

Article

Transfer Learning for Indoor Localization Algorithm Based on Deep Domain Adaptation

Jiahao Wang ¹ , Yifu Fu ¹, Hainan Feng ¹ and Junxiang Wang ^{2,*}

¹ School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu 610000, China; wangjh@uestc.edu.cn (J.W.); yf_yf_uestc@163.com (Y.F.); hainan.feng@std.uestc.edu.cn (H.F.)

² Technology Innovation Department of Tianfu Co-Innovation Center, University of Electronic Science and Technology of China, Chengdu 610000, China

* Correspondence: junxiang.wang@uestc.edu.cn; Tel.: +86-189-0801-1273

Abstract: In application, training data and test data collected via indoor positioning algorithms usually do not come from the same ideal conditions. Changes in various environmental conditions and signal drift can cause different probability distributions between the data sets. Existing positioning algorithms cannot guarantee stable accuracy when facing these issues, resulting in dramatic reduction and the infeasibility of the positioning accuracy of indoor location algorithms. Considering these restrictions, domain adaptation technology in transfer learning has proven to be a promising solution in past research in terms of solving the inconsistent probability distribution problems. However, most localization algorithms based on transfer learning do not perform well because they only learn a shallow representation feature, which can only slightly reduce the domain discrepancy. Based on the deep network and its strong feature extraction ability, it can learn more transferable features for domain adaptation and achieve better domain adaptation effects. A Deep Joint Mean Distribution Adaptation Network (DJMDAN) is proposed to align the global domain and relevant subdomain distributions of activations in multiple domain-specific layers across domains to achieve domain adaptation. The test results demonstrate that the performance of the proposed method outperforms the comparison algorithm in indoor positioning applications.



Citation: Wang, J.; Fu, Y.; Feng, H.; Wang, J. Transfer Learning for Indoor Localization Algorithm Based on Deep Domain Adaptation. *Sensors* **2023**, *23*, 9334. <https://doi.org/10.3390/s23239334>

Academic Editors: Marco Passafiume, Stefano Maddio, Giovanni Collodi and Binghao Li

Received: 16 October 2023

Revised: 14 November 2023

Accepted: 20 November 2023

Published: 22 November 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: indoor localization; transfer learning; deep domain adaptation network

1. Introduction

With the dawn of the era of digital intelligence, the importance of accurately acquiring location information between devices and users to optimize device–person interaction has become paramount. Consequently, indoor localization has risen to be one of the most critical technological components, finding extensive applications across various areas such as smart energy management, smart home systems, and points of interest identification [1–3]. It is often necessary to utilize alternative signal sources for indoor localization, such as Bluetooth Low Energy (BLE) [4], Ultra-wide Band (UWB) [5], Zigbee [6], Radio Frequency Identification (RFID) [7], Wi-Fi [8], etc. The majority of the above require the additional arrangement of specialized devices within the localization space, leading to issues such as high deployment costs and lack of convenience. BLE can turn all mobile devices carrying Bluetooth modules into signal transceiver devices, so its cost is very low and has higher accuracy than Wi-Fi. However, due to the poor stability of Bluetooth itself and its vulnerability to environmental interference, Bluetooth is more suitable as an auxiliary signal source to participate in the positioning process. Therefore, Wi-Fi signals have gained the favor of numerous researchers due to their ease of acquisition and widespread presence in various types of indoor scenarios.

Many localization methods based on Wi-Fi signals use Received Signal Strength Indication at the link layer as a data source. RSSI can only describe the overall superimposed

result of the distorted signals received from different paths. Channel State Information can distinguish the specific conditions of signals propagating through different paths, so it is more stable than the RSSI on a timescale but has strong specificity over space [9]. Although CSI can provide fine-grained information, it will vary with the passage of time or changes in the environment, such as humidity, temperature, and furnishings [10]. It will cause domain shifts and different probability distributions between the training data collected during the offline stage and the test data obtained during the online stage. Directly using these data for training and locating will significantly reduce accuracy or even result in the failure of implementations. Reconstructing the model periodically in response to environmental changes can be a solution to mitigate the interference of domain shift, but the resampling process is costly and infeasible in real applications.

To solve the domain shift problem, transfer learning has been enrolled in many solutions and proven to provide strong robustness and high accuracy. But in real applications, this method still does not perform well because they only learn shallow representation features to find a transformation form in a low-dimensional manifold to make the subspaces of the source and target domains similar after mapping, which can only slightly reduce the domain discrepancy [11–15]. Recent research has shown that through the strong feature extraction ability of deep neural networks, more transferable features for domain adaptation can be learned from raw data. Therefore, applying deep neural networks to domain adaptation methods can achieve better domain adaptation effects.

In this work, we propose a Deep Joint Mean Distribution Adaptation Network (DJMDAN) to align the global domain and relevant subdomain distributions of activations in multiple domain-specific layers across domains for domain adaptation. Our work can roughly achieve the same distribution after domain adaptation and retain the fine-grained information between different categories at the same time. The experiment results show that the proposed DJMDAN method can effectively mitigate the interference in the data distribution caused by domain shift and improve localization accuracy in dynamic environments.

The main contributions of this work can be summarized as follows. First, we present an improved domain adaptation method, which performs global and local adaptation of the data domain at the same time under the premise of avoiding the loss of fine-grained information of the data as much as possible. Second, the deep TL method is introduced into indoor localization to reduce the overhead caused by resampling fingerprint points and solve the problem caused by the instability of CSI data due to time and environmental changes. Third, we implement a localization system based on the proposed method and prove the accuracy and robustness in complex and changeable indoor environments by evaluating the performance of this system in a real environment.

The rest of this paper is structured as follows. Section 2 discusses the relevant work of the proposed method. Section 3 provides a general overview of DJMDAN architecture and introduces its domain adaptation loss function components. In Section 4, we evaluated the proposed system on data collected from real-world scenarios and compared it with other transfer learning methods. Finally, Section 5 concludes the paper and discusses future work.

2. Related Work

Localization in dynamic environments can be considered a domain adaptation problem. When conducting positioning, usually, all features of the entire sample will be input into the model. This can cause feature values sensitive to environmental changes to vary significantly in distribution, thus rendering the model trained in the source domain inapplicable in the target domain. Therefore, to solve this problem, some researchers have proposed methods based on feature selection. The main idea of this kind of method is to reduce the impact of environmental changes on positioning by selecting features with strong robustness (i.e., insensitive to environmental changes). In the early stage of positioning transfer learning, the team of Sinno Jialin Pan [16] proposed to learn a low-dimensional latent feature space through dimensionality reduction so that the distribution of source

domain data and target domain data in this space is approximate. Appropriate samples are selected from each fingerprint point for localization [17]. The method developed in [18] introduces a new relevance measure to select channels to be used as link features, thereby reducing the localization classification error when using the random forest classifier. On the other hand, after the device changes, the different number of access points will also seriously affect the localization accuracy. The main idea behind pseudo-labeling is to identify correlations between the source and target domain distributions, using source domain labels to annotate target domain data with pseudo labels. To solve the positioning offset problem after floor changes, the authors generate pseudo labels for the unlabeled target domain data, facilitating the mapping between domains [19]. In order to eliminate the impact of environmental changes on the localization results, Sun Zhuo et al. [20] established a dimensionality reduction method, which learns a mapping between a source data set and a target data set in a low-dimensional space.

To overcome the loss caused by shallow feature adaptation, researchers tried to apply deep learning in transfer learning and utilize the strong feature extraction ability of deep networks to achieve more transferable local features from data to optimize domain adaptation performance [21], and they hope to achieve better knowledge transfer results compared with those shallow domain adaptation algorithms. Most of the transfer learning methods based on deep domain adaptation can be mainly divided into two branches: the statistic moment matching-based method and the adversarial loss-based method.

The method of deep domain adaptation mainly depends on matching and minimizing the difference between two domains by aligning the first-order or second-order moment of domain features. Therefore, the network can learn the domain-invariant features. In this process, solutions relying on characteristics of Maximum Mean discrepancy (MMD) [22–27], Central Moment Discrepancy (CMD) [28], or covariance in a second-order statistic [29] have been widely researched. Another meaningful adversarial loss-based deep domain adaptation method makes use of data from different domains and keeps data labels non-discriminatory [30–33]. It makes use of the generative adversarial network (GAN) and tries to align the source domain and target domain at the data distribution level under the condition of unsupervised target domain.

In fact, most deep transfer learning methods are used in the field of image, and there are few related studies applied in the field of indoor localization. The deep transfer learning method is introduced into indoor localization [34], and the authors show through experiments that for a trained localization model, its feature extraction layer can be directly transferred to other models. It only needs the fully connected layer to be retrained to achieve the same baseline accuracy as that of the basic model without transfer. AdapLoc [35] aligns semantically with the source domain by marking a small number of CSI fingerprints as their positions in the target domain. This approach alleviates the alignment confusion issue to some extent. However, it does not align both inter-domain and intra-domain at the same time and uses Euclidean distance as the measure of domain features, which cannot adequately reflect the difference in the probability distribution of deep features.

Most of the existing deep domain adaptation algorithms only consider the global domain adaptation to solve the domain shift issue, that is, aligning the global data without considering the relationship between the two subdomains. Although this method mitigates the interference of domain shift on data distribution to a certain extent, the mapped data are often prone to confusion, making the data unable to be accurately classified. FitLoc [36] completes cross-room transfer by combining FLDA with Bregman divergence. It takes into account the fine-grained alignment problem, but the transfer ability is relatively weak.

To solve this problem, the proposed DJMDAN can adapt to the overall distribution difference between the two domains' data and retain the fine-grained information between different categories of data at the same time. It effectively solves the confused alignment issue while ensuring the domain adaptation effect to achieve greater environmental adaptability and higher accuracy of indoor localization.

3. DJMDAN Algorithm

3.1. Model Architecture

The DJMDAN uses a deep neural network to improve domain adaptation performance by learning and updating the network by comparing the Multi-Kernel Maximum Mean Discrepancy (MK-MMD) [37] and Local Maximum Mean Discrepancy (LMMD) [38] of two domains output by the fully connected layer, and at the same time takes into account the classification loss of the source domain. The main process is shown in Figure 1. The proposed DJMDAN is based on a one-dimensional convolutional neural network, and the whole network is composed of three parts: feature extractor, domain adaptation module and classifier. The specific network architecture is shown in Figure 2.

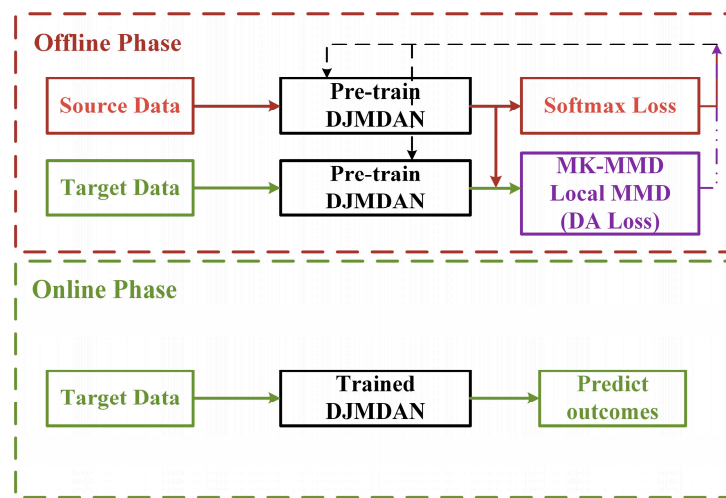


Figure 1. Flowchart of the DJMDAN algorithm.

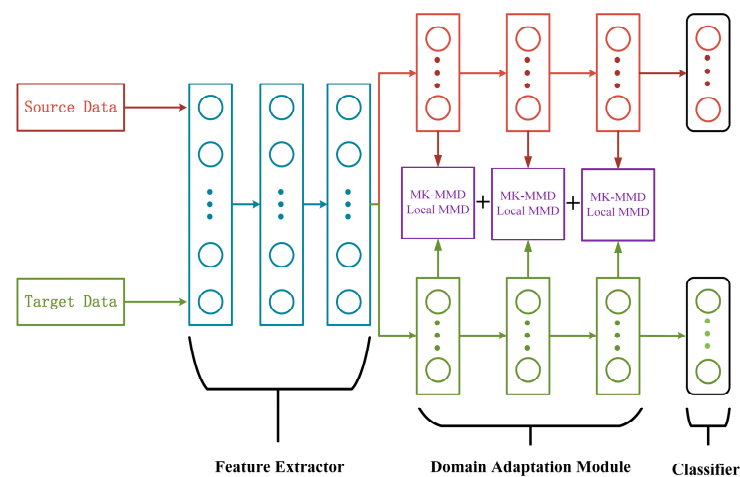


Figure 2. Diagram of the DJMDAN network structure.

As listed in Table 1, the feature extractor uses a three-layer composite layer, including convolution and pooling. It is used to extract more transferable features from the original data. The domain adaptation module consists of three fully connected layers. There is a domain adaptation layer to calculate the network's feature transfer capability after each fully connected layer, and the last fully connected layer is used for the classification layer.

Table 1. Network structure parameters.

Layer	Configuration
Conv1	(5,8), Stride = 2, Padding = 0
Maxp1	5, Stride 1
Conv2	(5,16), Stride = 2, Padding = 0
Maxp2	5, Stride 1
Conv3	(5,32), Stride = 1, Padding = 0
Max3	5, Stride = 1
FC1	512
FC2	256
FC3	128

$\mathcal{D}_{src}$ represents the source domain data collected in the offline phase, while $\mathcal{D}_{tar}$ represents the unlabeled data collected in the online phase, and y_{src} is the corresponding label. With the classification error and the domain adaptation error combined, the final loss function of DJMDAN is shown in (1), where ℓ_{clf} is the classification loss, and ℓ_m, ℓ_c are the domain adaptation loss at the marginal and conditional probability distributions, respectively. α and β are the weighting factors that balance the two parts.

$$\ell_{total} = \ell_{clf}(\mathcal{D}_{src}, y_{src}) + \alpha \ell_m(\mathcal{D}_{src}, \mathcal{D}_{tar}) + \beta \ell_c(\mathcal{D}_{src}, \mathcal{D}_{tar}) \quad (1)$$

3.2. Domain Adaptation Loss Function

Most of the existing domain adaptation solutions in deep transfer learning localization focus on adapting to the distribution differences of both domains from a global perspective. Although these methods effectively mitigate domain shift affection, the local geometric characteristics of the data are also ignored. They do not consider the relationship between the same category of data in the two domains, resulting in the loss of fine-grained information in the mapped data. It leads to the confused alignment problem of two domains, finally affecting the accuracy of the locating algorithm. To solve this problem, DJMDAN takes the MK-MMD and LMMD to achieve the adaptation to both global probability distribution and local probability distribution of two domains on the premise of avoiding fine-grained information loss as far as possible.

3.2.1. Marginal Probability Distribution Adaptation

The MMD metric was often used as a measurement method in previous deep transfer learning methods based on statistical moment matching. However, MMD only uses a single kernel function to calculate the distribution difference between the two domains after mapping, so it cannot dynamically obtain the optimal kernel function according to the difference of the output data when calculating the distribution difference of the multi-layer network output data. As a result, the accuracy and adaptability of previous deep transfer methods are limited. To this end, MK-MMD is introduced as a measure of the difference between marginal probability distributions. MK-MMD originated from Multi-Kernel Learning (MKL) [39], which is a linear combination of multiple MMD based on different kernel functions. The composite kernel function composed of multiple basis kernel functions has multifarious advantages of different kernel functions. Therefore, compared with MMD with a single fixed kernel, MK-MMD has a stronger feature mapping

ability, so it can help the network achieve better domain adaptation. The specific calculation formula of MK-MMD is shown in (2).

$$\begin{aligned}\ell_m(\mathcal{D}_{src}, \mathcal{D}_{tar}) &= \left\| \frac{1}{n_s} \sum_{i=1}^{n_s} \phi(x_{src}) - \frac{1}{n_t} \sum_{j=1}^{n_t} \phi(x_{tar}) \right\|_{\mathcal{H}}^2 \\ &= \frac{1}{n_s^2} \sum_{i=1}^{n_s} \sum_{j=1}^{n_s} (x_{src}, x_{src}) - \frac{2}{n_s n_t} \sum_{i=1}^{n_s} \sum_{j=1}^{n_t} (x_{src}, x_{tar}) \\ &\quad + \frac{1}{n_t^2} \sum_{i=1}^{n_t} \sum_{j=1}^{n_t} (x_{tar}, x_{tar})\end{aligned}\quad (2)$$

Since the kernel function of MK-MMD is composed of multiple different basis kernel functions, its kernel function $\kappa(\cdot)$ can be specifically expressed as (3)

$$\mathcal{K} \triangleq \left\{ \kappa(\cdot) = \sum_{i=1}^M \mu_i k_i : \sum_{i=1}^M \mu_i = 1, \mu_i \geq 0, \forall i \right\} \quad (3)$$

where M represents the number of basis kernel functions k_i that form the MK-MMD kernel function, and μ_i represents the weight value corresponding to the basis kernel function k_i . Common basis kernel functions are listed in Table 2.

Table 2. Common Basis Kernel Functions.

Kernel Name	Expression	Parameters
Linear kernel	$K(x_i, x_j) = x_i^T \cdot x_j$	
Radial basis kernel	$K(x_i, x_j) = e^{-\gamma \ x_i - x_j\ ^2}$	$\gamma = \frac{1}{2\sigma^2}, \sigma > 0$
Polynomial kernel	$K(x_i, x_j) = (\gamma x_i^T x_j + k)^n$	$\gamma > 0, k > 0$

Domain adaptation modules are considered to be added after each of the multiple fully connected layers to calculate the difference in the marginal probability distribution of the output, which can better reduce the difference in marginal distribution.

We assume that the output of two domains after the K_{th} layer, respectively, is $\{m_a^{sk}\}_{a=1}^{n_s}$ and $\{m_b^{tk}\}_{b=1}^{n_t}$. The loss function for the entire difference in the marginal probability distribution can be expressed as (4).

$$\begin{aligned}\ell_m &= \sum_{l=k1}^{k2} \left\| \frac{1}{n_s} \sum_{a=1}^{n_s} \phi(m_a^{sk}) - \frac{1}{n_t} \sum_{b=1}^{n_t} \phi(m_b^{tk}) \right\| \\ &= \frac{1}{n_s^2} \sum_{l=k1}^{k2} \left[\sum_{a=1}^{n_s} \sum_{b=1}^{n_s} \kappa(m_a^{sk}, m_b^{sk}) \right] \\ &\quad - \frac{2}{n_s n_t} \sum_{l=k1}^{k2} \left[\sum_{a=1}^{n_s} \sum_{b=1}^{n_t} \kappa(m_a^{sk}, m_b^{tk}) \right] \\ &\quad + \frac{1}{n_t^2} \sum_{l=k1}^{k2} \left[\sum_{a=1}^{n_t} \sum_{b=1}^{n_t} \kappa(m_a^{tk}, m_b^{tk}) \right]\end{aligned}\quad (4)$$

3.2.2. Conditional Probability Distribution Adaptation

The above adaptation method based on marginal probability distribution only solves the problem of the overall distribution alignment of the two domains from the perspective of global changes and ignores the fine-grained information between data categories in the mapping process, which leads to the loss of information in the mapped data. To further improve the final output accuracy, the difference in conditional probability distribution between the two domains still needs to be adapted by aligning the distribution of the subset data belonging to the same category in both domains. Therefore, we used LMMD rather than MMD as the measurement method to calculate the distribution difference in the subset

data of different categories, respectively. The weighted sum of them is the final conditional distribution probability difference, as is shown in (5).

$$LMMD_{\mathcal{H}}(p, q) = \frac{1}{C} \sum_{c=1}^C \left\| \sum_{i=1}^{n_s} \omega_i^{sc} \phi(x_i) - \sum_{j=1}^{n_t} \omega_j^{tc} \phi(x_{tar}) \right\|_{\mathcal{H}}^2 \quad (5)$$

In the above equation, ω_i^{sc} and ω_j^{tc} represent the weight of data belonging to certain category C , and $\sum_{i=1}^{n_s} \omega_i^{sc} = \sum_{j=1}^{n_t} \omega_j^{tc} = 1$. Taking the source domain data x_k as an example, the corresponding weight ω_k^c is calculated as follows.

$$\omega_k^c = \frac{y_{kc}}{\sum_{(x_i, y_i) \in \mathcal{D}_{src}} y_{ic}} \quad (6)$$

where y_{kc} is the c_{th} element in the corresponding label vector of x_k . The original labels can be transformed into a one-hot vector to calculate the corresponding weights ω_i^{sc} because there are labels in the source domain. However, there are no labels of the target domain in the indoor localization field, so the weights ω_j^{tc} for the target domain data $\mathcal{D}_{tar}$ without labels can be obtained via the soft labels output by the network.

$$\begin{aligned} \ell_c &= \sum_{l=k1}^{k2} \frac{1}{C} \sum_{c=1}^C \left\| \sum_{a=1}^{n_s} \omega_a^{sc} \phi(m_a^{sk}) - \sum_{b=1}^{n_t} \omega_b^{tc} \phi(m_b^{tk}) \right\|_{\mathcal{H}}^2 \\ &= \sum_{l=k1}^{k2} \frac{1}{C} \sum_{c=1}^C \left[\sum_{a=1}^{n_s} \sum_{b=1}^{n_t} \omega_a^{sc} \omega_b^{tc} k(m_a^{sk}, m_b^{tk}) \right] \\ &\quad + \sum_{l=k1}^{k2} \frac{1}{C} \sum_{c=1}^C \left[\sum_{a=1}^{n_t} \sum_{b=1}^{n_t} \omega_a^{tc} \omega_b^{tc} k(m_a^{tk}, m_b^{tk}) \right] \\ &\quad - 2 \sum_{l=k1}^{k2} \frac{1}{C} \sum_{c=1}^C \left[\sum_{a=1}^{n_s} \sum_{b=1}^{n_t} \omega_a^{sc} \omega_b^{tc} k(m_a^{sk}, m_b^{tk}) \right] \end{aligned} \quad (7)$$

To better mitigate the difference in the conditional distribution, it is also necessary to calculate the conditional probability distribution difference in the multi-layer domain adaptation layer. Assuming that the outputs of two domains in the k_{th} layer of the network are $\{m_a^{sk}\}_{a=1}^{n_s}$ and $\{m_b^{tk}\}_{b=1}^{n_t}$, to better compute the mapping function $\phi(\cdot)$, the loss function $\ell_{conditional}(\mathcal{D}_{src}, \mathcal{D}_{tar})$ for the entire difference in conditional probability distribution can be expressed as (7).

4. Experiments

To verify the localization effect of the DJMDAN algorithm in a dynamic environment, we conducted comparative experiments with existing localization algorithms based on transfer learning and traditional machine learning algorithms in real scenarios.

4.1. Experimental Environments

The length and width of the experimental area is about $7 \times 7 \text{ M}^2$, and the actual photo and plan graph are shown in Figure 3.

A commercial router, TP-Link WDR7660, with three transmit antennas, was selected as the signal transmission device. The signal-receiving device was a Lenovo ThinkPad X201i laptop computer with Ubuntu 12.0.4 system and an Intel 5300 network card. The whole localization scene was divided into 32 grid-fingerprint points with 1 M spacing. The placements of devices and fingerprint points are shown in Figure 3b.

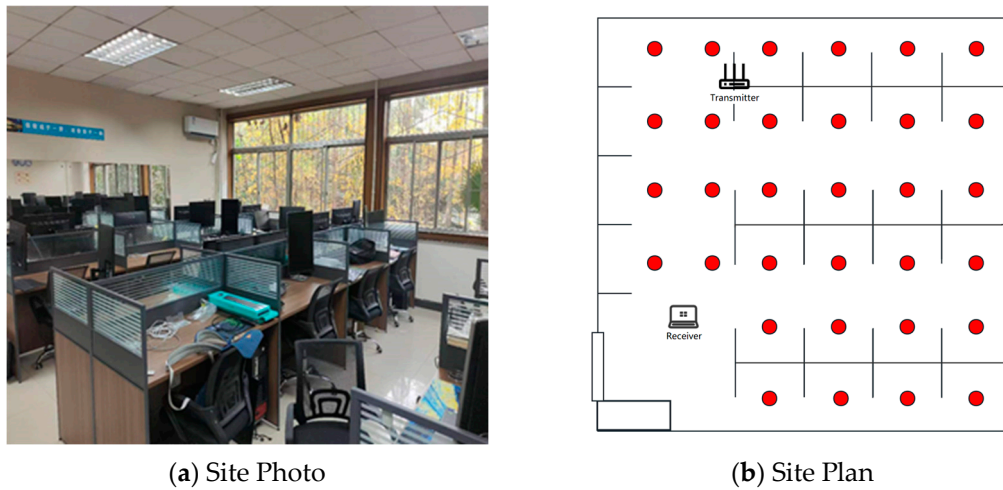


Figure 3. Environment setup.

4.2. Evaluation Metrics and Comparison Algorithms

4.2.1. Evaluation Criteria

Two criteria are mainly considered here to better evaluate the performance of localization algorithms.

Average Error Distance (AED): The arithmetic mean of the random error of all measurements taken in the experiments; the calculation formula is as follows.

$$AED = \frac{1}{K} \sum_{i=1}^K \sqrt{(x_{it} - x_{ip})^2 + (y_{it} - y_{ip})^2} \quad (8)$$

where (x_{it}, y_{it}) is the real coordinate value of the point to be measured, (x_{ip}, y_{ip}) is the predicted coordinate value, and K is the total number of samples in the test set.

Cumulative Distribution Functions (CDF): The probability distribution of the random variable X. In indoor localization, it is often used to reflect the proportion of the sample number in the overall sample number when the localization error value is less than a certain standard, and its expression is as follows:

$$F_D(d) = P(D \leq d) \quad (9)$$

4.2.2. Comparison Algorithms and Parameters

To evaluate the DJDMAN algorithm, we compare several algorithms mainly used in this field, such as KNN, TCA, 1D-CNN, DAN and DeepCORAL (which only uses the covariance of the fully connected layer output two domain data for the domain adaptation depth transfer method) [29].

Among them, 1DCNN is a deep learning algorithm without a domain adaptation module. Since the deep transfer learning method does not have a fixed basic network structure, the 1DCNN with three convolution-pooling layers and three fully connected layers is used as the basic network. 1DCNN is added as a benchmark algorithm to explore the influence of the domain adaptation method on the indoor localization effect. DAN is a depth transfer method that only uses the marginal probability distribution difference of data output by a fully connected layer for calculation, while DeepCORAL only uses the covariance.

For traditional machine learning methods, such as KNN and TCA, the K is set to 1. For deep learning methods, all 1D convolutional neural networks are used as their basic networks, and their parameters are shown in Table 1. In addition, the number of training rounds in the deep methods was set to 100, the learning rate was set to 0.001, the batch size was set to 128, and Adam was selected as the optimizer. To better suppress the noise

activation problem in the early stage of training, the weight functions of two probability distributions were updated dynamically and gradually [40] in the experiment; therefore, $\alpha = \beta = 2/1 + e^{-K\delta}$, where K is a fixed constant equal to 10, and $\delta \in [0, 1]$ increases linearly with training progress.

The DJMDAN uses four basic kernels, which are the linear kernel, polynomial kernel, Gaussian kernel and Laplacian kernel. As pointed out in [41,42], the weights of kernels in MK-MMD are set the same for better results, so it is not necessary to learn different weights. Therefore, we set the weights in multi-kernel MMD $\mu_i = 1/4$ in this paper.

4.3. Results and Analysis

4.3.1. Results under Varying Temporal Conditions

To verify the effectiveness of DJMDAN with the change of time conditions, we collected CSI data within 30 days in the experimental environment for comparison. The data on the first day were collected as the fingerprint database data, that is, the source domain data in transfer learning. The data collected in the remaining days were used as the test data to represent the target domain data in transfer learning. Figure 4 shows the change in the average error distance for each localization algorithm over 30 days.

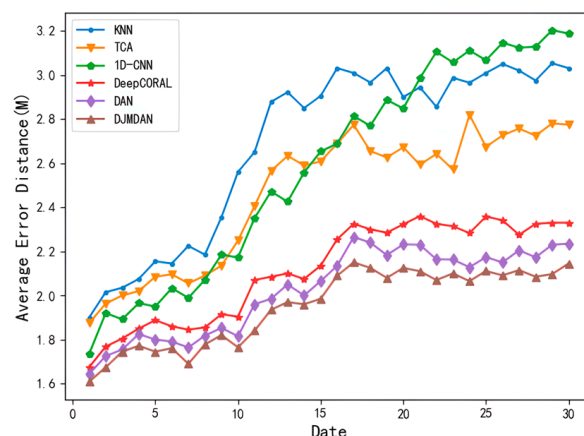


Figure 4. Comparison of the average positioning error of each positioning algorithm within 30 days.

From the results, due to the strong feature extraction ability of deep networks, all deep learning methods maintain lower localization error in the early stage of the experiment. However, with the passage of time conditions, the domain shift level of CSI signals collected at different times gradually increases, and the localization error of the algorithm with a one-dimensional convolutional neural network begins to be gradually higher than that of other transfer learning methods and even worse than that of traditional machine learning methods. Although the localization error of the algorithm based on deep transfer learning increases under the condition of time change, its localization accuracy is still better than algorithms based on traditional transfer learning methods.

All deep learning methods use the same basic network, so they are consistent in terms of model scale. Among them, the proposed DJMDAN achieves a more accurate localization effect than two classical deep domain adaptation algorithms in the experiment. This is because DAN only considers the marginal probability distribution difference of two domains as the loss function when performing deep domain adaptation, which ignores the important factor of the conditional probability distribution difference. However, DeepCORAL only uses the covariance of two domains as the loss function, and the domain adaptation ability is also limited.

The DJMDAN performs domain adaptation operation on the multi-layer fully connected layer from the perspective of the joint probability distribution difference, which is more effective in mitigating the domain shift phenomenon of the two domains. As a result, better localization results are achieved.

The cumulative distribution function diagram can better show the overall distribution of localization errors. As shown in Figure 5, the localization error of each algorithm on the 22nd day with larger discrimination and longer period among the localization errors of the above algorithms were selected to analyze its cumulative distribution. The result illustrates the CDF plot of the localization error for various transfer learning algorithms. The sample probability of localization results shows that the proposed DJMDAN algorithm is better than other algorithms in the same accuracy range under changing time conditions.

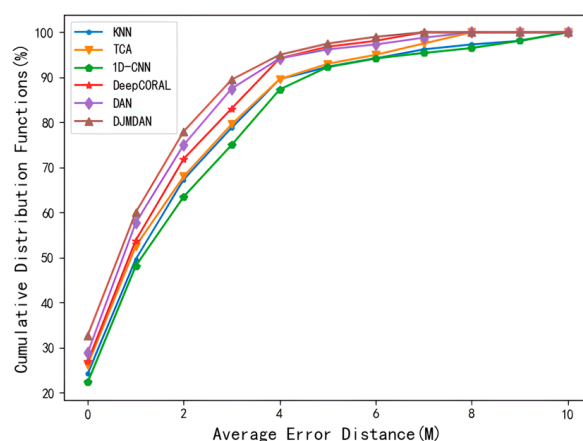


Figure 5. Comparison plot of the cumulative distribution function of error for each localization algorithm on day 22nd.

4.3.2. Results under Varying Environmental Conditions

To examine the effectiveness of DJMDAN under other environmental conditions, we conducted experiments on three cases, which are door and window switches in the laboratory, the position of indoor lockers, and the condition changes of both changes at the same time.

As shown in Figure 6, Figure 6a shows the opening of the door and window of the laboratory, Figure 6b shows the change in the placement of the lockers in the laboratory, and Figure 6c shows the change of both. Firstly, the CSI signal and location information of the fingerprint points were collected to form a fingerprint database without any indoor environmental conditions changes, and then the CSI signals of several points to be measured were collected as the location data for testing after each environmental condition was changed. Figure 7 shows the error of each algorithm under three conditions. Our solution shows the smallest average error distances in all tests.

The test error in Figure 7c corresponding to the superimposed change in the environmental conditions was taken for further comparative analysis, and the error cumulative distribution function of each algorithm is shown in Figure 8. The results show that under the change of other environmental conditions, the sample probability of localization results of the proposed DJMDAN algorithm is better than other algorithms within the same limited accuracy range.

According to the experimental results, it can be seen that the proposed DJMDAN can obtain a better domain adaptation effect than the traditional transfer learning methods and achieves greater indoor locating accuracy because of the abstract features extracted using a deep one-dimensional convolutional neural network. In addition, DJMDAN considers the overall domain adaptation and the local perspective of the data simultaneously so that the fine-grained information of each category is retained in the domain adaptation process. Therefore, DJMDAN can maintain better robustness and obtain higher localization accuracy to resist environmental changes.

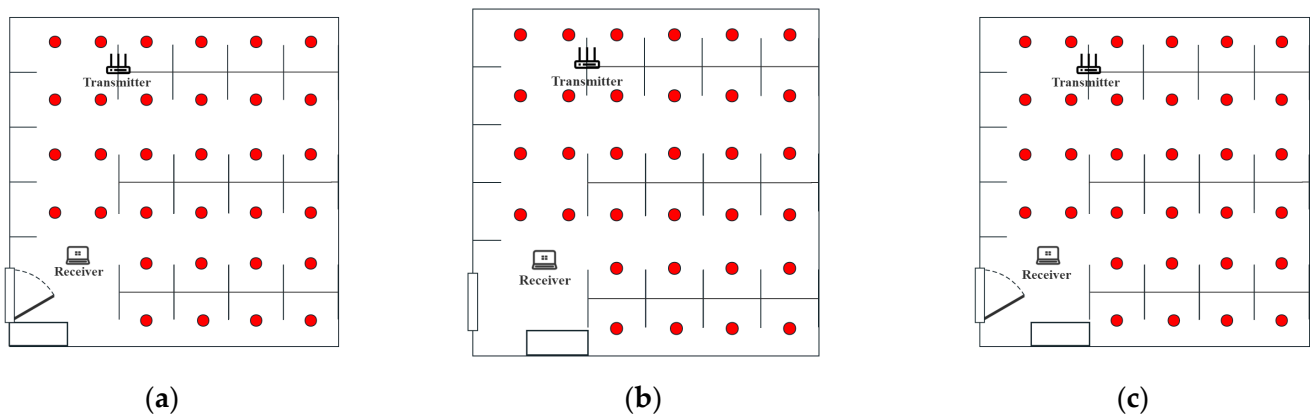


Figure 6. Changes in environmental conditions. (a) Laboratory Door and Window Configuration. (b) Laboratory Locker Placement Adjustment. (c) Combined Changes in Laboratory Door, Window, and Locker Placement.

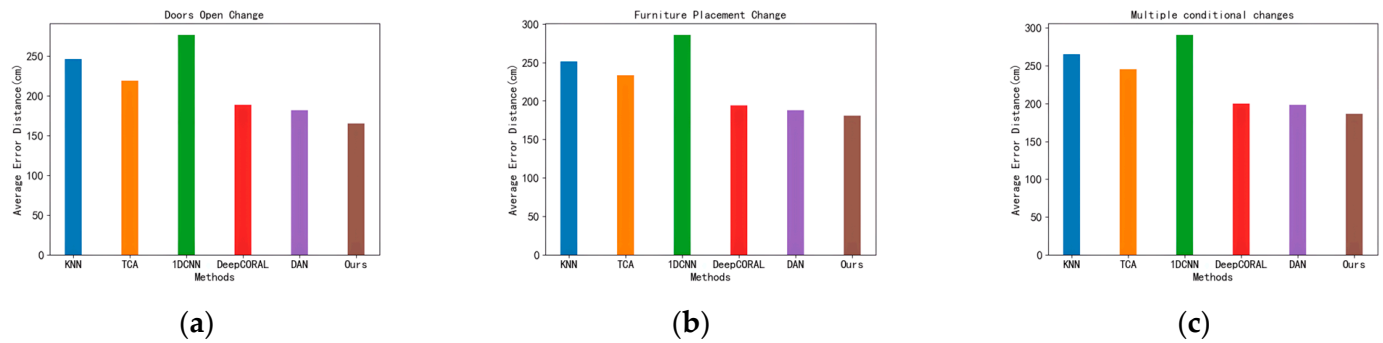


Figure 7. Results of each localization algorithm under the three environmental conditions. (a) Doors Open Change. (b) Furniture Placement Change. (c) Multiple conditional changes.

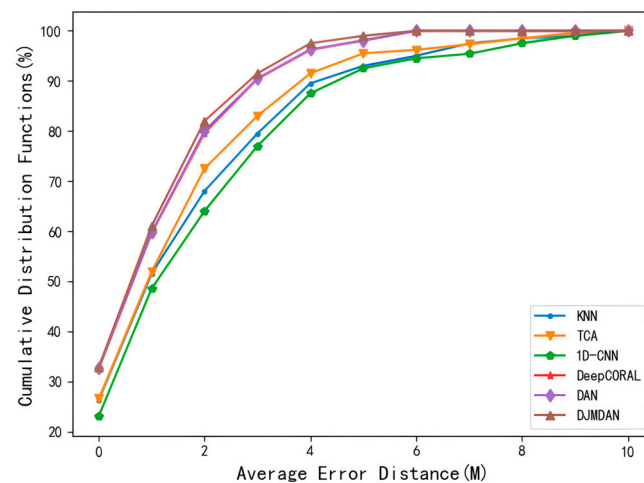


Figure 8. Results of cumulative distribution function for each localization algorithm under superimposed change in environmental conditions.

5. Conclusions

In this paper, a deep domain adaptation localization approach for Wi-Fi CSI signals using transfer learning has been proposed. It can maintain data domain adaptation, both global and local, by adapting the marginal and conditional probability distribution of two-domain data. The proposed method is an update based on DAN, and experiments

have shown that our method outperforms models of the same scale that do not use transfer learning. It also surpasses DAN and other transfer learning methods based on MMD metrics and methods using CORAL metrics. Experimental results show that it can effectively solve the domain shift problem and maintain a relatively ideal localization accuracy in complex and changing indoor environments.

Moving forward, there are two potential areas for further exploration and improvement. Firstly, the current domain adaptation operation is still based on the overall perspective of data, and we could consider introducing some metric functions based on the internal geometric characteristics of the data for domain adaptation work. Secondly, we are currently using a relatively basic one-dimensional convolutional neural network structure. It would be beneficial to select deeper network models that are more suitable for CSI signals as the basic model for deep domain adaptation localization algorithms.

Author Contributions: Conceptualization, J.W. (Jiahao Wang) and Y.F.; methodology, Y.F. and J.W. (Junxiang Wang); software, J.W. (Jiahao Wang) and Y.F.; validation, Y.F. and H.F.; formal analysis, J.W. (Jiahao Wang) and Y.F.; investigation, J.W. (Junxiang Wang) and H.F.; resources, J.W. (Jiahao Wang); data curation, Y.F.; writing—original draft, J.W. (Jiahao Wang) and Y.F.; writing—review and editing, Y.F. and H.F.; visualization, J.W. (Junxiang Wang) and Y.F.; supervision, J.W. (Jiahao Wang) and Y.F.; project administration, J.W. (Jiahao Wang) and J.W. (Junxiang Wang); funding acquisition, J.W. (Jiahao Wang) and J.W. (Junxiang Wang) All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the UESTC-ZHIXIAOJING Joint Research Center of Smart Home under Grant [H04W210180]; Neijiang Technology Incubation and Transformation Funds under Grant [2021KJFH004]; the Science and Technology Program of Sichuan Province, China, under Grants [2022YFG0212] and [2021YFG0024].

Institutional Review Board Statement: No humans or animals were involved in this study.

Informed Consent Statement: No humans or animals were involved in this study.

Data Availability Statement: Data are contained within the article.

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

References

1. Tekler, Z.D.; Low, R.; Yuen, C.; Blessing, L. Plug-Mate: An IoT-based occupancy-driven plug load management system in smart buildings. *Build. Environ.* **2022**, *223*, 109472. [\[CrossRef\]](#)
2. Thakur, N.; Han, C.Y. Multimodal approaches for indoor localization for ambient assisted living in smart homes. *Information* **2021**, *12*, 114. [\[CrossRef\]](#)
3. Low, R.; Tekler, Z.D.; Cheah, L. An end-to-end point of interest (POI) conflation framework. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 779. [\[CrossRef\]](#)
4. Tekler, Z.D.; Low, R.; Gunay, B.; Andersen, R.K.; Blessing, L. A scalable Bluetooth Low Energy approach to identify occupancy patterns and profiles in office spaces. *Build. Environ.* **2020**, *171*, 106681. [\[CrossRef\]](#)
5. Djovic, S.; Stojanovic, I.; Jovanovic, M.; Nikolic, T.; Djordjevic, G.L. Fingerprinting-assisted UWB-based localization technique for complex indoor environments. *Expert Syst. Appl.* **2021**, *167*, 114188. [\[CrossRef\]](#)
6. Alhmiedat, T.; Salem, A.O. A hybrid range-free localization algorithm for zigbee wireless sensor networks. *Int. Arab J. Inf. Technol.* **2017**, *14*, 647–653.
7. El-Absi, M.; Abbas, A.A.; Abuelhaija, A.; Zheng, F.; Solbach, K.; Kaiser, T. High-accuracy indoor localization based on chipless RFID systems at THz band. *IEEE Access* **2018**, *6*, 54355–54368. [\[CrossRef\]](#)
8. Njima, W.; Bazzi, A.; Chafii, M. DNN-based indoor localization under limited dataset using GANs and semi-supervised learning. *IEEE Access* **2022**, *10*, 69896–69909. [\[CrossRef\]](#)
9. Feng, X.; Nguyen, K.A.; Luo, Z. A survey of deep learning approaches for WiFi-based indoor positioning. *J. Inf. Telecommun.* **2022**, *6*, 163–216. [\[CrossRef\]](#)
10. Wang, Y.; Lei, Y.; Zhang, Y.; Yao, L. A robust indoor localization method with calibration strategy based on joint distribution adaptation. *Wirel. Netw.* **2021**, *27*, 1739–1753. [\[CrossRef\]](#)
11. Pan, S.J.; Tsang, I.W.; Kwok, J.T.; Yang, Q. Domain adaptation via transfer component analysis. *IEEE Trans. Neural Netw.* **2011**, *22*, 199–210. [\[CrossRef\]](#) [\[PubMed\]](#)
12. Long, M.; Wang, J.; Ding, G.; Sun, J.; Yu, P.S. Transfer feature learning with joint distribution adaptation. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 2200–2207.

13. Wang, J.; Chen, Y.; Hao, S.; Feng, W.; Shen, Z. Balanced distribution adaptation for transfer learning. In Proceedings of the 2017 IEEE International Conference on Data Mining (ICDM), Orleans, LA, USA, 18–21 November 2017; pp. 1129–1134.
14. Wang, J.; Feng, W.; Chen, Y.; Yu, H.; Huang, M.; Yu, P.S. Visual domain adaptation with manifold embedded distribution alignment. In Proceedings of the 26th ACM International Conference on Multimedia, Seoul, Republic of Korea, 22–26 October 2018; pp. 402–410.
15. Wang, J.; Chen, Y.; Feng, W.; Yu, H.; Huang, M.; Yang, Q. Transfer learning with dynamic distribution adaptation. *ACM Trans. Intell. Syst. Technol. (TIST)* **2020**, *11*, 1–25. [\[CrossRef\]](#)
16. Pan, S.J.; Shen, D.; Yang, Q.; Kwok, J.T. Transferring Localization Models across Space. In Proceedings of the AAAI 2008, Chicago, IL, USA, 13–17 July 2018; pp. 1383–1388.
17. Mager, B.; Lundrigan, P.; Patwari, N. Fingerprint-based device-free localization performance in changing environments. *IEEE J. Sel. Areas Commun.* **2015**, *33*, 2429–2438. [\[CrossRef\]](#)
18. Yang, J.; Zou, H.; Zhou, Y.; Xie, L. Learning gestures from WiFi: A Siamese recurrent convolutional architecture. *IEEE Internet Things J.* **2019**, *6*, 10763–10772. [\[CrossRef\]](#)
19. Wang, H.Y.; Zheng, V.W.; Zhao, J.; Yang, Q. Indoor localization in multi-floor environments with reduced effort. In Proceedings of the 2010 IEEE International Conference on Pervasive Computing and Communications (PerCom), Mannheim, Germany, 29 March–2 April 2010; pp. 244–252.
20. Sun, Z.; Chen, Y.; Qi, J.; Liu, J. Adaptive localization through transfer learning in indoor Wi-Fi environment. In Proceedings of the 2008 Seventh International Conference on Machine Learning and Applications, San Diego, CA, USA, 11–13 December 2008; pp. 331–336.
21. Yosinski, J.; Clune, J.; Bengio, Y.; Lipson, H. How transferable are features in deep neural networks? *Adv. Neural Inf. Process. Syst.* **2014**, *27*, 3320–3328.
22. Ghifary, M.; Kleijn, W.B.; Zhang, M. Domain adaptive neural networks for object recognition. In Proceedings of the PRICAI 2014: Trends in Artificial Intelligence: 13th Pacific Rim International Conference on Artificial Intelligence, Gold Coast, Australia, 1–5 December 2014; Proceedings 13; Springer International Publishing: Berlin/Heidelberg, Germany, 2014; pp. 898–904.
23. Long, M.; Cao, Y.; Wang, J.; Jordan, M. Learning transferable features with deep adaptation networks. In Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 97–105.
24. Long, M.; Zhu, H.; Wang, J.; Jordan, M.I. Deep transfer learning with joint adaptation networks. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 2208–2217.
25. Wang, L.; Shao, Y.; Guo, X. An adaptive localization approach based on deep adaptation networks. In Proceedings of the 2019 International Conference on Control, Automation and Information Sciences (ICCAIS), Hangzhou, China, 24–27 October 2019; pp. 1–5.
26. Zhu, Y.; Zhuang, F.; Wang, D. Aligning domain-specific distribution and classifier for cross-domain classification from multiple sources. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 5989–5996.
27. Schwendemann, S.; Amjad, Z.; Sikora, A. Bearing fault diagnosis with intermediate domain based layered maximum mean discrepancy: A new transfer learning approach. *Eng. Appl. Artif. Intell.* **2021**, *105*, 104415. [\[CrossRef\]](#)
28. Zellinger, W.; Grubinger, T.; Lughofer, E.; Natschlager, T.; Saminger-Platz, S. Central moment discrepancy (cmd) for domain-invariant representation learning. *arXiv* **2017**, arXiv:1702.08811.
29. Sun, B.; Saenko, K. Deep coral: Correlation alignment for deep domain adaptation. In Proceedings of the Computer Vision–ECCV 2016 Workshops, Amsterdam, The Netherlands, 8–10 October 2016; Proceedings, Part III 14. Springer International Publishing: Berlin/Heidelberg, Germany, 2016; pp. 443–450.
30. Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; Marchand, M.; Lempitsky, V. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.* **2016**, *17*, 1–35.
31. Hoffman, J.; Tzeng, E.; Park, T.; Zhu, J.-Y.; Isola, P.; Saenko, K.; Efros, A.; Darrell, T. Cycada: Cycle-consistent adversarial domain adaptation. In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; pp. 1989–1998.
32. Wu, Y.; Zhao, R.; Ma, H.; He, Q.; Du, S.; Wu, J. Adversarial domain adaptation convolutional neural network for intelligent recognition of bearing faults. *Measurement* **2022**, *195*, 111150. [\[CrossRef\]](#)
33. Soleimani, E.; Nazerfard, E. Cross-subject transfer learning in human activity recognition systems using generative adversarial networks. *Neurocomputing* **2021**, *426*, 26–34. [\[CrossRef\]](#)
34. Li, P.; Cui, H.; Khan, A.; Raza, U.; Piechocki, R.; Doufexi, A.; Farnham, T. Deep transfer learning for WiFi localization. In Proceedings of the 2021 IEEE Radar Conference (RadarConf21), Atlanta, GA, USA, 8–14 May 2021; pp. 1–5.
35. Zhou, R.; Hou, H.; Gong, Z.; Chen, Z.; Tang, K.; Zhou, B. Adaptive device-free localization in dynamic environments through adaptive neural networks. *IEEE Sens. J.* **2020**, *21*, 548–559. [\[CrossRef\]](#)
36. Chang, L.; Chen, X.; Wang, Y.; Fang, D.; Wang, J.; Xing, T.; Tang, Z. FitLoc: Fine-grained and low-cost device-free localization for multiple targets over various areas. *IEEE/ACM Trans. Netw.* **2017**, *25*, 1994–2007. [\[CrossRef\]](#)
37. Gretton, A.; Sejdinovic, D.; Strathmann, H.; Balakrishnan, S.; Pontil, M.; Fukumizu, K.; Sriperumbudur, B.K. Optimal kernel choice for large-scale two-sample tests. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1205–1213.

38. Zhu, Y.; Zhuang, F.; Wang, J.; Ke, G.; Chen, J.; Bian, J.; Xiong, H.; He, Q. Deep subdomain adaptation network for image classification. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 1713–1722. [[CrossRef](#)]
39. Orabona, F.; Jie, L.; Caputo, B. Multi Kernel Learning with Online-Batch Optimization. *J. Mach. Learn. Res.* **2012**, *13*, 227–253.
40. Ganin, Y.; Lempitsky, V. Unsupervised domain adaptation by backpropagation. In Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 1180–1189.
41. Venkateswara, H.; Eusebio, J.; Chakraborty, S.; Panchanathan, S. Deep hashing network for unsupervised domain adaptation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5018–5027.
42. Long, M.; Zhu, H.; Wang, J.; Jordan, M.I. Unsupervised domain adaptation with residual transfer networks. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 136–144.

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.