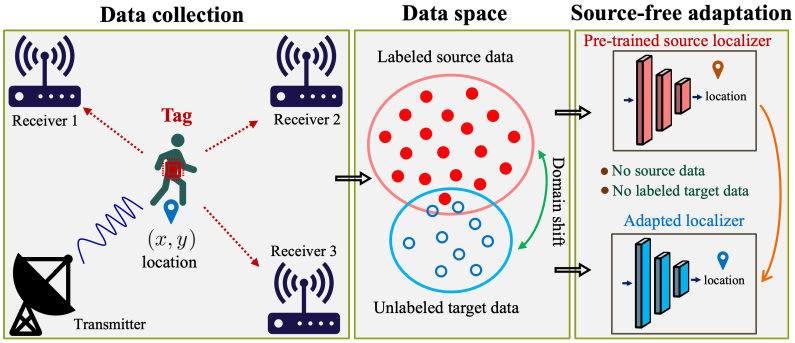
# MTLoc: A Confidence-Based Source-Free Domain Adaptation Approach For Indoor Localization

Negar Mehregan, Berk Bozkurt, Eric Granger,
Mohammadjavad Hajikhani, and Mohammadhadi Shateri

arXiv:2503.14767v2 [eess.SP] 16 Dec 2025

***Abstract*— Indoor localization using radio-frequency identification (RFID) has benefited from deep learning, yet models trained in a labeled source environment often degrade when deployed in a different, unlabeled target environment. Unsupervised domain adaptation (UDA) aims to mitigate this distribution shift by aligning a source-trained model with target-domain data. In practice, the source dataset is frequently unavailable at adaptation time due to privacy and resource constraints. This motivates source-free domain adaptation (SFDA); however, most SFDA methods have been developed for classification, and extending them to indoor localization (regression) is challenging, especially when target datasets are small and noisy. Motivated by above limitations, we introduce MTLoc, a source-free mean-teacher approach for indoor localization. MTLoc includes a student and a teacher network: the student network is updated using noisy target data with teacher-generated pseudo-labels. The teacher network maintains stability through exponential moving averages. To further ensure robustness, we propose a correction mechanism in which the teacher's pseudo-labels are refined using k-nearest neighbor correction. MTLoc allows for self-supervised learning on target data, facilitating effective adaptation to dynamic and noisy indoor environments. Validated using real-world data from our experimental setup with INLAN Inc., our results[1] show that MTLoc achieves high localization accuracy under challenging conditions, significantly reducing distance error compared to baselines. On average, it reduces MAE by 20.0% on Cross and 22.5% on Square datasets. With confidence correction, these improvements reach 23.9% and 28.2% respectively.**




***Index Terms*— Source-free domain adaptation, Indoor localization, Regression, Deep learning, Pseudo-labeling.**

IEEE Sensors Council

NOMENCLATURE

| | |
|---|---|
| $X, Y$ | Random variables |
| $x, y$ | Realizations for $X, Y$ |
| $p(x)$ | Probability distribution of $X$ |
| $p(x \mid y)$ | Conditional distribution of $X$ given $Y$ |
| $\mathbb{E}[X]$ | Expectation of the random variable $X$ |
| EMA ($\alpha$) | Exponential moving average with parameter $\alpha$ |
| GRL | Gradient Reversal Layer |
| $k$-NN | $k$-nearest-neighbor |

## I. Introduction

ROBUST and reliable localization systems aim to revolutionize positioning in fields like the Internet of Things [1], and Augmented Reality [2]. Various indoor localization techniques are being studied, in which WiFi-based methods are attracting the most attention due to their compatibility, cost-effectiveness, and wide applicability [3]–[5].

As GPS signals are proven to be ineffective for localization due to signal deflection and multipath-related issues [6], different technologies are introduced, with qualifications varying based on the type of tag applied. Notably, ultra-wideband (UWB) tags [7], [8] with high-precision accuracy and low power usage, radio frequency identification (RFID) tags [9], [10] with lower cost and long-range communication, and Bluetooth low energy (BLE) tags [11], [12], a medium range, energy-efficient technology, are used in localization models.

Tag-based tracking models are used to monitor key assets' location, ensure safety, and localize objects for healthcare applications [13]–[15]. A typical indoor positioning system includes receivers, transmitters, and tags for data collection,

N. Mehregan, E. Granger, and M. Shateri are with the LIVIA, Department of Systems Engineering, ETS Montreal, Canada (e-mails: negar.mehregan.1@ens.etsmtl.ca, eric.granger@etsmtl.ca, mohammadhadi.shateri@etsmtl.ca).
B. Bozkurt and M. Hajikhani are with INLAN Inc., Montreal, Canada (e-mails: b.bozkurt@inlantech.com, m.hajikhani@inlantech.com).

[1]Code and datasets: https://github.com/negarmehregan/Indoor-Localization/tree/main

where a signal is transmitted to the receiver after interacting with the tagged object. In various studies, through using receivers, different types of data were acquired from tags including received signal strength [16]–[18], Phase, Channel Impulse Response [19], [20], channel state information (CSI) [21], [22], angle of arrival [23]–[25], and radio frequency [26], [27]. However, the inaccuracies caused by signal attenuation and distortion remain a challenge, particularly in complex environments in which multipath components and non-line-of-sight effects stand out as the most concerning [20], [28]. To address these issues, various deep learning-based techniques are proposed for indoor localization.

Previous studies highlight the impact of environmental change on data [29], [30]. As mentioned by [31], CSI data is affected by any changes in the environment, hence, a localization system might not perform well in a new environment. Therefore, unsupervised domain adaptation (UDA) methods have been proposed, where the source domain model is fine-tuned to be applicable in a distinct but related target domain without access to ground truth. Given a pre-trained model utilizing labeled source data collected from Warehouse A, the objective is to evaluate its performance in a different environment, Warehouse B. The primary challenge lies in the fact that the new domain (Warehouse B) may produce substantially different object-related signals, possibly due to variations in the positions of transmitters and receivers. Retraining the model is infeasible due to the lack of annotated data from Warehouse B, falling under the framework for UDA.

Nevertheless, UDA raises concerns regarding data privacy, particularly when sensitive information is present in the source domain [32]. To avoid data leakage, source-free domain adaptation (SFDA) has been proposed, which enables robust adaptation without access to source data. SFDA not only respects privacy concerns but also reduces computational costs and facilitates storage and transfer [33], [34]. Additionally, regulatory constraints or corporate policies often restrict data transfer, further motivating SFDA as a practical approach [34]. The scalability of SFDA also supports adaptation across diverse environments [34], [35]. While most SFDA methods target classification, TASFAR [36] is a recent regression-based approach that estimates the target label distribution by modeling predictive uncertainty via a label-density map. It generates pseudo-labels for target samples to finetune the source model.

To the best of our knowledge, no method for SFDA has been proposed for indoor localization. Of the current SFDA methods, the mean teacher framework is particularly well-suited for regression-based indoor localization. In practice, target datasets for indoor localization are often limited and noisy, making domain adaptation reliant on highly confident pseudo-labels. Researchers such as Yeonguk et al. [37] have used this technique due to its EMA-based stabilization, which produces stable pseudo labels through consistency regularization. Although it is a simple approach, it ensures efficient performance in source-free scenarios while leveraging self-supervision.

Motivated by these characteristics, we introduce mean teacher localization (MTLoc) for SFDA, a self-supervised method leveraging a mean teacher framework. This approach uses a teacher-student architecture, where both networks are initialized with a localizer pre-trained on source data. The student model is fine-tuned on noisy target domain data using pseudo labels generated by the teacher network, enabling effective adaptation without access to source data. To further improve performance, we propose an enhanced model, MT-Loc with confidence, where the teacher's uncertain pseudo-labels are corrected using $k$-nearest neighbors ($k$-NN), and the student model is trained exclusively on the most confident ones. This refinement improves reliability by focusing on high-certainty predictions, reducing variations, and enhancing robustness.

The **main contributions** of this paper are as follows.

- We present the MTLoc, an SFDA method for indoor localization, by leveraging a self-supervised knowledge distillation framework to adapt from small and noisy target data.
- An improved confidence-based variant of MTLoc is proposed, where uncertain pseudo labels generated by the teacher network are refined using a $k$-NN strategy to compute a weighted average of the $k$ most confident pseudo labels. This ensures that the student network is trained exclusively on high-confidence labels, enhancing overall model robustness.
- A comprehensive data collection process was conducted to develop three datasets—Ceiling (source), Cross, and Square (targets)—at the INLAN Inc. testing facility. This setup involved four receivers, a transmitter equipped with dipole antennas, and an RFID tag, providing a robust evaluation environment for the proposed method. MTLoc is validated on this real-world data and significantly outperforms the state-of-the-art UDA and SFDA models.

This paper is organized as follows. Sections II provides related work. Section III defines the SFDA problem and baseline. Section IV presents our proposed confidence-based SFDA method. Section V details data collection, experimental protocol, performance measures, and results analysis.

## II. Related Work

Over the years, several approaches have been proposed for indoor localization. Some rely on kernel-based methods, such as Zhang et al. [38], who developed an integrated multiple-kernel extreme learning machine that uses temporal and spatial CSI features for device-free localization in cluttered environments. However, AI-based techniques have become the dominant trend. For instance, with MFFALoc [21], CSI fingerprints are fed into a convolutional neural network (CNN), tackling multipath and non-line-of-sight effects. Rao et al. [21] further remove environmental noise by utilizing waveform diversity and detection, and implement a feature fusion system to merge multi-antenna feature matrices, resulting in fine-grained fingerprints. For example, MFFALoc [21] use a CNN on CSI fingerprints to tackle multipath and non-line-of-sight effects. In addition, Rao et al. [21] improve fingerprint quality by using waveform diversity and detection to suppress environmental noise, and by fusing multi-antenna feature

matrices to construct fine-grained representations. In parallel, Zhang et al. [39] address domain adaptation by using online learning, where fingerprint augmentation and a sequential extreme learning machine with a forgetting mechanism allow the model to adapt dynamically to environmental changes. Some studies, such as Chen et al. [40], apply 1D-CNN for RF-based environment classification, while others, like Yan et al. [41], use 2D deep CNNs for CSI feature extraction to improve localization accuracy. Se-Loc [42] is another CNN-driven approach with a security-enhanced semi-supervised indoor localization technique, integrating a denoising autoencoder (AE) to improve fingerprinting. This model employs a Pearson correlation coefficient-based access point selection system to eliminate contaminated access points. Cao et al. [43] proposed a confidence-aware mean teacher framework for metallographic image semantic segmentation, which adaptively calibrates pseudo-labels with class-aware thresholds and improves data augmentation through confidence-driven patch mixing.

Some research works, such as [44], use AE for spatial feature extraction. Likewise, in [45] and [28], a variational AE is applied to extract key position-related features and reduce environmental noise to solve localization and multipath challenges. Some works merge recurrent neural networks with AE, such as [46], where a stacked AE is combined with an attention-based long short-term memory network, enhancing localization accuracy. Liu et al. [47] use gated recurrent units for feature extraction, while Chen et al. [48] apply a deep LSTM with a local feature extractor to process RSSI fingerprints, capturing temporal dependencies through a regression layer. Generative models have also been explored for data augmentation [49]–[51], improving localization accuracy.

In fault diagnosis settings, Wang et al. [52] provide a comprehensive survey of domain adaptation, focusing on both homogeneous methods, including metric learning, adversarial learning, and feature reconstruction, along with heterogeneous settings, source-free, domain generalization, partial, open-set, and universal domain adaptation methods.

Currently, UDA approaches can be classified into three main categories [53]: (a) domain distribution alignment [54]–[56], (b) adversarial learning to extract domain-invariant representation [57]–[59], and (c) model fine-tuning through self-supervision [60]–[62]. In indoor localization, most UDA approaches rely on adversarial learning through gradient reversal layers (GRL). Examples include Zhang et al. [63], who use a semi-supervised graph convolutional network with RSSI data; Xu et al. [64], who propose a Gaussian noise–infused co-teaching mechanism; Zhang et al. [65], who develop dynamic adversarial adaptation networks to eliminate inconsistencies in global and local discriminators; and Prasad et al. [66], who apply an AE for dimension reduction alongside a GRL to create domain-invariant features.

Discrepancy-based models aligning source and target distributions have also been explored, such as [67], which uses the Grassmann manifold for dimensionality reduction while preserving data integrity, combined with maximum mean discrepancy for accuracy. Hybrid mechanisms combining generative and GRL-based approaches also exist, such as [68] and [31], where generative models (variational AE and adversarial AE, respectively) are used for augmentation. In [68], a domain-adaptive classifier includes a feature extractor with classification-reconstruction layers, while in [31], a CNN feature extractor is coupled with GRL-based discriminator and predictor networks.

As discussed earlier, SFDA overcomes the limitations of UDA regarding privacy, storage, and transfer constraints. Fang et al. [33] provide a detailed survey of SFDA in image-based studies, highlighting its efficiency and scalability. However, no SFDA methods have yet been applied to indoor localization, leaving an open gap that this paper addresses.

## III. Problem Definition and Baseline Overview

### A. Problem Formulation:

Consider an indoor localization framework consisting of $N$ receivers and a single transmitter. The transmitter emits signals that interact with objects within the environment, and these signals are subsequently captured by the receivers. The objective is to develop an object localization system by analyzing the signals recorded by the receivers. Let $Z$ denote the random variable representing the signals recorded by the receivers, and let $(X, Y)$ represent the joint random variables corresponding to the true location associated with the recorded signal $Z$. The goal is to learn an object localizer defined as the mapping $\mathcal{M}(\cdot;\theta) : \mathcal{Z} \to \mathcal{X} \times \mathcal{Y}$ parameterized by $\theta$, which produces an approximation $(\hat{X}, \hat{Y})$ of the true location for each observed signal $Z$, i.e $(\hat{X}, \hat{Y}) = (Z;\theta)$. To optimize the parameters $\theta$ of the localizer, we employ Bayesian optimization by maximizing the log-likelihood of the true location given the estimated location as follows:

$$\theta^* = \arg\max_{\theta} \log p_{X,Y|\hat{X},\hat{Y}}(x, y \mid \hat{x}, \hat{y}), \tag{1}$$

where depending on the assumed statistical model of the localization errors, this maximum likelihood estimation leads to different loss functions.

*Gaussian Error Model* [69]: Assuming the localization errors $(X - \hat{X}, Y - \hat{Y})$ follow a bivariate Gaussian distribution with zero mean and covariance matrix $\Sigma$, the negative log-likelihood function becomes:

$$-\log p(x, y \mid \hat{x}, \hat{y}) \propto \begin{pmatrix} x - \hat{x} \\ y - \hat{y} \end{pmatrix}^{\top} \Sigma^{-1} \begin{pmatrix} x - \hat{x} \\ y - \hat{y} \end{pmatrix}. \tag{2}$$

If $\Sigma$ is proportional to the identity matrix, the expression simplifies, and maximizing the likelihood presented in Equ. (1) is equivalent to the following problem:

$$\theta^* = \arg\min_{\theta} \mathbb{E}_{p_{X,Y,Z}} \left[ (X - \hat{X})^2 + (Y - \hat{Y})^2 \right], \tag{3}$$

where the expectation is taken over the joint distribution of $p_{X,Y,Z}$.

*Laplacian Error Model* [69]: Alternatively, considering the localization errors follow a bivariate Laplacian distribution and assuming independence between error in $X$ and $Y$, the joint probability density function can be written as follows:

$$p(x, y \mid \hat{x}, \hat{y}) = \left( \frac{1}{\sqrt{2}b} \right)^2 \exp\left( -\frac{|x - \hat{x}| + |y - \hat{y}|}{\sqrt{2}b} \right), \tag{4}$$

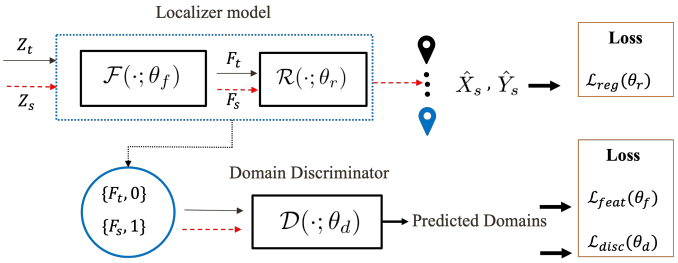



Fig. 1: *System overview:* The source data $Z_s$ and the unlabeled target data $Z_t$ are processed through our Localizer model, which comprises a feature extractor $\mathcal{F}(\cdot;\theta_f)$ and a regressor $\mathcal{R}(\cdot;\theta_r)$. The source data $Z_s$ is used to generate predictions, while the feature extractor $\mathcal{F}(\cdot;\theta_f)$ is trained to deceive the discriminator $\mathcal{D}(\cdot;\theta_d)$.

where $b > 0$ is the scale parameter. In this case, the negative log-likelihood function is simplified to:

$$-\log p(x, y \mid \hat{x}, \hat{y}) = \frac{1}{\sqrt{2}b}\left(|x-\hat{x}|+|y-\hat{y}|\right) + 2\log(\sqrt{2}b). \tag{5}$$

Ignoring the constant terms, maximizing the likelihood in Equ. (1) corresponds to the following problem:

$$\theta^* = \arg\min_{\theta} \mathbb{E}_{p_{X,Y,Z}}\left[|X-\hat{X}|+|Y-\hat{Y}|\right]. \tag{6}$$

In this work, either formulation (3) or (6) could be used for optimizing the parameters $\theta$ of localizer $\mathcal{M}(\cdot;\theta)$, however, we applied equation (6). After training $\mathcal{M}(\cdot;\theta)$ using data from a source domain $\mathbf{D}_s$ characterized by the joint distribution $p^s_{X,Y,Z}$, we aim to adapt it to a new target (operational) domain $\mathbf{D}_t$, where the joint distribution is $p^t_{X,Y,Z} \neq p^s_{X,Y,Z}$. The discrepancy between the source and target domains poses a challenge, as the localizer trained on $\mathbf{D}_s$ may perform poorly on $\mathbf{D}_t$ due to distribution shifts. Our objective is to adapt $\mathcal{M}(\cdot;\theta)$ to perform well in the target domain using an unlabeled $\mathbf{D}_t$.

### B. Baselines:

In this study, we considered several baselines including a source-only pre-adaptation model and an oracle approach trained on labeled target datasets [70], [71]. Among UDA approaches created for indoor localization, adversarial UDA and discrepancy-based models are prominent, with the state-of-the-art focusing predominately on the use of GRL-based adversarial UDA techniques. Thereby, we evaluated an adversarial UDA method that leverages a GRL-based model. This approach is inspired by FreeLoc [31], an adaptation of Domain-Adversarial Neural Network (DANN) [72], which facilitates domain alignment through adversarial training, and serves as another baseline. As illustrated in Fig. 1, the framework assumes access to a labeled source dataset, $\mathbf{D}_s = \{z_s^{(i)}, (x_s^{(i)}, y_s^{(i)})\}_{i=1}^{N_s}$ with distribution $p^s_{X,Y,Z}$, and an unlabeled target dataset, $\mathbf{D}_t = \{z_t^{(j)}\}_{j=1}^{N_t}$ with distribution $p^t_Z$. Here, $z$ represents the signals received by the receivers, while $(x, y)$ denotes the object location in the corresponding domain.

This framework leverages a localizer model that comprises two core components: a feature extractor $\mathcal{F}(\cdot;\theta_f)$, which maps input signals to a latent feature space, and a regressor $\mathcal{R}(\cdot;\theta_r)$, which predicts the object's location based on the extracted features. Additionally, a domain discriminator $\mathcal{D}(\cdot;\theta_d)$ is incorporated to facilitate domain adaptation. During the adaptation process, the feature extractor processes received signals from both the source and target domains, while the domain discriminator is trained to distinguish between features from these domains. Concurrently, the regressor is optimized using labeled source data to predict object locations accurately. The feature extractor, however, is adversarially trained to both confuse the domain discriminator and preserve localization accuracy for the regressor.

The training objective is formalized such that the feature extractor aims to align the source and target domains in the feature space. This alignment occurs when the domain discriminator can no longer differentiate between the two domains, thus enabling the regressor trained on source data to generalize effectively to the target domain.

$$\min_{\theta_f,\theta_r}\max_{\theta_d} \mathbb{E}\left[\log \mathcal{D}(\mathcal{F}(Z_s))\right] + \mathbb{E}\left[\log(1-\mathcal{D}(\mathcal{F}(Z_t)))\right] + \mathbb{E}\left[|X_s-\hat{X}_s|+|Y_s-\hat{Y}_s|\right]. \tag{7}$$

The first two terms represent the adversarial objective, where the feature extractor $\mathcal{F}$ seeks to mislead the domain discriminator $\mathcal{D}$. The final term represents the regression loss, encouraging the regressor $\mathcal{R}$ to minimize the distance between the predicted and true object locations for the source domain. Through this adversarial training, feature alignment between the source and target domains is achieved, enabling robust location prediction in the target domain. It should be mentioned that for training each model, the following loss functions are defined based on Equ. (7) where the expectations are approximated empirically. More precisely, let $\{z_s^{(b)}, (x_s^{(b)}, y_s^{(b)})\}_{b=1}^{B}$ and $\{z_t^{(b)}\}_{b=1}^{B}$ be a set of $B$ data samples from source and target respectively. The loss functions for training each network in Fig. 1 are presented in equations (8)(9)(10).

$$\mathcal{L}_{\text{reg}}(\theta_r) = \mathbb{E}_{p^s_{X,Y,Z}}\left[|X_s-\hat{X}_s|+|Y_s-\hat{Y}_s|\right] \approx \frac{1}{B}\sum_{b=1}^{B}\left[|x_s^{(b)}-\hat{x}_s^{(b)}|+|y_s^{(b)}-\hat{y}_s^{(b)}|\right], \tag{8}$$

$$\mathcal{L}_{\text{disc}}(\theta_d) = -\mathbb{E}_{p^s_Z}\left[\log \mathcal{D}(\mathcal{F}(Z_s))\right] - \mathbb{E}_{p^t_Z}\left[\log(1-\mathcal{D}(\mathcal{F}(Z_t)))\right] \approx \frac{-1}{B}\sum_{b=1}^{B}\left[\log \mathcal{D}(\mathcal{F}(z_s^{(b)})) + \log(1-\mathcal{D}(\mathcal{F}(z_t^{(b)})))\right], \tag{9}$$

$$\mathcal{L}_{\text{feat}}(\theta_f) = \mathcal{L}_{\text{reg}} - \mathcal{L}_{\text{disc}}. \tag{10}$$

The detailed training procedure of the feature extractor, regressor, and discriminator for our baseline model adversarial UDA is given in Algorithm 1 which illustrates how these networks are updated depending on their functionality. In the next section, our proposed method for indoor localization will be discussed. Unlike state-of-the-art approaches, our UDA

method is source-free, meaning that it does not rely on access to source data $\mathbf{D}_s$ for adaptation.

**Algorithm 1:** Domain Adaptation with Access to Source Data

**Input:** Labeled Source data $Z_s$, Target data $Z_t$
**Initialize:** Feature Extractor $\mathcal{F}(\cdot;\theta_f)$, Regressor $\mathcal{R}(\cdot;\theta_r)$, and Discriminator $\mathcal{D}(\cdot;\theta_d)$
**while** not converged **do**
  Extract features: $F_s = F(Z_s;\theta_f)$, $F_t = F(Z_t;\theta_f)$
  Update $\theta_r$ by $\nabla_{\theta_r}\mathcal{L}_{reg}$(Equ. (8))
  Train Discriminator $\mathcal{D}(\cdot;\theta_d)$ on $(F_s, F_t)$
  Update $\theta_F$ by $\nabla_{\theta_f}\mathcal{L}_{feat}$ (Equ. (10)) Received Signal Strength Indicator data
  Update $\theta_D$ by $\nabla_{\theta_d}\mathcal{L}_{disc}$ (equation (9))
**end while**

As additional UDA baselines, we consider several alignment techniques. ADDA [73] performs adversarial marginal alignment by freezing the source encoder and training a target encoder adversarially against a domain discriminator so that target features match the source distribution, then reusing the source regressor. FiDO [29], an autoencoder-inspired model, enforces reconstruction-based invariance by pairing a feature extractor and localizer with a parallel reconstruction head; it is trained on labeled source and unlabeled target data to learn domain-invariant features. DAFI [30] applies class-conditional adversarial alignment with a shared feature extractor and localizer, using two GRL-trained domain discriminators (one feature-level and one class-conditioned). In addition, we adapted SHOT [74], an SFDA method originally designed for classification, for the regression task.

## IV. The Proposed MTLoc Method

In this study, an SFDA approach is proposed for indoor localization, called mean teacher localization (MTLoc), illustrated in Fig. 2. As the current approaches are conditioned on the availability of the source domain data, making them not feasible in real-world scenarios [33], we developed a source-free model. In this technique, two models are trained to ensure the quality and effectiveness of the training. This approach implements two networks: 1) teacher network $\mathcal{T}(\cdot;\theta_t)$ [33] 2) student network $\mathcal{S}(\cdot;\theta_s)$ [33] where the Teacher model is developed to have similar insights to the Students', aiding in efficient generalization to new data. In the first stage, the Teacher and student networks are initialized using a pre-trained source localizer, $\mathcal{M}(\cdot;\theta)$ with a regression loss, minimizing the average absolute differences from the predicted and ground-truth parameters is shown in the Equ. (8).

In this approach, the teacher network receives the unlabeled target data $Z_t$, while the student network receives an augmented version by using an additive Gaussian noise $\mathcal{N}(0,\sigma^2)$ with a pre-determined variance $\sigma^2$ [75]. teacher network $\mathcal{T}(\cdot;\theta_t)$ generates pseudo labels, $(X'_t, Y'_t)$, for each sample received from $Z_t$. These newly derived pseudo labels are considered as the ground truth for Student model $\mathcal{S}(\cdot;\theta_s)$ which would be used while proceeding with predictions, $(\hat{X}_t, \hat{Y}_t)$. During the training process, the teacher network's parameters are updated by applying an Exponential Moving

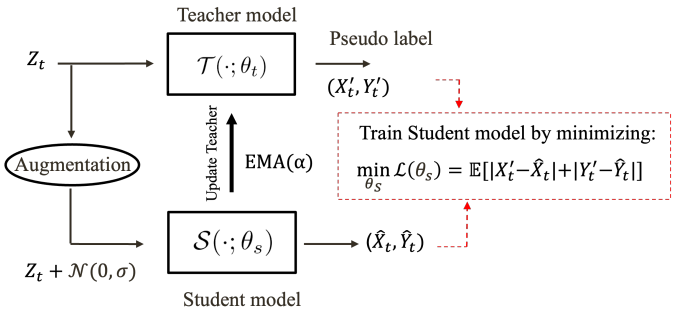



Fig. 2: *MTLoc: The unlabeled target data ($Z_t$) is passed to the Teacher model $\mathcal{T}(\cdot;\theta_t)$ to generate pseudo labels $(X'_t, Y'_t)$, while undergoing an augmentation process, through Gaussian Noise, allowing the Student model $\mathcal{S}(\cdot;\theta_s)$ to proceed with the prediction $\hat{X}_t, \hat{Y}_t$ while minimizing the Knowledge Distillation Loss $L_{KD}$.*

Average (EMA) process based on the Students', with a pre-determined decay rate $\alpha$. As the student network is trained to imitate the Teacher's performance and replicate its generated pseudo labels, by applying the loss function (11), any data degradation would be minimized, resulting in a stable and aligned network.

$$\begin{aligned}\mathcal{L}(\theta_s) &= \mathbb{E}\left[|X'_t - \hat{X}_t| + |Y'_t - \hat{Y}_t|\right] \\ &\approx \frac{1}{B}\sum_{b=1}^{B}\left[|x_t'^{(b)} - \hat{x}_t^{(b)}| + |y_t'^{(b)} - \hat{y}_t^{(b)}|\right].\end{aligned} \tag{11}$$

**Algorithm 2:** Confidence-based source-free UDA with self-supervised knowledge distillation

**Given:** A localizer with parameters $\mathcal{M}(\cdot;\theta)$, (unlabeled) target data $z_t$, and decay rate $\alpha$
Initialize parameters of Teacher $\mathcal{T}(\cdot;\theta_t)$ & Student $\mathcal{S}(\cdot;\theta_s)$ with $\mathcal{M}(\cdot;\theta)$
**for** number of adaptation steps **do**
  Determine confident scores, correct labels (Algorithm 3)
  Get a batch of target data $\{z_t^{(b)}\}_{b=1}^{B}$ with associated Pseudo Labels $\{(x'_t, y'_t)^{(b)}\}_{b=1}^{B}$.
  Do data augmentation at the Student's input $\{z_t^{(b)} + \epsilon^{(b)}\}$ where $\epsilon^{(b)} \sim \mathcal{N}(0,\sigma)$
  Update Student by $\nabla_{\theta_s}[\mathcal{L}(\theta_s)]$ (see equation 11)
  Update Teacher using EMA: $\theta_t \leftarrow \alpha \cdot \theta_s + (1-\alpha)\cdot\theta_t$
**end for**

To enhance the performance of our MTLoc model, we developed an enhanced approach, built on a confidence-based methodology, illustrated as Fig. 3, called MTLoc with confidence, in which every unconfident sample $z_t$ and its generated pseudo labels undergo a correctional process, modifying the uncertain predictions. This aids the Student model in being trained on more confident labels. In our correction module, $\mathcal{T}(\cdot;\theta_t)$ generates pseudo labels $(x'_t, y'_t)$ using a noisy version of the unlabeled target data $z_t$. By repeating this process, the standard deviation $\sigma$ of the measurements is determined. Each sample is tested 10 times, and these repeated trials provide the basis for calculating the standard deviations for both $x$ and $y$.

These uncertain variables would be corrected by using a $k$NN-based technique, if the standard deviation of generated

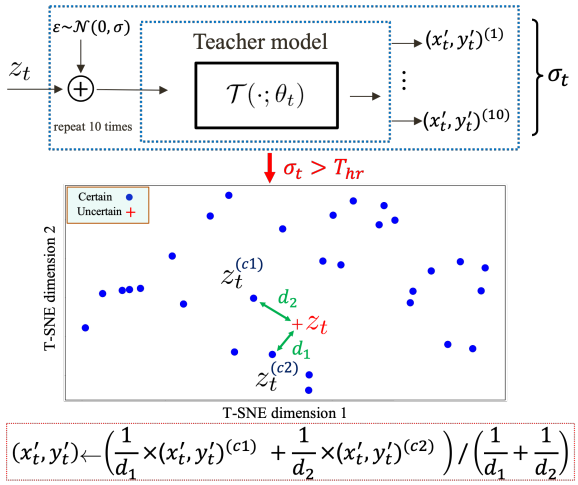



Fig. 3: *MTLoc with confidence: $z_t$ together with the generated pseudo labels $(x'_t, y'_t)$ go through a correction process in which the uncertain labels would be identified based on a threshold $(T_{hr})$ and modified using $k$-NN (with $k = 2$). The t-SNE plot shown above highlights an uncertain sample $z_t$ being corrected using two nearest certain ones. The newly modified pseudo label $(x'_t, y'_t)$ is considered as ground truth for our Student model to generate predictions and update the Teacher model with EMA while applying distillation loss.*

labels is higher than an assigned Threshold $(T_{hr})$. For additional details regarding the implementation of the threshold, please refer to Appendix C. In this mechanism, $k$NN is applied in the input space, extracting $k$ nearest confident samples. Thereafter, an uncertain pseudo label $(x'_t, y'_t)$ would be modified by performing a weighted average of the $k$ confident neighboring labels as illustrated below:

$$(x'_t, y'_t) \leftarrow \frac{\sum_{i=1}^{K} \frac{1}{d_i} \times (x'_t, y'_t)^{(c_i)}}{\sum_{i=1}^{K} \frac{1}{d_i}}, \tag{12}$$

with $d_i$ being the Euclidean Distance between the uncertain and the i-th closest confident samples while the $(x'_t, y'_t)^{(c_i)}$ is the associated confident pseudo label.

**Algorithm 3:** Confidence scores and label generation process for SFDA approach

1: **Input:** Unlabeled target data $z_t$, Teacher Model $\mathcal{T}(\cdot;\theta_t)$
2: Compute $(x'_t, y'_t)$ for 10 runs with additive noise $N(0,\sigma)$
3: Identify uncertain pseudo labels as those with $\sigma_t > T_{hr}$ where $\sigma_t$ is the standard deviation of the 10 runs and $T_{hr}$ is a pre-determined threshold.
4: **for** each uncertain sample $z_t$ **do**
5: Find the $k$-nearest confident samples by applying $k$-nearest neighbors to (input) data space.
6: Correct the Pseudo Label $(x'_t, y'_t)$ by using the weighted average of the labels of the confident samples (see equation 12)
7: **end for**

The training along with the specifics of the pseudo label correctional process of our introduced confidence-based MTLoc model is detailed in Algorithms 2 and 3.

## V. Results and Discussion

### A. Experimental Setup

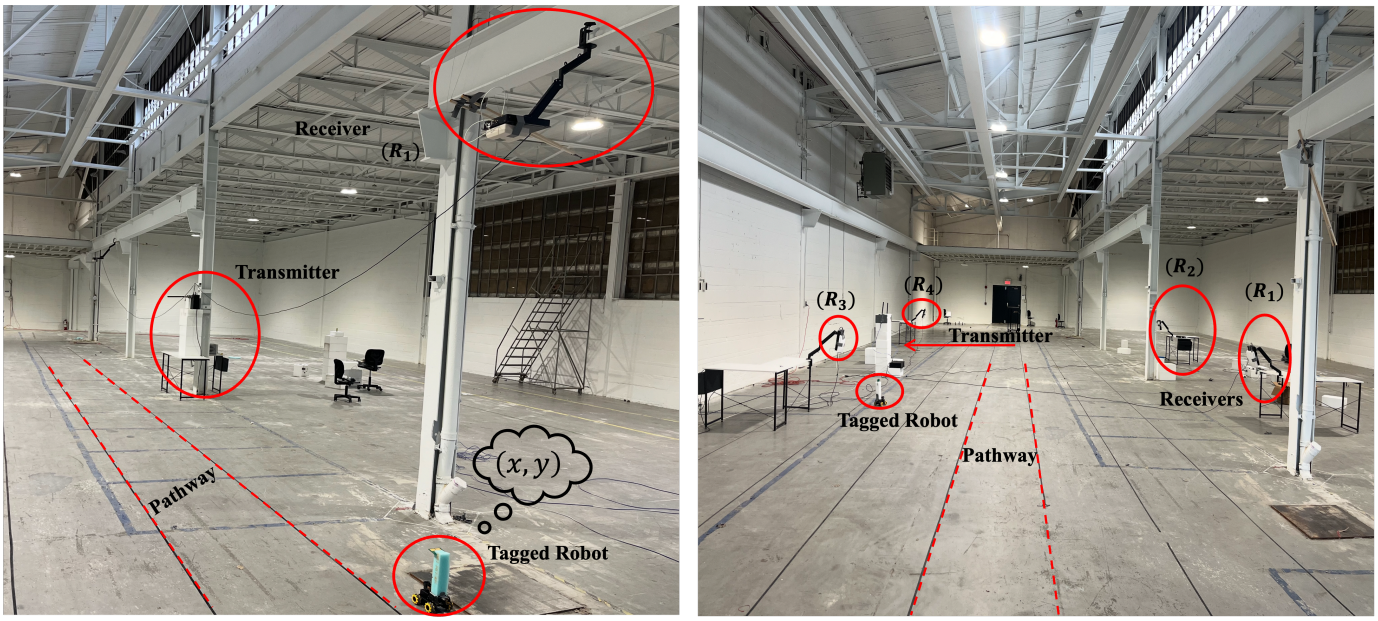
#### 1) Environment and devices

The experiments were performed in INLAN's testing facility [76] where a transmitter and four receivers are deployed in different configurations that are specified in the datasets provided (see Fig. 4). The transmitter is equipped with a dipole stick antenna (2 stick antennas for cross and square datasets) with 2.15 dBi gain, and each receiver is equipped with a patch antenna with 16 dBi gain. Note that for the locations of the devices, the bottom left corner of the warehouse is considered as the origin, all units are in terms of meters and the positions are given in the form $(x, y)$ where $x$ and $y$ denotes the horizontal and vertical axes, respectively. While both cross-environment and within-environment domain shifts are possible, in this work we focus on the latter by treating changes in sensor placement and layout as distinct domains. Configuration details are as follows:

- Ceiling configuration: Transmitter is located at $(8.5, 13)$ meter, receivers are located at the points $(8.5, 5), (0.5, 13.6), (8.5, 21.9), (17.5, 14)$. The transmitter antenna is placed parallel to the ground at a height of 1.8 m. Receivers are mounted on the ceiling at a height of 3 m with each patch antenna parallel to the ground. The tag is also parallel to the ground and elevated at a height of 0.36 m.
- Square configuration: Transmitter is located at $(5, 9)$ meter, receivers are located at the points $(0, 1.5), (0, 15), (11, 14), (11, 2.5)$. Transmitter antennae separated by 0.4 m are placed perpendicular to the ground at a height of 1.8 m. Receivers are mounted on tables at a height of 1.4 m with each patch antenna perpendicular to the ground at a slight angle to capture the tag's movement. The tag is also perpendicular to the ground and elevated at a height of 0.4 m.
- Cross configuration: Transmitter is located at $(5, 9)$ meter, receivers are located at the points $(0, 8), (6.5, 17.5), (12.5, 8.8), (6.3, 0)$. Transmitter antennae separated by 0.4 m are placed perpendicular to the ground at a height of 1.8 m. Receivers are mounted on tables at a height of 1.4 m with each patch antenna perpendicular to the ground at a slight angle to capture the tag's movement. The tag is also perpendicular to the ground and elevated at a height of 0.4 m.

#### 2) Operational specifications

The dataset utilized in this paper was obtained using the INLAN RFID system. Unlike conventional RFID systems, which are limited to a sub-10-meter reading range, the INLAN system enables ultra-long-range reading capability over 30 meters, positioning RFID technology as a viable solution for real-time localization. This extended range is achieved by introducing harmonic RFID tags in conjunction with a distributed bi-static configuration, in contrast to the conventional mono-static setup used in RFID readers. In this setting, a transmitter operates at the 915 MHz frequency band, illuminating the RFID tag. INLAN harmonic RFID tags receive this 915 MHz signal and respond at 1.83 GHz. The response signal is captured by multiple receivers distributed across the monitoring environment. Each receiver has three heterodyne lines, which measure the signal's strength. In the presented




(a) (b)

Fig. 4: Experimental setup of INLAN's Ceiling and Square datasets where (a) is the source domain formation with receivers positioned on the ceiling, and a transmitter activating RFID tag, located in the robot moving on pre-determined pathways and (b) the target domain's setup with receivers installed in the square position.

experiments, two of the three lines measure the received power at both x- and y-polarizations. The measured data is then transferred back to the transmitter, which also acts as the system's gateway.

*3) Data collection*

Data collection is performed using a laptop running INLAN's proprietary software, connected to the transmitter via an ethernet cable. An RFID tag mounted on a car follows a pre-programmed path (comprising straight lines spaced a meter apart for the experiments), with its movement synchronized using the software to generate ground truth data. As the car moves at a constant speed of approximately 0.28 m/sec, power readings (in dBm) from each receiver (with two polarizations each) are recorded. The time interval between each reading pair of a given receiver is approximately 0.12 seconds. For the square and cross configurations, the transmitter's output is periodically switched between two antennas, and the maximum reading is recorded, effectively extending the time between consecutive readings to 0.25 seconds.

## B. Comparison with Baseline Models

As previously mentioned, we analyzed our indoor localization frameworks developed based on SFDA, called MTLoc, and MTLoc with confidence, with seven baseline models: source-only, DANN, ADDA, DAFI, FiDO, SHOT and Oracle (see Table I). We utilized INLAN's Ceiling, Cross, and Square datasets, having 12273, 1485, and 1562 samples, respectively. Under this within-environment setup, the Ceiling configuration serves as the source domain, whereas Cross and Square serve as target domains induced solely by sensor-placement changes within the same environment. The data was split into training and testing sets with an 80-20 ratio. To ensure robust evaluation, we used 5-fold cross-validation by further dividing the training set into training and validation subsets. Then, the performance of the models was assessed using the held-out test data. To identify the optimal hyperparameters, we examined the error distances corresponding to various validation losses. Through this analysis, we found that for our Cross dataset, the combination of an alpha value of 0.8, a k value of 2, and x and y threshold coefficients of 8 and 4, respectively, yielded the smallest distance error. Similarly, for the Square dataset, $x$ and $y$ threshold coefficients of 4 and 2 resulted in the minimum distance error. The implementation specifics are introduced below, and respective architectures are described in Appendix A.

**Localizer:** Our source model is trained using a one-dimensional CNN incorporating 9 layers, with INLAN's Ceiling dataset, employed in our MTLoc and baseline models.

**MTLoc:** This approach is based on the mean teacher technique using Student and teacher networks, initialized by our localizer, where our Teacher model is updated by applying EMA with an alpha of 0.7.

**MTLoc with confidence:** Our enhanced approach follows MTLoc's methodology, using a confidence-based $k$-NN method ($k$=2) and an alpha of 0.8 to adjust uncertain pseudo labels to the nearest confident ones.

**DANN:** Our UDA baseline is created using a localizer initialized by our localizer and a discriminator encompassing three layers of 1D-CNN.

**ADDA:** Adversarial learning based technique, aligning target features to the source using domain discriminator.

**FiDO:** An autoencoder-based method that promotes domain-

TABLE I: Performance of our MTLoc model on two target datasets (Cross & Square), compared to source-only, DANN, ADDA, SHOT, DAFI, FiDO, and oracle. Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) values were obtained over 10 runs (excluding the source-only approach, which was pre-trained on the source dataset and evaluated on the target set).

| **Method** | **Source-free** | **Dataset** | **MAE (m) ↓** | | | **RMSE (m) ↓** | | |
|---|---|---|---|---|---|---|---|---|
| | | | **(x)** | **(y)** | **(d)** | **(x)** | **(y)** | **(d)** |
| Source-only | | Cross | 12.34 | 15.50 | 26.96 | 15.93 | 23.14 | 27.29 |
| | | Square | 12.37 | 16.05 | 16.58 | 11.19 | 14.72 | 17.59 |
| DANN (inspired by FreeLoc [31]) | ✗ | Cross | $2.22 \pm 0.01$ | $7.59 \pm 0.01$ | $8.18 \pm 0.01$ | $2.81 \pm 0.01$ | $8.95 \pm 0.01$ | $9.30 \pm 0.01$ |
| | ✗ | Square | $2.32 \pm 0.01$ | $7.64 \pm 0.01$ | $8.26 \pm 0.01$ | $2.91 \pm 0.01$ | $8.99 \pm 0.01$ | $9.37 \pm 0.01$ |
| ADDA [73] | ✗ | Cross | $3.29 \pm 1.56$ | $4.58 \pm 0.80$ | $6.15 \pm 1.34$ | $3.70 \pm 1.57$ | $5.40 \pm 1.01$ | $6.65 \pm 1.43$ |
| | ✗ | Square | $3.46 \pm 1.68$ | $4.56 \pm 1.03$ | $6.24 \pm 1.52$ | $3.82 \pm 1.67$ | $5.38 \pm 1.16$ | $6.72 \pm 1.56$ |
| SHOT [74] | ✗ | Cross | $2.96 \pm 0.29$ | $4.14 \pm 0.08$ | $5.55 \pm 0.29$ | $3.48 \pm 0.33$ | $4.91 \pm 0.16$ | $6.02 \pm 0.31$ |
| | ✗ | Square | $3.05 \pm 0.29$ | $4.14 \pm 0.03$ | $5.59 \pm 0.23$ | $3.58 \pm 0.35$ | $4.87 \pm 0.05$ | $6.05 \pm 0.23$ |
| DAFI [30] | ✗ | Cross | $2.59 \pm 0.64$ | $5.48 \pm 1.57$ | $6.46 \pm 1.70$ | $2.99 \pm 0.69$ | $6.50 \pm 1.80$ | $7.17 \pm 1.86$ |
| | ✗ | Square | $2.83 \pm 0.75$ | $5.28 \pm 1.32$ | $6.47 \pm 1.07$ | $3.24 \pm 0.78$ | $6.28 \pm 1.61$ | $7.17 \pm 1.30$ |
| FiDO [29] | ✗ | Cross | $4.51 \pm 0.77$ | $4.54 \pm 0.61$ | $6.91 \pm 0.95$ | $4.96 \pm 0.82$ | $5.59 \pm 0.93$ | $7.48 \pm 1.17$ |
| | ✗ | Square | $4.29 \pm 1.56$ | $4.90 \pm 1.40$ | $7.10 \pm 1.20$ | $4.64 \pm 1.59$ | $5.83 \pm 1.53$ | $7.65 \pm 1.22$ |
| MTLoc (Our Model) | ✓ | Cross | $1.85 \pm 0.27$ | $4.58 \pm 0.60$ | $\underline{5.23} \pm \underline{0.54}$ | $2.18 \pm 0.28$ | $5.43 \pm 0.82$ | $\underline{5.86} \pm \underline{0.74}$ |
| | ✓ | Square | $1.76 \pm 0.10$ | $4.53 \pm 0.25$ | $\underline{5.13} \pm \underline{0.23}$ | $2.09 \pm 0.13$ | $5.41 \pm 0.39$ | $\underline{5.80} \pm \underline{0.35}$ |
| MTLoc with confidence | ✓ | Cross | $1.85 \pm 0.24$ | $4.31 \pm 0.39$ | $\mathbf{4.98 \pm 0.37}$ | $2.18 \pm 0.26$ | $5.11 \pm 0.53$ | $\mathbf{5.57 \pm 0.50}$ |
| | ✓ | Square | $1.76 \pm 0.16$ | $4.13 \pm 0.07$ | $\mathbf{4.75 \pm 0.06}$ | $2.09 \pm 0.18$ | $4.78 \pm 0.09$ | $\mathbf{5.22 \pm 0.07}$ |
| Oracle | | Cross | $0.61 \pm 0.05$ | $1.03 \pm 0.08$ | $1.00 \pm 0.07$ | $0.78 \pm 0.06$ | $1.46 \pm 0.14$ | $1.43 \pm 0.14$ |
| | | Square | $0.41 \pm 0.02$ | $0.78 \pm 0.04$ | $0.73 \pm 0.03$ | $0.54 \pm 0.02$ | $1.07 \pm 0.05$ | $1.00 \pm 0.05$ |

invariant features by training a reconstruction head on unlabeled target data.

**DAFI:** A GRL-based UDA framework with dual domain heads that minimizes target conditional entropy and enforces logit-space triplet alignment, while predicting continuous $(x, y)$ as the $q$-weighted average of class centers.

**SHOT:** An SFDA method for regression that freezes the regressor, adapts the feature extractor under an EMA teacher, and enforces prediction-statistics matching with CORAL-based feature alignment.

**Source-only:** In this pre-adaptation baseline, we tested our unlabeled target datasets (INLAN's Cross and Square sets) using our source localizer.

**Oracle:** In this approach, we fine-tuned our localizer with labeled target data.

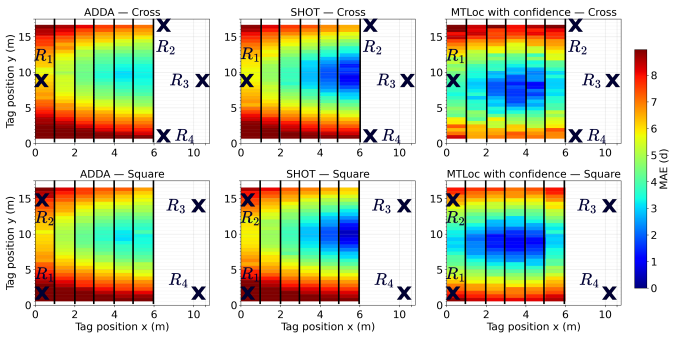



Fig. 5: Heatmaps of mean absolute distance error for MTLoc with confidence and the strongest baselines (ADDA, SHOT) on the Cross and Square target datasets. Receiver locations $(R_1 - R_4)$ are indicated.

As shown in Table I, the MAE results for both the $x$ and $y$ values of the source-only model are notably high. This outcome can be due to a significant shift in the environment between training and deployment, often referred to as a domain or distributional shift. The Oracle method, which utilizes labeled data to fine-tune the source model, achieves significantly lower error, indicating the upper bound of adaptation performance. In contrast, other methods such as DANN, ADDA, DAFI, FiDO, and SHOT fall between these extremes. As the $x$ values represent the distance between pathways, we analyze the $y$ values and their combined distance $d$ to obtain a more comprehensive evaluation. Therefore, we report MAE for all methods. With the DANN baseline, MAE is reduced considerably compared to the source-only model, motivating further investigation into source-free adaptation. Our MTLoc model yields substantial improvements over all considered baselines. Compared with the baselines (ADDA, DANN, SHOT, DAFI, FiDO), MTLoc reduces MAE on average by 20.0% and 22.5% for Cross and Square respectively. Further improvements are observed with MTLoc with Confidence, which achieves average reductions of 23.9% on Cross and 28.2% on Square. Compared with individual baselines, MTLoc with Confidence achieves MAE reductions in the range of 10.3%–27.9% on Cross and 15.0%–33.1% on Square. Additionally, the standard deviation of MAE is significantly lower, with relative reductions of approximately 56.9% on Cross and 92.6% on Square, indicating improved robustness and consistency across runs. To better present the performance of our approach, we generate heatmaps, as shown in Fig.5 for our best model (MTLoc with confidence) alongside two strong baselines (ADDA and SHOT). Overall, the results show that MTLoc maintains lower localization error across both datasets

(Cross & Square), confirming its robustness and consistency in cross-domain scenarios. Notably, a confidence-based $k$-NN correction was applied during adaptation. The confidence module is active only during adaptation, ensuring there is no additional computational cost at inference. The deployed model runs identically to MTLoc during testing. While average error reductions may be modest in some cases, we consistently observe smaller standard deviations, particularly for the distance metric MAE, indicating improved stability across runs. These properties make MTLoc practical and robust for SFDA in real-world indoor localization.

### C. Ablation Studies

In this section, we observe the impact of hyper-parameter values, including the EMA's alpha, $k$ of our $k$-NN-based correctional model, and the threshold we set for our $x$ and $y$-predicted pseudo labels. For this experiment, our MTLoc with confidence model is applied on the same datasets, testing $k$ values of 1, 2, and 3, along with alpha values of 0.6, 0.7, 0.75, and 0.8. Additionally, we applied $x$ and $y$ coefficients for thresholds of 3, 4, 5, and 4, 5, and 6, respectively. The validation and test losses were observed using 5-fold cross-validation. With an alpha of 0.8 and a $k$ value of 2 obtaining optimal results on the validation set for both the Cross and Square datasets using threshold coefficients of (8,4) and (4,2), we conducted an ablation study to assess the impact of $k$ and alpha, as shown in Tables II&III.

TABLE II: *Ablation Results on $\alpha$:* The test MAE values and validation distance errors are shown for the Cross and Square datasets at varying $\alpha$ values. The $x$ and $y$ threshold coefficients are set to (8,4) for the Cross and (4,2) for the Square datasets.

| $k$ | Dataset | MAE (m) ↓ | | | RMSE (m) ↓ | | |
|---|---|---|---|---|---|---|---|
| | | (x) | (y) | (d) | (x) | (y) | (d) |
| 1 | Cross | $1.86 \pm 0.25$ | $4.78 \pm 0.46$ | $5.42 \pm 0.42$ | $2.20 \pm 0.27$ | $5.67 \pm 0.68$ | $6.10 \pm 0.60$ |
| | Square | $1.78 \pm 0.20$ | $4.21 \pm 0.15$ | $4.82 \pm 0.15$ | $2.09 \pm 0.22$ | $4.90 \pm 0.18$ | $5.33 \pm 0.18$ |
| 2 | Cross | $1.85 \pm 0.24$ | $4.31 \pm 0.39$ | $\mathbf{4.98 \pm 0.37}$ | $2.18 \pm 0.26$ | $5.11 \pm 0.53$ | $\mathbf{5.57 \pm 0.50}$ |
| | Square | $1.76 \pm 0.16$ | $4.13 \pm 0.07$ | $\mathbf{4.75 \pm 0.06}$ | $2.09 \pm 0.18$ | $4.78 \pm 0.09$ | $\mathbf{5.22 \pm 0.07}$ |
| 3 | Cross | $1.85 \pm 0.23$ | $4.58 \pm 0.47$ | $5.24 \pm 0.40$ | $2.18 \pm 0.25$ | $5.49 \pm 0.63$ | $5.91 \pm 0.55$ |
| | Square | $1.79 \pm 0.16$ | $4.21 \pm 0.09$ | $4.84 \pm 0.14$ | $2.14 \pm 0.17$ | $4.89 \pm 0.16$ | $5.34 \pm 0.18$ |
| 4 | Cross | $2.19 \pm 0.42$ | $4.38 \pm 0.30$ | $5.22 \pm 0.37$ | $2.54 \pm 0.46$ | $5.18 \pm 0.40$ | $5.79 \pm 0.44$ |
| | Square | $1.72 \pm 0.11$ | $4.25 \pm 0.18$ | $4.84 \pm 0.19$ | $2.05 \pm 0.12$ | $4.97 \pm 0.28$ | $5.38 \pm 0.27$ |

TABLE III: *Ablation Results on $\alpha$:* The test MAE values and validation distance errors are shown for the Cross and Square datasets at varying $\alpha$ values. The $x$ and $y$ threshold coefficients are set to (8,4) for the Cross and (4,2) for the Square datasets.

| $\alpha$ | Dataset | MAE (m) ↓ | | | RMSE (m) ↓ | | |
|---|---|---|---|---|---|---|---|
| | | (x) | (y) | (d) | (x) | (y) | (d) |
| 0.70 | Cross | $1.96 \pm 0.31$ | $4.62 \pm 0.47$ | $5.32 \pm 0.39$ | $2.29 \pm 0.34$ | $5.49 \pm 0.70$ | $5.97 \pm 0.59$ |
| | Square | $1.74 \pm 0.11$ | $4.46 \pm 0.31$ | $5.05 \pm 0.34$ | $2.07 \pm 0.14$ | $5.31 \pm 0.47$ | $5.70 \pm 0.48$ |
| 0.75 | Cross | $2.06 \pm 0.36$ | $4.32 \pm 0.24$ | $5.10 \pm 0.31$ | $2.40 \pm 0.38$ | $5.08 \pm 0.30$ | $5.63 \pm 0.32$ |
| | Square | $1.79 \pm 0.18$ | $4.30 \pm 0.26$ | $4.93 \pm 0.27$ | $2.12 \pm 0.20$ | $5.06 \pm 0.35$ | $5.49 \pm 0.34$ |
| 0.80 | Cross | $1.85 \pm 0.24$ | $4.31 \pm 0.39$ | $\mathbf{4.98 \pm 0.37}$ | $2.18 \pm 0.26$ | $5.11 \pm 0.53$ | $\mathbf{5.57 \pm 0.50}$ |
| | Square | $1.76 \pm 0.16$ | $4.13 \pm 0.07$ | $\mathbf{4.75 \pm 0.06}$ | $2.09 \pm 0.18$ | $4.78 \pm 0.09$ | $\mathbf{5.22 \pm 0.07}$ |
| 0.90 | Cross | $4.47 \pm 1.39$ | $5.51 \pm 1.06$ | $7.69 \pm 1.63$ | $4.91 \pm 1.39$ | $6.70 \pm 1.27$ | $8.34 \pm 1.68$ |
| | Square | $2.93 \pm 0.57$ | $5.00 \pm 0.28$ | $6.28 \pm 0.56$ | $3.38 \pm 0.63$ | $6.15 \pm 0.36$ | $7.03 \pm 0.56$ |

### D. Discussion

MTLoc is the first SFDA method developed for indoor localization, addressing the limited availability of source data under privacy or resource constraints. The approach outperforms state-of-the-art baselines, including UDA methods (DANN, ADDA, FiDO, and DAFI) and SHOT, a source-free classification method adapted for regression. MTLoc introduces several promising foundations for further extension. While the MTLoc model with confidence shows robust performance, it still falls short of an oracle model having access to labeled target data. A future direction could be studying the case where a small amount of labeled target samples is available during adaptation. In this setting, semi-supervised and weakly supervised domain adaptation techniques could be applied. In many real-world scenarios, it is feasible to collect a few labeled reference tags in the target domain. By including 5 to 10 labeled samples can significantly improve the model's alignment with the target environment. These labeled points can serve as anchors to guide the model's predictions toward the correct locations and help stabilize pseudo-labels. This, in turn, reduces the domain shift between source and target distributions. By using a small amount of supervision in an originally unsupervised SFDA setting, this extension could improve model accuracy while maintaining the low computational cost that makes MTLoc practical for large-scale deployment.

## VI. Conclusions

Data privacy, storage, transfer costs, and domain shifts are critical challenges in UDA, often affecting model performance in localization tasks. To address these concerns, an SFDA model was proposed for indoor localization that preserves data privacy and reduces transfer costs. In particular, the new MTLoc model is initialized by a localizer with the architecture of a 1D-CNN. MTLoc utilizes a student-teacher architecture that improves stability through an MT mechanism, where the teacher network generates pseudo labels. An enhanced certainty-based approach called "MTLoc with confidence" selectively uses highly confident pseudo labels during training to improve performance. This technique refines the teacher network's pseudo labels by including the two nearest confident labels. In our experimental validation, the MTLoc with Confidence model achieves lower localization error than baseline methods, including DANN, ADDA, DAFI, FiDO, and the SFDA method SHOT. The confidence-based approach notably reduced errors in the $y$ dimension and improved stability in both $x$ and $y$ with small variations, resulting in a more robust and reliable model.

## Acknowledgment

This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC), and Mitacs, with additional computational resources provided by the Digital Research Alliance of Canada.

## References

[1] Y. Etiabi and E. M. Amhoud. Federated distillation based indoor localization for iot networks. *IEEE Sensors Journal*, 24(7), 2024.

[2] H. Liu, H. Xue, L. Zhao, D. Chen, Z. Peng, and G. Zhang. Magloc-ar: Magnetic-based localization for visual-free augmented reality in large-scale indoor environments. *IEEE TVCG*, 29(11), 2023.

[3] Y. Zhao, J. Xu, J. Wu, J. Hao, and H. Qian. Enhancing camera-based multimodal indoor localization with device-free movement measurement using wifi. *IEEE IoTJ*, 7(2), 2020.

[4] J. H. Seong and D. H. Seo. Selective unsupervised learning-based wi-fi fingerprint system using autoencoder and gan. *IEEE IoTJ*, 7(3), 2020.

[5] Z. Liu, X. Wang, Z. Chen, M. Zhao, S. Zhang, and J. Li. 3d-fingerprint augment based on super-resolution for indoor 3d wifi localization. In *WCSP*, 2021.

[6] X. Weng, K.V. Ling, H. Liu, and K. Cao. Towards end-to-end gps localization with neural pseudorange correction. In *FUSION*, 2024.

[7] Y. Li, B. Yu, and L. Huang. An indoor uwb localization method based on adaptive channel bias estimation. *IEEE Sensors Journal*, 25(1):1339–1349, 2025.

[8] Y. Huang, S. Mazuelas, F. Ge, and Y. Shen. Indoor localization system with nlos mitigation based on self-training. *IEEE TMC*, 22(7), 2023.

[9] Z. Gao, Y. Ma, K. Liu, X. Miao, and Y. Zhao. An indoor multi-tag cooperative localization algorithm based on nmds for rfid. *IEEE Sensors Journal*, 17(7):2120–2128, 2017.

[10] L. Yang, Y. Chen, X.-Y. Li, C. Xiao, M. Li, and Y. Liu. Tagoram: Real-time tracking of mobile rfid tags to high precision using cots devices. In *MobiCom*, 2014.

[11] T. M. T. Dinh, N. S. Duong, and K. Sandrasegaran. Smartphone-based indoor positioning using ble ibeacon and reliable lightweight fingerprint map. *IEEE Sensors Journal*, 20(17):10283–10294, 2020.

[12] Y. Yu, R. Chen, L. Chen, X. Zheng, D. Wu, W. Li, and Y. Wu. A novel 3-d indoor localization algorithm based on ble and multiple sensors. *IEEE IoTJ*, 8(11), 2021.

[13] W. Zhu, J. Cao, Y. Xu, L. Yang, and J. Kong. Fault-tolerant rfid reader localization based on passive rfid tags. *IEEE TPDS*, 25(8), 2014.

[14] C.-S. Cheng, H. H. Chang, Y.-T. Chen, T.-H. Lin, P.-C. Chen, C.-M. Huang, H. S. Yuan, and W. C. Chu. Accurate location tracking based on active rfid for health and safety monitoring. In *ICBBE*, 2009.

[15] W.-H. Chen, H. H. Chang, T.-H. Lin, P.-C. Chen, L.-K. Chen, S. J. Hwang, D. H. Yen, H. S. Yuan, and W. C. Chu. Dynamic indoor localization based on active rfid for healthcare applications: A shape constraint approach. In *BMEI*, 2009.

[16] H. Huang, J. Yang, X. Fang, H. Jiang, and L. Xie. Varifi: Variational inference for indoor pedestrian localization and tracking using imu and wifi rss. *IEEE IoTJ*, 10(10), 2023.

[17] Z. Xu, B. Huang, B. Jia, and G. Mao. Enhancing wifi fingerprinting localization through a co-teaching approach using crowdsourced sequential rss and imu data. *IEEE IoTJ*, 11(2), 2024.

[18] A. K. Panja, S. Biswas, S. Neogy, and C. Chowdhury. Dimensionality reduction through multiple convolutional channels for rss-based indoor localization. *IEEE Sensors Journal*, 24(22):37482–37491, 2024.

[19] D.-H. Kim, A. Farhad, and J.-Y. Pyun. Uwb positioning system based on lstm classification with mitigated nlos effects. *IEEE IoTJ*, 10(2), 2023.

[20] H. Yang, Y. Wang, C. K. Seow, M. Sun, M. Si, and L. Huang. Uwb sensor-based indoor los/nlos localization with support vector machine learning. *IEEE Sensors Journal*, 23(3):2988–3004, 2023.

[21] X. Rao, Z. Luo, Y. Luo, Y. Yi, G. Lei, and Y. Cao. Mffaloc: Csi-based multifeatures fusion adaptive device-free passive indoor fingerprinting localization. *IEEE IoTJ*, 11(8), 2024.

[22] X. Liu, R. Wu, H. Zhang, Z. Chen, Y. Liu, and T. Qiu. Graph temporal convolutional network-based wifi indoor localization using fine-grained csi fingerprint. *IEEE Sensors Journal*, 25(5):9019–9033, 2025.

[23] L. Terças, H. Alves, C. H. M. de Lima, and M. Juntti. Bayesian-based indoor factory positioning using aoa, tdoa, and hybrid measurements. *IEEE IoTJ*, 11(12), 2024.

[24] C. Huang, Z. Tian, W. He, K. Liu, and Z. Li. Spotlight: A 3-d indoor localization system in wireless sensor networks based on orientation and rssi measurements. *IEEE Sensors Journal*, 23(21):26662–26676, 2023.

[25] H. Ye, B. Yang, Z. Long, and C. Dai. A method of indoor positioning by signal fitting and pdda algorithm using ble aoa device. *IEEE Sensors Journal*, 22(8):7877–7887, 2022.

[26] R. Elbakly and M. Youssef. Robust low-overhead rf-based localization for realistic environments. *IEEE TMC*, 21(6), 2022.

[27] E. Fernando, O. De Silva, G. K. I. Mann, and R. Gosine. Toward developing an indoor localization system for mavs using two or three rf range anchors: An observability based approach. *IEEE Sensors Journal*, 22(6):5173–5187, 2022.

[28] T. Wang, Y. Li, J. Liu, K. Hu, and Y. Shen. Multipath-assisted single-anchor localization via deep variational learning. *IEEE TWC*, 2024.

[29] X. Chen, H. Li, C. Zhou, X. Liu, D. Wu, and G. Dudek. FiDo: Ubiquitous fine-grained wifi-based localization for unlabelled users via domain adaptation. Association for Computing Machinery, 2020.

[30] H. Li, X. Chen, J. Wang, D. Wu, and X. Liu. DAFI: Wifi-based device-free indoor localization via domain adaptation. *Proceedings of the ACM on IMWUT*, 5(4), 2021.

[31] D. Yan, F. Shang, P. Yang, F. Han, Y. Yan, and X.-Y. Li. freeloc: Wireless-based cross-domain device-free fingerprints localization to free user's motions. *IEEE IoTJ*, 2024.

[32] Y. Kim, D. Cho, and S. Hong. Towards privacy-preserving domain adaptation. *IEEE SPL*, 27, 2020.

[33] Y. Fang, P.-T. Yap, W. Lin, H. Zhu, and M. Liu. Source-free unsupervised domain adaptation: A survey. *Neural Networks*, 164, 2024.

[34] J. Li, Z. Yu, Z. Du, L. Zhu, and H. T. Shen. A comprehensive survey on source-free domain adaptation. *IEEE TPAMI*, 46(8), 2024.

[35] J. Pei, Z. Jiang, A. Men, L. Chen, Y. Liu, and Q. Chen. Uncertainty-induced transferability representation for source-free unsupervised domain adaptation. *IEEE TIP*, 32, 2023.

[36] Tianlang He, Zhiqiu Xia, Jierun Chen, Haoliang Li, and S.-H. Gary Chan. Target-agnostic source-free domain adaptation for regression tasks. In *2024 IEEE 40th ICDE*, 2024.

[37] Y. Yu, S. Shin, S. Back, M. Ko, S. Noh, and K. Lee. Domain-specific block selection and paired-view pseudo-labeling for online test-time adaptation. In *CVPR*, 2024.

[38] Jie Zhang, Yanjiao Li, and Wendong Xiao. Integrated multiple kernel learning for device-free localization in cluttered environments using spatiotemporal information. *IEEE IoTJ*, 8(6):4749–4761, 2021.

[39] Jie Zhang, Jianqiang Xue, Yanjiao Li, and Simon L. Cotton. Leveraging online learning for domain-adaptation in wi-fi-based device-free localization. *IEEE TMC*, 24(8):7773–7787, 2025.

[40] Z. Chen, M. I. AlHajri, M. Wu, N. T. Ali, and R. M. Shubair. A novel real-time deep learning approach for indoor localization based on rf environment identification. *IEEE Sensors Letters*, 4(6):1–4, 2020.

[41] J. Yan, L. Wan, W. Wei, X. Wu, W. P. Zhu, and D. P. K. Lun. Device-free activity detection and wireless localization based on cnn using channel state information measurement. *IEEE Sensors Journal*, 21(21):24482–24494, 2021.

[42] Q. Ye. Se-loc: Security-enhanced indoor localization with semi-supervised deep learning. *IEEE TNSE*, 10(5), 2023.

[43] Y. Cao, B. Luo, Y. Chen, L. Xu, and C. Ding. Confidence-aware mean teacher for semi-supervised metallographic image semantic segmentation. *Computational Materials Science*, 249:113645, 2025.

[44] Y. Ruan, L. Chen, X. Zhou, Z. Liu, X. Liu, G. Guo, and R. Chen. ipos-5g: Indoor positioning via commercial 5g nr csi. *IEEE IoTJ*, 10(10), 2023.

[45] Y. Li, S. Mazuelas, and Y. Shen. A variational learning approach for concurrent distance estimation and environmental identification. *IEEE TWC*, 22(9), 2023.

[46] S. L. Ayinla, A. A. Aziz, and M. Drieberg. Salloc: An accurate target localization in wifi-enabled indoor environments via sae-alstm. *IEEE Access*, 12, 2024.

[47] M. Liu, X. Liao, Z. Gao, and Q. Li. Ft-loc: A fine-grained temporal features-based fusion network for indoor localization. *IEEE IoTJ*, 11(3), 2024.

[48] Z. Chen, H. Zou, J. Yang, H. Jiang, and L. Xie. Wifi fingerprinting indoor localization using local feature-based deep lstm. *IEEE Systems Journal*, 14(2):3001–3010, 2020.

[49] S. A. Junoh and J.-Y. Pyun. Augmentation of fingerprints for indoor ble localization using conditional gans. *IEEE Access*, 12, 2024.

[50] Q. Li, H. Qu, Z. Liu, N. Zhou, W. Sun, S. Sigg, and J. Li. Af-dcgan: Amplitude feature deep convolutional gan for fingerprint construction in indoor localization systems. *IEEE TETCI*, 5(3), 2021.

[51] W. Wu. A meta-learning approach for device-free indoor localization. *IEEE Communications Letters*, 27(3), 2023.

[52] C. Wang, Z. Wang, Q. Liu, H. Dong, W. Liu, and X. Liu. A comprehensive survey on domain adaptation for intelligent fault diagnosis. *Knowledge-Based Systems*, 327:114109, 2025.

[53] J. Li, Z. Yu, Z. Du, L. Zhu, and H. T. Shen. A comprehensive survey on source-free domain adaptation. *IEEE TPAMI*, 46(8):5743–5762, 2024.

[54] M. Long, Y. Cao, J. Wang, and M. Jordan. Learning transferable features with deep adaptation networks. In *ICML*, 2015.

[55] S. Cui, S. Wang, J. Zhuo, L. Li, Q. Huang, and Q. Tian. Towards discriminability and diversity: Batch nuclear-norm maximization under label insufficient situations. In *CVPR*, 2020.

[56] C. Shui and et al. Towards more general loss and setting in unsupervised domain adaptation. *IEEE TKDE*, 35(10):10140–10150, Oct. 2023.

[57] M. Long, Z. Cao, J. Wang, and M. I. Jordan. Conditional adversarial domain adaptation. In *NeurIPS*, 2018.

[58] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *CVPR*, 2017.

[59] J. Li, M. Jing, H. Su, K. Lu, L. Zhu, and H. T. Shen. Faster domain adaptation networks. *IEEE TKDE*, 34(12):5770–5783, Dec. 2022.
[60] Q. Lian, F. Lv, L. Duan, and B. Gong. Constructing self-motivated pyramid curriculums for cross-domain semantic segmentation: A non-adversarial approach. In *CVPR*, 2019.
[61] Y. Zou, Z. Yu, B. Kumar, and J. Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *ECCV*, 2018.
[62] J. Li, Z. Du, L. Zhu, Z. Ding, K. Lu, and H. T. Shen. Divergence-agnostic unsupervised domain adaptation by adversarial attacks. *IEEE TPAMI*, 44(11):8196–8211, Nov. 2022.
[63] M. Zhang, Z. Fan, R. Shibasaki, and X. Song. Domain adversarial graph convolutional network based on rssi and crowdsensing for indoor localization. *IEEE IoTJ*, 10(15), 2023.
[64] Z. Xu, B. Huang, B. Jia, and G. Mao. Enhancing wifi fingerprinting localization through a co-teaching approach using crowdsourced sequential rss and imu data. *IEEE IoTJ*, 11(2), 2024.
[65] L. Zhang, S. Wu, T. Zhang, and Q. Zhang. Automatic radio map adaptation for robust indoor localization with dynamic adversarial learning. *IEEE TII*, 2024.
[66] G. Prasad, A. Pandey, and S. Kumar. Domain adaptation for localization using combined autoencoder and gradient reversal layer in dynamic iot environment. *IEEE TNSE*, 11(1), 2024.
[67] L. Zhang, S. Wu, T. Zhang, and Q. Zhang. Robloc: Robust wireless localization with dynamic self-adaptive learning. *IEEE IoTJ*, 11(10), 2024.
[68] X. Chen, H. Li, C. Zhou, X. Liu, D. Wu, and G. Dudek. Fidora: Robust wifi-based indoor localization via unsupervised domain adaptation. *IEEE IoTJ*, 9(12), 2022.
[69] C. M. Bishop and N. M. Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.
[70] S. Varailhon, M. Aminbeidokhti, M. Pedersoli, and E. Granger. Source-free domain adaptation for yolo object detection. In *ECCV*, 2024.
[71] A. Belal, A. Meethal, F. P. Romero, M. Pedersoli, and E. Granger. Multi-source domain adaptation for object detection with prototype-based mean teacher. In *WACV*, 2024.
[72] Y. Ganin and V. Lempitsky. Unsupervised domain adaptation by backpropagation. In *Proc. of the 32nd Int. Conf. Mach. Learn.*, 2015.
[73] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *CVPR*, pages 7167–7176, July 2017.
[74] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? Source hypothesis transfer for unsupervised domain adaptation. In *ICML*, 2020.
[75] T. Huang, Y. Zhang, M. Zheng, S. You, F. Wang, C. Qian, and C. Xu. Knowledge diffusion for distillation. *NeurIPS*, 36, 2024.
[76] N. Mehregan, B. Bozkurt, E. Granger, M. Hajikhani, and M. Shateri. Inlan-indoor localization. IEEE DataPort, 2025.

## VII. Appendix A: Models Implementation

In this section, implementation details of the MTLoc and MTLoc with confidence, along with the three baselines, Adversarial UDA, source-only, and oracle models, are provided. In this study, a localizer, pre-trained on the INLAN's Ceiling dataset, was designed and used in our source-only, UDA baseline and proposed SFDA approaches. An Adam optimizer with a learning rate of 0.001 was used for the optimization.

**Source-only:** In our source-only model, we used a localizer, having a feature extractor and a regressor.

*Feature Extractor:*

- Conv1D (filters = 64, kernel size = 2, activation = 'ReLU')
- Dropout (rate = 0.2)
- Conv1D (filters = 128, kernel size = 2, activation = 'ReLU')
- Dropout (rate = 0.2)
- Flatten

*Regressor:*

- Dense (units = 128, activation = 'ReLU')
- Dropout (rate = 0.2)
- Dense (units = 64, activation = 'ReLU')
- Dense (units = 2)

**Adversarial UDA:** In our Adversarial UDA baseline, we used a localizer having a feature extractor and regressor with the same structure as our source-only model, together with a discriminator.

*Discriminator:*

- Dense (units = 128, activation = 'ReLU')
- Dense (units = 64, activation = 'ReLU')
- Dense (units = 1, activation = 'Sigmoid')

**MTloc:** In our MTLoc model, our teacher and student networks have the same structure as our localizer. As the teacher network receives INLAN's Cross and Squares datasets, the Student model obtains noisy versions of them as unlabeled target data with a variance of 0.1. By applying EMA with an alpha of 0.7, the teacher networks' parameters aim to be close to the student's parameters. In our confidence-based model, following the same methodology, a threshold was defined in which the average of the predicted two-dimensional positional coordinates $(x, y)$ is added to 5 and 3 times their respective standard deviation, correcting uncertain pseudo labels by applying $k$NN ($k$=2).

**Oracle:** In the Oracle model, we finetune our localizer, previously presented, on labeled target data (INLAN's Cross and Square datasets).

## VIII. Appendix B: Threshold Ablation Study

In this section, threshold hyperparameters of our MTLoc with confidence model are experimentally analyzed on both the Cross and Square datasets, to better understand the behavior of the proposed approach. Table A-4 presents the average test results of each experiment across 10 runs, along with the corresponding standard deviations obtained via 5-fold cross-validation. To define a confidence threshold for the positional coordinates, we independently scale the standard deviations of the $x$- and $y$-coordinate predictions with respect to their means and respective uncertainty distributions. This scaling factor was regarded as a tunable hyperparameter, with values of 4, 5, 6, and 8 evaluated for the $x$ positional coordinate, and 2, 3, and 4 for the $y$ positional coordinate. In our approach, using the cross and square datasets, we derived matrices of dimensions $1485 \times 1$ and $1562 \times 1$ for the $x$ and $y$ coordinates, respectively. Within the correction module, each training sample was processed 10 times, with random noise added in each iteration. This process resulted in matrices of dimensions $1485 \times 10 \times 2$ and $1562 \times 10 \times 2$, where the last dimension represents the $x$ and $y$ coordinates. After obtaining the outputs, we calculated the standard deviation across the 10 noise-augmented iterations for each sample, yielding $\sigma_x$ and $\sigma_y$ for the $x$ and $y$ coordinates, respectively. Our threshold is defined as:

$$T_{hr} = \mu + c \cdot \sigma, \tag{13}$$

where $\mu$ is the average of all the $\sigma_x$ or $\sigma_y$, $\sigma$ is the standard deviation of all $\sigma_x$ or $\sigma_y$, and $c$ is the scaling coefficient ($c_x$

or $c_y$). This threshold identifies samples with high uncertainty, enabling the correction module to modify these pseudo labels specifically. This approach ensures focus on improving the most unconfident predictions, thereby improving its overall robustness. Table IV shows an ablation study on threshold coefficients when $k$ and alpha are 2 and 0.8, respectively.

TABLE A-4: *Performance of MTLoc model on two target datasets (Cross & Square) over threshold coefficients $c_x$ and $c_y$, for $k = 2$ and $\alpha = 0.8$.*

| $(c_x, c_y)$ | Dataset | MAE (m) ↓ | | | RMSE (m) ↓ | | |
|---|---|---|---|---|---|---|---|
| | | (x) | (y) | (d) | (x) | (y) | (d) |
| (4,2) | Cross | $1.78 \pm 0.13$ | $4.75 \pm 0.58$ | $5.36 \pm 0.51$ | $2.13 \pm 0.13$ | $5.65 \pm 0.80$ | $6.05 \pm 0.72$ |
| | Square | $1.76 \pm 0.16$ | $4.13 \pm 0.07$ | $\mathbf{4.75 \pm 0.06}$ | $2.09 \pm 0.18$ | $4.78 \pm 0.09$ | $\mathbf{5.22 \pm 0.07}$ |
| (4,3) | Cross | $1.96 \pm 0.31$ | $4.41 \pm 0.37$ | $5.23 \pm 0.32$ | $2.30 \pm 0.34$ | $5.22 \pm 0.52$ | $5.72 \pm 0.49$ |
| | Square | $1.78 \pm 0.17$ | $4.26 \pm 0.17$ | $4.92 \pm 0.22$ | $2.11 \pm 0.18$ | $5.01 \pm 0.27$ | $5.43 \pm 0.28$ |
| (4,4) | Cross | $1.90 \pm 0.38$ | $4.56 \pm 0.35$ | $5.24 \pm 0.34$ | $2.18 \pm 0.40$ | $5.42 \pm 0.50$ | $5.88 \pm 0.45$ |
| | Square | $1.85 \pm 0.30$ | $4.19 \pm 0.10$ | $4.86 \pm 0.22$ | $2.13 \pm 0.37$ | $4.93 \pm 0.12$ | $5.35 \pm 0.25$ |
| (5,2) | Cross | $2.03 \pm 0.35$ | $4.57 \pm 0.56$ | $5.32 \pm 0.50$ | $2.38 \pm 0.40$ | $5.45 \pm 0.75$ | $5.97 \pm 0.67$ |
| | Square | $1.77 \pm 0.12$ | $4.34 \pm 0.25$ | $4.95 \pm 0.26$ | $2.10 \pm 0.15$ | $5.10 \pm 0.40$ | $5.52 \pm 0.38$ |
| (5,3) | Cross | $2.05 \pm 0.33$ | $4.61 \pm 0.75$ | $5.37 \pm 0.60$ | $2.41 \pm 0.38$ | $5.43 \pm 1.04$ | $5.97 \pm 0.87$ |
| | Square | $1.86 \pm 0.18$ | $4.26 \pm 0.20$ | $4.94 \pm 0.17$ | $2.21 \pm 0.20$ | $5.04 \pm 0.30$ | $5.48 \pm 0.25$ |
| (5,4) | Cross | $1.81 \pm 0.19$ | $4.68 \pm 0.55$ | $5.36 \pm 0.47$ | $2.14 \pm 0.21$ | $5.66 \pm 0.96$ | $6.07 \pm 0.85$ |
| | Square | $1.82 \pm 0.15$ | $4.39 \pm 0.35$ | $5.00 \pm 0.33$ | $2.10 \pm 0.16$ | $5.25 \pm 0.45$ | $5.63 \pm 0.43$ |
| (6,2) | Cross | $2.03 \pm 0.35$ | $4.34 \pm 0.39$ | $5.10 \pm 0.34$ | $2.38 \pm 0.36$ | $5.12 \pm 0.59$ | $5.66 \pm 0.49$ |
| | Square | $1.76 \pm 0.16$ | $4.33 \pm 0.19$ | $4.94 \pm 0.20$ | $2.09 \pm 0.17$ | $5.09 \pm 0.29$ | $5.50 \pm 0.25$ |
| (6,3) | Cross | $1.99 \pm 0.34$ | $4.57 \pm 0.24$ | $5.29 \pm 0.19$ | $2.32 \pm 0.36$ | $5.46 \pm 0.37$ | $5.95 \pm 0.27$ |
| | Square | $1.71 \pm 0.08$ | $4.18 \pm 0.15$ | $4.78 \pm 0.12$ | $2.03 \pm 0.07$ | $4.89 \pm 0.20$ | $5.29 \pm 0.17$ |
| (6,4) | Cross | $1.75 \pm 0.10$ | $4.47 \pm 0.47$ | $5.12 \pm 0.40$ | $2.07 \pm 0.12$ | $5.36 \pm 0.70$ | $5.75 \pm 0.62$ |
| | Square | $1.79 \pm 0.09$ | $4.50 \pm 0.25$ | $5.12 \pm 0.28$ | $2.07 \pm 0.10$ | $5.30 \pm 0.39$ | $5.70 \pm 0.37$ |
| (8,4) | Cross | $1.85 \pm 0.24$ | $4.31 \pm 0.39$ | $\mathbf{4.98 \pm 0.37}$ | $2.18 \pm 0.26$ | $5.11 \pm 0.53$ | $\mathbf{5.57 \pm 0.50}$ |
| | Square | $1.71 \pm 0.05$ | $4.24 \pm 0.11$ | $4.83 \pm 0.11$ | $2.03 \pm 0.06$ | $4.98 \pm 0.17$ | $5.38 \pm 0.16$ |

## IX. Appendix C: Adapting SHOT To Regression

In this section, we discuss one of the baselines, SHOT [74], a classification-based SFDA method adapted to the regression setting, as shown in Fig. A-6. In the original setting, SHOT minimizes prediction entropy to ensure feature alignment. However, entropy is not applicable to regression tasks, as predictions are continuous positional coordinates. To address this, we maintain the original SHOT framework: a feature extractor $\mathcal{F}(\cdot;\theta_s)$ and a frozen, source-pretrained regressor $\mathcal{G}(\cdot;\theta_g)$. The model is trained to predict tag locations $y$ for unlabeled target data. During adaptation, only the feature extractor $\mathcal{F}(\cdot;\theta_s)$ is trained, while $\mathcal{G}(\cdot;\theta_g)$ remains frozen.

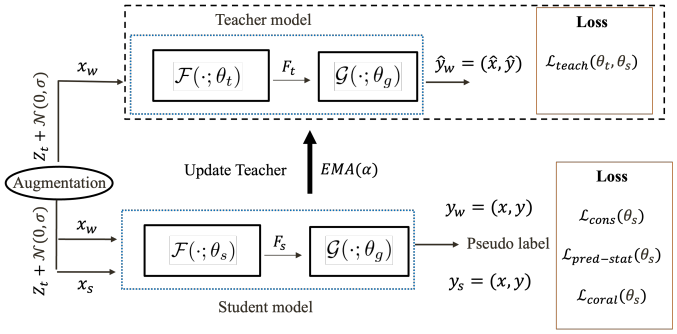



Fig. A-6: *Adaptation of SHOT for regression. The dashed black box indicates the teacher model, which can be optionally disregarded. Target inputs are augmented into two views: a weak view with small Gaussian noise $\mathcal{N}(0, 0.01^2)$ and a strong view with element-wise random masking ($p = 0.10$) and Gaussian noise $\mathcal{N}(0, 0.05^2)$.*

For each unlabeled target sample $z_t$, two augmented views are created: a weak view $x_w = a_w(z_t)$ (small Gaussian noise, $\mathcal{N}(0, 0.01^2)$ and a strong view $x_s = a_s(z_t)$ (element-wise random masking with probability 0.10 having additive Gaussian noise, $\mathcal{N}(0, 0.05^2)$, with $y_w$ and $y_s$ as their predictions. To ensure prediction consistency across views, we minimize Eq. 14:

$$\mathcal{L}_{\text{cons}} = \mathbb{E}_{x\sim\mathcal{D}_t}\left[\|y_w - y_s\|_2^2\right]. \tag{14}$$

To increase the model's stability during training, a second encoder $\mathcal{F}(\cdot;\theta_t)$ is used as a teacher model, updated via EMA: $\theta_t \leftarrow \text{EMA}(\theta_s)$. In this student–teacher framework, the teacher encoder receives the weak, noisy data $x_w$ and produces target predictions. To ensure alignment between the student's prediction on the strong view and the teacher's outputs, we apply the teacher loss in Eq. 15:

$$\mathcal{L}_{\text{teach}} = \mathbb{E}_{x\sim\mathcal{D}_t}\left[\|\hat{y}_s - \hat{y}_t\|_2^2\right]. \tag{15}$$

Furthermore, as the source-domain prediction statistics and feature covariances are computed during pre-training, statistical alignment between source and target is ensured by matching the first- and second-order moments (mean and variance) of the student's predictions across domains via the following prediction-statistics loss (Eq. 16):

$$\mathcal{L}_{\text{pred-stat}} = \|\mu_t - \mu_s\|_2^2 + \left\|\sigma_t^2 - \sigma_s^2\right\|_2^2. \tag{16}$$

Additionally, to align feature-level covariance statistics across source features $\mathcal{F}_s$ and target features $\mathcal{F}_t$, we use the CORAL loss in Eq. 17 to align feature distributions across domains in this source-free setting:

$$\mathcal{L}_{\text{coral}} = \|\text{Cov}(\mathcal{F}_t) - \text{Cov}(\mathcal{F}_s)\|_F^2, \tag{17}$$

where $\|\cdot\|_F$ is the Frobenius norm. The total loss is defined as follows:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{cons}}\mathcal{L}_{\text{cons}} + \lambda_{\text{teach}}\mathcal{L}_{\text{teach}} + \lambda_{\text{stat}}\mathcal{L}_{\text{pred-stat}} + \lambda_{\text{coral}}\mathcal{L}_{\text{coral}}.$$

This loss formulation balances consistency, statistical alignment, and teacher-guided supervision for effective SFDA. We use a learning rate of $10^{-3}$ and an exponential moving average coefficient of 0.995. All loss weights are treated as hyperparameters and selected on a held-out validation split: $\lambda_{\text{cons}} = 1.0$, $\lambda_{\text{teach}} = 0.5$, $\lambda_{\text{stat}} = 0.10$, and $\lambda_{\text{coral}} = 0.02$.