DATA ANALYTICS AND MACHINE LEARNING

ML-WiGR: a meta-learning-based approach for cross-domain device-free gesture recognition

Zhenyue Gao 1,2 · Jianqiang Xue 1 · Jianxing Zhang 1 · Wendong Xiao 1,2

Accepted: 24 March 2022 / Published online: 11 May 2022
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2022

Abstract
Accurate sensing and understanding of gestures can improve the quality of human–computer interaction and show great theoretical significance and application potentials in the fields of smart home, assisted medical care and virtual reality. WiFi channel state information (CSI)-based device-free wireless gesture recognition requires no sensors and has a series of advantages such as permission for non-line-of-sight scenario, low cost, preserving for personal privacy and working in the dark night. Although most of the current WiFi CSI-based gesture recognition approaches can achieve good performance, they are difficult to adapt to the new domains. Therefore, this paper proposes ML-WiGR, a novel approach for device-free gesture recognition in cross-domain applications. ML-WiGR applies convolutional neural networks (CNN) and long short-term memory (LSTM) neural networks as the basic model for gesture recognition to extract spatial and temporal features. Combined with the meta-learning training mechanism, ML-WiGR can dynamically adjust the learning rate and meta-learning rate in training process adaptively and optimize the initial parameters of a basic model for gesture recognition, only using a few samples and several iterations to adapt to the new domain. In the experiments, the approach is tested under a variety of scenarios. The results show that ML-WiGR can achieve comparable performance against existing approaches with only a small number of samples for training in cross-domains.

Keywords Device-free gesture recognition · WiFi channel state information · Meta-learning · Deep learning

1 Introduction

In recent years, with the development of artificial intelligence and computer technology, as well as people’s increasing attention to the quality of life, device-free wireless sensing technology has gradually become an important research topic of intelligent applications. The device-free wireless sensing technology can utilize the wireless radio frequency signal in the environment to sense the state of human body without attaching any electronic device. Compared with traditional technologies, such as videos (Neethu et al. 2020), wearables (Tramontano et al. 2019), ultrasounds (Chen et al. 2017) and other technologies (Qin et al. 2019), it has a series of advantages such as permission for non-line-of-sight scenario, low cost, preserving for personal privacy and working in the dark night.

WiFi is a common radio frequency technology and usually can be employed for intelligent sensing. For example, Halperin et al. released the CSI Tool (Halperin et al. 2011) to extract WiFi CSI (Channel state information) from commercial network interface cards, which greatly facilitates the acquisition of CSI from commercial WiFi devices. WiFi-based sensing technology with CSI has been extensively studied for various applications, including sleep detection (Gu et al. 2019), fall detection (Ding and Wang 2020), gesture recognition (Tang et al. 2021), WiFi imaging (Adib et al. 2015), crowd detection (Xi et al. 2014), lip recognition (Wang et al. 2016), daily behavior detection (Wang et al. 2015), breathing and heartbeat detection (Abdelnasser et al. 2015;
Liu et al. 2015), gait recognition (Zhang et al. 2019), indoor localization (Zhang et al. 2020a, b) and trajectory tracking (Wu et al. 2021).

Accurate sensing and understanding of gestures can improve the quality of human–computer interaction and has great theoretical significance and application potentials in the fields of smart home, assisted medical care and virtual reality. As shown in Fig. 1, when the human hands moving in the sensing area of WiFi, it will disturb the normal propagation of the signal and there will be some phenomena such as reflection, refraction, diffraction and scattering. By analyzing the changes of received WiFi signal, the state of human hand can be inferred.

We collectively refer to the factors related to gesture recognition as domains and it is called crossing domains when the domains change. Although many WiFi-based gesture recognition systems have performed well in the fixed domain, such as WiGest (Abdelnasser et al. 2015), WiG (He et al. 2015), WiGeR (Al-qaness and Li 2016), WiFinger (Li et al. 2016), Smokey (Zheng et al. 2016) and WiMU (Venkatnarayan et al. 2018), they are difficult to adapt to the new domains when the person or environment changes. The performance of a classifier trained with original signal features in one domain usually drops sharply in another domain.

At present, there are some works researching cross-domain problems. Widar3.0 (Zheng et al. 2019) extracts the body–coordinate velocity profile (BVP) feature from the Doppler frequency shift (DFS) of WiFi CSI and can perform well with zero effort in cross-domain tasks. But Widar3.0 needs more network links to get better performance, which may consume more resources of devices. CrossSense (Zhang et al. 2018) and EI (Jiang et al. 2018) use transfer learning and adversarial learning methods to deal with cross-domain problems, respectively. However, they both need to pay more effort on collecting samples in the new domains to get better performance.

In order to improve the ability to adapt to the new domains, a new approach needs to be suggested to solve the cross-domain problem in WiFi-based device-free wireless sensing, hoping that a smaller cost of collecting samples and training models is required to get better performance with less network links when facing with the changes of domains.

Meta-learning can learn some general knowledge from a series of tasks in source domain which are similar to the target task. The knowledge can help new tasks iterate quickly and achieve better performance with few samples. Thus, in this paper in order to address the cross-domain problem, we proposed ML-WiGR, a meta-learning approach for cross-domain gesture recognition with WiFi CSI. ML-WiGR mainly contains two aspects: the improved recognition mechanism and the model structure.

We modified the training mechanism of Reptile (Nichol et al. 2018), a first-order meta-learning algorithm, to improve the performance to deal with cross-domain tasks. The new meta-learning strategy introduces online adaptive adjustment strategies of learning rate and meta-learning rate to avoid the expensive hyperparameter adjustment problem during training. ML-WiGR trains the initial parameters of the basic model so that it only needs a small number of gradient steps with a small amount of training data from a new domain to generate a new model that can adapt to the cross-domain tasks. Meanwhile, we designed a simple but practical neural network model for gesture recognition. It applies convolutional neural networks (CNN) and long short-term memory (LSTM) networks as a basic model to extract the spatial–temporal characteristics of BVP that extracted from DFS of CSI and it can perform well with the improved meta-learning mechanism.

The rest of this paper is arranged as follows. Section 2 reviews the related work. Section 3 gives the description and formulation of the problem. Section 4 presents the details of the proposed ML-WiGR, including the recognition mechanism and the model structure. Sect. 5 validates the proposed approach with extensive experiments. Finally, Sect. 6 concludes this paper.

2 Related works

2.1 Device-free WiFi-based gesture recognition

WiFi-based device-free wireless sensing has attracted a great amount of interest as it provides a ubiquitous sensing solution by using the pervasive WiFi infrastructure (Wu et al. 2017). The interests of WiFi have experienced from received signal strength indicator (RSSI) Abdelnasser et al. (2015) and Zhang et al. (2020b) to CSI which is accepted by more recent applications Wu et al. (2021); Tang et al. (2021); Ding and Wang (2020). The existing work is mainly divided into two categories: pattern-based and model-based recognition solutions. Different gestures or activities will exert different influence on surrounding wireless signals; however, the influence patterns are so weak and complex that many sys-
tems are designed based on machine learning. For example, WiG He et al. (2015) uses statistical features and SVM to recognize four gestures. The recognition rate on the LOS path reaches 92%, and the recognition rate on the NLOS path reaches 88%. WiFinger Li et al. (2016) uses a series of signal processing methods and the KNN algorithm to realize the recognition of 9 American sign languages and realize finger-level gesture recognition with an accuracy of 90.4%. In general, although the existing machine learning approaches can enable WiFi-based device-free gesture recognition to get well performance in the fixed domain, they have limitations when used in complex cross-domain scenarios, including lack of consistent features and the needs of a large number of samples.

There are some works to deal with cross-domain problems. Widar3.0 Zheng et al. (2019) extracts the BVP feature from the DFS of WiFi CSI, and it can perform well with zero effort in cross-domain tasks. But Widar3.0 needs more links to get better performance, which may consume more costs of devices. CrossSense Zhang et al. (2018) and EI Jiang et al. (2018) use transfer learning and adversarial learning methods to deal with cross-domain problems, respectively, but need to pay more cost for collecting samples in new domain to get better performance. Therefore, a new approach is required to solve cross-domain issues with less network links and a small number of samples.

### 2.2 Meta-learning

In recent years, with the development of artificial intelligence, deep learning techniques has achieved remarkable successes in various scenarios, including recommender systems Wu et al. (2019, 2021), power system Luo et al. (2019), autonomous vehicle Kebria et al. (2019), service computing Wu et al. (2020, 2021). Although these advances and solutions Luo et al. (2020), ample challenges remain to be solved, for example, the large amounts of data and training that are needed to achieve good performance. These requirements severely constrain the ability of deep neural networks to learn new concepts quickly. Therefore, meta-learning Sung et al. (2018) and Hu et al. (2018) has been suggested as one strategy to overcome these challenges. This kind of methods is used for training on a series of tasks which are similar to the target task, and the meta-learner can learn some general knowledge in multiple epochs of task iterations. The knowledge can help the training of the new tasks iterate quickly and achieve better performance.

Currently, meta-learning techniques mainly include metric-based, model-based and optimization-based approaches Huisman et al. (2021). The metric-based meta-learning Sung et al. (2018) and Chen et al. (2020) requires no adjustments for the test tasks, but may achieve bad performance when the distributions of test sets far away from train sets. Due to the flexibility of the system, the model-based methods Kuznetsova et al. (2020) have wider applicability than most metric-based meta-learning, but have worse performance for many supervised tasks. Compared with above two methods, the optimization-based meta-learning approaches, such as MAML Finn et al. (2017), LEO Anderson et al. (2018) and meta-learner LSTM Baydinet al. (2017), can achieve better performance when the tasks are more widely distributed. However, it needs to optimize the base model for each task with more expensive computation resources. In order to improve the performance for cross-domain tasks of device-free gesture recognition, in this paper optimization-based meta-learning approach is chosen as the learner to classify the category of gestures.

### 3 Problem description and formulation

#### 3.1 Cross-domains

Although many previous systems can get better performance in the fixed conditions, the problem of crossing domains need to be considered before deployment. Taking WiG as an example, it has a good performance in the source domain, but the accuracy drops significantly when it is applied to other domains directly.

Different gestures will exert different influence on surrounding WiFi signals. However, the influence patterns are so weak and complex that if we could not find the most consistent characteristics from the changing patterns of gestures in different domains, the performance may be degraded significantly.

The problems are often caused by the changes of the environment or the person in the sensing area. In this paper, we classify the cross-domain problems as single cross-domain and multiple cross-domain problems. Single cross-domain problem involves only one factor changing in the environment. Figure 2 shows the single cross-domain problems caused by the changing of different single factors. In real scene, it is often faced with multiple cross-domain issues, that is, there are multiple factors changing together. Therefore, when verifying the feasibility of the approach, a variety of cross-domain influences need to be considered.

#### 3.2 Problem description

In this paper, we hope that when facing with the changes of domains, only low effort is needed in collecting samples and training models to get better performance. Traditionally, a machine learning algorithm usually focuses on a special task $T$. In the field of gesture recognition based on WiFi CSI, we assume that a basic task is to apply a classifier to determine the
labels of gestures in certain domains. For example, we have \( K \) samples of \( L \) labels with the following representations:

\[
D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_K, y_K)\}, \quad x_i \in \mathbb{R}^d, \quad y_i \in \{0, 1, \ldots, L\}
\]

The main goal of training model is to establish a classifier \( f \), where the sample \( x_i \) is used as the input of the model, and the estimated value of the corresponding label \( y_i \) is the output of the model. In a general supervised learning scenario, the number of labels \( L \) is limited, the number of samples \( K \) is large, and all samples are simply divided into two parts: a training set \( D_{\text{train}} \) and a testing set \( D_{\text{test}} \). When facing with cross-domain problems, we hope to learn a better classifier in a domain with only a small number of samples and less training cost. At this time, the number of samples \( K \) will be small, and the problem can be regarded as a few-shot learning problem. When a small number of samples are directly applied to this problem, it will cause serious overfitting.

### 3.3 Problem formulation

To solve the above problem, meta-learning mechanism is introduced and the proposed approach no longer focuses on a specific gesture classification task, but serves to formulate a meta-task model \( F \). It can obtain the ability to estimate the new task \( T_N \) by learning from tasks in the task set \( T = \{T_1, T_2, T_3, \ldots, T_{N-1}\} \). The classification task \( N \) is different from the other \( N - 1 \) tasks which are in different domains.

In the field of applying meta-learning for WiFi-based gesture recognition, we defines a specific task \( T_i \) as a classification task in a certain domain, which need to distinguish \( L \) different categories of gestures. There is a set of tasks \( T = \{T_1, T_2, T_3, \ldots, T_{N-1}\} \), where each task \( T_i \) contains \( L \) labels of gestures. The number of each gestures \( k \) is very small, generally \( k < 20 \). But these samples are usually easy to obtain. When the same gestures appearing in a new domain, these samples constitute a new classification task \( T_N \). At this time, for each category of gestures in the new domain, only \( k \) samples are available. The main goal of modeling is to complete the final classification task \( T_N \) by learning \( L \times k \) new samples and obtain the classifier model \( f_N \). In task \( T_N \), \( L \times k \) samples can constitute a support set, denoted as: \( S_u = \{(x_1, y_1), (x_2, y_2), \ldots, (x_{L \times k}, y_{L \times k})\}, x_i \in \mathbb{R}^d, y_i \in \{0, 1, \ldots, L\} \), the samples to be classified constitute the testing set.

Since \( k \) is small, only \( L \times k \) samples are not enough to complete the task \( T_N \). However, learning experience can be borrowed from similar tasks \( T_i \) in other domains. The \( L \times k \) samples in \( T_i \) can constitute a sample set, denoted as \( S_u = \{(x_1, y_1), (x_2, y_2), \ldots, (x_{L \times k}, y_{L \times k})\}, x_i \in \mathbb{R}^d, y_i \in \{0, 1, \ldots, L\} \). The samples with labels to be verified in \( T_i \) constitute the query set. Here, the sample set and query set of the task \( T_i \) simulate the support set and test set of the final task \( T_N \), respectively. The difference is that the sample set and query set are sampled from easily available labeled datasets.

A set of tasks \( T_{\text{train}} \) is constructed, which contains a large number of tasks similar to the test set in structure. The task set \( T_{\text{test}} \) contains tasks such as \( T_N \) and is used as a set of tasks for testing. From the perspective of meta-learning, \( T_{\text{train}} \) and \( T_{\text{test}} \) are the training set and test set in the meta-task model \( F \), respectively. Therefore, \( T_{\text{train}} \) and \( T_{\text{test}} \) can be expressed as meta-training set and meta-testing set, respectively. Figure 3 depicts a schematic diagram of dataset division when ML-WiGR crosses different positions in gesture recognition. In this figure, samples at different
locations are used for meta-training across locations, and samples at new locations are used for meta-testing.

4 ML-WiGR

4.1 Approach overview

ML-WiGR is an approach for cross-domain problem in WiFi-based gesture recognition. Figure 4 shows its overall structure. This approach uses a deep neural network model to mine the spatial and temporal characteristics of CSI and improves the training mechanism of Reptile which is a first-order meta-learning method. An online adaptive adjustment strategy of learning rate and meta-learning rate is introduced to deal with the problem that the original Reptile requires expensive hyperparameter adjustment during training. By training the initial parameters of the basic model, it only needs a small number of samples and a few steps of gradient descent to generate a new model to adapt to new task.

4.2 Training mechanism

ML-WiGR first constructs a series of tasks with only a small number of samples in the source domains. The model trained by these samples will be used to evaluate the supervisory on the query set, and the supervision signal will be used to optimize the original network. The model will be more general after learning. In this process, the optimized model of the previous task can be directly used for the training of the next task. After multiple iterations, the model can be applied for evaluation on the test set. ML-WiGR adopts a first-order meta-learning algorithm to speed up the training for new tasks, and it is not easy to cause overfitting to the original data because of using stepwise optimization method in the previous training of similar tasks. In the training process, the original Reptile is sensitive to the choice of hyperparameters. It has two important hyperparameters, namely task-level learning rate $\alpha$ and meta-learning rate $\beta$, which correspond to the learning rate of the inner loop and outer loop, respectively. Increasing any hyperparameter grid search computation by an order will consume more time and resources. In addition, a good value of $\alpha$ and $\beta$ in Reptile is more important than any algorithms based on traditional stochastic gradient descent (SGD), because only a small number of samples are available in few-shot learning problems.

In the original Reptile algorithm, given model parameters $\theta$, SGD is used to adjust the parameters $\theta$ to adapt to the new task $T_t$

$$\theta'_t = \theta - \alpha \nabla_{\theta} L_{\text{train}(t)} (f_{\theta})$$  \hspace{1cm} (1)

where $\alpha$ is the task-level learning rate, $t$ is the task number, and $T_{\text{train}(t)}$ and $T_{\text{test}(t)}$ represent the training and test set of task $t$. These tasks are sampled from the defined task distribution $p(T)$. The goal of meta-learning is

$$\begin{align*}
\min_{\theta} L_{T_{\text{test}(t)}} (f_{\theta}) \\
= L_{T_{\text{test}(t)}} (f_{\theta - \beta \nabla_{\theta} L_{\text{train}(t)} (f_{\theta})})
\end{align*}$$ \hspace{1cm} (2)

The model aims to optimize the parameters $\theta$ so that it can adapt to new tasks with only one or several steps of stochastic gradient descent operations. For the optimization in (2), we have

$$\theta_t = \theta - \beta \nabla_{\theta} L_{T_{\text{test}(t)}} (f_{\theta'_t})$$ \hspace{1cm} (3)

where $\beta$ is the meta-learning rate. Since $\nabla_{\theta} L_{T_{\text{test}(t)}} (f_{\theta'_t})$ is a second derivative, $\theta - \theta'_t$ is used to replace it in Reptile (Nicol et al. 2018), denote as

$$\theta_t = \theta - \beta (\theta - \theta'_t)$$ \hspace{1cm} (4)

Therefore, only the first derivative is required when calculate $\theta'_t$. But in order to find the appropriate hyperparameters for each training stage, we need to add updating rules for $\alpha$ and $\beta$.

First of all, our goal is to update $\alpha$ toward the optimal value $\alpha^*_t$ for a single task to minimize the loss in the training process $L_{T_{\text{train}(t)}} (f_{\theta})$, where $i$ is the number of iterations in a single task. We assume that the optimal $\alpha$ value does not change much during each iteration, so that it can be estimated from $\alpha_{t-1}$. Therefore, perform one step gradient descent in the previous $\alpha_{t-1}$, and obtain the gradient

$$\frac{\partial L_{T_{\text{train}(t)}} (f_{\theta_{i-1}})}{\partial \alpha} = \frac{\partial L_{T_{\text{train}(t)}} (f_{\theta_{i-1}})}{\partial \theta_{t-1}} \cdot \frac{\partial \theta_{t-1}}{\partial \alpha}$$

$$= \nabla_{\theta_{t-1}} L_{T_{\text{train}(t)}} (f_{\theta_{i-1}}) \cdot \frac{\partial (\theta_{t-1} - \alpha_{t-1} \nabla_{\theta_{t-1}} L_{T_{\text{train}(t)}} (f_{\theta_{i-1}}))}{\partial \alpha}$$

$$= \nabla_{\theta_{t-1}} L_{T_{\text{train}(t)}} (f_{\theta_{i-1}}) \cdot \nabla_{\theta_{t-1}} L_{T_{\text{train}(t)}} (f_{\theta_{i-1}})$$ \hspace{1cm} (5)

Obtain $\alpha_i$ as follows

$$\alpha_i = \alpha_{i-1} + \alpha_{\text{hyperlr}} \nabla_{\theta_{i-1}} L_{T_{\text{train}(t)}} (f_{\theta_{i-1}}) \cdot \nabla_{\theta_{i-1}} L_{T_{\text{train}(t)}} (f_{\theta_{i-1}})$$

$$= \alpha_{i-1} + \alpha_{\text{hyperlr}} \nabla_{\theta_{i-1}} L_{T_{\text{train}(t)}} (f_{\theta_{i-1}}) \cdot \nabla_{\theta_{i-1}} L_{T_{\text{train}(t)}} (f_{\theta_{i-1}})$$ \hspace{1cm} (6)

where $\alpha_{\text{hyperlr}}$ is the hyperlearning rate of $\alpha$.

At the same time, we need to update the meta-learning rate $\beta$. Similar to $\alpha$, update $\beta$ toward the optimal value $\beta^*$ to minimize the loss during training $L_{T_{\text{test}(t)}} (f_{\theta'_t})$. It is assumed
that the optimal $\beta$ value does not change much during each iteration, so that it can be estimated from $\beta_{(i-1)}$ via
\[ \frac{\partial \mathcal{L}_{\text{test}(i-1)}(f_{\theta_{i-1}})}{\partial \beta} = \frac{\partial \mathcal{L}_{\text{test}(i-1)}(f_{\theta'_{i-1}})}{\partial \theta_{i-1}} \cdot \frac{\partial \theta_{i-1}}{\partial \beta} \]
\[ = -\nabla_{\theta_{i-1}} \mathcal{L}_{\text{test}(i-1)}(f_{\theta'_{i-1}}) \cdot \nabla_{\theta_{i-2}} \mathcal{L}_{\text{test}(i-2)}(f_{\theta'_{i-2}}) \]  

According to Equation (4), the hypergradient of $\beta$ is
\[ \frac{\partial \mathcal{L}_{\text{test}(i-1)}(f_{\theta'_{i-1}})}{\partial \beta} = - (\theta_{i-1} - \theta'_{i-1}) \cdot (\theta_{i-2} - \theta'_{i-2}) \]
Obtain $\beta_t$ as follows

$$\beta_t = \beta_{t-1} + \beta_{\text{hyperlr}} \left( \theta_{t-1} - \theta'_{t-1} \right) \cdot \left( \theta_{t-2} - \theta'_{t-2} \right)$$

The pseudocode of ML-WiGR is given in Algorithm 1.

Algorithm 1 ML-WiGR

Require: distribution over tasks $p(T)$
Require: initial learning rates $\alpha_0, \beta_0$
Require: hypergradient learning rates $\alpha_{\text{hyperlr}}, \beta_{\text{hyperlr}}$

1: Randomly initialize $\theta$
2: while not done do
3: Sample task $T_{\text{train(t)}} \sim p(T)$
4: for iteration = 1, 2, ..., do
5: Compute loss $L_{\text{train}}(f_\theta)$
6: Compute the gradient of $\alpha$ according to (5)
7: Update $\alpha$ according to (6)
8: Compute adapted parameters with gradient descent: $\theta'_t = \theta_t - \alpha_t \nabla f_{\theta_t} L_{\text{train}}(f_\theta)$
9: end for
10: Compute the gradient of $\beta$ according to (7)
11: Update $\beta$ according to (8)
12: Update $\theta_t = \theta_{t-1} - \beta_t \left( \theta_{t-1} - \theta'_{t-1} \right)$
13: end while

ML-WiGR modified the original Reptile and adaptively adjusts the learning rate $\alpha$ and the meta-learning rate $\beta$. The solution is based on the hypergradient descent (HD) (Baydinet al. 2017) algorithm, which automatically updates the learning rate by performing gradient descent on the learning rate based on the original optimization step. This algorithm does not require additional gradient calculations and only needs to store the gradients in the previous optimization steps.

4.3 Recognition mechanism

ML-WiGR applies a deep neural network as a basic model to mine spatial and temporal features in the tasks of gesture recognition, of which overall structure is shown in Fig. 4. The model consists of a convolutional neural network (CNN) and a recurrent neural network (RNN), which are used for spatial feature extraction and temporal modeling, respectively.

4.3.1 Spatial feature extraction

BVP that extracted from the DFS of CSI is used as the input of this approach. The input data of the basic model is similar to a set of image sequences. Each BVP describes the energy distribution of speed in a short time interval. CNN can identify simple patterns in the data and then generate more complex patterns from these simple patterns in higher-level layers. Therefore, in order to fully understand the generated BVP data, we need to extract the spatial features from a single BVP and model the time dependence of the entire series. A simple model is designed to learn higher-level features.

4.3.2 Temporal modeling

In addition to the local spatial features in each BVP, the BVP sequence also contains dynamic connections with time. RNN can understand context of the input features and model complex time dynamic sequences. LSTM layers are added after the CNNs. Finally, fully connected layers with Softmax as the activation function is applied to output a category vector of length $L$ of which value represent the probability of each labels.

5 Performance

This section shows the dataset and the performance of the proposed approach.

5.1 Experiment setting

In this paper, we use the Widar3.0 gesture recognition dataset (Zheng et al. 2019) of Tsinghua University to verify our proposed gesture recognition method. The dataset contains raw CSI data and extracted signal features (DFS and BVP), including about 260,000 action examples collected in 75 different scenarios (including different positions, orientations and environments). The experiments in Widar3.0 are mainly carried out in 3 indoor environments: an empty classroom equipped with tables and chairs, a spacious hall and an office equipped with sofas, tables and other furniture.

The $L$-way $K$-shot cross-domain problem is defined as follows: select $L$ categories from known gestures in the source domains, provide the model with $K$ different instances about each of $L$ labels, and evaluate the ability of the model about classifying the new instances of these $L$ categories in the target domain. In the experiment, we consider a 5-way 5-shot scenario, using 5 samples to quickly learn 5 labels of gesture classification with only 10 steps of gradient descent.

We implement the proposed approach with PyTorch and run it on a computer with Intel Xeon E5-1620 CPU, NVIDIA GeForce GTX 1080 Ti GPU and 16GB RAM. We extracted the BVP from DFS of four links.

5.2 Evaluation for crossing domains

Crossing domains is often caused by changes of many factors such as environments, locations, orientations and personnel. Specifically, single crossing domain is often caused by only one factor, and multiple crossing domains is caused by multiple factors. In order to evaluate the feasibility of the proposed approach in detail, we decompose cross-domain issues into crossing single domain and crossing multiple domains, and then discuss them separately.
5.2.1 Crossing single domain

Firstly consider the single cross-domain problem. In the experiment, we strictly control variables about the domains and select five categories of gestures for experiments, such as push and pull, sweep, clap, slide and draw zigzag. Specifically, when facing a problem about crossing rooms, all actions come from the same person with the same orientation in the sensing area. Figures 5, 6, 7 and 8 show the recognition performance when crossing different domains with 5-way 5-shot scenario and the performances of crossing orientations and crossing persons drop slightly.

5.2.2 Crossing multiple domains

Next, consider the cross-domain problem when multiple factors change simultaneously. Specifically, we also select five categories of gestures for experiments, such as push and pull, sweep, clap, slide and draw zigzag. When the user, location, orientation and residential room are all changed, we try to quickly learn 5 categories of gesture classification with only 5 samples. Figure 9 shows the performance of the proposed approach when there are multiple factors changing at the same time. The displayed accuracy will be slightly reduced.

5.2.3 Comparison of different sample sizes

Different sample sizes will influence the performance of the approach. In order to verify whether the proposed approach has learned “how to learn,” we also select above five gestures and 1, 5 and 10 samples are selected separately from each category of these gestures for experiment. The selected samples constitute 5-way-1shot, 5-way-5shot and 5way-10shot tasks, respectively. Table 1 shows the performance of the proposed approach under different number of samples. From the table, we can find that as the sample size increases, the recognition accuracy will also improve. The approach can still achieve better performance when the number of sample is
5.2.4 Compared with the state-of-the-art works

In addition to comparing the classification capabilities of the proposed approach when using different numbers of samples, we also compare the proposed approach against several state-of-the-art approaches including Widar3.0, EI, CARM and WiG, where the first two are feasible for cross-domain recognition. Figure 10 shows the performance comparison of these approaches. Specifically, Widar3.0 applies advanced BVP which is a cross-domain feature and does not require additional data or model retraining from new domains. EI incorporates an adversarial network and specializes the training loss to additionally exploit characteristics of unlabeled data in target domains. Both Widar3.0 and EI have capabilities of crossing domains. But Widar3.0 needs more links to get better performance, which may consume more resources of devices. EI requires to pay more cost for collecting samples. CARM uses DFS profiles as learning features and adopts HMM model. WiG uses statistical features and SVM to recognize gestures. Both features and learning models of CARM and WiG do not have cross-domain capabilities so that could not find the most consistent characteristics from the changing patterns of gestures in different domains, so the performance may be degraded significantly. Although our proposed approach requires additional training data in new domains, it only needs to pay the cost of collecting a few samples and acceptable training to obtain a satisfactory performance improvement.

5.2.5 Compared with different features

Although the learner has ability of crossing domain, it is also important for the input features to adapt to new domain. Different features will make difference to the performance of the model. We need to examine which input features are more suitable for our model. The CSI raw data and DFS are often chosen as the inputs of some system. Thus, they are also considered as the input to validate ML-WiGR. Figure 11 shows the results of ML-WiGR with different input features. It can be seen that the model has worse performance when input unprocessed CSI raw data which does not have cross-domain capabilities. Both DFS and BVP have cross-domain capabilities, and BVP is more suitable as the input of ML-WiGR.
base model for each task with more expensive computation overhead. Therefore, it is necessary to improve the speed of convergence for training. ML-WiGR is extended from Reptile algorithm and introduces online adaptive adjustment strategy of learning rate and meta-learning rate to solve the problem of the original Reptile for expensive hyperparameter adjustment during training. Compared with Reptile in Fig. 12, we can see that ML-WiGR shows fast convergence in training phase so that it can save the computation burden.

5.2.7 Discussions

It can be seen from the above experiments that although ML-WiGR requires additional training data for the new domains, it only needs to pay the cost of collecting a few samples for acceptable training to obtain satisfactory performance improvement. Especially combining with BVP, a kind of consistent features of changes, ML-WiGR can improve the performance even more in cross-domain tasks. How to get the samples for cross-domain training may influence the performance of our proposed approach, and it may get worse performance when choosing similar samples. It is deserved further study to use samples by considering their similarities.

6 Conclusions

This paper proposes ML-WiGR, a meta-learning approach for WiFi-based device-free gesture recognition. It applies CNN and LSTM deep neural networks as a basic model to extract the spatial–temporal characteristics of BVPs. We present a modified meta-learning strategy and introduce online adaptive adjustment strategies of learning rate and meta-learning rate to avoid the expensive hyperparameter adjustment problem during training. ML-WiGR trains the initial parameters of the basic model for generating a new model to adapt to cross-domain classification tasks. The experimental results show that ML-WiGR can achieve superior performance against existing approaches with only a small number of samples for training with cross-domains.

In this paper, a few samples is used for training the model of gesture recognition in crossing domains, and future work can be done to further enhance the performance of ML-WiGR by considering the similarities of the collected samples. To extend ML-WiGR to more complex non-line-of-sight scenarios and multi-target gesture recognition are also deserved for further studies.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s00500-022-07110-y.

Funding This work is supported in part by the National Key Research and Development Program of China (grant no. 2017YFB1401203), National Natural Science Foundation (NNSF) of China under Grant 62173032, and Foshan Science and Technology Innovation Program (grant no. BK20AF005).

Data availability The data that supports the findings of this study are openly available in Widar3.0 at https://doi.org/10.1145/3307334.3326081, reference number (Zheng et al. 2019).

Declarations

Conflict of interest The authors declare that we have no conflicts of interests regarding the publication of this article.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Informed consent I consent the journal to review the paper. I inform that the manuscript has not been submitted to other journal for simultaneous consideration. The manuscript has not been published previously. The study is not split up into several parts to increase the quantity of submissions and submitted to various journals or to one journal over time. No data have been fabricated or manipulated (including images) to support my conclusions. No data, text or theories by others are presented as if they were of my own. Proper acknowledgements to other works are provided, and I use no material that is copyrighted. I consent to submit the paper, and I have contributed sufficiently to the scientific work and I am responsible and accountable for the results.

References

Abdelnasser H, Harras KA, Youssef M (2015a) Ubibreathe: a ubiquitous non-invasive wifi-based breathing estimator. In: Proceedings of the 16th ACM international symposium on mobile Ad Hoc networking and computing, pp 277–286

Abdelnasser H, Youssef M, Harras KA (2015b) Wigest: a ubiquitous wifi-based gesture recognition system. In: 2015 IEEE conference on computer communications (INFOCOM), IEEE, pp 1472–1480

Adib F, Hsu CY , Mao H et al (2015) Capturing the human figure through a wall. ACM Trans Gr 34(6):1–13

Al-qaness MAA, Li F (2016) Wiger: Wifi-based gesture recognition system. ISPRS Int J Geo Inf 5(6):1–13

Anderson P, He X, Buehler C et al. (2018) Bottom-up and top-down attention for image captioning and visual question answering. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 6077–6086
Baydin AG, Cornish R, Rubio DM, et al (2017) Online learning rate adaptation with hypergradient descent. arXiv preprint arXiv:1703.04782

Chen H, Li F, Wang Y (2017) Echotrack: acoustic device-free hand tracking on smart phones. In: IEEE INFOCOM 2017-IEEE conference on computer communications, IEEE, pp 1–9

Chen J, Zhan LM, Wu XM et al (2020) Variational metric scaling for metric-based meta-learning. In: Proceedings of the AAAI conference on artificial intelligence, pp 3478–3485

Ding J, Wang Y (2020) A wifi-based smart home fall detection system using recurrent neural network. IEEE Trans Consum Electron 66(4):308–317

Finn C, Abbeel P, Levine S (2017) Model-agnostic meta-learning for fast adaptation of deep networks. In: International conference on machine learning, PMLR, pp 1126–1135

Gu Y, Zhang X, Liu Z et al (2019) Wifi-based real-time breathing and heart rate monitoring during sleep. In: 2019 IEEE global communications conference (GLOBECOM), IEEE, pp 1–6

Halperin D, Hu W, Sheth A et al. (2011) Tool release: gathering 802.11n traces with channel state information. ACM SIGCOMM Comput Commun Rev 41(1):53–53

He W, Wu K, Zou Y et al. (2015) Wig: Wifi-based gesture recognition system. In: 2015 24th international conference on computer communication and networks (ICCCN), IEEE, pp 1–7

Hu H, Gu J, Zhang Z et al. (2018) Relation networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3588–3597

Huisman M, van Rijn JN, Plaat A (2021) A survey of deep meta-learning. Artif Intell Rev 7:1–59

Jiang W, Miao C, Ma F et al. (2018) Towards environment independent device free human activity recognition. In: Proceedings of the 24th annual international conference on mobile computing and networking, pp 289–304

Kebría PM, Khorasvi A, Salaken SM et al (2019) Deep imitation learning for autonomous vehicles based on convolutional neural networks. IEEE/CAA J Automatica Sinica 7(1):82–95

Kuznetsova A, Rom H, Alldrin N et al (2020) The open images dataset v4. Int J Comput Vision 128(7):1956–1981

Li H, Yang W, Wang J et al. (2016) Wifinger: talk to your smart devices with finger-grained gesture. In: Proceedings of the 2016 ACM international joint conference on pervasive and ubiquitous computing, pp 250–261

Liu J, Wang Y, Chen Y et al. (2015) Tracking vital signs during sleep leveraging off-the-shelf wifi. In: Proceedings of the 16th ACM international symposium on mobile Ad Hoc networking and computing, pp 267–276

Luo X, Yuan Y, Chen S et al (2020) Position-transitional particle swarm optimization-incorporated latent factor analysis. IEEE Trans Knowl Data Eng 2:159

Luo Y, Zhao S, Yang D et al (2019) A new robust adaptive neural network backstepping control for single machine infinite power system with tsc. IEEE/CAA J Automatica Sinica 7(1):48–56

Neethu P, Suguna S, Rathi D (2020) An efficient method for human hand gesture detection and recognition using deep learning convolutional neural networks. Soft Comput 7:1–10

Nichol A, Achiam J, Schulman J (2018) On first-order meta-learning algorithms. arXiv preprint arXiv:1803.02999

Qin Z, Du J, Han G et al. (2019) Lol: localization-free online keystroke tracking using acoustic signals. Soft Comput 23(21):11,063–11,075

Sung F, Yang Y, Zhang L, et al. (2018) Learning to compare: Relation network for few-shot learning. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1199–1208

Tang Z, Liu Q, Wu M et al (2021) Wifi csi gesture recognition based on parallel lstm-fcn deep space-time neural network. China Commun 18(3):205–215

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.