Scalable Multi-Floor Wi-Fi Localization via Active and Transfer Learning

Mohamed Ahmed
Centre for Applied AI
Sheridan College
Oakville, Ontario, Canada
ahmed76@sheridancollege.ca

Joseph Chun-Chung Cheung
AI Consultant
Markham, Ontario, Canada
jcheungca@yahoo.ca

Tarek El Salti
School of Applied Computing
Sheridan College
Oakville, Ontario, Canada
tarek.elsalti@sheridancollege.ca

Mahmoud Elboghdady
Centre for Applied AI
Sheridan College
Oakville, Ontario, Canada
mahmoud.elboghdady@sheridancollege.ca

Ahmed Ibrahim
Centre for Applied AI
Sheridan College
Oakville, Ontario, Canada
ahmed.ibrahim@sheridancollege.ca

***Abstract*—Wi-Fi fingerprinting is a widely used approach for indoor localization, but its scalability is limited by the cost of constructing and maintaining radio maps, particularly in large multi-floor environments. This paper proposes a unified framework that integrates Active Learning, denoising representation learning, and Transfer Learning to reduce data collection and training requirements while maintaining localization accuracy. On a source floor, Active Learning identifies informative reference points for efficient radio map construction, while a Denoising Supervised Autoencoder Regressor (DSAR) learns robust representations from noisy RSS measurements. Knowledge acquired on the source floor is then transferred to structurally similar target floors through projection-based adaptation, reducing the need for exhaustive site surveys. Experiments conducted on a real-world multi-floor Wi-Fi dataset demonstrate that the proposed approach improves localization accuracy while substantially reducing deployment effort. Active Learning reduces positioning error by up to 39%, while Transfer Learning decreases training data and training time requirements by up to 58% and 70%, respectively. Overall, the framework reduces the number of required reference points by 33.3% and significantly lowers data collection and training costs, demonstrating a practical and scalable solution for multi-floor indoor localization.**

***Index Terms*—micro-location; Wi-Fi localization; indoor localization; machine learning; generative AI**

I. Introduction

Indoor localization has become a key component of modern intelligent environments, enabling applications such as indoor navigation, asset tracking, and context-aware services in large buildings. Among the various approaches, Wi-Fi Received Signal Strength (RSS)-based fingerprinting remains one of the most widely adopted techniques due to its reliance on existing infrastructure and compatibility with commodity mobile devices.

In fingerprinting-based localization, RSS measurements are collected at known reference points (RPs) to construct a radio map, and a machine learning (ML) model is trained to infer a user's location by matching new measurements to this map. The positioning accuracy of fingerprinting depends on the information contained in the collected RSS measurements, and recent work has proposed methods to enrich this information [1]. Nevertheless, fingerprinting suffers from a fundamental limitation: the need for dense and labor-intensive data collection to construct accurate radio maps [2]. This challenge is further amplified in multi-floor environments, where site surveys must be repeated for each floor, making large-scale deployment costly and impractical. This motivates the need to reuse spatial information across structurally similar floors, enabling knowledge transfer through transfer learning techniques and reducing redundant data collection. Moreover, RSS measurements are highly sensitive to environmental and temporal variations (e.g., human presence, furniture changes), causing radio maps to quickly become outdated, degrading localization accuracy and requiring frequent data recollection and model retraining [2], [3].

To address these limitations, prior works have explored techniques to improve data efficiency and robustness. In particular, Active Learning (AL) reduces data collection effort by selecting informative RPs [4]–[7], while representation learning methods, such as autoencoder-based models, improve robustness to noisy RSS measurements [8], [9]. However, these approaches are typically studied in isolation. Active Learning methods focus on selecting informative samples, but do not explicitly address signal noise and variability. In contrast, representation learning approaches, such as Denoising Autoencoders (DAE), improve robustness to noise but assume sufficiently dense data and do not account for reduced spatial coverage.

This limitation reduces their effectiveness in real-world deployments, where both data efficiency and robustness are required simultaneously. In particular, sparse sampling introduces spatial gaps that Active Learning alone cannot resolve, while representation learning methods are not designed to compensate for incomplete coverage. In this work, we adopt a Denoising Supervised Autoencoder Regressor (DSAR), i.e., a

979-8-3195-0167-7/26/$31.00 ©2026 IEEE

DAE-based encoder coupled with a regressor, to jointly learn robust representations and localization mappings.

Therefore, a unified approach that jointly addresses data efficiency and robustness is needed. Moreover, the effectiveness of different transfer learning strategies for adapting learned representations under sparse sampling remains underexplored.

To the best of our knowledge, this is the first work to jointly integrate a novel Active Learning strategy, DAE-based representation learning, and transfer learning for scalable multi-floor indoor localization under sparse sampling conditions, yielding significant gains in accuracy and efficiency.

This motivates the following research question: *How do different transfer learning strategies impact localization performance when combined with Active Learning and DAE-based representation learning under sparse and noisy RSS conditions?*

To address the limitations of data-intensive and environment-sensitive Wi-Fi fingerprinting for multi-floor indoor localization, this paper makes the following contributions:

1) We propose a regression-based Active Learning strategy that leverages ensemble uncertainty, quantified using the coefficient of variation of ensemble predictions, to identify informative reference points (RPs) and improve sampling efficiency and localization performance.
2) We develop a unified cross-floor localization framework that integrates Active Learning, DAE, and Transfer Learning to reduce data collection and training effort while maintaining localization accuracy across multiple floors.

II. Literature Review

Research on cross-floor indoor localization remains relatively limited compared to general indoor positioning. Existing approaches typically rely on floor-specific fingerprint databases or independent model training for each floor, resulting in significant site-survey costs. More recent studies have explored knowledge transfer across floors and buildings, but often assume dense fingerprint collection and do not explicitly address sampling efficiency [10], [11]. Consequently, the literature is largely fragmented across the underlying techniques used to improve scalability and robustness, including Active Learning, representation learning, and Transfer Learning. This section reviews these research directions and highlights the remaining gaps relevant to cross-floor indoor localization.

A. Active Learning

Active Learning (AL) is widely studied to reduce data collection effort by selecting the most informative samples for labeling. Prior studies have applied uncertainty-based and variance-driven sampling strategies to wireless and localization problems, demonstrating improved efficiency compared to random sampling [5], [7], [12]. In indoor localization, AL has been used to guide data collection by identifying positions where additional samples improve model accuracy or update outdated radio maps [11], [13]. While these approaches reduce labeling cost, they primarily focus on sample selection and do not address robustness to noisy RSS measurements or performance under sparse sampling.

B. Representation Learning

Representation learning methods, particularly Denoising Autoencoders (DAEs), have been introduced to improve robustness to noise in RSS data by learning compact latent representations [14]. In indoor localization, stacked and supervised autoencoder architectures are commonly used to model RSS fingerprints and enhance localization performance under signal variability [8], [15], [16]. However, these approaches generally assume dense and well-distributed training data and are not explicitly designed to operate under constrained sampling scenarios.

C. Transfer Learning

Transfer learning has attracted significant attention as a mechanism to improve scalability by reusing knowledge across environments [17], [18]. In the context of cross-floor indoor localization, transfer learning offers an attractive alternative to constructing independent radio maps for each floor, potentially reducing survey effort and deployment cost. In indoor localization, prior work has explored transferring models across time, space, and devices using both classical and deep learning approaches [10]. Recent studies highlight that learned representations can generalize across related domains, particularly in early network layers [19]. However, existing transfer learning approaches typically assume the availability of sufficiently large datasets and do not explicitly consider adaptation under sparse sampling conditions.

D. Summary and Research Gap

Overall, existing work addresses Active Learning, representation learning, and transfer learning largely in isolation. AL focuses on reducing sampling cost, representation learning improves robustness to noise, and transfer learning enhances scalability across environments. However, the combined effect of sparse sampling, noisy RSS measurements, and cross-floor adaptation remains largely unexplored. Although a limited number of studies have investigated cross-floor indoor localization, existing approaches generally rely on dense fingerprint collection and have not jointly addressed sampling efficiency, representation robustness, and cross-floor knowledge transfer. As a consequence, this limits the practicality of existing approaches in real-world deployments. This gap motivates the proposed unified framework presented in the next section, which jointly integrates these components to enable efficient and robust indoor localization under constrained data collection scenarios.

III. Methodology

A. Framework Overview

The framework operates under the following assumptions: (1) the user's floor is correctly identified prior to localization; (2) source and target floors exhibit structural similarity

to enable Reference Point (RP) projection and adjustment; (3) prior localization coverage requirements are available to estimate the number of RPs needed, thereby defining the sampling budget; (4) the set of observable Access Points (APs) remains stable over time; and (5) Received Signal Strength (RSS) measurements capture variations due to typical user orientations.

An RP is a predefined location where RSS measurements are collected, and RP projection denotes the mapping of selected RPs from a source floor onto a structurally similar target floor.

Figure 1 presents an overview of the three main pipelines in the proposed framework: Active Learning, Transfer Learning, and Location Inference.

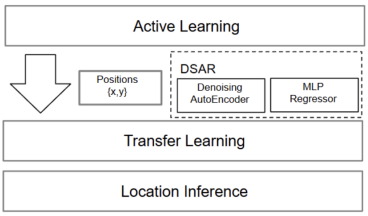


Fig. 1: Overview of the three main pipelines.

The main objective of the Active Learning (AL) process is to identify the most informative Reference Points (RPs) required to support accurate localization on the reference floor (see Fig. 1), whose RSS samples are used to train the Denoising Supervised Autoencoder Regressor (DSAR) model that is subsequently incorporated into a transfer learning framework to enable target-floor localization. In this stage, a reduced radio map is constructed using projected reference points, minimizing exhaustive RP searching and RSS sampling requirements. Finally, the adapted models are deployed for real-time localization, completing the end-to-end pipeline. The following sections detail each stage of the proposed pipeline and describe the key components involved.

B. Source-Floor Localization Model and Active Learning

In this work, a Denoising Supervised Autoencoder Regressor (DSAR) is employed as the localization model [16]. The DSAR combines a Denoising Autoencoder (DAE) with a Multilayer Perceptron (MLP) regressor to jointly learn robust feature representations and spatial mapping (see Fig. 2). The DAE encodes the input RSS vectors mixed with noise into a lower-dimensional latent representation that captures the underlying structure of the signal space while mitigating noise. This latent representation is simultaneously used by the decoder to reconstruct the input signal, and the MLP regressor to estimate spatial coordinates. The model is trained using a joint objective that combines reconstruction loss and regression loss, allowing both components to be optimized simultaneously. This design enables the DSAR model to remain robust under noisy and sparse RSS conditions, which are typical in real-world deployments. Once trained, the DSAR model and its learned parameters are stored and used for transfer learning and localization on target floors.

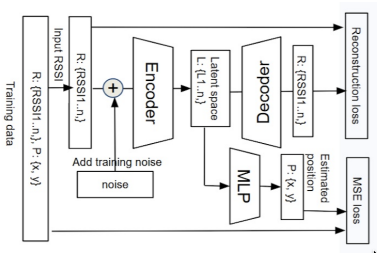


Fig. 2: Denoising Supervised Autoencoder Regressor (DSAR).

Once the reference floor (a.k.a. source-floor) is identified, an Active Learning strategy is employed to construct a set of informative Reference Points (RPs) (see Fig. 3). The objective is to identify the most informative set of RPs that can support accurate localization while reducing RSS data collection effort. The selected RPs along with the collected RSS samples

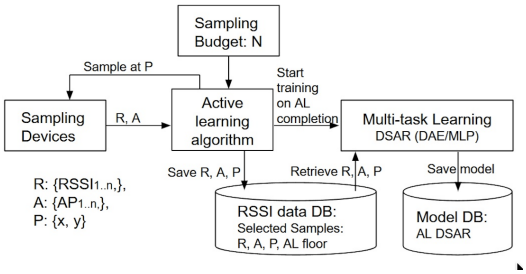


Fig. 3: Active Learning-based RP and DSAR model training.

are subsequently used to create a reduced radio map and to train the localization model. By focusing on informative sampling locations, the AL process aims to maximize model performance under a constrained sampling budget. At the end of this stage, a trained DSAR model and the selected RPs on the reference floor are produced and passed to the transfer learning stage.

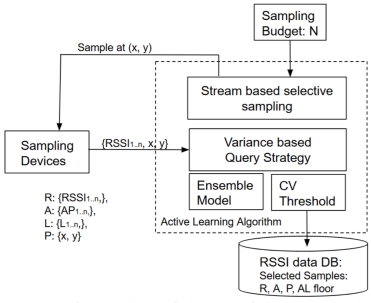


Fig. 4: Active Learning algorithm with ensemble uncertainty.

The Active Learning algorithm identifies the most informative sampling locations based on model uncertainty. To initialize the process, a small set of spatially diverse RPs is selected from different regions of the floor to ensure adequate coverage (see Fig. 4). At each iteration, a candidate location is evaluated by collecting RSS samples and generating multiple location predictions using an ensemble regressor. The

variability among these predictions is quantified using the Coefficient of Variation (CV), which reflects the level of model uncertainty. Higher CV values indicate greater disagreement among the models and, consequently, higher informativeness. If the CV exceeds a predefined threshold, the candidate location is selected and added to the training dataset. Otherwise, it is discarded. This process continues until the predefined sampling budget is reached. This strategy prioritizes locations that improve the model's understanding of the RSS–location relationship, resulting in a more informative and efficient radio map. For completeness, Algorithm 1 presents the pseudocode of the proposed approach. Lines 1–2 show that four distinct locations are initially sampled to prime the ensemble regressor. At each iteration, the currently accepted samples are used to train the ensemble model (Lines 9–11). A candidate coordinate is then randomly selected from the set of unvisited locations (Line 12), and its corresponding RSSI samples are collected (Line 13). The ensemble regressor is leveraged to produce multiple predictions (one per tree), forming a distribution of predicted positions (Lines 14–18). The mean and standard deviation of this distribution are computed component-wise. Since each prediction is a 2D coordinate, the magnitudes of the mean and standard deviation vectors are used to derive a scalar Coefficient of Variation (CV) (Lines 19–21), which quantifies the disagreement among ensemble members. As shown in Lines 22–26, if the CV exceeds a predefined threshold, the candidate position is considered informative and added to the RSSI database. The set of selected coordinates is updated accordingly, contributing to the stopping condition (Line 3). Otherwise, the candidate is rejected. This algorithm runs in O(n) where n is the number of available coordinates in $C_{available}$.

Algorithm 1 Active Learning Algorithm

Require: Target number of points N, starting coordinates C_{start}, available coordinates $C_{available}$, CV threshold τ, tree-based ensemble model M, dataset D with RSSI fingerprints $(RSSI_1, RSSI_2, \ldots, RSSI_n, x, y)$
Ensure: Selected coordinates $C_{selected}$
1: $C_{selected} \leftarrow C_{start}$
2: $C_{rejected} \leftarrow \emptyset$
3: **while** $|C_{selected}| < N$ **do**
4: $C_{visited} \leftarrow C_{selected} \cup C_{rejected}$
5: $C_{unvisited} \leftarrow C_{available} \setminus C_{visited}$
6: **if** $C_{unvisited} = \emptyset$ **then**
7: **break**
8: **end if**
9: $X \leftarrow$ RSSI samples in D belonging to coordinates in $C_{selected}$
10: $y \leftarrow$ true (x, y) positions in D belonging to coordinates in $C_{unvisited}$
11: Train tree-based ensemble model M using (X, y)
12: Randomly select coordinate c from $C_{unvisited}$
13: $X_c \leftarrow$ RSSI samples in D belonging to coordinate c
14: $P \leftarrow \emptyset$
15: **for** each tree t in M **do**
16: $\hat{Y}_t \leftarrow t$.predict(X_c)
17: Add $\hat{Y}_t$ to P
18: **end for**
19: $\mu \leftarrow \|\text{mean}(P)\|$
20: $\sigma \leftarrow \|\text{std}(P)\|$
21: $CV \leftarrow \dfrac{\sigma}{\mu}$
22: **if** $CV \geq \tau$ **then**
23: Add c to $C_{selected}$
24: **else**
25: Add c to $C_{rejected}$
26: **end if**
27: **end while**
28: **return** $C_{selected}$

C. Cross-Floor Transfer Learning

First, the Reference Points (RPs) selected on the reference floor are projected onto the target floor to verify their validity, i.e., their physical existence within the target floor layout (see Fig. 5). If a projected RP does not exist on the target floor (e.g., it falls outside the accessible area), an alternative RP is generated at a nearby feasible location. Once all required RPs are confirmed, Wi-Fi RSS measurements are collected at these positions. Rather than training the DSAR model from scratch, transfer learning is applied by initializing the target floor model with parameters learned from the reference floor and then fine-tuning the model using the newly collected target floor RSS samples. To account for differences in Access Point (AP) availability between floors (in our environment, the reference and target floors have 450 and 370 APs, respectively), the boundary layers of the Denoising Autoencoder (DAE) are replaced—specifically, the first encoder layer and the final decoder layer—while the intermediate layers are frozen. The Multilayer Perceptron (MLP) regressor layers are configured using different strategies (i.e., replace, freeze, or unfreeze) to determine the optimal adaptation setup. This approach enables the model to retain learned representations from the DAE while allowing the regressor sufficient flexibility to adapt to spatial and signal variations on the target floor. At the end of this process, the adapted DSAR model for the target floor is saved and made available to support localization during system deployment.

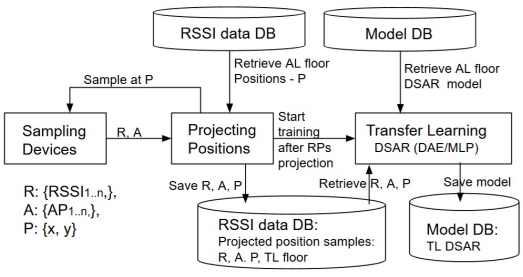


Fig. 5: Target floor transfer learning workflow for DSAR.

D. Online Localization

During system operation, localization is performed using the models trained and adapted during the offline stages. We assume that the target floor has been identified beforehand, as described in Section III-A. Given an incoming Wi-Fi RSS sample from a user device, the corresponding floor-specific

DSAR model is selected. The Denoising Autoencoder (DAE) first transforms the RSS input into a latent representation. Subsequently, the Multilayer Perceptron (MLP) regressor uses this representation to estimate the user's spatial coordinates. The online localization phase requires only a single forward pass through the DSAR model. Consequently, the computational overhead is low and suitable for real-time deployment. A detailed latency and runtime evaluation is left for future work.

IV. Experimental Design

To evaluate the proposed framework, we adopt a two-stage experimental design aligned with the system pipeline: (i) Active Learning (AL), which selects the most informative subset of reference points (RPs) on the source floor, and (ii) Transfer Learning (TF), which adapts the trained model to the target floor under constrained data availability using the selected RPs.

A. Dataset and Environment

Experiments are conducted on a multi-floor Wi-Fi RSS dataset collected in the S-Building at Sheridan College. The 3rd floor serves as the source floor with approximately 110 reference points (RPs), each collecting 30 RSS measurements using a single device with unconstrained user orientation. After filtering for known Access Point (AP) MAC addresses, 450 APs are retained. The 4th floor serves as the target floor and contains 370 APs, of which 360 are shared with the source floor and 10 are floor-specific. RSS data are collected at the projected RPs. Both floors share a similar architectural layout, and RSS measurements are collected only along the corridors, which are surrounded by rooms. Consequently, most measurements are obtained under non-line-of-sight (NLOS) conditions, as signals are attenuated by walls separating the corridors from the surrounding rooms and by occupants present during data collection. Together, these provide a realistic environment that allows us to emulate the online interactive process effectively.

B. Active Learning Configuration and Evaluation

Active Learning (AL) is used to select an informative subset of Reference Points (RPs) under a constrained sampling budget using a Random Forest (RF)–based uncertainty measure, quantified by the Coefficient of Variation (CV). RPs with higher uncertainty (i.e., higher CV) are considered more informative and are prioritized for selection. The selected RPs are used to construct a reduced radio map for localization. AL performance is evaluated using Root Mean Square Error (RMSE) and sampling efficiency across CV thresholds (0.08, 0.16, 0.24, 0.32, and 0.40) and RP budgets ($N = 10, 20,$ and 30), with random sampling as a baseline for comparison. For each configuration, a 70/30 train–test split is used. The training set is used for RF-based RP selection and model training, while unselected coordinates are returned to the test set for evaluation. Results are reported as the mean and standard deviation across repeated runs. The selected CV thresholds are informed by Statistics Canada data-quality guidelines [20], where CV values below 16% indicate acceptable variability, 16–33% indicate marginal variability, and values above 33% indicate high variability. In the AL setting, however, higher CV values correspond to greater model uncertainty and therefore more informative sampling locations.

C. Transfer Learning Configuration and Evaluation

Transfer learning (TF) is performed using the representative AL configuration ($N = 20$, $CV = 0.16$), selected for its balance between localization accuracy and sampling efficiency, as observed in the Active Learning evaluation. The corresponding RPs are projected onto the target floor to construct a reduced radio map. The Denoising Supervised Autoencoder Regressor (DSAR) model trained on the source floor is used to initialize the target model. To account for differences in AP dimensionality, the encoder input and decoder output layers are replaced, while intermediate layers are frozen to preserve learned representations. Within the DSAR model, the Multilayer Perceptron (MLP) regressor is adapted using multiple strategies: U0 (fully frozen), U1–U2 (progressively unfreezing the first one or two layers), and R1–R2 (replacing the first layer, with R2 additionally unfreezing the second layer). These configurations enable systematic evaluation of different levels of model flexibility and knowledge reuse. Evaluation is conducted under constrained data and training settings. The number of training epochs is varied from 5 to 50 in steps of 5 to analyze convergence behavior, while the transfer size, defined as the fraction of RSS samples used per RP, is adjusted from 0.2 to 0.95 in increments of 0.2 to assess data efficiency. Performance is measured using Root Mean Square Error (RMSE), evaluated from two perspectives: (i) RMSE versus epochs to assess convergence speed, and (ii) RMSE versus transfer size to evaluate data efficiency. A non-transfer baseline (T2), trained from scratch on the target floor for 50 epochs, is used for comparison. In addition, tolerance-based thresholds ($\pm5\%$, $\pm10\%$, $\pm15\%$) relative to the source-floor DSAR baseline RMSE are introduced to assess convergence efficiency and transfer size. Comparative analysis is conducted across all regressor adaptation strategies to evaluate the effectiveness of transfer learning under varying conditions. All models share identical architectures and training protocols to ensure fair comparison.

D. DSAR Model Configuration

The DSAR model employs a symmetric encoder–decoder architecture with a latent dimension of 96, followed by an MLP regressor. Each layer block consists of a linear layer, ReLU activation, batch normalization, and dropout (rate 0.2). The full layer dimensions are summarized in Table I. RSS samples for selected RPs are split 70/30 for model training; samples for unselected RPs are used for evaluation under different AL/TF strategies.

V. Results

A. Reference Floor (3rd Floor) Active Learning Results

For the Active Learning (AL) performance on the 3rd floor, as shown in Fig. 6 and Table II, AL generally outperforms

TABLE I: DSAR architecture layer dimensions.

Model	Input	Hidden	Output
Encoder	No. of APs × 256	256×128	128 × 96 (latent)
Decoder	96 (latent) × 128	128×256	256 × No. of APs
Regressor	96 (latent) × 64	64×32, 32×32, 32×16	16×2

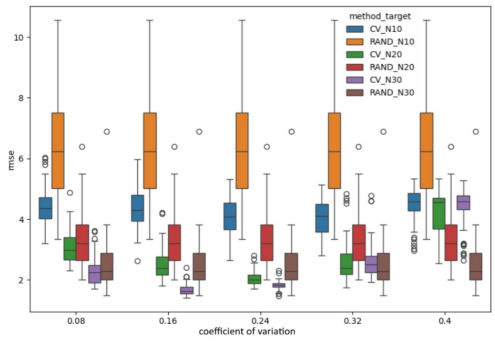


Fig. 6: RMSE of Active Learning vs. random sampling.

random sampling across most evaluated thresholds and target-RP settings, particularly at moderate CV values. Note that the reported RMSEs are computed from the ensemble model (i.e., the Random Forest (RF) regressor) predictions compared to the true positions. For 10 target points, AL maintains relatively stable RMSE across thresholds 0.08–0.32, ranging from 4.06 ± 0.61 to 4.45 ± 0.62, while consistently selecting the full target number of RPs up to threshold 0.32. Even at stricter thresholds (0.4), where the average number of selected RPs decreases slightly, AL still preserves similar performance. Across all thresholds, this corresponds to an approximate 31–37% RMSE reduction compared to random sampling. For 20 target RPs, AL demonstrates a clear improvement as the threshold increases from 0.08 to 0.24, with RMSE decreasing from 3.08 ± 0.57 to 2.05 ± 0.25, which is the best performance observed for this setting. Beyond this point, performance begins to degrade as the stricter threshold reduces the number of selected RPs below the target. Nevertheless, AL outperforms random sampling for most thresholds up to 0.32, achieving an approximate 9–39% RMSE reduction, with the strongest improvement occurring at a CV threshold of 0.24. A similar trend is observed for 30 target RPs, where AL achieves its best performance at threshold 0.16, reaching an RMSE of 1.69 ± 0.21. Performance remains strong at threshold 0.24, despite selecting fewer RPs on average, but deteriorates at higher thresholds as the number of selected RPs drops substantially, leading to inferior accuracy compared to random sampling for CV values ≥ 0.32. In contrast, at low and moderate thresholds (0.08–0.24), AL outperforms random sampling, achieving an approximate 8–33% RMSE reduction. Overall, the results suggest that moderate thresholds (around 0.16–0.24) provide the best balance between sample quality and sufficient RP coverage, particularly for higher target-RP settings. At very high thresholds, the AL strategy becomes overly selective, leading to fewer selected RPs and a corresponding decline in localization performance.

B. Target Floor (4th Floor) DSAR Results

As noted before, we continue the transfer learning evaluation using the Active Learning configuration at N = 20, CV = 0.16, selected for its balance between localization accuracy and sampling efficiency. To assist the evaluation, the RMSE result (2.06 m) along with its tolerance variations at +5% (2.16 m), +10% (2.27 m), and +15% (2.37 m) achieved by the source-floor DSAR model using the same AL configuration is used as the baseline for comparison with the target-floor DSAR performance. Rather than reporting all raw data (a total of 300 combinations), results are first aggregated over the ten transfer-epoch and five transfer-size settings. For RMSE-versus-epoch analysis, the reported values correspond to the mean RMSE averaged across all transfer sizes. Conversely, for RMSE-versus-size analysis, the reported values correspond to the mean RMSE averaged across all epoch settings. Fig. 7

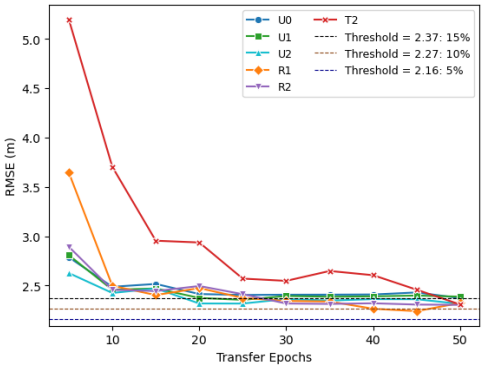


Fig. 7: TL vs T2 (RMSE over Transfer Epochs).

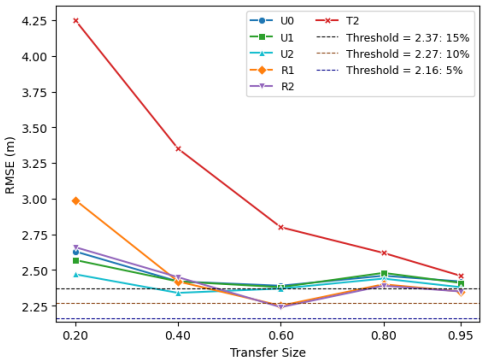


Fig. 8: TL vs T2 (RMSE over Transfer Size).

and Table III evaluates the training-time budget, analyzing how many epochs each strategy requires to achieve competitive performance: each row reports the mean RMSE averaged over all transfer sizes at the given epoch count. The comparison columns ($*_vs_15\%$) demonstrate the deviation from the $+15\%$ baseline. Negative values indicate that the RMSE has dropped below the threshold, while positive values indicate that performance remains above it. Furthermore, values lower than -5% imply that the RMSE has also crossed the $+10\%$ benchmark. Among the evaluated methods, from the U family, U2 is the first transfer learning strategy to surpass the $+15\%$

TABLE II: Active Learning results compared to the random-sampling baseline.

Methods	N	CV	RMSE mean	RMSE std	N mean	N std	Sel. rate mean	Sel. rate std
AL	10	0.08	4.450	0.622	10.00	0.000	0.957	0.064
AL	10	0.16	4.316	0.615	10.00	0.000	0.821	0.117
AL	10	0.24	4.110	0.619	10.00	0.000	0.604	0.149
AL	10	0.32	4.055	0.614	10.00	0.000	0.288	0.146
AL	10	0.40	4.464	0.604	8.14	1.666	0.088	0.046
AL	20	0.08	3.077	0.574	20.00	0.000	0.903	0.067
AL	20	0.16	2.524	0.529	20.00	0.000	0.624	0.100
AL	20	0.24	2.054	0.246	19.76	0.687	0.263	0.079
AL	20	0.32	2.633	0.729	15.20	2.626	0.132	0.023
AL	20	0.40	4.283	0.813	8.36	2.337	0.073	0.020
AL	30	0.08	2.323	0.510	30.00	0.000	0.869	0.058
AL	30	0.16	1.689	0.206	30.00	0.000	0.477	0.092
AL	30	0.24	1.822	0.157	23.36	2.337	0.204	0.023
AL	30	0.32	2.648	0.643	14.70	2.573	0.128	0.022
AL	30	0.40	4.409	0.723	8.00	2.185	0.070	0.019
RAND	10	N/A	6.462	1.784	10.00	0.000	1.000	0.000
RAND	20	N/A	3.379	0.902	20.00	0.000	1.000	0.000
RAND	30	N/A	2.512	0.863	30.00	0.000	1.000	0.000

TABLE III: TL vs T2 (RMSE over Number of Epochs).

Epochs	R1	R2	U0	U1	U2	T2	R1_vs_15	R2_vs_15	U0_vs_15	U1_vs_15	U2_vs_15	T2_vs_15
5	3.64	2.89	2.79	2.81	2.63	5.20	61.91%	25.42%	20.24%	21.28%	12.48%	137.22%
10	2.50	2.46	2.49	2.46	2.42	3.70	6.24%	4.26%	5.73%	4.26%	2.68%	64.76%
15	2.40	2.44	2.52	2.47	2.47	2.95	1.65%	3.62%	7.14%	4.92%	4.90%	28.40%
20	2.47	2.50	2.41	2.37	2.32	2.94	5.10%	6.12%	2.17%	0.25%	-2.48%	27.53%
25	2.37	2.41	2.40	2.35	2.32	2.57	0.25%	2.10%	1.71%	-0.69%	-2.52%	9.76%
30	2.34	2.32	2.41	2.39	2.36	2.55	-1.23%	-2.50%	1.83%	1.21%	-0.29%	8.65%
35	2.34	2.31	2.41	2.39	2.35	2.65	-1.50%	-2.69%	1.85%	0.84%	-1.03%	13.51%
40	2.26	2.32	2.41	2.39	2.36	2.60	-5.08%	-2.38%	1.95%	1.04%	-0.36%	11.43%
45	2.24	2.31	2.43	2.40	2.36	2.46	-6.26%	-3.01%	2.94%	1.42%	-0.39%	4.31%
50	2.32	2.30	2.38	2.39	2.31	2.31	-2.24%	-3.16%	0.42%	0.88%	-2.68%	-2.98%

threshold, achieving an RMSE of 2.32 m after 20 epochs and maintaining performance below this level throughout training, followed by U1 crossing at 25 epochs (2.35 m) and R2 at 30 epochs (2.32 m). Further improvements are observed with R1, which reaches an RMSE of 2.26 m after 40 epochs and becomes the first method to cross the $+10\%$ benchmark.

In contrast, the non-transfer model (T2) only reaches the same threshold after 50 epochs, requiring approximately three times more training iterations than U2. This highlights a significant convergence advantage, where transfer learning achieves comparable accuracy with significantly reduced training time. Overall, these results demonstrate that transfer learning not only provides competitive localization accuracy but, more importantly, significantly accelerates convergence, enabling efficient training with fewer epochs while maintaining or exceeding baseline performance. While Fig. 7 traces performance against the training-time budget, Fig. 8 and Table IV turn to the data budget, examining how much target-floor data each strategy requires to remain competitive. Each row reports the mean RMSE averaged over all epochs at the given transfer size, and the comparison columns ($*_vs_15\%$) follow the convention introduced in Table III. The ordering along the size axis mirrors the layer-depth hierarchy of the strategies. Again, from the U family, U2 was the first to cross the $+15\%$ threshold, reaching 2.34 m at $size = 0.4$, that is, only 40% of the target-floor data. The R family reaches its lowest values at $size = 0.6$, where R1 and R2 record 2.25 m and 2.24 m, both clearing the -5% mark and making $size = 0.6$ the single point on the data-fraction axis where any strategy crosses the $+10\%$ benchmark. U0 and U1 improve as well, reaching their lowest RMSE at 0.6, but do not drop below the $+15\%$ line at any fraction. The non-transfer model (T2) behaves exactly as expected for a from-scratch model, improving steadily as more data is added, reaching its lowest RMSE of 2.46 m at $size = 0.95$, yet it fails to reach the $+15\%$ line. A non-monotonic anomaly is captured at $size = 0.8$ for all the transfer learning models (i.e., the issue is not applicable to T2); more detail is covered in the Discussion Section.

Summary: Taken together, and as summarized in Table V, the proposed Transfer Learning matches the from-scratch baseline (T2) in RMSE performance while reaching it at a fraction of its training time and data cost. The R family delivers the closest match to the source floor; R2 reaches 2.08 m in just 15 epochs on 60% of the target-floor data. The R family is therefore the stronger choice for saving training time, with a 20–70% reduction in epochs compared to T2. The U family, in contrast, has a greater impact on data, saving 37–58% compared to T2. T2 itself performed exactly as expected, improving with more data and more epochs.

VI. Discussion

A. Active Learning

The objective of Active Learning (AL) in this work is to identify an effective set of RPs on one floor that yields

TABLE IV: TL strategies (RMSE over normalization size).

Size	R1	R2	U0	U1	U2	T2	R1_vs_15	R2_vs_15	U0_vs_15	U1_vs_15	U2_vs_15	T2_vs_15
0.20	2.99	2.66	2.63	2.57	2.47	4.25	30.15%	14.13%	12.67%	9.76%	4.90%	91.31%
0.40	2.42	2.45	2.42	2.42	2.34	3.35	2.48%	3.93%	2.48%	2.48%	-1.41%	47.62%
0.60	2.25	2.24	2.39	2.38	2.37	2.80	-5.78%	-6.26%	1.02%	0.53%	0.05%	20.92%
0.80	2.40	2.39	2.46	2.48	2.44	2.62	1.50%	1.02%	4.42%	5.39%	3.45%	12.18%
0.95	2.35	2.35	2.42	2.41	2.38	2.46	-0.92%	-0.92%	2.48%	1.99%	0.42%	4.42%

TABLE V: Each strategy at its best operating point.

Method	Best size	Best epoch	Best RMSE (m)	Epochs saved	Data saved
R1	0.60	40	2.15	20%	37%
R2	0.60	15	2.08	70%	37%
U0	0.60	35	2.22	30%	37%
U1	0.40	25	2.23	50%	58%
U2	0.40	25	2.20	50%	58%
T2	0.95	50	2.09	–	–

strong localization performance and project these selected RPs onto other floors with similar layouts. While RP data collection is still required on the other floors, this approach removes the need to repeatedly perform the RP search and selection process for each floor, thereby significantly reducing the overall sampling time and effort involved. Accordingly, parameters such as the sampling budget N, the Coefficient of Variation (CV), and the resulting selection rate (i.e., positions accepted divided by the total number of positions sampled) are central to evaluating AL effectiveness, as they directly govern the trade-off between data-collection cost and localization accuracy. As shown in Fig. 6 and Table II, AL generally achieves lower RMSE and lower variability than random sampling when evaluated using an equivalent number of RPs. This indicates that AL selects more informative RPs, which better enhance the ensemble (i.e., Random Forest) model's understanding of the floor's RSSI radio map. As a result, AL delivers more reliable and consistent localization performance, whereas random sampling exhibits comparatively lower localization accuracy with higher variability. For all N, as the CV threshold increases, AL RMSE improves until it reaches a peak (CV_{N10}@0.32, CV_{N20}@0.24, CV_{N30}@0.16), after which performance degrades. This is because, as the threshold increases, fewer RPs satisfy the CV criterion, removing non-informative positions and decreasing RMSE. However, at excessively high thresholds, AL becomes overly selective, sharply reducing the number of eligible informative positions and resulting in performance degradation instead. When $N = 10$, the performance gain achieved by increasing the CV threshold is not as significant as it is for $N = 20$ and $N = 30$. The primary reason is that at a smaller N, the covered area per RP increases. This larger coverage area imposes a fundamental constraint on localization performance, irrespective of the CV threshold. Nevertheless, compared to random sampling at $N = 10$, even at a lower CV (0.08), AL still provides a 30% improvement in RMSE performance. As the CV threshold increases, the selection rate decreases, meaning fewer sampled positions meet the selection criterion. Therefore, more sampling effort is required to identify a sufficient number of informative RPs, increasing the overall data-collection cost. This is because the informativeness of a location is not known beforehand, and identifying suitable positions becomes increasingly difficult as the CV threshold increases. Our experimental results show that in the moderate CV range (corresponding to CV ≈ 0.16), AL achieves the best balance between localization accuracy and sampling effort (i.e., with a selection rate of approximately 0.5, indicating a moderate and manageable sampling effort), offering near-optimal RMSE for a given sampling budget. This operating point serves as a practical configuration for deployment.

B. Transfer Learning

The experimental results show that the transfer-learning models achieve the predefined performance thresholds using substantially fewer training epochs and less training data than the non-transfer baseline. This suggests that the DSAR model trained at the source floor has successfully extracted the needed features from the corresponding RSS data, and the transfer learning process is equally successful in applying the learned parameters in the target floor model. This can be further illustrated with U0, where the encoder's input and the decoder's output are changed, but the regressor coefficients are frozen. Note that the encoder and decoder changes are needed because of the reduction in the number of APs on the target floor. Although U0 does not reach the +15% benchmark, it still offers a respectable performance (0.42% or 1 cm above the target) under such a limited condition. By changing the first two (out of seven) layers of the regressor, the transferred models can readily achieve the benchmark thresholds (see Tables III and IV).

On further study with U0, we observed that the RMSE drops rapidly during the early epochs before stabilizing. Since the regressor remains frozen in the U0 configuration, this initial improvement can be attributed primarily to the autoencoder. A similar trend is observed across the other transfer-learning strategies and the non-transfer baseline (T2), where performance converges around epoch 10 before diverging, suggesting that the autoencoder drives the initial gain while the regressor contributes subsequent refinement.

To further investigate this behavior, we visualize the latent space using UMAP. Encoder latent variables from the R1 model (60% training size, 45 epochs) are first clustered into five groups using k-means and then projected into two dimensions. As shown in Fig. 9, meaningful structure emerges by epoch 10 and becomes more refined by epoch 45. The corresponding localization scatter plots in Fig. 10 exhibit a similar progression. These changes are reflected in the RMSE, which decreases from 6.42 m to 2.51 m and 2.25 m at epochs

1, 10, and 45, respectively. The large improvement between epochs 1 and 10, followed by a further refinement thereafter, supports the observation that the autoencoder provides the dominant initial gain, while the regressor contributes further performance improvements.

Comparing the two families, Fig. 7 shows that the U family shares a similar pattern where the RMSE continues to drop from 5 to 20 epochs, then settles steadily around the same level. Among the U family members, U2 offers the best overall performance, suggesting that the additional trainable layer provides useful adaptation flexibility. The R family shares a different pattern. The RMSE drops from 5 to 10 epochs, then goes up and down until the RMSE settles at 30 epochs. After that, R2 tends downward steadily until the end, while R1 continues to adjust. The second-layer blocks in R2 seem to offer the capability to stabilize the performance. Given the above, the U family has managed to settle at an earlier stage, while the R family provides a better opportunity to gain additional RMSE performance. This is likely due to how the regressor chooses its coefficients. In the unfreezing scenario, the coefficients are initialized with the optimum values identified on the reference floor; large deviations from these optimum values may be difficult to achieve during adaptation. On the other hand, replacing (i.e., randomly initializing the coefficients from scratch) offers an opportunity for the regressor to seek better coefficients to integrate with the adjusted latent space. Considering the above factors, the two-layer approach is preferred over the one-layer approach for potential overall performance gain (RMSE, epoch, and training-size reduction) and model stability. R2 is preferred over U2 for better RMSE performance while maintaining competitive epoch and training-size reduction. However, if training-size reduction is the primary goal, then U2 is still the preferred choice, as noted in the Results Section.

Regarding the anomaly (transfer size $= 0.8$) identified in the Results Section, on close examination of the results, all the transfer models in fact experienced RMSE performance degradation after reaching the peak performance. This phenomenon is mainly due to the overfitting of the DSAR. Given that the source DSAR model is already well trained, the transferred models, after reaching their peak learning, have started memorizing idiosyncrasies of the target-floor data instead of learning transferable characteristics of the target-floor environment. We can also confirm this by examining when the overfitting happened in the transfer learning models. U2 has inherited all the model coefficients, and it has the greatest flexibility to learn in the U family. As a result, it is the quickest to reach the peak and also the quickest to experience overfitting, i.e., at transfer size 0.6. U0 and U1 have less learning flexibility, and the R family has added complexity (i.e., a new regressor layer); they all take longer to learn, and therefore the overfitting is deferred to transfer size 0.8. On the other hand, T2 is a brand-new model that has no inherited coefficients. Hence, no overfitting is observed.

A practical outcome of this study is a simple 60% guideline: the strongest R2 and U2 configurations achieve near-optimal performance using approximately 60% of the training data and training time required by the source-floor model, providing a useful deployment guideline.

C. Final Accounting

Does the proposed combined Active Learning and Transfer Learning strategy save training epochs and data? We answer this using the RPs selected via Active Learning specifically for the above Transfer Learning study as an example. In Active Learning, we use $N = 20$, $CV = 0.16$ to perform the selection. The selection rate is 0.624, i.e., 32 positions are sampled to obtain the 20 RPs. Each RP takes 30 scans, and the 12 dropped positions take another 30 scans each, resulting in 960 scans. The DSAR model is trained with 50 epochs. On the target floor, we have 20 RPs, and with R2 we need 360 ($20 \times 0.6 \times 30$) scans and 15 epochs for training. As a result, a total of 40 RPs, 1320 scans, and 65 epochs are needed. The RMSE performance is 2.08 m. To produce a comparable RMSE result with random sampling, we need at minimum $N = 30$ with RMSE 2.512 m according to Table II. For two floors, a total of 60 RPs, 1800 ($30 \times 2 \times 30$) scans, and 100 epochs are needed. Comparing the two, the proposed R2 strategy achieves a 33.3% reduction in RPs, 26.7% in training data, and 35% in training epochs. With the same saving in RPs, using the U2 strategy, which offers better data saving but consumes more epochs, the saving is 33.3% and 25% in training data and epochs, respectively. Although the evaluation is conducted on a single multi-floor building, the framework targets environments with similar layouts commonly found in real-world deployments. Therefore, the observed gains are likely to generalize.

VII. Conclusion

Wi-Fi fingerprinting enables practical indoor localization, yet its reliance on dense and repeated site surveys limits scalability. In this work, we addressed this challenge by proposing a regression-based Active Learning strategy and unifying it with a Denoising Supervised Autoencoder Regressor (DSAR), and projection-based transfer learning into a single, end-to-end framework, and by translating its behavior into actionable deployment guidelines.

Our findings establish two key operating principles for practical use. First, in Active Learning, a moderate uncertainty threshold (CV ≈ 0.16) is observed to provide the best trade-off between localization accuracy and sampling effort, maintaining a manageable selection rate. Second, in Transfer Learning, an empirical $\approx 60\%$ guideline using approximately 60% of the training data and training time of the reference model emerges as a reliable operating point for effective adaptation. Within this setting, the R family offers stronger gains in training efficiency, reducing training epochs by 20–70%, while the U family provides greater data efficiency, reducing training requirements by 37–58%.

The current framework assumes static user positions, consistent with many RSS fingerprinting-based indoor localization studies. User mobility may introduce temporal fluctuations

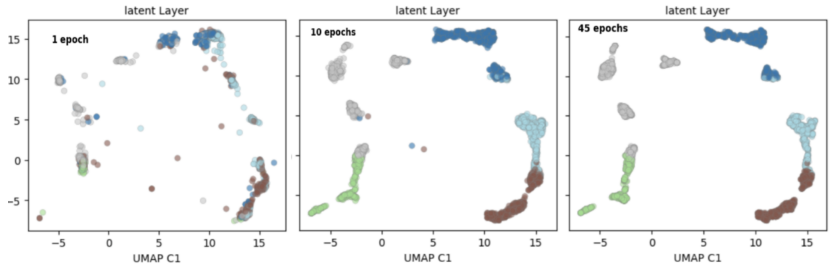


Fig. 9: UMAP projection of the encoder latent space shown at 1, 10, and 45 transfer epochs.

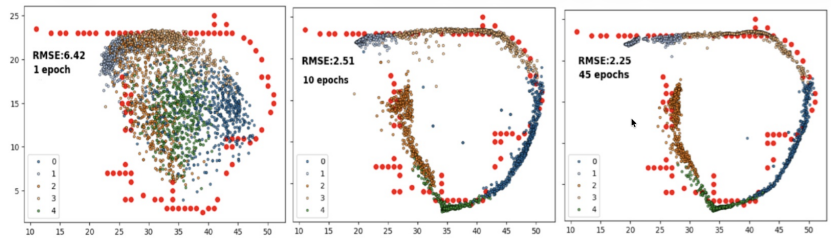


Fig. 10: Localization scatter plots corresponding to Fig. 9 at 1, 10, and 45 transfer epochs.

in RSS measurements, which could affect localization performance. As future work, we plan to extend the framework to support mobile users. Additionally, we will evaluate the framework across a wider range of environments to further assess its generalizability.

Acknowledgment

This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) through the Applied Research and Development (ARD) program, and by TELUS. The authors used ChatGPT for language editing and clarity improvement and have reviewed the content, taking full responsibility for it.

References

[1] N. Gonzalez Diaz, E. Zola, and I. Martín-Escalona, "Assessing the impact of coupling RTT and RSSI measurements in fingerprinting Wi-Fi indoor positioning," in *Proceedings of the Int'l ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM '23)*. Montreal, QC, Canada: ACM, 2023, pp. 19–26.

[2] M. Ahmed, T. E. Salti, J. Chun-Chung Cheung, F. Derakhshan, K. Zheng, and R. Bhatelia, "Low-error indoor positioning via synthetic rssi augmentation and zx–wknn hybrid model," in *2025 International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, 2025, pp. 609–618.

[3] S.-H. Lee and D.-H. Seo, "Scalable wi-fi fingerprinting localization by deep similarity network," *IEEE Internet of Things Journal*, vol. 12, no. 4, pp. 4197–4206, 2025.

[4] C. Sun and M. Hashemi, "Efficient user localization in wireless networks using active deep learning," in *2021 55th Asilomar Conference on Signals, Systems, and Computers*, 2021, pp. 1602–1606.

[5] F. Garbuglia, W. Raes, J. De Bruycker, N. Stevens, D. Deschrijver, and T. Dhaene, "Bayesian active learning for received signal strength-based visible light positioning," *IEEE Photonics Journal*, vol. 14, no. 6, pp. 1–8, 2022.

[6] J. Zhang, J. Katz-Samuels, and R. Nowak, "GALAXY: Graph-based active learning at the extreme," in *Proceedings of the 39th International Conference on Machine Learning (ICML)*, ser. Proceedings of Machine Learning Research, vol. 162. PMLR, 2022, pp. 26 223–26 238.

[7] B. Settles, "Active learning literature survey," University of Wisconsin–Madison, Computer Sciences Technical Report 1648, Jan. 2009, revised January 26, 2010.

[8] A. Kargar-Barzi, E. Farahmand, N. Taheri Chatrudi, A. Mahani, and M. Shafique, "An edge-based wifi fingerprinting indoor localization using convolutional neural network and convolutional auto-encoder," *IEEE Access*, vol. 12, pp. 85 050–85 060, 2024.

[9] K. Kim and J. Lee, "Adaptive scheme of denoising autoencoder for estimating indoor localization based on rssi analytics in ble environment," *Sensors*, vol. 23, no. 12, 2023.

[10] Y. Etiabi, W. Njima, and E. M. Amhoud, "Femloc: Federated meta-learning for adaptive wireless indoor localization tasks in iot networks," *IEEE Internet of Things Journal*, vol. 11, no. 22, pp. 36 991–37 007, 2024.

[11] L. T. Nguyen and J. Zhang, "Wi-fi fingerprinting through active learning using smartphones," in *Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication (UbiComp '13 Adjunct)*. Zurich, Switzerland: ACM, 2013, pp. 969–976.

[12] N. Soltani, J. Zhang, B. Salehi, D. Roy, R. Nowak, and K. Chowdhury, "Learning from the best: Active learning for wireless communications," *IEEE Wireless Communications*, vol. 31, no. 4, pp. 177–183, 2024.

[13] J. Barrows, V. Radu, M. Hill, and F. Ciravegna, "Active learning with data distribution shift detection for updating localization systems," in *2021 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2021, pp. 1–8.

[14] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th International Conference on Machine Learning (ICML)*. ACM, 2008, pp. 1096–1103.

[15] I. Neupane, S. Shahrestani, and C. Ruan, "Wi-Fi RSS fingerprinting-based indoor localization in large multi-floor buildings," *Electronics*, vol. 15, no. 1, p. 183, 2026.

[16] Y.-H. Wang, T.-W. Yang, C.-F. Chou, and I.-C. Chang, "Wi-fi dsar: Wi-fi based indoor localization using denoising supervised autoencoder," in *2021 30th Wireless and Optical Communications Conference (WOCC)*, 2021, pp. 188–192.

[17] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[18] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He, "A comprehensive survey on transfer learning," *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, 2021.

[19] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Advances in Neural Information Processing Systems (NIPS)*, vol. 27, 2014, pp. 3320–3328.

[20] Statistics Canada, "Survey of principals (sop)," Statistics Canada, Ottawa, Ontario, Canada, Survey Documentation, Integrated Meta Data Base (IMDB), Record No. 5065, accessed: 2026-06-09.