound on the absolute difference between $\tilde{\mathbf{r}}(\mathcal{L}_m, T_3)$ and $\mathbf{r}(\mathcal{L}_m, T_3)$, which is near to zero. Primarily, we want to confirm the truth that the RSS-GMM consistency is correct in RKHS. Notably, [51] has precisely certificated the feasibility of the consistency in RKHS. Thence, based on existing T_1 and T_2 data, we can exploit mathematical formulas to express the GMM consistency as follows,

$$\begin{aligned}\|\mathbf{r}(\mathcal{L}_m, T_1) - \tilde{\mathbf{r}}(\mathcal{L}_m, T_1)\| &\leq \xi, \\ \|\mathbf{r}(\mathcal{L}_m, T_2) - \tilde{\mathbf{r}}(\mathcal{L}_m, T_2)\| &\leq \xi,\end{aligned}\tag{A1}$$

where $\xi \rightarrow 0$.

And we consider a mapping function $\Phi(\cdot)$ ⁷ ($\mathbb{C}^{N_i \times 1} \rightarrow \mathcal{H}$) to express the consistency. Simultaneously, the mapping function has proved the property that implies Frechet differentiability in [53]. We can depict the underlying distribution function $\Phi(\cdot)$ as follows,

$$\tilde{\mathbf{r}}(\mathcal{L}_m, t) = \Phi(\mathbf{r}(\mathcal{L}_m, t)).\tag{A2}$$

$\mathbb{C}^{N_i \times 1} \rightarrow \mathcal{H}$

Furthermore, we obtained from the Frechet differentiability of the mapping function $\Phi(\cdot)$, and the differential in $\gamma_k = \{(\alpha_k(t), \mu_k(t), \Sigma_k(t))\}$ is a linear function from $\mathbb{C}^{N_i \times 1}$ to RKHS $\mathcal{H}$. The differentiability can be mathematically expressed as,

$$d\Phi_\theta = \sum_{k=1}^K \frac{\partial \Phi}{\partial \theta}.\tag{A3}$$

Let $\frac{\partial \Phi}{\partial \theta}$ denote the vector $\left\{ \frac{\partial \Phi}{\partial \theta} \right\}_{k=1, \dots, K}$ of elements in RKHS $\mathcal{H}$. Indeed, the mapping function's properties (continuity, derivability) make it possible to minimize the distances via gradient descent approaches.

Simultaneously, the differentiability in θ can be calculated by the following formula,

$$\Phi(\cdot)_{\theta+h} = \Phi(\cdot)_\theta + \mathcal{B}h + o(h),\tag{A4}$$

where $\mathcal{B}$ is a linear operator, and $o(h) \rightarrow 0$.

According to the differentiability in equation A4, θ at T_3 can derived as

$$\theta(T_3) = \theta(T_2) + \frac{T_3 - T_2}{T_2 - T_1} [\theta(T_2) - \theta(T_1)].\tag{A5}$$

So far, according to all parameters satisfying the continuity expression, we get the fitted $\tilde{\mathbf{r}}(\mathcal{L}_m, T_3)$ also satisfies the next equation,

$$\|\mathbf{r}(\mathcal{L}_m, T_3) - \tilde{\mathbf{r}}(\mathcal{L}_m, T_3)\| \leq \xi.\tag{A6}$$

We can derive $\tilde{\mathbf{r}}(\mathcal{L}_m, T_3)$ based on the GMM consistency at T_3 . Thereby, we can acquire $\mathbf{r}(\mathcal{L}_m, T_3)$ and fulfill the up-to-data fingerprint database $\widehat{\mathcal{DB}}_{T_3}$ from the derivation process. To re-establish the fingerprint database in a real localization system, we estimate the entire fingerprint database $\widehat{\mathcal{DB}}_{T_3}$ employing the raw data for $\mathcal{DB}_{T_2}$ and $\mathcal{DB}_{T_1}$. Hence,

$$\widehat{\mathcal{DB}}_{T_3} = \Phi(\mathcal{DB}_{T_2}, \mathcal{DB}_{T_1}), \forall \Phi(\cdot) \in \mathcal{H},\tag{A7}$$

$$T_2 - T_1 \gg T_3 - T_2.\tag{A8}$$

⁷ The function is continuous and differentiable on $\mathcal{H}$ [52].

However, the bottleneck is obtaining the underlying distribution $\Phi(\cdot)$ in the above-formed solution process. In the paper, we use the temporal MMD criterion [54–57] to express the discrepancy between the two databases so that we derive the mapping distribution. Therefore, the underlying distribution function $\Phi(\cdot)$ to correspond to the database migration from the time domain can be calculated by the defined MMD minimization in problem 3. Combining the above results, we have completed the derivation.

Appendix B. Proof of Corollary 1

We rely on the relationship between $\mathbf{r}(\mathcal{L}_m, t)$ and the position $\mathcal{L}_m$ to obtain the CRLB location error.

$$\begin{aligned}\mathbb{J}_{(\mathcal{L}_m)} &= -E \left\{ \left[\frac{\partial \ln P(\hat{\mathcal{L}}_m; \mathbf{r}(\mathcal{L}_m, t))}{\partial \hat{\mathcal{L}}_m} \right]^2 \right\} \\ &= \begin{bmatrix} \mathbb{J}_{\hat{x}_m \hat{x}_m} & \mathbb{J}_{\hat{x}_m \hat{y}_m} \\ \mathbb{J}_{\hat{y}_m \hat{x}_m} & \mathbb{J}_{\hat{y}_m \hat{y}_m} \end{bmatrix}.\end{aligned}\quad (\text{A9})$$

Then the CRLB of $\mathcal{L}_m$ can be expressed by (A10) through mathematical calculation,

$$Q_{\theta} = \frac{\mathbb{J}_{\hat{x}_m \hat{x}_m} + \mathbb{J}_{\hat{y}_m \hat{y}_m}}{\mathbb{J}_{\hat{x}_m \hat{x}_m} \mathbb{J}_{\hat{y}_m \hat{y}_m} - \mathbb{J}_{\hat{x}_m \hat{y}_m}^2}, \quad (\text{A10})$$

where

$$\mathbb{J}_{\hat{x}_m \hat{x}_m} = \sum_{v=1}^{N_i} \left(\frac{\hat{x}_m - x_v}{d_{vm}^2} \right)^2, \quad (\text{A11})$$

$$\mathbb{J}_{\hat{x}_m \hat{y}_m} = \mathbb{J}_{\hat{y}_m \hat{x}_m} = \sum_{v=1}^{N_i} \left(\frac{(\hat{x}_m - x_v)(\hat{y}_m - y_v)}{d_{vm}^4} \right)^2, \quad (\text{A12})$$

$$\mathbb{J}_{\hat{y}_m \hat{y}_m} = \sum_{v=1}^{N_i} \left(\frac{\hat{y}_m - y_v}{d_{vm}^2} \right)^2. \quad (\text{A13})$$

And d_{lm} denotes the distance from $\mathcal{L}_m$ to the v -th (x_v, y_v) AP, which can be given by,

$$d_{lm} = \sqrt{(x_m - x_v)^2 + (y_m - y_v)^2}. \quad (\text{A14})$$

Appendix C. Derivation of Each Item

By expanding the matrix in 28, each item can be calculated separately as follows, the 1st derivative for the matrix element

$$\frac{\partial \ln P(\mathbf{r}(\mathcal{L}_m, t); \theta)}{\partial \alpha_k(t)} = \sum_{n_i=1}^{N_i} \sum_{k=1}^K \left[-\frac{N_i}{2} \ln(2\pi) - \frac{1}{2} |\Sigma_k(t)| - \frac{(\tilde{r}^{N_i}(\mathcal{L}_m, t) - \mu_k(t))^2}{2\Sigma_k(t)} \right]. \quad (\text{A15})$$

$$\frac{\partial \ln P(\mathbf{r}(\mathcal{L}_m, t); \theta)}{\partial \mu_k(t)} = - \sum_{n_i=1}^{N_i} \frac{\alpha_k(t) (\tilde{r}^{N_i}(\mathcal{L}_m, t) - \mu_k(t))}{\Sigma_k(t)}. \quad (\text{A16})$$

$$\frac{\partial \ln P(\mathbf{r}(\mathcal{L}_m, t); \theta)}{\partial \Sigma_k(t)} = \sum_{n_i=1}^{N_i} \alpha_k(t) \left[-\frac{1}{2} + \frac{(\tilde{r}(\mathcal{L}_m, t) - \mu_k(t))^2}{2\Sigma_k^2(t)} \right]. \quad (\text{A17})$$

and the 2nd derivative is

$$\frac{\ln P(\tilde{\mathbf{r}}(\mathcal{L}_m, t); \boldsymbol{\theta})}{\partial \alpha_k(t) \partial \alpha_k(t)} = 0. \quad (\text{A18})$$

$$\frac{\ln P(\tilde{\mathbf{r}}(\mathcal{L}_m, t); \boldsymbol{\theta})}{\partial \alpha_k(t) \partial \mu_k(t)} = \sum_{n_i=1}^{N_i} \frac{\tilde{r}^{N_i}(\mathcal{L}_m, t) - \mu_k(t)}{\Sigma_k(t)}. \quad (\text{A19})$$

$$\frac{\ln P(\tilde{\mathbf{r}}(\mathcal{L}_m, t); \boldsymbol{\theta})}{\partial \alpha_k(t) \partial \Sigma_k(t)} = \sum_{n_i=1}^{N_i} \left[-\frac{1}{2} + \frac{(\tilde{r}^{N_i}(\mathcal{L}_m, t) - \mu_k(t))^2}{2\Sigma_k^2(t)} \right]. \quad (\text{A20})$$

$$\frac{\ln P(\tilde{\mathbf{r}}(\mathcal{L}_m, t); \boldsymbol{\theta})}{\partial \mu_k(t) \partial \alpha_k(t)} = - \sum_{n_i=1}^{N_i} \frac{\tilde{r}^{N_i}(\mathcal{L}_m, t) - \mu_k(t)}{\Sigma_k(t)}. \quad (\text{A21})$$

$$\frac{\ln P(\tilde{\mathbf{r}}(\mathcal{L}_m, t); \boldsymbol{\theta})}{\partial \mu_k(t) \partial \mu_k(t)} = \sum_{n_i=1}^{N_i} \frac{\alpha_k(t)}{2\Sigma_k(t)}. \quad (\text{A22})$$

$$\frac{\ln P(\tilde{\mathbf{r}}(\mathcal{L}_m, t); \boldsymbol{\theta})}{\partial \mu_k(t) \partial \Sigma_k(t)} = \sum_{n_i=1}^{N_i} \frac{\alpha_k(t) (\tilde{r}^{N_i}(\mathcal{L}_m, t) - \mu_k(t))}{2\Sigma_k^2(t)}. \quad (\text{A23})$$

$$\frac{\ln P(\tilde{\mathbf{r}}(\mathcal{L}_m, t); \boldsymbol{\theta})}{\partial \Sigma_k(t) \partial \alpha_k(t)} = \sum_{n_i=1}^{N_i} \left[-\frac{1}{2} + \frac{(\tilde{r}^{N_i}(\mathcal{L}_m, t) - \mu_k(t))^2}{2\Sigma_k^2(t)} \right]. \quad (\text{A24})$$

$$\frac{\ln P(\tilde{\mathbf{r}}(\mathcal{L}_m, t); \boldsymbol{\theta})}{\partial \Sigma_k(t) \partial \mu_k(t)} = - \sum_{n_i=1}^{N_i} \frac{\alpha_k(t) (\tilde{r}^{N_i}(\mathcal{L}_m, t) - \mu_k(t))}{\Sigma_k^2(t)}. \quad (\text{A25})$$

$$\frac{\ln P(\tilde{\mathbf{r}}(\mathcal{L}_m, t); \boldsymbol{\theta})}{\partial \Sigma_k(t) \partial \Sigma_k(t)} = - \sum_{n_i=1}^{N_i} \frac{\alpha_k(t) (\tilde{r}^{N_i}(\mathcal{L}_m, t) - \mu_k(t))^2}{4\Sigma_k^3(t)}. \quad (\text{A26})$$

Then the expectation for each item on the diagonal is calculated as follows,

$$\mathcal{G}(\alpha_k(t), \alpha_k(t)) = \mathcal{G} \left\{ \frac{[\partial \ln P(\tilde{\mathbf{r}}(\mathcal{L}_m, t); \boldsymbol{\theta})]}{\partial \alpha_k(t) \partial \alpha_k(t)} \right\}^2 = 0. \quad (\text{A27})$$

$$\mathcal{G}(\mu_k(t), \mu_k(t)) = \mathcal{G} \left\{ \frac{[\partial \ln P(\tilde{\mathbf{r}}(\mathcal{L}_m, t); \boldsymbol{\theta})]}{\partial \mu_k(t) \partial \mu_k(t)} \right\}^2 = N_i \frac{\alpha_k(t)}{2\Sigma_k(t)}. \quad (\text{A28})$$

$$\mathcal{G}(\Sigma_k(t), \Sigma_k(t)) = \mathcal{G} \left\{ \frac{[\partial \ln P(\tilde{\mathbf{r}}(\mathcal{L}_m, t); \boldsymbol{\theta})]}{\partial \Sigma_k(t) \partial \Sigma_k(t)} \right\}^2 = - \sum_{n_i=1}^{N_i} \frac{\alpha_k(t) (\tilde{r}^{N_i}(\mathcal{L}_m, t) - \mu_k(t))^2}{4\Sigma_k^3(t)}. \quad (\text{A29})$$

So we let $\mathbb{J}_{\mathcal{E}(\gamma_k, \gamma_k)}$ express the FIM for $\mathcal{E}(\gamma_k, \gamma_k)$.

Appendix D. Proof of Corollary 2

We denote the CRLB error analysis derived from transfer learning and error analysis derived from GMM by $CRLB_1$ and $CRLB_2$, respectively, at T_3 . And we conclude the corollary 2; the following is the derivation process.

Firstly, we can derive $CRLB_1$ by the parameters obtained from the GMM fitted in T_1 and T_2 . According to A5,

$$\alpha_k(T_3) = \alpha_k(T_2) + \frac{T_3 - T_2}{T_2 - T_1} [\alpha_k(T_2) - \alpha_k(T_1)]. \quad (\text{A30})$$

$$\mu_k(T_3) = \mu_k(T_2) + \frac{T_3 - T_2}{T_2 - T_1} [\mu_k(T_2) - \mu_k(T_1)]. \quad (\text{A31})$$

$$\Sigma_k(T_3) = \Sigma_k(T_2) + \frac{T_3 - T_2}{T_2 - T_1} [\Sigma_k(T_2) - \Sigma_k(T_1)]. \quad (\text{A32})$$

Based on the fitted GMM model, $CRLB_2$ is the bound derived in $\boldsymbol{\theta}(T_3)$, which can be expressed as,

$$\text{Cov}_{\theta(T_3)} \geq \mathbb{J}_{\theta(T_3)}^{-1} \quad (\text{A33})$$

We can acquire the CRLB for the actual RSS parameters analysis in $\theta(T_3)$, which can be expressed by,

$$\begin{aligned} \text{CRLB}_2 &= \mathcal{Q}_{\theta(T_3)} \\ &= \sqrt{\text{tr}\{\mathbb{J}_{\theta(T_3)}^{-1}\}} \end{aligned} \quad (\text{A34})$$

In the case of known time conditions $T_2 - T_1 \gg T_3 - T_2$, the consistency theorem is satisfied according to the parameters evaluated by CRLB. When the total number of samples gradually increases, the estimator $\hat{\theta}$ gradually converges to the actual value θ , and the following inferences are made:

$$\begin{aligned} \|\text{CRLB}_1 - \text{CRLB}_{\text{actual}}\| &\leq \zeta \\ \|\text{CRLB}_2 - \text{CRLB}_{\text{actual}}\| &\leq \zeta \end{aligned} \quad (\text{A35})$$

where ζ denotes a constant close to zero. And CRLB_1 is derived from transfer learning at T_3 . CRLB_2 is the error analysis obtained after deriving from time-domain-related parameters in T_1 and T_2 . $\text{CRLB}_{\text{actual}}$ is the CRLB that derived from actual parameters at T_3 .

Then, we can derive the conclusion,

$$\|\text{CRLB}_1 - \text{CRLB}_2\| \leq \zeta \quad (\text{A36})$$

References

1. Zhou, Baoding, et al. "Cooperative Indoor Localization Using Mobile Robot Anchors via Factor Graph Optimization." *IEEE Internet Things J.* (2025).
2. Zuo, Yiping, et al. "Mobile blockchain-enabled secure and efficient information management for indoor positioning with federated learning." *IEEE Trans. Mob. Comput.* 23.12 (2024): 12176-12194.
3. Fayyad, Jamil, et al. "Deep learning sensor fusion for autonomous vehicle perception and localization: A review." *Sensors* 20.15 (2020): 4220.
4. Dauptain, Xavier, et al. "Conception of a high-level perception and localization system for autonomous driving." *Sensors* 22.24 (2022): 9661.
5. Ušinskas, Vygantas, et al. "Towards autonomous driving: technologies and data for vehicles-to-everything communication." *Sensors* 24.11 (2024): 3411.
6. Ren, Yanbing, et al. "DistPreserv: Maintaining user distribution for privacy-preserving location-based services." *IEEE Trans. Mob. Comput.* 22.6 (2022): 3287-3302.
7. Zhu, Lipeng, et al. "A tutorial on movable antennas for wireless networks." *IEEE Commun. Surv. Tutorials* (2025).
8. Saunders, Simon R., and Alejandro A. Aragón-Zavala. *Antennas and propagation for wireless communication systems*. John Wiley & Sons, 2024.
9. Jiang, Wei, et al. "Indoor and outdoor seamless positioning method using UWB enhanced multi-sensor tightly-coupled integration." *IEEE Trans. Veh. Technol.* 70.10 (2021): 10633-10645.
10. Wu, Chong, et al. "RF-SLAM: UHF-RFID based simultaneous tags mapping and robot localization algorithm for smart warehouse position service." *IEEE Trans. Ind. Inf.* 19.12 (2023): 11765-11775.
11. Bianchi V, Ciampolini P, De Munari I. RSSI-based indoor localization and identification for ZigBee wireless sensor networks in smart homes[J]. *IEEE Trans. Instrum. Meas.*, 2018, 68(2): 566-575.
12. Zhang B, Sifaou H, Li G Y. CSI-fingerprinting indoor localization via attention-augmented residual convolutional neural network[J]. *IEEE Trans. Wireless Commun.*, 2023, 22(8): 5583-5597.
13. Mackey, Andrew, et al. "Improving BLE beacon proximity estimation accuracy through Bayesian filtering." *IEEE Internet Things J.* 7.4 (2020): 3160-3169.
14. Liu, Hui, et al. "Survey of wireless indoor positioning techniques and systems." *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.* 37.6 (2007): 1067-1080.

15. Morar, Anca, et al. "A comprehensive survey of indoor localization methods based on computer vision." *Sensors* 20.9 (2020): 2641.
16. Abuhoureyah, Fahd, Yan Chiew Wong, and Ahmad Sadhiqin Mohd Isira. "Trajectory mapping through channel state information by triangulation method and fine-tuning." *J. Eng. Appl. Sci.* 71.1 (2024): 196.
17. Qin, Junping, et al. "Indoor trajectory tracking scheme based on Delaunay triangulation and heuristic information in wireless sensor networks." *Sensors* 17.6 (2017): 1275.
18. Li, Ze, et al. "TriLoc: Toward accurate indoor localization with assistance of microwave reflections." *IEEE Trans. Microwave Theory Tech.* 71.6 (2022): 2734-2747.
19. Shamaei, Kimia, and Zaher M. Kassas. "A joint TOA and DOA acquisition and tracking approach for positioning with LTE signals." *IEEE Trans. Signal Process.* 69 (2021): 2689-2705.
20. Hsiao, An-Hung, et al. "SALC: Skeleton-Assisted Learning-Based Clustering for Time-Varying Indoor Localization." *IEEE Trans. Network Sci. Eng.* 11.1 (2023): 439-452.
21. Huang, Huakun, et al. "Wi-Fi Sensing Based on Deep Supervised Dictionary Learning for Robust Device-Free Localization." *IEEE Trans. Veh. Technol.* (2025).
22. Rafique, Hamaad, et al. "A characteristics-based least common multiple algorithm to optimize magnetic field-based indoor localization." *IEEE Internet Things J.* (2024).
23. Farahsari, Pooyan Shams, et al. "A survey on indoor positioning systems for IoT-based applications." *IEEE Internet Things J.* 9.10 (2022): 7680-7699.
24. Zhu, Xiaoqiang, et al. "Indoor intelligent fingerprint-based localization: Principles, approaches and challenges." *IEEE Commun. Surv. Tutorials* 22.4 (2020): 2634-2657.
25. Zang, Yiming, et al. "A novel optical localization method for partial discharge source using ANFIS virtual sensors and simulation fingerprint in GIL." *IEEE Trans. Instrum. Meas.* 70 (2021): 1-11.
26. Wei, Chen, et al. "Fingerprint-based localization and channel estimation integration for cell-free massive MIMO IoT systems." *IEEE Internet Things J.* 9.24 (2022): 25237-25252.
27. Li, Qiao, et al. "Automatic indoor radio map construction and localization via multipath fingerprint extrapolation." *IEEE Trans. Wireless Commun.* 22.9 (2023): 5814-5827.
28. Liu, Yutong, Linghe Kong, and Guihai Chen. "Data-oriented mobile crowdsensing: A comprehensive survey." *IEEE Commun. Surv. Tutorials* 21.3 (2019): 2849-2885.
29. Gong, Xiaodong, et al. "An enhanced indoor positioning solution using dynamic radio fingerprinting spatial context recognition." *IEEE Internet Things J.* 10.2 (2022): 1297-1309.
30. Zhou, Rong, et al. "A wi-fi indoor positioning method based on an integration of emdt and wknn." *Sensors* 22.14 (2022): 5411.
31. Tan, Chuanqi, et al. "A survey on deep transfer learning." *International conference on artificial neural networks. Cham: Springer International Publishing*, 2018.
32. Pan, Sinno Jialin, et al. "Domain adaptation via transfer component analysis." *IEEE Trans. Neural Networks* 22.2 (2010): 199-210.
33. Long, Mingsheng, et al. "Transfer feature learning with joint distribution adaptation." *Proceedings of the IEEE international conference on computer vision*. 2013.
34. Long, Mingsheng, et al. "Learning transferable features with deep adaptation networks." *International conference on machine learning*. PMLR, 2015.
35. Redner, Richard A., and Homer F. Walker. "Mixture densities, maximum likelihood and the EM algorithm." *SIAM review* 26.2 (1984): 195-239.
36. Xie, Ning, Zhuoyuan Li, and Haijun Tan. "A survey of physical-layer authentication in wireless communications." *IEEE Commun. Surv. Tutorials* 23.1 (2020): 282-310.
37. Zhang, Menglin, et al. "Two-stage stochastic optimal operation of integrated electricity and heat system considering reserve of flexible devices and spatial-temporal correlation of wind power." *Appl. Energy* 275 (2020): 115357.
38. Reynolds, Douglas. "Gaussian mixture models." *Encyclopedia of biometrics*. Springer, Boston, MA, 2015. 827-832.
39. Bond S R, Hoeffler A, Temple J R W. GMM estimation of empirical growth models[J]. *Available at SSRN* 290522, 2001.
40. Yu, L., Yang, T., & Chan, A. B. (2018). Density-preserving hierarchical EM algorithm: Simplifying Gaussian mixture models for approximate inference. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41(6), 1323-1337.
41. Moon H R, Phillips P C B. GMM estimation of autoregressive roots near unity with panel data[J]. *Econometrica*, 2004, 72(2): 467-522.

42. Conley, Timothy G. "GMM estimation with cross sectional dependence." *Journal of econometrics* 92.1 (1999): 1-45.
43. Andrews D W K. Consistent moment selection procedures for generalized method of moments estimation[J]. *Econometrica*, 1999, 67(3): 543-563.
44. Barni, Mauro, et al. "A quasi-Euclidean norm to speed up vector median filtering." *IEEE Trans. Image Process.* 9.10 (2000): 1704-1709.
45. Guo, Gongde, et al. "KNN model-based approach in classification." OTM Confederated International Conferences" On the Move to Meaningful Internet Systems". Berlin, Heidelberg: Springer Berlin Heidelberg, 2003.
46. Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[J]. *Advances in neural information processing systems*, 2012, 25.
47. Peleg, S., & Porat, B. (2002). The Cramer-Rao lower bound for signals with constant amplitude and polynomial phase. *IEEE Trans. Signal Process.*, 39(3), 749-752.
48. Kay S M. Fundamentals of statistical signal processing: estimation theory[M]. *Prentice-Hall, Inc.*, 1993.
49. Shen Y, Win M Z. On the accuracy of localization systems using wideband antenna arrays[J]. *IEEE Trans. Commun.*, 2010, 58(1): 270-280.
50. Gui, Linqing, et al. "A Cramer–Rao lower bound of CSI-based indoor localization." *IEEE Trans. Veh. Technol.* 67.3 (2017): 2814-2818.
51. Patwari, Neal, et al. "Locating the nodes: cooperative localization in wireless sensor networks." *IEEE Signal Process Mag.* 22.4 (2005): 54-69.
52. Carrasco M, Florens J P. On the asymptotic efficiency of GMM[J]. *Econom. Theory*, 2014, 30(2): 372-406.
53. Carrasco M, Florens J P. Generalization of GMM to a continuum of moment conditions[J]. *Econom. Theory*, 2000, 16(6): 797-834.
54. Gretton, Arthur, et al. "A kernel two-sample test." *The journal of machine learning research* 13.1 (2012): 723-773.
55. Borgwardt, Karsten M., et al. "Integrating structured biological data by kernel maximum mean discrepancy." *Bioinf.* 22.14 (2006): e49-e57.
56. Yan, Hongliang, et al. "Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017.
57. Jia, Xiaodong, et al. "Assessment of data suitability for machine prognosis using maximum mean discrepancy." *IEEE Trans. Ind. Electron.* 65.7 (2017): 5872-5881.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.