Wi-Learner: Towards One-shot Learning for Cross-Domain Wi-Fi based Gesture Recognition

CHAO FENG, Northwest University, Shaanxi International Joint Research Centre for the Battery-Free Internet of Things, China
NAN WANG, Northwest University, Shaanxi International Joint Research Centre for the Battery-Free Internet of Things, China
YICHENG JIANG, Zhejiang University, School of Art and Archaeology, China
XIA ZHENG, Zhejiang University, School of Art and Archaeology, China
KANG LI *, Northwest University, School of Information Science and Technology, China
ZHENG WANG, School of Computing, University of Leeds, United Kingdom
XIAOJIANG CHEN*, Northwest University, Shaanxi International Joint Research Centre for the Battery-Free Internet of Things, China

Contactless RF-based sensing techniques are emerging as a viable means for building gesture recognition systems. While promising, existing RF-based gesture solutions have poor generalization ability when targeting new users, environments or device deployment. They also often require multiple pairs of transceivers and a large number of training samples for each target domain. These limitations either lead to poor cross-domain performance or incur a huge labor cost, hindering their practical adoption. This paper introduces Wi-Learner, a novel RF-based sensing solution that relies on just one pair of transceivers but can deliver accurate cross-domain gesture recognition using just one data sample per gesture for a target user, environment or device setup. Wi-Learner achieves this by first capturing the gesture-induced Doppler frequency shift (DFS) from noisy measurements using carefully designed signal processing schemes. It then employs a convolution neural network-based autoencoder to extract the low-dimensional features to be fed into a downstream model for gesture recognition. Wi-Learner introduces a novel meta-learner to "teach" the neural network to learn effectively from a small set of data points, allowing the base model to quickly adapt to a new domain using just one training sample. By so doing, we reduce the overhead of training data collection and allow a sensing system to adapt to the change of the deployed environment. We evaluate Wi-Learner by applying it to gesture recognition using the Widar 3.0 dataset. Extensive experiments demonstrate Wi-Learner is highly efficient and has a good generalization ability, by delivering an accuracy of 93.2% and 74.2% – 94.9% for in-domain and cross-domain using just one sample per gesture, respectively.

CCS Concepts: • **Human-centered computing → Ubiquitous and mobile computing**.

*Corresponding author

Authors' addresses: Chao Feng, Northwest University, Shaanxi International Joint Research Centre for the Battery-Free Internet of Things, China, chaofeng@stumail.nwu.edu.cn; Nan Wang, Northwest University, Shaanxi International Joint Research Centre for the Battery-Free Internet of Things, China, nan_wang@stumail.nwu.edu.cn; Yicheng Jiang, Zhejiang University, School of Art and Archaeology, China, jiangyicheng@zju.edu.cn; Xia Zheng, Zhejiang University, School of Art and Archaeology, China, zhengxia@zju.edu.cn; Kang Li , Northwest University, School of Information Science and Technology, China, likang@nwu.edu.cn; Zheng Wang, School of Computing, University of Leeds, United Kingdom, z.wang5@leeds.ac.uk; Xiaojiang Chen, Northwest University, Shaanxi International Joint Research Centre for the Battery-Free Internet of Things, China, xjchen.nwu.edu.cn.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2022 Association for Computing Machinery.
2474-9567/2022/9-ART114 $15.00
https://doi.org/10.1145/3550318

Additional Key Words and Phrases: Gesture recognition, Wi-Fi, Deep learning, Domain adaption

ACM Reference Format:
Chao Feng, Nan Wang, Yicheng Jiang, Xia Zheng, Kang Li, Zheng Wang, and Xiaojiang Chen. 2022. Wi-Learner: Towards One-shot Learning for Cross-Domain Wi-Fi based Gesture Recognition. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 3, Article 114 (September 2022), 27 pages. https://doi.org/10.1145/3550318

1 INTRODUCTION

Gesture recognition underpins many human-computer interaction applications. With the capability of gesture recognition, one can interact with today's digital devices in a contact-free manner. Wireless signals like Wi-Fi [26, 41, 47], RFID [43, 62], acoustic [22, 52] and millimeter radar [34, 46] are emerging as a powerful modality for building wireless sensing systems for gesture recognition. Compared with traditional solutions that build around wearable devices [12, 37], and smartphones [30], wireless sensing has the advantage of not requiring instrumenting the users (i.e., device-free), and being less privacy intrusive than other infrastructure-based solutions such as video monitoring [11, 48]. Among these signals, a Wi-Fi-based solution is particularly attractive due to its low cost and the ubiquitous Wi-Fi devices around us.

There is a growing interest in developing Wi-Fi-based gesture recognition systems using machine learning to learn a decision model from training samples [3, 14, 18, 25, 26]. While giving good results in specific environments, the efficacy of existing learning-based approaches depends on a range of *domain factors*, like the multipath of deployed environments, the user characteristics, the user's location and orientation, and the Wi-Fi transceiver setups. Experience and studies show that even a small change in the domain factor [56], such as moving or adding furniture, or changing the position and distance of wireless devices or the location where the activity is performed, can significantly degrade the performance of a Wi-Fi-based gesture recognition solution. Fundamentally, this is because domain factor change can violate a core assumption of machine learning - the training and test time examples are identically and independently drawn from the same distribution (i.i.d.). The change in the domain factors can change multipath, completely changing the wireless signal pattern, causing the incoming test distribution to diverge from the model training samples', leading to poor recognition accuracy.

One way to improve the robustness of a learning-based sensing system is to make sure the model training samples cover all possible target domains. Unfortunately, doing so is infeasible due to the extensive user involvement for data acquisition. Some recent works utilize adversarial learning [4, 38] or transfer learning [24] to improve the generalization ability of the sensing model. These approaches, while important, are unlikely to cover all possible domain factors seen during deployment time. Another possibility is to find features that are robust to the environment by using multiple transceivers [10, 61]. However, it is very hard, if not impossible, to find such universal features across various domains with different device deployments, user characteristics and gestures. These drawbacks call for a new approach for constructing learning-based gesture recognition systems.

We present Wi-Learner, a cross-domain learning-based gesture recognition system. Wi-Learner has the benefit of being low-cost because it relies on just one Wi-Fi transceiver pair. It can adapt to a new domain by using just one gesture sample for each target domain, seen at test time. This *one-shot* learning capability significantly reduces the data acquisition overhead. While being low-cost and low-overhead, we show that Wi-Learner is highly accurate and can quickly adapt to changes of environments, users, location and orientation of the user, and device deployments. The cross-domain learning feature of Wi-Learner allows one to build a robust gesture recognition system with a small set of data samples. As illustrated in Fig. 1, such a capability allows the sensing system to adapt to the change of user identity, environment, as well as the location, orientation, and deployment of the Wi-Fi transceiver.

Realizing our goal requires overcoming several technical challenges. Wi-Learner realizes gesture recognition using the Doppler frequency shift (DFS) information. Prior work has shown that different performed gestures

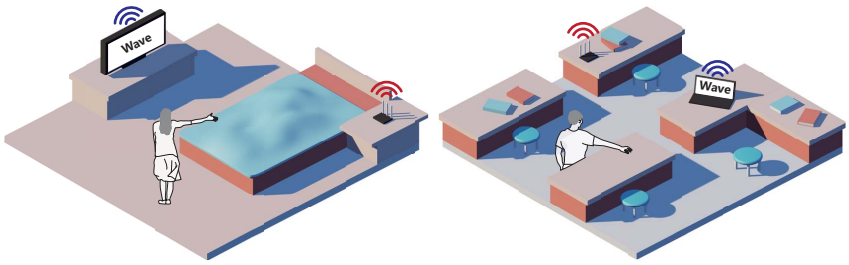


Fig. 1. Example application of Wi-Learner, Where one male and one female perform the same gesture in different environments (e.g., home, lab), and they stand at different locations and orientations relative to different transceivers' deployments.

induce unique DFS change patterns and can be used to build an effective gesture recognition system [51]. The DFS patterns can be observed in the DFS spectrograms extracted from the channel state information (CSI) using commercial off-the-shelf Wi-Fi transceivers, making DFS suitable for Wi-Learner. Unfortunately, the CSI measurements are inherently noise in real-life deployment. Such noise can manifest in the CSI amplitude and phase readings – both are essential for obtaining accurate DFS spectrograms. For example, prior studies show that the CSI phase readings can vary from 0 to 2π across different wireless packets, and the CSI amplitude readings contain much impulse noise [8]. Such a noise level can severely affect the accuracy of gesture recognition. Moreover, indoor environments normally incur a heavy multipath effect. It causes excessive irrelevant interference exists in the extracted DFS spectrograms. This issue is amplified on Wi-Learner because it has to rely on just one pair of Wi-Fi transceivers. Furthermore, as the learning algorithm of Wi-Learner operates on minimal training samples (one sample per gesture for each new domain), it must find ways to make best use of the available data samples to maximize the information gain. Failing to do so will degrade the resulting performance and generalization ability of Wi-Learner, discouraging wide-scale adoption.

We introduce a set of signal processing methods, model architectures and learning strategies to overcome the aforementioned technical challenges. To minimize the noise in the CSI amplitude and phase readings, Wi-Learner first employs a discrete wavelet transform (DWT) algorithm to denoise the CSI amplitude readings. Then, it leverages the observation that the two antennas at the same receiver share identical phase offsets to cancel out random phase offsets in the CSI phase readings. Finally, Wi-Learner applies a series of signal processing schemes including weight-based conjugate multiplication, antenna selection and spectrogram enhancement to obtain the cleaned DFS spectrograms.

To extract DFS change representation from the DFS spectrograms, Wi-Learner employs a lightweight autoencoder scheme. Our autoencoder builds on a convolution neural network architecture that is shown to be effective in learning wireless signal representation [16]. Our key insight is to use an encoder-decoder strategy to guide the network to focus on extracting gesture-induced features while discarding the irrelevant information in the DFS spectrograms. Our model is trained to produce a latent representation to capture gesture-dependent characteristics. To improve the noise-resistance of the autoencoder, we further introduce a Gaussian noise generation module to mimic the signal measurement noise. Later in Section 6, we show that our autoencoder scheme is highly effective in extracting useful representation from the CSI readings, which in turn allows one to build an accurate downstream model for gesture recognition based on the extracted information.

To tackle the domain adaptation challenge with limited target samples, Wi-Learner employs a novel meta-learner framework to "teach" our base sensing model to learn effectively from a few data points. To do so, we first use the training dataset collected from a variety of domains to initialize the model. We then use a small amount of data (one sample per gesture) collected from each new domain to fine-tune the model parameters. To

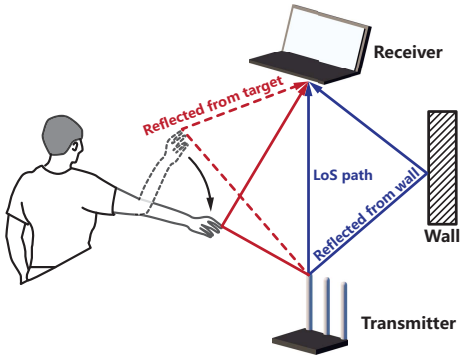


Fig. 2. Multipaths caused by the human target movement

this end, Wi-Learner first designs a base model to extract deep gesture-dependent features from DFS change representation. It then adopts a task generation scheme to generate different learning tasks to mimic domain variations using the limited training dataset, where different tasks contain different domains' gestures. Then, Wi-Learner leverages these tasks to train the base model, guiding the neural network to focus on extracting information that is most likely to improve the learning quality. By doing so, Wi-Learner enables fast domain adaption with one-shot learning while retaining the desired performance with a good generalization ability.

We implement Wi-Learner on a pair of commodity Wi-Fi transceivers and evaluate the system performance on a public dataset, Widar3.0 [61]. Experimental results demonstrate that Wi-Learner is highly efficient and has a good generalization ability, by delivering an accuracy of 93.2% and 74.2% – 94.9% for in-domain and cross-domain using just one sample per gesture, respectively. In addition, we utilize two other Wi-Fi datasets (SignFi [26] and WiAR [13]) with different sensing tasks to demonstrate the generalizability of Wi-Learner.

Contribution. This work makes following technical contributions:

- We present a learning-based solution that relies on one pair of Wi-Fi transceivers and one-shot learning but can deliver accurate and reliable cross-domain gesture recognition (Section 6).
- We introduce a lightweight autoencoder to derive useful Wi-Fi signal representation to support gesture recognition (Sections 4.2).
- We show how a novel meta-learner can be developed to achieve fast domain adaptation with one sample per gesture (Section 4.3).

2 BACKGROUND AND MOTIVATION

In this section, we first introduce the channel statement information (CSI) channel used for RF-based activity recognition. Then, we depict the problem scope of this work. Finally, some preliminary experiments are conducted to showcase the practical cross-domain issues in gesture recognition systems.

2.1 Modeling the CSI Channel

Typically, in an indoor environment, transmitted signals bounce off different objects and these reflected copies are inter-wined at the receiver. As depicted in Fig. 2, the Wi-Fi signals not only travel along a direct path but also multiple reflection paths (e.g., walls and the human target). Thus, the channel frequency response can be

expressed as:

$$H(f,t) = \sum_{i=1}^{N} a_i(f,t)\, e^{-j2\pi f \frac{v_i t}{c}} \tag{1}$$

where f and t are the frequency and arrival time of subcarriers, N is the number of paths, v_i denotes the i^{th} path's length change velocity, and c is the speed of light. Due to the propagation paths can be divided into static paths and dynamic paths [51], we rewrite Equation (1) as:

$$H(f,t) = H_s(f,t) + H_d(f,t) = H_s(f) + \sum_{i \in N_d} a_i(f,t)\, e^{-j2\pi f \frac{v_i t}{c}} \tag{2}$$

where $H_s(f,t)$ represents the sum of all static paths' responses and N_d denotes the number of dynamic paths. Based on Equation (2), we can observe that once a human activity (e.g., performing gestures) occurs, the length of dynamic paths will change, thereby introducing a Doppler frequency shift ($f\frac{v_i t}{c}$) to the i^{th} path signal. Please note that the speed of reflection path length change is not the real velocity of the human target. Since different gestures have different trajectories of hand components, which results in different target reflection path length change patterns of Wi-Fi signals, we can employ the CSI measurements to derive the Doppler change patterns induced by gestures and use them to infer different gestures [51].

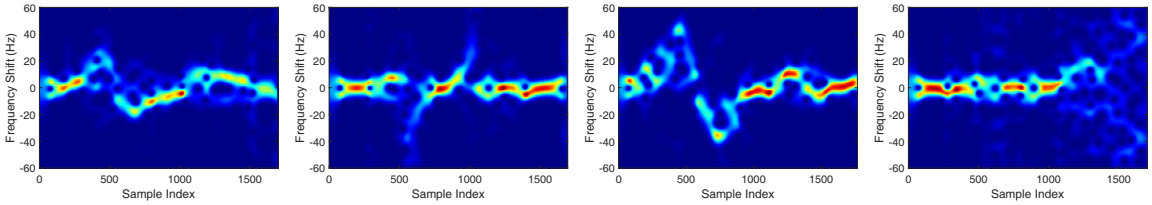
2.2 Problem Scope

This paper aims to deliver accurate cross-domain gesture recognition using just one-shot per gesture in the target domain. We here provide a general description of the problem terms we used as follows: 1) *Domain factors:* We term these factors uncorrelated with gestures as *domain factors*, and this paper mainly focuses on five factors: *environment*, *user*, *location* and *orientation* of the user and *device deployment*. Note that each domain factor could involve different numbers of domains. For example, if the dataset contains 5 different locations, it means that the location factor has 5 location domains. 2) *Cross-domain:* It means that the domains in the training dataset and test dataset are different, and they have no intersection. It is worth noting that there maybe one, two or more domain factors are changed in the training dataset and test dataset. In this work, Wi-Learner explores the one-cross-domain gesture recognition performance where only a single domain factor changes, and also studies multi-cross-domain gesture recognition when multiple domain factors simultaneously change. For example, *cross-location* means that only location domain labels between the training dataset and test dataset are different while the labels of other domain factors are the same. 3) *In-domain:* This term indicates the domains in the training dataset include the domains in the test dataset. 4) *Shot:* It refers to the number of labeled samples per gesture, i.e., *one-shot* means one labeled sample per gesture. 5) *Training dataset:* It refers to the sample set used to train the model. In this work, we assume that the training dataset consists of samples from different domains (e.g., user, orientation, location, device deployment and environment), which can be extracted from different public Wi-Fi datasets, such as Widar 3.0. 6) *Test dataset:* It refers to the sample set used to test the trained model.

2.3 Practical Cross-domain Issues

In this part, we intend to exploit the experimental measurements from the public Widar3.0 dataset [1] to showcase practical cross-domain issues existed in current wireless sensing systems. For simplicity, we here use two different domain factors, environments and transceiver layouts, as an example. Note that when exploring the impact induced by a certain domain, we ensure the other domains are the same. Specifically, we first select a performer's (User 2) two different gesture measurements (e.g., drawing zigzag and sliding). Then, we respectively choose the same performer's "zigzag" gesture measurements from a different room and a different device deployment to

[1]This public dataset can be obtained from Widar3.0 [61]



(a) Performing the "ZigZag" gesture by user 2.
(b) Performing the "sliding" gesture by user 2.
(c) Performing the "ZigZag" gesture by user 2 in a different room.
(d) Performing "ZigZag" by user 2 in a different transceiver layout.

Fig. 3. The DFS spectrograms under different cases.

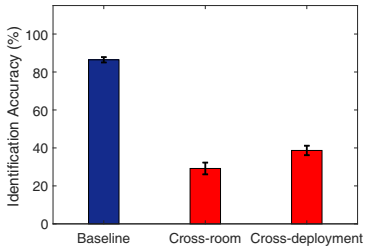


Fig. 4. The gesture recognition accuracy under different cases.

mimic two different domains. Next, we employ the Short-Term Fourier Transform (STFT) to these measurements for obtaining the DFS spectrograms.

The results are illustrated in Fig. 3. From Fig. 3(a) and Fig. 3(b), we can see that different gestures have different DFS change patterns in the DFS spectrograms. This observation indicates that the DFS spectrograms can be used to recognize different gestures, which is in line with the literature [31]. Based on Fig. 3(c) and Fig. 3(d), we can observe that the DFS spectrograms of the same gesture are greatly different between two domains. It implies that the DFS spectrograms carry adverse domain information irrelevant to gestures, which would lead to wrong gesture recognition.

To quantify the gesture recognition performance in different domains, we first use six gestures' measurements in a fixed domain as the baseline, and respectively use the measurements from a different environment and transceivers layout as two test domains. Then, we employ the DSF spectrogram as the input into the deep learning model proposed in the study [26] to recognize gestures. Fig. 4 shows the average accuracy in each domain, we can observe that the average accuracy of the baseline is 86.4%, yet the accuracy respectively drops to 29.2% and 38.6% in the other two domains, which implies that the recognition performance in a domain different from the training domain significantly decreases. Thus, an accurate and robust cross-domain gesture recognition scheme is highly desired.

3 SYSTEM OVERVIEW

Wi-Learner is a low-cost and accurate cross-domain gesture recognition system built on commodity Wi-Fi devices. It only consists of a pair of transceivers. The core idea of Wi-Learner is to make the trained model learn to learn, thereby achieving fast adaption to a new domain with a few samples. The system architecture is depicted in Fig. 5, which involves the following modules:

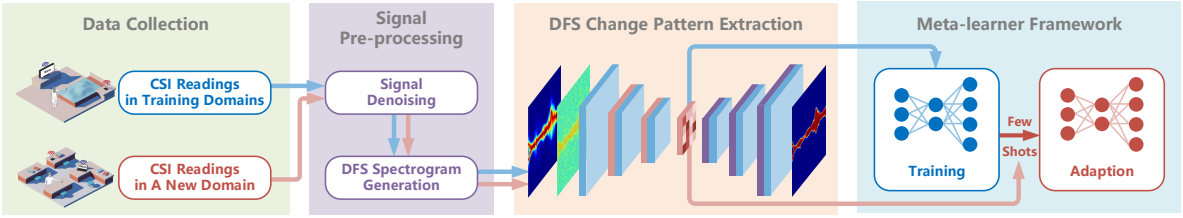


Fig. 5. System overview of Wi-Learner.

Data Collection Module: Wi-Learner first collects raw CSI readings when different users perform gestures in different domains as the training dataset, which can be obtained from the public Wi-Fi dataset (e.g., Widar 3.0). Then, Wi-Learner collects the measurements in a new domain as the target dataset.

Signal Pre-processing Module: Due to the hardware imperfection and environmental noise, the raw CSI readings can not be directly used for feature extraction. To remove environmental noise from raw amplitude readings, Wi-Learner applies a DWT algorithm. Then, Wi-Learner employs an antenna selection scheme to select a suitable antenna pair and performs a weight-based conjugate multiplication between two antennas to remove random phase offsets. After that, Wi-Learner employs a spectrogram enhancement method to obtain a useful DFS spectrogram.

DFS Change Pattern Extraction Module: To efficiently extract distinct features from DFS spectrograms for gesture representation, we adopt a lightweight convolution autoencoder. In addition, we add a gaussian noise generation module into the autoencoder to improve the generalization capability of feature extraction.

Meta-learner Framework Module: To achieve good gesture performance in a new domain and avoid intensive re-collection, we develop a novel meta-learner framework. Specifically, we first design a new base model, and employ a task generation scheme to provide multiple tasks in the training dataset. Then, we employ these tasks to teach the base model to learn a task rapidly. Next, a few samples from target domain are used to adapt the model. Finally, Wi-Learner outputs the predictions of gestures.

4 SYSTEM DESIGN

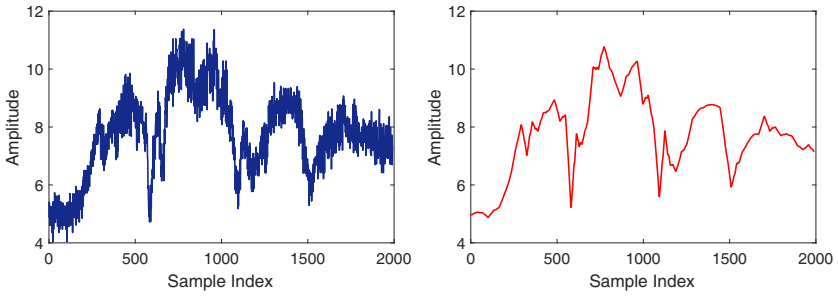
In this section, we elaborate the system design of Wi-Learner to recognize users' gestures through a pair of Wi-Fi devices. The major parts of Wi-Learner are three modules, the signal pre-processing module, DFS change pattern extraction module and meta-learner framework module.

4.1 Signal Pre-processing

In this sub-section, the system performs a set of pre-processing steps to obtain the useful DFS Spectrogram.

4.1.1 Signal Denoising. Due to the environmental noise and hardware imperfection, the collected CSI amplitude readings are corrupted, as shown in Fig. 6(a). To alleviate this impact, we first adopt the DWT algorithm [8] to decompose the raw signal into different level's detail coefficients and approximation coefficients. Then, a detail coefficient threshold is applied to each level for discarding the clutter. Finally, we use all the processed coefficients to reconstruct the signal. The denoised amplitude readings are illustrated in Fig. 6(b). Note that we select the Daubechies 3 wavelet (dB3) to decompose the signal, and reduce the signal to 6 levels.

4.1.2 DFS Spectrogram Extraction. In practice, it is challenging to directly extract accurate Doppler shifts from the raw CSI. This is because the commodity Wi-Fi transmitter and receiver are not synchronized and introduce



(a) Raw amplitude readings. (b) Denoised amplitude readings.

Fig. 6. The amplitude readings with/without signal denosing.

an unknown phase offset in each CSI measurement $\hat{H}$:

$$\hat{H} = \left(H_s\left(f\right) + \sum_{i \in N_d} a_i\left(f,t\right) e^{-j2\pi f \frac{v_i t}{c}} \right) e^{-j(\eta+\beta)} \tag{3}$$

where $(\eta + \beta)$ is the phase offset induced by the carrier frequency offset, sampling frequency offset and packet detection delay [55].

To remove this phase offset, prior work [51] adopts a CSI power scheme. However, the power-based method could remove the information of signs of Doppler shifts, thereby losing the gesture directions information. Instead of using it, we adopt a weight-based conjugate multiplication scheme to overcome this issue. The key intuition is that two antennas at the same receiver have the same phase offset. This scheme has two benefits: 1) eliminating the phase offset while retaining the information of gesture direction; 2) magnifying the amplitude of the dynamic path. Specifically, denote the CSI of the m^{th} antenna as $H_1\left(f,t\right)$, we perform a conjugate multiplication operation between two antennas, we have:

$$\begin{aligned} H_1\left(f,t\right)\bar{H}_2\left(f,t\right) &= \left(H_{s,1}\left(f\right) + \sum_{i \in N_{d,1}} a_i\left(f,t\right) e^{-j2\pi f \frac{v_i t}{c}} \right) \left(\bar{H}_{s,2}\left(f\right) + \sum_{n \in N_{d,2}} a_n\left(f,t\right) e^{j2\pi f \frac{v_n t}{c}} \right) \\ &= H_{s,1}\left(f\right)\bar{H}_{s,2}\left(f\right) + \sum_{i \in N_{d,1}, n \in N_{d,2}} a_i\left(f,t\right) a_n\left(f,t\right) e^{-j2\pi f \frac{(v_i - v_n)t}{c}} \\ &+ H_{s,1}\left(f\right) \sum_{n \in N_{d,2}} a_n\left(f,t\right) e^{j2\pi f \frac{v_n t}{c}} + \bar{H}_{s,2}\left(f\right) \sum_{i \in N_{d,1}} a_i\left(f,t\right) e^{-j2\pi f \frac{v_i t}{c}} \end{aligned} \tag{4}$$

where $\bar{H}$ is the conjugate operation, $N_{d,1}$ and $N_{d,2}$ denote the dynamic paths for the first and second antenna, respectively.

After that, the random phase offset is successfully removed. However, during this process, the Doppler components induced by gesture's movement are corrupted. As presented in Equation (4), we can see that $H_1\left(f,t\right)\bar{H}_2\left(f,t\right)$ consists of three components:

- **Static component.** $H_{s,1}\left(f\right)\bar{H}_{s,2}\left(f\right)$, which is the product of two antennas' static responses. It does not introduce any Doppler shifts. However, due to the power of this component, containing the direct path, is large, it could interfere with Doppler shift estimation. Thus, we can employ a high-pass filter to remove this component.

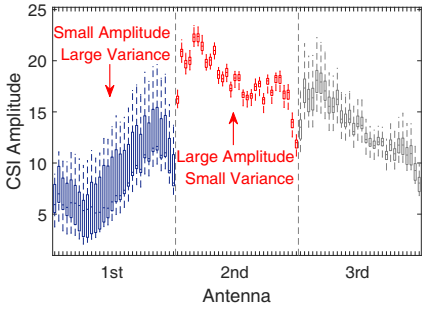


Fig. 7. Antenna selection.

- **Cross component.** $\sum_{i \in N_{d,1}, n \in N_{d,2}} a_i(f,t)\, a_n(f,t) e^{-j2\pi f \frac{(v_i - v_n)t}{c}}$, which is the product of two antennas' dynamic responses. As a result, this component's value is very small. In addition, since it contains the difference of Doppler shifts, we employ a high-pass filter to eliminate it and avoid obfuscating real Doppler shifts.
- **Target component.** The remaining two terms are two products of static responses of one antenna and dynamic responses of another antenna. They both contain the Doppler shifts that we care about. Due to the two antennas are close to each other, they have similar absolute Doppler shift values but with opposite sign. Thus, only one term is the true Doppler shift. Here, the fourth term $\bar{H}_{s,2}(f) \sum_{i \in N_{d,1}} a_i(f,t)\, e^{-j2\pi f \frac{v_i t}{c}}$ is the correct Doppler shift, and we need to remove the third term to prevent it from interfering with Doppler shifts of interest.

As discussed above, to extract Doppler shifts of interest, one dilemma we faced is how to remove the disturbance components, such as the first three terms in Equation (4). To eliminate the impacts of the first two terms, we employ a high-pass filter. As to the third term, we first adopt an antenna selection scheme to determine the proper antenna pair, and then introduce a weight-based method to eliminate interference. The insight of the antenna selection scheme is that selecting one antenna with the largest CSI variance and one antenna with the largest CSI amplitude. The rationale behind it is that the CSI with larger variances is generally sensitive to the target's activity, resulting in larger dynamic responses, and the CSI with higher amplitude usually has strong static paths, resulting in larger static responses. Hence, we design a ratio coefficient ρ_m to characterize the properties :

$$\rho_m = \frac{1}{K} \sum_{k=1}^{K} \frac{var\left(|H_m(f_k,t)|\right)}{mean\left(|H_m(f_k,t)|\right)}, m \in [1,3] \tag{5}$$

where *var* and *mean* denote the variance and mean value of amplitude readings for the m^{th} antenna of the k^{th} subcarrier. With this calculation, we select the antenna pair with the highest and lowest ratio coefficient. As shown in Fig. 7, we can see that the 1^{th} antenna has the largest variances with small amplitudes, while the 2^{th} antenna has the largest amplitudes with small variances. By calculating the ratio coefficient, the 1^{th} and 2^{th} antennas are selected as the first and second antennas in Equation (4).

Through above steps, we can alleviate the impact of the third term with reverse Doppler information. To further eliminate the impact of the third term, a weight-based method is introduced. The basic intuition is to minimize the static responses of $H_{s,1}(f)$ and magnify the static responses of $H_{s,2}(f)$. Thus, we employ two

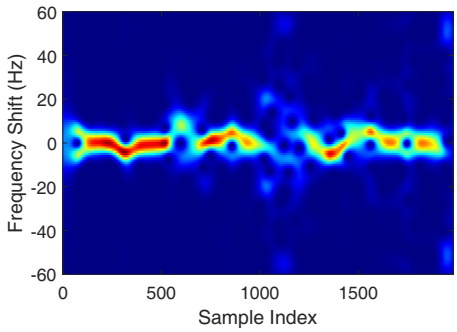


Fig. 8. DFS spectrogram of the gesture "Zigzag" performed by user 1.

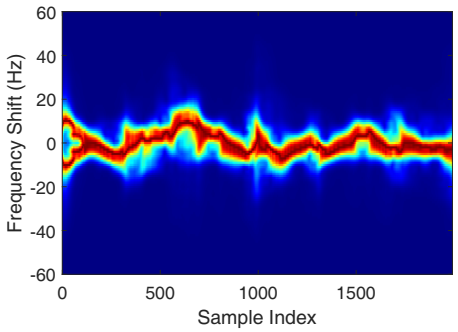


Fig. 9. Enhanced DFS spectrogram of the gesture "Zigzag" performed by user 1.

weights ε_1 and ε_2 into Equation (4), and we have:

$$\hat{H} = \left(\left(\left| H_1\left(f_k, t\right) \right| - \varepsilon_1 \right) e^{j \angle H_1(f_k,t)} \right) \cdot \left(\left(\left| H_2\left(f_k, t\right) \right| + \varepsilon_2 \right) e^{-j \angle H_2(f_k,t)} \right) \tag{6}$$

where $\varepsilon_1 = \min\left(\left| H_1\left(f_k, t\right) \right| \right)$ and $\varepsilon_2 = \kappa \cdot \max\left(\left| H_2\left(f_k, t\right) \right| \right)$, κ is an empirical value of 30. By adding two weights into $H_1\left(f_k, t\right)$ and $H_2\left(f_k, t\right)$, we can effectively reduce the responses of the third term and increase the responses of the fourth term we desired, thereby enabling to obtain accurate Doppler shifts information from $\hat{H}$.

Finally, to extract the DFS spectrogram from $\hat{H}$, We adopt the STFT to obtain Doppler shifts spectrogram. Fig. 8 illustrates the DFS spectrogram of the gesture "ZigZag" performed by user 1, we can see that the Doppler shifts tend to alternate between positive and negative during the entire movement, which can indicate the gesture's trajectory and enable us to accurately recognize different gestures. To obtain a high quality spectrogram, we employ a spectrogram enhancement scheme based on the seam carving algorithm [21]. The enhanced DFS spectrogram is shown in Fig. 9, we can clearly see the fluctuation tendency of Doppler shifts .

4.2 DFS Change Pattern Extraction

Since different gestures cause different DFS change patterns, which are reflected in the DFS spectrograms, our objective is to efficiently derive DFS change patterns from them. However, to achieve it, multiple issues are needed to be tackled. First, performing a gesture can not be assumed as a single point's movement. It means that we can not directly use a threshold-based method to extract the Doppler change curve [50]. Otherwise, some gesture-related information would be lost. Second, the whole DFS spectrogram contains substantial irrelevant interference due to the effect of environment multi-paths, and its data dimension is high. Directly utilizing the spectrogram as a DFS change pattern to recognize gestures would incur performance degradation and high time overhead.

To address this problem, we design a lightweight denoised convolution autoencoder network to efficiently extract DFS change patterns. The basic idea is that the encoder abstracts the information from the input data (DFS spectrogram) to the latent representation (DFS change patterns), and the decoder tries best to reconstruct the input data from the latent representation. The encoder-decoder strategy guides the network to focus on extracting gesture-induced features while discarding the irrelevant information in the DFS spectrogram. Different from traditional autoencoder networks [28], which is used to reconstruct the original image data, we aim to reduce the dimension of the RF data and capture the latent distinguished features. In addition, we introduce a noise generation module to improve the generalization capability of feature extraction. This network structure

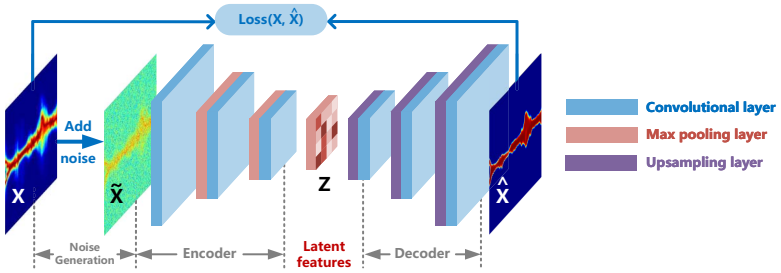


Fig. 10. The lightweight denoised convolution autoencoder network.

is illustrated in Fig. 10, which consists of three modules: a noise generation module, an encoder module and a decoder module.

Noise generation module: Although a DWT denoise algorithm is used to alleviate the noise in the DFS spectrogram, there still have a certain of environmental noise and device noise induced by device diversity. To mitigate this impact, we design a noise generation module to emulate these noises as an additive gaussian noise [33, 35], which aims to train an autoencoder model is robust to derive noise-free features. Now, consider the input data is X, we can obtain a distorted output $\tilde{X}$ as follows:

$$\tilde{X} \sim X + N\left(0, s^2\right)$$

where X denotes the DSF spectrogram, and s is the standard deviation of the guassian noise, which is a pre-selected hyperparameter.

Encoder module: Based on the distorted $\tilde{X}$, we then feed it into the encoder module. The goal of encoder module is to abstract the $\tilde{X}$ into a latent space Z, and the latent space Z is employed to characterize the distinct features of the input data. Thus, we have:

$$Z = Encoder\left(\tilde{X}, w_{\tilde{X}}\right)$$

where the *Encoder* consists of three convolution layers and three max-pooling layers, $w_{\tilde{X}}$ is the parameters to be learned.

Decoder module: After obtaining Z, the decoder module aims to reconstruct the input data X from the latent space Z. Hence, we put the Z into the decoder as follows:

$$\hat{X} = Decoder\left(Z, w_Z\right)$$

where $\hat{X}$ is the reconstructed data, the *Decoder* includes three up-polling layers and three convolution layers, w_Z is the parameters.

To ensure the latent space Z is able to represent the distinct features of the input data X and robust to the noises, the difference between the original input data X and reconstructed data $\hat{X}$ should be minimized. Therefore, we design a loss function as follows:

$$Loss\left(X, \hat{X}\right) = \frac{1}{b_n} \sum_{i=1}^{b_n} \left(X_i - \hat{X}_i\right)^2$$

where b_n is the size of the batch. We optimize this loss function on the training set by iteratively using the backward propagation to minimize the loss value. Note that this process is only trained one time. Once the model is trained well, we can only use the trained encoder to derive the latent features Z for the test data.

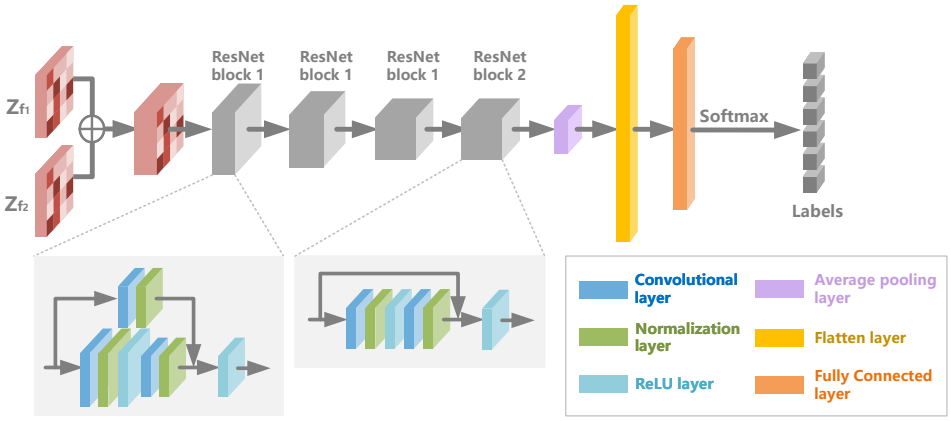


Fig. 11. Structure of the base network.

Based on the proposed autoencoder scheme, we can feed the DFS spectrogram into the encoder and derive the latent features to represent the DFS change patterns. With this model, we achieve the following benefits: 1) we can automatically capture gesture related information without human involvement (e.g., manually design threshold to obtain features) ; 2) The latent features significantly reduce the data dimension of DFS spectrogram while keeping gesture related features, which can reduce the time overhead and enable an accurate gesture recognition.

4.3 Meta-learner Framework

Upon acquiring the DFS change patterns, Wi-Learner needs to use them to identify different gestures. A straightforward method is to use a simple convolutional neural network. This method, however, only works well in a fixed domain. In other words, when the target domain is different from the trained domain, the accuracy is significantly decreased. This is because the extracted features contain substantial domain information (e.g., environment, location and device deployment) irrelevant to gestures. Thus, to tackle this issue, we introduce a new meta-learner framework. The basic intuition is to teach the model to learn how to learn, and enable Wi-Learner to have a fast domain adaption capability in a new domain by only using a few samples per class. Different from the traditional meta-learning framework [9] used for image recognition, our novel meta-learner framework carefully designs a base network and a task generation scheme for wireless signals. The new meta-learner framework mainly consists of four parts: base network, task generation, parameter updating and domain adaption. Next, we will present the details of each part, respectively.

4.3.1 Base Network. The goal of the base network is to learn feature representations and recognize different gestures from the obtained DFS change patterns. However, one practical problem we faced is how to select appropriate CSI subcarriers' DFS change patterns as the input. Since the CSI usually has multiple subcarriers (i.e., 30), and some of them have strong correlations. In other words, they contain redundant information. If we directly use DFS change patterns from all subcarriers, the computational complexity of the base network is significantly increased. Thus, we employ the principal component analysis (PCA) algorithm to select the top two principal components (i.e., f_1, f_2). The main reason behind empirically choosing the first two principal components is to achieve a good trade-off between classification performance and computational complexity.

Once determining the indexes of subcarriers, we employ the corresponding DFS change patterns Z_{f_1} and Z_{f_2} as the input of the base network. Fig. 11 shows the structure of the base network, which consists of a feature

extractor module and a gesture recognizer module. Specifically, to fully combine the information from different subcarriers, we first perform a concatenation operation as follows:

$$Z_{concat} = Z_{f_1} \oplus Z_{f_2}$$

where $\oplus$ denotes concatenation operation.

Next, to learn the high-level features of gesture's motion and reduce gesture-irrelevant information, we design a residual DenseNet including four residual blocks. Each block adopts a residual architecture, as shown in Fig.11. The reason for choosing residual architecture is that it contains two paths (one path is the input travels multiple convolution layers and another path is that the input freely passes through which is referred as to shortcut connection [15]) that can guide the network to learn the difference between them, thereby removing some gesture-irrelevant information. Meanwhile, this architecture can prevent overfitting and speed up convergence. In addition, adopting multiple residual blocks is to derive deep-level features concealed in the input. By using the residual DenseNet, we have:

$$G = Dense(Z_{concat}, w_G)$$

where G is the learned features, $Dense(\cdot)$ denotes the residual DenseNet, and w_G is the learned parameters. The learned feature G is fed into the gesture recognizer module to predict the labels of performed gestures. It first goes through a fully connected layer to map the feature representation G into a latent space. We then employ a softmax layer to calculate the probability $\hat{y}$. Finally, a gesture's label is predicted.

4.3.2 Task Generation Scheme. Although the aforementioned base network can effectively extract feature representations, the features are still domain-dependent, which causes the base model to not work well in a new domain. To handle this problem, our key insight is to generate a large number of tasks to mimic different domain variations, teaching the base network how to adapt to a new domain with a few samples per class. However, it is non-trivial to generate multiple diverse and realistic domain variations given the limited training dataset. To deal with this issue, instead of randomly generating different tasks [9] based on the training dataset, we introduce a novel task generation scheme to generate substantial domain variations, which includes two stages.

In the first stage, we perform a data augmentation operation for generating two new kinds of synthesized domain factors on the training dataset [2]. Specifically, this augmentation operation consists of two signal transformations of the original data. The first transformation is to add different levels of noise to the signals without changing the length of them, which aims to mimic the signals are polluted by environment noise and hardware noise, we term it as noise domain factor. The second transformation is to stretch or compress the signals for simulating different velocities of the performed gestures, we term it as velocity domain factor. Note that the gesture label of each transformation is the same as the original signal. Through this step, the number of domains in the training dataset is augmented. Notice that the goal of this augmentation operation is only to make our model robust to noise and velocity factors.

In the second stage, since the domain variations based on the augmented training dataset are still limited compared to the practical domain variations, we thus perform a task generation on the augmented training dataset to generate substantial single-domain tasks (one domain per task) and multi-domain tasks (multiple domains per task) for mimicking multiple diverse and realistic domain variations. To be specific, consider the training dataset involves three domain factors (e.g., users (U), locations (L) and orientations (O)), each factor contains R domains. So, there have R^3 domains, ranging from "$U_1L_1O_1$" to "$U_RL_RO_R$". Note that there is no intersection of data for each domain. For single-domain tasks, we generate each task by sampling each gesture's data from the same domain. For multi-domain tasks, at least two gestures' data in a task is sampled from two different domains. That is to say, the gestures' data in a multi-domain task can be sampled from two or three different domains.

[2]We assume that the training dataset consists of the samples from different domains (e.g., user, orientation, location, device deployment and environment), which can be extracted from public Wi-Fi datasets, such as Widar 3.0.

ALGORITHM 1: Wi-Learner's Base Network Training

Input: Training dataset $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, step size ζ, meta step size ξ
Input: number of examples used for inner gradient update k
Output: Trained parameters θ
Randomly initialize θ
while *not done* **do**
  $T \leftarrow$GenerateTask(S)
  for $T_i \in T$ **do**
    $S_{T_i} \leftarrow k$ support samples from T_i
    $Q_{T_i} \leftarrow n-k$ query samples from T_i where $S_{T_i} \cap Q_{T_i} = \emptyset$
    Evaluate $\nabla_\theta Loss_{T_i}(\theta)$ with S_{T_i} via $Loss_{T_i}(\theta) = \sum_{j=1}^{m_{S_{T_i}}} y_j \log\left(\hat{y}_j\right) + \left(1-y_j\right)\log\left(1-\hat{y}_j\right) + \kappa o$
    Compute adapted parameters with gradient descent: $\theta_{T_i} \leftarrow \theta - \zeta\nabla_\theta Loss_{T_i} f(\theta)$
    Use $\theta_i{}'$ for one iteration on Q_{T_i}, compute the loss $Loss_{T_i}(\theta_{T_i})$
  end
  Update $\theta \leftarrow \theta - \xi\nabla_\theta \sum_{T_i \in S} Loss_{T_i}\left(\theta - \zeta\nabla_\theta Loss_{T_i}\left(\theta_{T_i}\right)\right)$
end

Through the above task generation scheme, we can obtain multiple diverse and realistic domain variations with the limited training dataset, supporting Wi-Learner can teach the model to effectively learn from a few samples, thereby delivering a good cross-domain result, even crossing multiple domains simultaneously.

4.3.3 Parameter Updating. After generating multiple tasks from the training dataset, our next goal is to employ them for training the parameters of the based network. Toward this end, Wi-Learner adopts an optimization-based training algorithm to perform parameter updating. The basic idea is to seek the effective initial model parameters θ, enabling the model to rapidly adapt to a new domain with a few samples. Specifically, we first select training samples from each task T_i, then divide them into a support set S_{T_i} and a query set Q_{T_i} without overlapping samples, both of them only contain N samples per class. It is worth noting that the value of N is usually small, such as one or two samples per class. The reason is to simulate a few samples we can obtain from a new domain.

Next, we use the support set to train the task-specific parameters θ_{T_i} of the base network. This aims to mimic the adaption process of the base model to learn the knowledge from a new domain with a few samples. Afterward, the query set is used to evaluate the task performance and iteratively update the initial parameters θ. Specifically, we use a cross-entropy loss to calculate the loss of each task as follows:

$$Loss_{T_i}(\theta) = \sum_{j=1}^{m_{S_{T_i}}} y_j \log\left(\hat{y}_j\right) + \left(1-y_j\right)\log\left(1-\hat{y}_j\right) + \kappa o$$

where $m_{S_{T_i}}$ denotes the number of samples for the task S_{T_i}, κ is the hyper-parameter, and o is a regularizer term for avoiding overfitting. Then, we perform a gradient adjustment operation with one step to the initial model parameters θ, we have:

$$\theta_{T_i} = \theta - \zeta\nabla_\theta Loss_{T_i}(\theta)$$

where ζ represents the step size, it is a hyper-parameter to control the model learning rate. Generally, with a larger value of ζ, the rate of the adaption process can be faster.

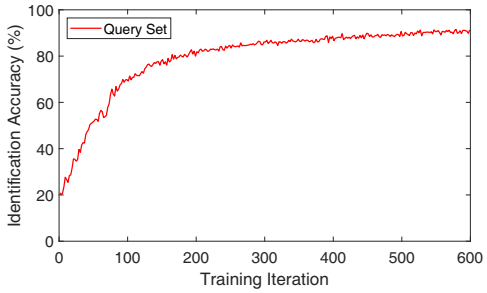


Fig. 12. Learning curve of the query set in the training phase.

To find the parameters θ that can minimize the sum loss of all tasks with a small number of gradient steps, we define a meta-objective function as follows:

$$\min_{\theta} \sum_{T_i \in S} Loss_{T_i} \left(\theta - \zeta \nabla_{\theta} Loss_{T_i} (\theta)\right)$$

Then, we employ stochastic gradient descent (SGD) to optimize the meta-objective function, the parameters θ are updated as follows:

$$\theta \leftarrow \theta - \xi \nabla_{\theta} \sum_{T_i \in S} Loss_{T_i} \left(\theta - \zeta \nabla_{\theta} Loss_{T_i} (\theta)\right)$$

where the meta step size ξ is a hyper-parameter. The detailed algorithm is presented in Algorithm 1.

Through this optimization process, the parameters θ continue to learn knowledge from diverse tasks, and eventually are sensitive to different tasks. To demonstrate the parameters θ can learn knowledge, we plot the learning curves of the support set and the query set in the training phase. As shown in Fig 12, we can see that as the number of iterations increases, the accuracy of the query set continually rises. This result confirms Wi-Learner can teach the model to learn how to learn with only a few samples in the target domain. The rationale behind it is that the support set is new data for the model, which aims to update the parameter θ to let the model learn the features and achieve a better performance in the query set. Finally, the optimized parameters θ can be used as effective initial values of the base network, and enable the based network to have a fast adaption capability to a new domain. Note that with more different tasks, the model has better generalization performance.

4.3.4 New Domain Adaptation. To achieve accurate gesture recognition in a new domain, we need to perform domain adaption by using a few samples per class in a new domain to calibrate the parameters of the trained dataset. This is because when the domain changes, it causes the incoming test distribution to diverge from the model training samples, thereby leading to poor recognition accuracy. Specifically, we first use the optimized parameters θ as the initial values to the base model. Then, the few samples in a new domain are employed to update the parameters θ. As an example, the updated parameters in the i^{th} gradient descent step are as follows:

$$\theta_i \leftarrow \theta_{i-1} - \zeta \nabla_{\theta} Loss_D (\theta_{i-1})$$

where D denotes the samples collected in the new domain. After updating the parameters with a few gradient steps, Wi-Learner can successfully perform gesture recognition in a new domain. Note that the number of training epochs is very small, which enables real-time gesture recognition. (Detailed results in Section 6).

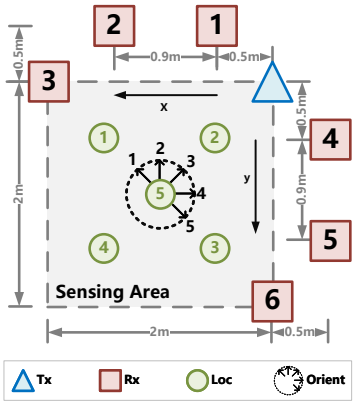


Fig. 13. A typical setup of devices and domains in the sensing area, figure modified from [61].

5 IMPLEMENTATION

We implement Wi-Learner[3] and evaluate the performance on a public dataset from Widar 3.0 [61]. The dataset consists of one transmitter and six receivers, each of them equips an Intel 5300 wireless NIC. The receivers all possess three antennas which are placed in a line. Then, the working frequency is 5.825 GHz, and the sample rate is 1000 packets per second. The gesture dataset of Widar 3.0 is collected from *16 users*, *5 locations*, *5 orientations*, and *6 receivers* in three different rooms. The deployment in the sensing area is illustrated in Fig. 13. Due to the non-uniform distribution of gestures and users across different domains, we mainly select 6 gestures and 5 users for overall performance distribution. The 6 gestures are sliding, drawing O, drawing zigzag, drawing N, drawing triangle and drawing rectangle, respectively. We refer to the selected data as dataset 1. Note that, we select the second receiver and the transmitter as one pair of Wi-Fi transceivers by default. Other receivers (1, 3, 4, 5, 6) are used to test whether our proposed solution can work well in different deployments. Besides, we also use two other Wi-Fi public datasets (e.g., SignFi [26] and WiAR [13]) with different sensing tasks to evaluate the generalizability of our system.

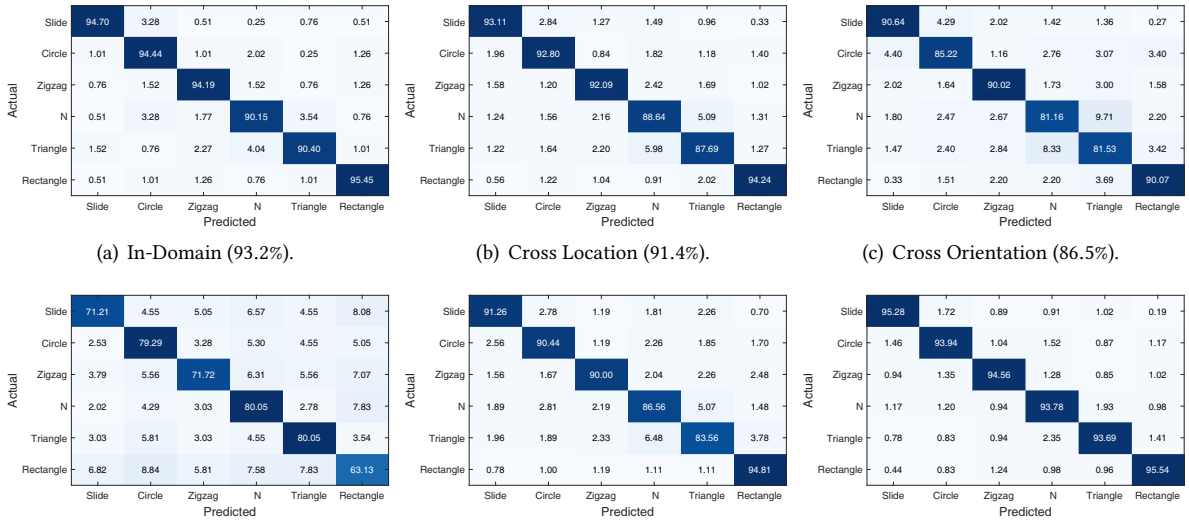
6 PERFORMANCE

6.1 Overall Performance

6.1.1 In-domain Gesture Recognition Accuracy. To evaluate the performance of in-domain gesture recognition, we take all domain factors into consideration. Specifically, we select 90 percentage samples of each domain for training and the remaining samples for testing on dataset 1. Fig. 14(a) shows the confusion matrix of 6 gestures, we clearly see that Wi-Learner achieves an average accuracy of 93.2%. Besides, Wi-Learner can achieve consistently high accuracy of over 90.1% for all gestures. These results indicate that Wi-Learner can effectively extract gesture-related features.

6.1.2 Cross-domain Gesture Recognition Accuracy. We now evaluate the cross-domain gesture recognition performance on dataset 1. Specifically, when evaluating each specific domain factor, we ensure the other domain factors are the same, and calculate the average accuracy of cases where one out of all domains is selected for testing and the rest of domain samples are for training. Note that Wi-Learner only employs one-shot per gesture in each target domain to adapt the trained network. Fig. 14(b), 14(c), 14(d), 14(e) and 14(f) illustrate the gesture

[3]Code is available at: https://github.com/Coolnerdn/WiLearner.



(a) In-Domain (93.2%). (b) Cross Location (91.4%). (c) Cross Orientation (86.5%).

(d) Cross Environment (74.2%). (e) Cross Person (89.4%). (f) Cross Deployment (94.4%).

Fig. 14. Confusion matrices of different settings when only using one-shot per gesture in the target domain.

confusion matrices under each specific domain factor including location, orientation, environment, person and transceiver's deployment, respectively. We can observe that Wi-Learner delivers an average accuracy ranging from 74.2% to 94.4% when crossing different domains using one-shot per gesture. It is worth noting that although the cross-environment average accuracy is only 74.2%, the accuracy has a significant improvement of 50.9% compared to the performance without performing domain adaption (23.3%). In addition, the accuracy can be improved by using more shots for domain adaption (Detailed results in Section 6.2.1). Overall, these results indicate that Wi-Learner can achieve robust and accurate cross-domain gesture recognition in a new domain.

6.1.3 Verification of the Meta-learner Framework. To verify the effectiveness of the meta-learner framework, we compare Wi-Learner to three baselines on dataset 1: 1) **Src:** only using the training dataset to train the base network, and there is no adaptation to the target domain; 2) **Tgt:** only employing one-shot per gesture from the target domain to train the model; 3) **Src+ Tgt:** using both the training dataset and the target domain's one-shot per gesture for training the deep learning model. Note that Wi-Learner uses the training dataset to train the base network and one-shot per gesture from the target domain to adapt the model. The results are shown in Fig. 15. We can see that Wi-Learner outperforms the other baselines when crossing different single domain factors.[4] This is because that the proposed meta-learner framework can teach Wi-Learner to learn how to learn from the training dataset, making Wi-Learner possess a fast adaption capability in a new domain. Thus, these results demonstrate the effectiveness of proposed meta-learner framework.

6.1.4 Verification of the Autoencoder Scheme. To assess the performance of the proposed denoised autoencoder (DAE) scheme for cross-domain gesture recognition, we run two benchmark experiments on dataset 1 with and without the scheme. Specifically, we respectively feed the generated DFS spectrograms and the DFS change

[4]Note that when evaluating each domain factor, we calculate the average accuracy of cases where one out of all domain instances are used for testing, while the rest domain instances are for training. For example, when considering the location factor, the accuracies over five locations are averaged and employed as the final cross-location accuracy.

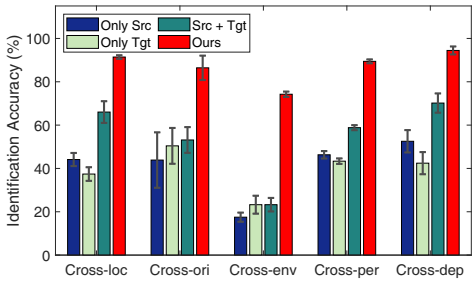


Fig. 15. The performance of the baselines and Wi-Learner.

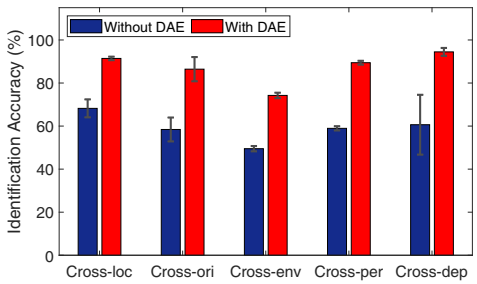


Fig. 16. The performance with/without the proposed DAE scheme.

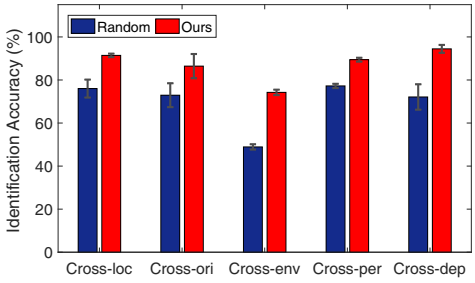


Fig. 17. The performance with task generation scheme for each domain factor.

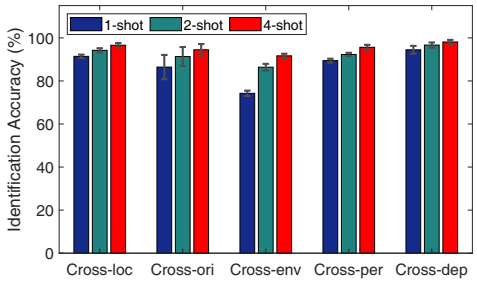


Fig. 18. The performance with varying numbers of shot for each domain factor.

pattern derived from the DAE scheme into the meta-learner framework in Section 4.3. The results are presented in Fig. 16. It is clear that the cross-domain performance with our proposed scheme is consistently better than directly using DFS spectrograms, which demonstrates the proposed DAE scheme is highly effective in extracting useful representation from DFS spectrograms and discarding the irrelevant information.

6.1.5 Verification of the Task Generation. To evaluate the effectiveness of task generation scheme on cross-domain gesture identification, we employ a random task generation method from the training dataset as a baseline, which is widely used in the meta-learning technology [17]. Specifically, we conduct a set of benchmark experiments by evaluating each specific domain factor on dataset 1 and use one-shot per gesture in each target domain to adapt the trained network. Fig. 17 plots the average cross-domain gesture recognition performance. The results depict our proposed scheme works better than the random-based scheme. For example, when crossing different domains, Wi-Learner at least has an increase of accuracy by 12.2% and up to 25.3% improvement. This implies that our scheme can generate multiple diverse and realistic tasks to mimic different domain variations, and help the base network learn how to adapt better to a new domain. In conclusion, the task generation scheme can effectively improve cross-domain performance.

6.2 Diverse Factors on Wi-Learner

In this subsection, we investigate the factors affecting the system performances.

6.2.1 Impact of Number of Shots. In this experiment, we change the number of shots collected in the target domain for adapting the trained model to evaluate the impact on cross-domain accuracy. Specifically, we vary

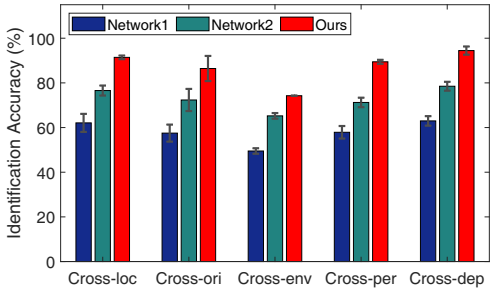


Fig. 19. The performance with varying different base networks.

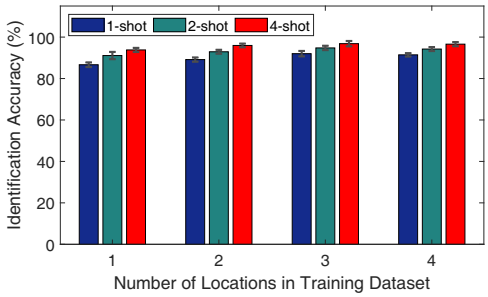


Fig. 20. The impact of training dataset diversity.

the shot number with 1, 2 and 4, which are frequently used in few-shot learning evaluations. For each case, we respectively evaluate each specific domain factor while keeping the other domain factors are the same. Fig. 18 shows the system performance of different shot numbers. We can see that the recognition accuracy increases for each domain factor as the number of shots increases. In particular, the average accuracy for crossing different environments increases from 74.3 to 91.7% when the number of shots varies from 1 to 4. This is because Wi-Learner can learn more target domain related information with more shots involved. Hence, Wi-Learner is able to achieve better cross-domain gesture recognition by using more shots in the adaption phase.

6.2.2 Impact of Different Base Networks. We now evaluate the system performance when the base network is changed. Specifically, we select two different architectures of the base network. One is the convolutional neural network (CNN) architecture, which is modified from SignFi [26]. The other is selecting the residual architecture, which merely contains one residual block. For each case, we respectively evaluate each specific domain factor while keeping the other domain factors are the same. As Fig. 19 shows, our proposed base network outperforms the other two architectures when using one-shot per gesture in the target domain. And we can observe that Wi-Learner increases the accuracy by around 24.7% – 31.5% compared with the CNN architecture. We believe this is because our base model is able to derive deep-level features concealed in the input.

6.2.3 Impact of Training Dataset Diversity. In this experiment, we explore how the number of locations in the training dataset impacts cross-location performance. Specifically, we vary the number of locations from 1 to 4 in the training dataset, and the collections from another location are used to the target dataset. Note that we keep other domain factors are the same in the training dataset and target dataset. The results are presented in Fig. 20. We can see that the average accuracy increases from 86.7% to 91.4% with one-shot as the number of locations varies from 1 to 4, and the performance is always above 86.7%. This is due to with more location domains in the training dataset, Wi-Learner can learn more knowledge from them to teach the trained model how to adapt to a new domain. Even the number of location domains in the training dataset is limited, Wi-Learner still can learn knowledge from other domain factors (e.g., orientation, environment, person and deployments) and deliver a good performance.

6.2.4 Impact of Crossing Multiple Target Domains. In practical scenarios, multiple domain factors may change simultaneously, which is very significant for the real-world adoption of Wi-Learner. Thus, we evaluate the system performance by changing the number of crossing domain factors. Since there are too many combinations of the five factors (i.e., location (L), orientation (O), environment (E), user (U) and transceiver deployment (D)), we select four combinations of these domain factors: L/O, L/O/U, L/O/U/D, L/O/E/U/D. Note that when testing the above combinations, we keep the other domains unchanged. In other word, the difference of domain factors

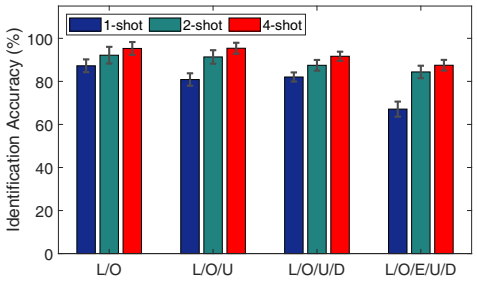


Fig. 21. The impact of crossing multiple factors.

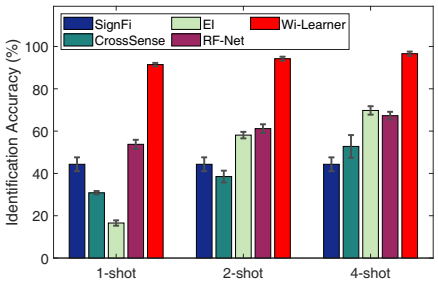


Fig. 22. Comparison of recognition models.

between the training data and the target data is only above mentioned domain factors. Fig. 21 illustrates the results, we can see that the identification accuracy decreases with more domain factors changes simultaneously. For example, when the number of domain factors varies from 2 to 5, the average accuracies with one-shot are 87.2%, 80.9%, 82.1% and 67.2%, respectively. These results are expected because with more domain factors change, Wi-Learner is harder to extract gesture-related features. In addition, we observe that the accuracies with 4-shot are increased to 95.3%, 95.1%, 91.7% and 87.5% as the number of domain factors varies from 2 to 5. Thus, we can employ more shots to improve the performance of crossing multiple target domains simultaneously.

6.2.5 Comparison with Existing Recognition Models. We compared our method Wi-Learner with four state-of-the-art gesture recognition methods, including the traditional CNN architecture (SignFi [26]), the transfer learning architecture (CrossSense [57]), the adversarial learning architecture (EI [16]), and the meta-learning architecture (RFNet [5]). For simplicity, we here only evaluate the cross-location performance on dataset 1 (Widar 3.0 dataset). Specifically, we calculate the average accuracy of cases where one out of all location instances are used for testing, while the rest of location instances are for training. Fig. 22 plots the results of five different methods by varying different numbers of shots for adaption. we can clearly see that Wi-Learner delivers better performance than the other four methodologies. In addition, we observe an interesting thing is that when only employing a few samples (e.g., 1-shot) from the target domain to adapt the model, CrossSence and EI have worse performance than the CNN method (e.g., SignFi), which does not have the cross-domain capability. This result indicates prior transfer learning methods and adversarial learning methods can not work well in a new domain only with a few target samples available. Instead, Wi-Learner can achieve an accurate cross-domain gesture recognition accuracy only using one-shot in the target domain.

6.2.6 Impact of Adaptation Overhead. In this experiment, we compare Wi-Learner with the transfer learning architecture and adversarial learning architecture to evaluate the adaption overhead in a new target domain, which is important to ensure real-time gesture recognition. Specifically, we here only consider cross-location performance on dataset 1, and we use four shots per gesture in the target domain to adapt each trained model. Fig. 23 shows the accuracy training curves for the target adaption process. As we can see, Wi-Learner significantly reduces the adaptation overhead compared to the two methods while delivering the highest accuracy. For example, Wi-Learner only requires 7 epochs to converge, the other two methods however need 20 epochs and 75 epochs to converge, respectively. Thus, these results confirm that Wi-Learner has the fastest convergence and retains a higher identification accuracy, which ensures a real-time application.

6.2.7 Generalizability for Other Datasets. We now assess the generalizability of Wi-Learner to the other Wi-Fi-based datasets for different sensing applications. Specifically, we select two public datasets: 1) **SignFi [26]:** it aims to recognize sign language gestures using Wi-Fi. We select a dataset that contains 7500 instances of 150

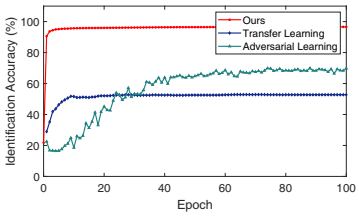
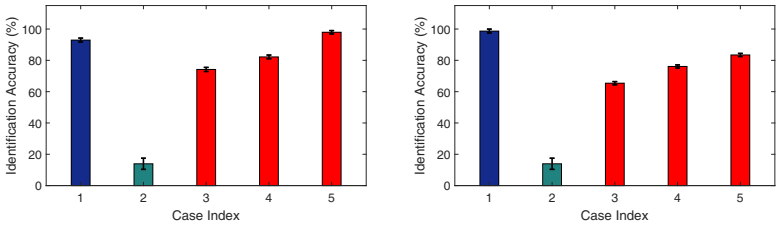


Fig. 23. The impact of adaptation overhead.



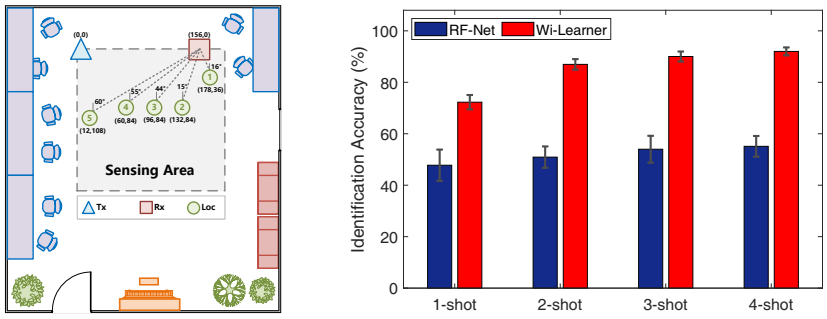
(a) SignFi dataset. (b) WiAR dataset.

Fig. 24. The performance on two public Wi-Fi sensing datasets.

sign gestures performed by 5 users in a lab environment and 1500 instances of 150 sign gestures performed by one of the 5 users in a home environment. We refer to these measurements as dataset 2. 2) **WiAR [13]:** it aims to identify different activities with Wi-Fi devices. This dataset includes 16 different activities performed by 10 users, and each activity is performed by 30 times. We refer to these measurements as dataset 3. Based on the two datasets, we respectively explore the cross-environment and cross-user issues on dataset 2 and dataset 3. For dataset 2, we select one user in a home environment as the target domain and the lab environment as the training domain. For dataset 3, we select one user as the target domain and the remaining users as the training domain. Then, we evaluate the performance with the following cases: 1) **Case 1:** Employing 80% measurements of the training dataset and target dataset for training and 20% for testing; 2) **Case 2:** only using the training dataset for training and using the target dataset for testing; 3) **Case 3, 4, 5:** using the training dataset for training and different numbers of shots per gesture (e.g., 1, 2, 4) from target dataset to adapt the trained model, respectively. Finally, the remaining target dataset is used for testing. Note that case 1 and 2 employ the model modified from SignFi, case 3, 4 and 5 apply our proposed deep learning model.

Fig. 24 plots the recognition performance in the two new datasets. We see that the accuracy drops sharply for case 2 compared with case 1. This is due to the received signals not only carry gesture information but also involve substantial domain information. When applying Wi-Learner's framework, we can observe there have 60.3% and 52.9% accuracy improvement with one-shot on the dataset 2 and dataset 3 compared with case 2, respectively. Besides, with more shots are used for adapting, the performance is better. Thus, Wi-Learner can work well for other datasets with different sensing tasks, which implies the good generalizability of Wi-Learner.

6.2.8 Generalizability for Real-world Scenarios. To evaluate how Wi-Learner can address the domain-dependent issue in real-world scenarios, we collected a large number of new gesture data in our lab environment. Note that there is a significant domain shift between the new dataset and Widar 3.0 dataset, as the two datasets collect CSI measurements under entirely different domain factors, such as the user, location, orientation, environment, device, and deployment. Specifically, we recruited five volunteers to perform five gestures, including "slide", "draw a circle", "draw zigzag", "push and pull", and "sweep". For each gesture, we ask each volunteer to perform 30 times at five locations, respectively. The detailed experimental setup is shown in Fig. 25(a). After collecting new gesture data, we employ the public dataset, WiDar 3.0, to train our model. Then, we use different shots per gesture for each user and location in our new dataset to adapt the trained model, and use the rest of the new dataset to evaluate the real-world performance. Fig. 25(b) illustrates the results of Wi-Learner and RF-Net. We can see that our method Wi-Learner outperforms RF-Net and can achieve a good cross-domain performance. In addition, as the number of shot increases, the accuracy of Wi-Learner continuously increases. For example, Wi-Learner respectively achieves an average accuracy of 72.01% and 92.23% when 1-shot and 4-shot are involved. The above results demonstrate that Wi-Learner can leverage a few samples per class in a new domain to effectively adapt



(a) Real-world Setup. (b) Result.

Fig. 25. The performance on the real-world scenario.

the model trained by the public dataset for achieving cross-domain gesture recognition, even if all domain factors change.

7 DISCUSSION

Naturally, there is room for future work and further improvement. We discuss a few points here.

Multi-user scenarios. Multi-user sensing is a well-known challenge in RF sensing because the reflection signals from multiple targets get mixed at the receiver, interfering with each other. The current version of Wi-Learner works well with a single target. When there are multiple targets, the system works well if the targets are far away from each other. If multiple targets are close to each other, it is challenging for our system to work. We believe some recent progress in multi-dimensional signal processing [54] can help to achieve multi-user sensing. It is a challenging yet interesting research direction to deal with multiple targets in our future work.

Continuous location changes. During our daily life, we are very likely to stand at different locations while performing gestures. In current version, Wi-Learner requires the user to provide one or a few training samples for achieving accurate gesture recognition once their location changes. This process, however, is cumbersome and seems to be not realistic. We believe it is possible to combine our proposed learning-based solution with a model-based solution (e.g., designing a location-independent MNP feature [10]) to overcome this shortcoming. We will leave this scenario as our future work.

Model learning overhead. To achieve accurate cross-domain gesture recognition, Wi-Learner requires a large amount of data from different domains as a training dataset to train the proposed model. Although the collection process of the training dataset is time-consuming and labour-intensive, we believe this issue will be resolved when more developers and researchers are devoted to building standard Wi-Fi public gesture datasets. In addition, the current version of Wi-Learner needs one sample per class in the target domain to update the model for achieving cross-domain gesture recognition. To avoid collecting data in the target domain, our future work is to employ more advanced frameworks such as zero-shot learning [39].

Generalization capability. With more different domain training data for task generation, Wi-Learner can possess a stronger generalization ability. The current implementation of Wi-Learner only employs some data from an open dataset as training data to generate different tasks, it causes a limited generalization ability to a new domain. We believe that Wi-Learner can achieve better cross-domain gesture recognition with more available standard Wi-Fi gesture datasets.

Unseen gesture recognition. In practical scenarios, there are usually occurring undefined gestures. However, the current version of Wi-Learner only focuses on the trained gestures and does not study an unseen gesture involves. This direction is challenging and promising, we leave it as our future work to explore how to adapt to a new gesture based on the learned knowledge of performed gestures.

8 RELATED WORK

In this section, we will discuss the related studies in gesture recognition and other cross-domain works.

8.1 Vision-based Gesture Recognition

Vision-based systems [2, 7, 29, 44, 60] utilize cameras to capture users' motion and perform gesture recognition. Although they can achieve high accuracy, they are highly dependent on the light condition and the viewing angle. Moreover, they would incur privacy invasion. In contrast, Wi-Learner is immune to lighting conditions and does not raise privacy issues.

8.2 Sensor-based Gesture Recognition

Due to the popularity of portable devices, many current systems focus on sensor-based (e.g., smartphone, wristband and smartwatch) gesture recognition [1, 19, 20, 42]. For example, Viband [20] exploits accelerometers to capture bio-acoustic data and perform hand gestures. Another work [42] performs gesture detection by using a smartwatch to track hand trajectory. Though promising, these methods require users to carry wearable devices, which are cumbersome and would raise a feeling of discomfort. Different from the above systems, Wi-Learner identifies different gestures in a non-intrusive manner that does not require users to carry any device.

8.3 RF-based Gesture Recognition

Extensive efforts have been devoted to achieving gesture recognition by employing various RF signals, including millimeter radar [32, 49], RFID [6, 36, 59, 62], and Wi-Fi [25–27, 31, 40]. Wang *et al.* [49] exploit Google's Soli radar sensors for dynamic gesture recognition. mmWrite [32] performs handwriting tracking by using a mmWave radio to sense the hand trajectory. Grfid [62] employs a weighted DTW scheme to achieve robust gesture recognition. These systems can achieve satisfactory results, they however require expensive devices. To avoid this issue, WiSee [31] performs whole-home gesture recognition by using Wi-Fi signals to detect the Doppler shifts induced by gestures. WiFinger [40] exploits the unique CSI patterns for fine-grained finger gesture recognition. SignFi [26] proposes a convolutional neural network to identify performed sign gestures with Wi-Fi devices. The major drawback of these methods is that their performance drops sharply when the new domain is not the same as the trained domain. In contrast, Wi-Learner can realize accurate gesture performance in a new domain.

8.4 Cross-domain Gesture Recognition

Since current mainstream gesture recognition systems can not achieve a good performance when the domain changes, many researchers have tried to address the cross-domain issue. Existing solutions for improving systems' adaption ability to domain variations (e.g., environment, person, location, and orientation) can be divided into two categories, including *domain-independent feature extraction* and *cross-domain adaptation.*

Domain-independent feature extraction. There are two major approaches to extract domain-independent features for gesture recognition. The first approach is to employ an adversarial network as domain discriminator to alleviate the impact of domains, so as to derive domain-independent features [4, 16, 21, 23, 38]. For example, EI [16] designs a conditional adversarial architecture to remove the environmental factors. CrossGR [23] extracts the user-agonistic activity features from the Wi-Fi channel information by using an adversarial network. Although these methods can alleviate the impact of domains, they usually need a substantial amount of samples from

the target domain, which is labour-intensive and time-consuming. The second approach is to utilize geometric models to capture domain-independent features [10, 61]. For example, Widar3.0 [61] leverages multiple links to obtain a domain-independent body-coordinate velocity profile. However, this method requires deploying multiple devices and knowing the accurate location of transceivers in advance. Gao *et al.* [10] employ two non-parallel RF links to design MNP features for achieving position-independent gesture recognition. Yet, it cannot adapt to new users. Different from them, Wi-Learner only needs a few samples in target domains such as one sample per gesture, and requires a pair of transceivers. In addition, Wi-Learner is robust to different domain factors including environment, person, location, orientation, and deployment factors.

Cross-domain adaption. The essence of this category is to transfer a domain-specific recognition model into a new domain by using new data in the new domain. One approach is transfer learning [45, 57, 58]. For example, CrossSense [57] adopts transfer learning to achieve cross-site sensing. TL-Fall [58] proposes a transfer learning scheme to make the fall detection model work well in the new environment with only a few labeled data. Although transfer learning methods can adapt recognition model to a new domain, it requires a large number of samples from the target domain to achieve a satisfying performance. To reduce the collection effort, some recent approaches adopt meta-learning frameworks [5, 53]. For example, RF-Net [5] uses a metric-based meta-learning framework to achieve cross-environment human activity recognition with two pairs of Wi-Fi devices. However, RF-Net only focuses on a single domain factor (e.g., different environments), and has a limited cross-domain performance with an average accuracy of about 60% using one shot per gesture in the target domain. Xiao *et al.* [53] adopt a few-shot learning framework to recognize unseen gestures, which however requires four receivers to transform existing gestures into virtual gestures. This process requires sophisticated knowledge, which cannot be extended to other tasks straightforwardly. Different from them, Wi-Learner can achieve robust performance across different domain factors using one sample per gesture and one pair of transceivers, and is easy to scale to other sensing tasks, which benefits from our a set of signal processing schemes and a new meta-learning framework.

9 CONCLUSION

This paper has presented Wi-Learner, a machine-learning-based cross-domain gesture recognition system. Wi-Learner is designed to be low infrastructure cost by using just one pair of Wi-Fi devices and low overhead by relying on just one sample per gesture when targeting a new domain for new users and device setup. Wi-Learner applies a series of signal processing to minimize the impact of the inherently noisy Wi-Fi signal measurements. It then uses a lightweight autoencoder to extract Wi-Fi signal representation for gesture recognition. To quickly adapt to changing domain factors like users and wireless device locations, Wi-Learner introduces a meta-learner to teach the decision model to maximize the information extracted from limited training samples during deployment. Extensive experiments across three public datasets demonstrate that Wi-Learner can realize accurate and robust cross-domain gesture recognition using contactless Wi-Fi signals.

ACKNOWLEDGMENTS

This work is supported by NSFC A3 Foresight Program Grant 62061146001. This work is also partially supported by the National Natural Science Foundation of China under Grant Nos.61972316 and 62172332, the Shaanxi International Science and Technology Cooperation Program under grant agreement 2020KWZ-013.

REFERENCES

[1] Gaoshuai Cao, Kuang Yuan, Jie Xiong, Panlong Yang, Yubo Yan, Hao Zhou, and Xiang-Yang Li. 2020. Earphonetrack: involving earphones into the ecosystem of acoustic motion tracking. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*. 95–108.

[2] Junming Chen, Jingjing Meng, Xinchao Wang, and Junsong Yuan. 2020. Dynamic graph cnn for event-camera based gesture recognition. In *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 1–5.

[3] Zhe Chen, Chao Cai, Tianyue Zheng, Jun Luo, Jie Xiong, and Xin Wang. 2021. RF-Based Human Activity Recognition Using Signal Adapted Convolutional Neural Network. *IEEE Transactions on Mobile Computing* (2021).
[4] Cao Dian, Dong Wang, Qian Zhang, Run Zhao, and Yinggang Yu. 2020. Towards Domain-independent Complex and Fine-grained Gesture Recognition with RFID. *Proceedings of the ACM on Human-Computer Interaction* 4, ISS (2020), 1–22.
[5] Shuya Ding, Zhe Chen, Tianyue Zheng, and Jun Luo. 2020. RF-net: A unified meta-learning framework for RF-enabled one-shot human activity recognition. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*. 517–530.
[6] Xiaoyi Fan, Wei Gong, and Jiangchuan Liu. 2018. Tagfree activity identification with rfids. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 1 (2018), 1–23.
[7] Biyi Fang, Jillian Co, and Mi Zhang. 2017. Deepasl: Enabling ubiquitous and non-intrusive word and sentence-level sign language translation. In *Proceedings of the 15th ACM Conference on Embedded Network Sensor Systems*. 1–13.
[8] Chao Feng, Jie Xiong, Liqiong Chang, Ju Wang, Xiaojiang Chen, Dingyi Fang, and Zhanyong Tang. 2019. WiMi: Target material identification with commodity Wi-Fi devices. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 700–710.
[9] Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*. PMLR, 1126–1135.
[10] Ruiyang Gao, Mi Zhang, Jie Zhang, Yang Li, Enze Yi, Dan Wu, Leye Wang, and Daqing Zhang. 2021. Towards Position-Independent Sensing for Gesture Recognition with Wi-Fi. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 2 (2021), 1–28.
[11] Georgia Gkioxari, Ross Girshick, Piotr Dollár, and Kaiming He. 2018. Detecting and recognizing human-object interactions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 8359–8367.
[12] Yu Guan and Thomas Plötz. 2017. Ensembles of deep lstm learners for activity recognition using wearables. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 2 (2017), 1–28.
[13] Linlin Guo, Lei Wang, Chuang Lin, Jialin Liu, Bingxian Lu, Jian Fang, Zhonghao Liu, Zeyang Shan, Jingwen Yang, and Silu Guo. 2019. Wiar: A public dataset for WiFi-based activity recognition. *IEEE Access* 7 (2019), 154935–154945.
[14] Xiaonan Guo, Jian Liu, Cong Shi, Hongbo Liu, Yingying Chen, and Mooi Choo Chuah. 2018. Device-free personalized fitness assistant using WiFi. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 4 (2018), 1–23.
[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.
[16] Wenjun Jiang, Chenglin Miao, Fenglong Ma, Shuochao Yao, Yaqing Wang, Ye Yuan, Hongfei Xue, Chen Song, Xin Ma, Dimitrios Koutsonikolas, et al. 2018. Towards environment independent device free human activity recognition. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*. 289–304.
[17] Gregory Koch, Richard Zemel, Ruslan Salakhutdinov, et al. 2015. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, Vol. 2. Lille.
[18] Hao Kong, Li Lu, Jiadi Yu, Yingying Chen, Linghe Kong, and Minglu Li. 2019. Fingerpass: Finger gesture-based continuous user authentication for smart homes using commodity wifi. In *Proceedings of the Twentieth ACM International Symposium on Mobile Ad Hoc Networking and Computing*. 201–210.
[19] Gierad Laput and Chris Harrison. 2019. Sensing fine-grained hand activity with smartwatches. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–13.
[20] Gierad Laput, Robert Xiao, and Chris Harrison. 2016. Viband: High-fidelity bio-acoustic sensing using commodity smartwatch accelerometers. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. 321–333.
[21] Chenning Li, Manni Liu, and Zhichao Cao. 2020. WiHF: Enable user identified gesture recognition with WiFi. In *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 586–595.
[22] Dong Li, Jialin Liu, Sunghoon Ivan Lee, and Jie Xiong. 2020. FM-track: pushing the limits of contactless multi-target tracking using acoustic signals. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*. 150–163.
[23] Xinyi Li, Liqiong Chang, Fangfang Song, Ju Wang, Xiaojiang Chen, Zhanyong Tang, and Zheng Wang. 2021. CrossGR: Accurate and Low-cost Cross-target Gesture Recognition Using Wi-Fi. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 1 (2021), 1–23.
[24] Chi Lin, Jiaye Hu, Yu Sun, Fenglong Ma, Lei Wang, and Guowei Wu. 2018. WiAU: An accurate device-free authentication system with ResNet. In *2018 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 1–9.
[25] Chi Lin, Tingting Xu, Jie Xiong, Fenglong Ma, Lei Wang, and Guowei Wu. 2020. WiWrite: An Accurate Device-Free Handwriting Recognition System with COTS WiFi. In *2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 700–709.
[26] Yongsen Ma, Gang Zhou, Shuangquan Wang, Hongyang Zhao, and Woosub Jung. 2018. Signfi: Sign language recognition using wifi. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 1 (2018), 1–21.

[27] Kai Niu, Fusang Zhang, Jie Xiong, Xiang Li, Enze Yi, and Daqing Zhang. 2018. Boosting fine-grained activity sensing by embracing wireless multipath effects. In *Proceedings of the 14th International Conference on emerging Networking EXperiments and Technologies*. 139–151.

[28] Aaron van den Oord, Nal Kalchbrenner, Oriol Vinyals, Lasse Espeholt, Alex Graves, and Koray Kavukcuoglu. 2016. Conditional image generation with pixelcnn decoders. *arXiv preprint arXiv:1606.05328* (2016).

[29] Munir Oudah, Ali Al-Naji, and Javaan Chahl. 2020. Hand gesture recognition based on computer vision: a review of techniques. *journal of Imaging* 6, 8 (2020), 73.

[30] Massimo Panella and Rosa Altilio. 2018. A smartphone-based application using machine learning for gesture recognition: Using feature extraction and template matching via Hu image moments to recognize gestures. *IEEE Consumer Electronics Magazine* 8, 1 (2018), 25–29.

[31] Qifan Pu, Sidhant Gupta, Shyamnath Gollakota, and Shwetak Patel. 2013. Whole-home gesture recognition using wireless signals. In *Proceedings of the 19th annual international conference on Mobile computing & networking*. 27–38.

[32] Sai Deepika Regani, Chenshu Wu, Beibei Wang, Min Wu, and KJ Ray Liu. 2021. mmWrite: Passive Handwriting Tracking Using a Single Millimeter Wave Radio. *IEEE Internet of Things Journal* (2021).

[33] Hamada Rizk, Ahmed Shokry, and Moustafa Youssef. 2019. Effectiveness of data augmentation in cellular-based localization using deep learning. In *2019 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 1–6.

[34] Panneer Selvam Santhalingam, Al Amin Hosain, Ding Zhang, Parth Pathak, Huzefa Rangwala, and Raja Kushalnagar. 2020. mmASL: Environment-Independent ASL Gesture Recognition Using 60 GHz Millimeter-wave Signals. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 1 (2020), 1–30.

[35] Souvik Sen, Božidar Radunovic, Romit Roy Choudhury, and Tom Minka. 2012. You are facing the Mona Lisa: Spot localization using PHY layer information. In *Proceedings of the 10th international conference on Mobile systems, applications, and services*. 183–196.

[36] Longfei Shangguan, Zimu Zhou, and Kyle Jamieson. 2017. Enabling gesture-based interactions with objects. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*. 239–251.

[37] Sheng Shen, He Wang, and Romit Roy Choudhury. 2016. I am a smartwatch and i can track my user's arm. In *Proceedings of the 14th annual international conference on Mobile systems, applications, and services*. 85–96.

[38] Cong Shi, Jian Liu, Nick Borodinov, Bruno Leao, and Yingying Chen. 2020. Towards Environment-independent Behavior-based User Authentication Using WiFi. In *2020 IEEE 17th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*. IEEE, 666–674.

[39] Richard Socher, Milind Ganjoo, Hamsa Sridhar, Osbert Bastani, Christopher D Manning, and Andrew Y Ng. 2013. Zero-shot learning through cross-modal transfer. *arXiv preprint arXiv:1301.3666* (2013).

[40] Sheng Tan and Jie Yang. 2016. WiFinger: Leveraging commodity WiFi for fine-grained finger gesture recognition. In *Proceedings of the 17th ACM international symposium on mobile ad hoc networking and computing*. 201–210.

[41] Raghav H Venkatnarayan, Griffin Page, and Muhammad Shahzad. 2018. Multi-user gesture recognition using WiFi. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*. 401–413.

[42] Tran Huy Vu, Archan Misra, Quentin Roy, Kenny Choo Tsu Wei, and Youngki Lee. 2018. Smartwatch-based early gesture detection 8 trajectory tracking for interactive gesture-driven applications. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 1 (2018), 1–27.

[43] Chuyu Wang, Jian Liu, Yingying Chen, Hongbo Liu, Lei Xie, Wei Wang, Bingbing He, and Sanglu Lu. 2018. Multi-touch in the air: Device-free finger tracking and gesture recognition via cots rfid. In *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE, 1691–1699.

[44] Huogen Wang, Pichao Wang, Zhanjie Song, and Wanqing Li. 2017. Large-scale multimodal gesture recognition using heterogeneous networks. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*. 3129–3137.

[45] Jindong Wang, Yiqiang Chen, Lisha Hu, Xiaohui Peng, and S Yu Philip. 2018. Stratified transfer learning for cross-domain activity recognition. In *2018 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE, 1–10.

[46] Jie Wang, Zhouhua Ran, Qinghua Gao, Xiaorui Ma, Miao Pan, and Kaiping Xue. 2021. Multi-person device-free gesture recognition using mmWave signals. *China Communications* 18, 2 (2021), 186–199.

[47] Lei Wang, Ke Sun, Haipeng Dai, Wei Wang, Kang Huang, Alex Liu, Xiaoyu Wang, and Qing Gu. 2019. WiTrace: Centimeter-Level Passive Gesture Tracking Using OFDM signals. *IEEE Transactions on Mobile Computing* (2019).

[48] Minsi Wang, Bingbing Ni, and Xiaokang Yang. 2017. Recurrent modeling of interaction context for collective activity recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3048–3056.

[49] Saiwen Wang, Jie Song, Jaime Lien, Ivan Poupyrev, and Otmar Hilliges. 2016. Interacting with soli: Exploring fine-grained dynamic gesture recognition in the radio-frequency spectrum. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. 851–860.

[50] Wei Wang, Alex X Liu, and Muhammad Shahzad. 2016. Gait recognition using wifi signals. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 363–373.

[51] Wei Wang, Alex X Liu, Muhammad Shahzad, Kang Ling, and Sanglu Lu. 2015. Understanding and modeling of wifi signal based human activity recognition. In *Proceedings of the 21st annual international conference on mobile computing and networking*. 65–76.

[52] Wei Wang, Alex X Liu, and Ke Sun. 2016. Device-free gesture tracking using acoustic signals. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*. 82–94.
[53] Rui Xiao, Jianwei Liu, Jinsong Han, and Kui Ren. 2021. OneFi: One-Shot Recognition for Unseen Gesture via COTS WiFi. In *Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems*. 206–219.
[54] Yaxiong Xie, Jie Xiong, Mo Li, and Kyle Jamieson. 2019. mD-Track: Leveraging multi-dimensionality for passive indoor Wi-Fi tracking. In *The 25th Annual International Conference on Mobile Computing and Networking*. 1–16. doi:10.1145/3300061.3300133.
[55] Jie Xiong, Karthikeyan Sundaresan, and Kyle Jamieson. 2015. Tonetrack: Leveraging frequency-agile radios for time-based indoor wireless localization. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*. 537–549.
[56] Shuangjiao Zhai, Zhanyong Tang, Petteri Nurmi, Dingyi Fang, Xiaojiang Chen, and Zheng Wang. 2021. RISE: robust wireless sensing using probabilistic and statistical assessments. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*. 309–322.
[57] Jie Zhang, Zhanyong Tang, Meng Li, Dingyi Fang, Petteri Nurmi, and Zheng Wang. 2018. CrossSense: Towards cross-site and large-scale WiFi sensing. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*. 305–320.
[58] Lei Zhang, Zhirui Wang, and Liu Yang. 2019. Commercial Wi-Fi based fall detection with environment influence mitigation. In *2019 16th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 1–9.
[59] Shigeng Zhang, Chengwei Yang, Xiaoyan Kui, Jianxin Wang, Xuan Liu, and Song Guo. 2019. Reactor: Real-time and accurate contactless gesture recognition with RFID. In *2019 16th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 1–9.
[60] Zhengyou Zhang. 2012. Microsoft kinect sensor and its effect. *IEEE multimedia* 19, 2 (2012), 4–10.
[61] Yue Zheng, Yi Zhang, Kun Qian, Guidong Zhang, Yunhao Liu, Chenshu Wu, and Zheng Yang. 2019. Zero-effort cross-domain gesture recognition with Wi-Fi. In *Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services*. 313–325.
[62] Yongpan Zou, Jiang Xiao, Jinsong Han, Kaishun Wu, Yun Li, and Lionel M Ni. 2016. Grfid: A device-free rfid-based gesture recognition system. *IEEE Transactions on Mobile Computing* 16, 2 (2016), 381–393.