Classical to Quantum Transfer Learning Framework for Wireless Sensing Under Domain Shift

Yingxin Shan†, Peng Liao†, Xuyu Wang‡, Lingling An§, and Shiwen Mao II
†Guangzhou Institute of Technology, Xidian University, Guangzhou, China.
‡Knight Foundation School of Computing & Information Sciences, Florida International University, Miami, FL, USA
§School of Computer Science and Technology, Xidian University, Xi'an, China.
II Department of Electrical and Computer Engineering, Auburn University, Auburn, AL, USA
Email: yxinshan@stu.xidian.edu.cn, pengl3@stu.xidian.edu.cn, xuywang@fiu.edu, an.lingling@gmail.com, smao@ieee.org

GLOBECOM 2023 - 2023 IEEE Global Communications Conference | 979-8-3503-1090-0/23/$31.00 ©2023 IEEE | DOI: 10.1109/GLOBECOM54140.2023.10437382

***Abstract*—To implement ubiquitous wireless sensing, the domain shift problem (e.g., different environments, users, devices) for machine learning based approaches should be addressed. Some existing methods are proven to be effective, such as transfer learning and domain adaptation. Meanwhile, quantum machine learning, a combination of quantum computing and machine learning, has attracted much attention. More importantly, quantum transfer learning (QTL) has been successful for certain applications, e.g., image classification. In this paper, we explore a classical to quantum (C2Q) framework to address the domain shift problem in wireless sensing by exploiting the great potential of QTL. Specifically, we first analyze the data shift problem in various types of wireless datasets by calculating the Kullback–Leibler (KL) divergence of different domains. Then, a QTL framework is designed to introduce importance weighting and adversarial strategies. We finally evaluate the proposed framework using the representative human activity recognition task on three wireless sensing datasets. Experimental results demonstrate the feasibility of the framework and its great potential for solving the domain shift problem in wireless sensing.**

***Index Terms*—wireless sensing, quantum machine learning, quantum neural networks, transfer learning.**

I. Introduction

Wireless sensing has become a research hotspot recently, which focuses on leveraging the existing wireless devices for sensing, such as the ability to locate the position of objects and detect the behavior of objects, e.g., falling, breathing, heartbeat, and gestures [1]–[3]. Many applications based on such sensing abilities are emerging. However, the wireless signal is susceptible to the surrounding environment, hardware impairments, interference, and noise, which are all time-varying. The deployment of these applications across environments, users, and devices is a big challenge in ubiquitous wireless sensing. For example, for the same task, a well-trained model on data collected on one device may perform poorly on data collected on another device. Since collecting and labeling wireless data is labor-intensive, costly, and error-prone, researchers have tried to gain some prior knowledge from some well-trained large models, which can guide the process of training on a small amount of data. This is an effective means to reduce data collection and training costs. Transfer learning (TL) [4] is such a strategy, which trains a good performing predictor on dataset A and then fine-tunes the model on dataset B to achieve good performance. It has been successfully applied in image classification and other fields [5], [6].

The above TL approach can also be extended to quantum machine learning (QML) [7], which is a recent advance of combining quantum computing and machine learning. Quantum computing has been shown to have a great potential for machine learning tasks, with its powerful computational capability and unique features (i.e., quantum entanglement) that cannot be achieved by classical learning. Similar to the neural network in classical machine learning, the parameterized quantum circuit, which is also a quantum neural network (QNN) [8], has been recently introduced as one of the most regular architectures for QML. Recent research has been motivated to focus on implementing existing ML tasks using QNNs, ranging from image classification [9] to transfer learning [10]. Limited by current quantum hardware that supports limited qubits, QML is still in its infancy and too early to demonstrate the quantum supremacy in accuracy and scalability. Therefore, the main focus of the current research direction is to identify possible challenges and demonstrate the potential of QNN-based QML for existing and emerging applications.

Following this trend, in this research we aim to re-design TL using QML, especially for wireless sensing tasks. In the noisy intermediate-scale quantum (NISQ) era [11], designing a TL network working on the real quantum circuit is very complex and time-consuming. Classical data uploading to a quantum circuit takes a lot of time and is constrained by the availability of quantum bits (qubits). Furthermore, quantum computers are very expensive. Thus, using a hybrid quantum-classical TL model will be more appealing for the current NISQ era. For example, the authors in [12] proposed a classical-to-quantum (C2Q) convolutional neural network (CNN), a TL model for three independent binary classification tasks with the MNIST data, which capitalized the advantages of CNN in the few-parameter regime to the full extent. Similar models have also been successfully applied to spoken command recognition [13]. In particular, an attempt was made in [14] to improve the robustness against domain shifts across Wi-Fi scanning sessions, which is the first attempt to use QTL in wireless sensing. However, this approach is more suitable for handling cases where the distributions in the dataset do not differ much, and their performance may drop significantly when the distributions vary a lot in wireless sensing.

In this paper, we leverage the C2Q model, which has exhibited many application advantages, to solve the domain shift problem in wireless sensing. Different from the prior works, various datasets of our wireless sensing, including radar and Wi-Fi datasets, are analyzed and the network architecture is adapted for their respective offset problems using the C2Q TL model. The contributions of this paper are three-fold, as summarized below:

- We analyze the data shift problem in various types of wireless datasets by calculating the KL divergence of different domains.
- We design a novel QML model for wireless sensing, where a variational quantum circuit is designed for the feature extractor. In the training process, importance weighting and an adversarial strategy are also introduced.
- We verify the feasibility of the proposed QML model for human activity recognition across sensors, users, and environments on various wireless datasets.

The rest of the paper is organized as follows. Section II introduces the background and motivation. Section III presents the system design while Section IV presents the experimental study. We finally conclude this paper in Section V.

II. Background and Motivation

A. Domain Shift in Wireless Sensing

The radio frequency (RF) signals received by wireless devices for sensing tasks are susceptible to environmental and subjective influences. An application model trained on a specific environment and subject is likely to perform poorly on another subject in a new environment. This mainly results from the domain shifts in wireless sensing. We will use Kullback-Leibler (KL) divergence [15] to represent the differences between domains (e.g., users, environments, and devices), which is widely used to describe the discrepancies between probability measures.

In this paper, three different types of public datasets are chosen to calculate KL divergence. (i) Ci4R [16]: collected from three different radars at 24 GHz, 77GHz, and 10GHz frequencies for a total of 11 different activities and ambulatory gaits. (ii) MCD [17], [18]: MCD-Gesture are collected by mmWave Radar in 750 domains, including six environments, 25 volunteers, and five locations. (iii) Widar 3.0 [19]: collected for six hand gestures in three different environments. As shown in Fig. 1, there are different degrees of offset in these datasets between domains. A larger value of KL represents a larger difference in distribution between two domains. It is obvious that data collected at different frequencies in the Ci4R have the largest differences in distribution, which could be caused by the differences in hardware. Difference is also considerable between the user domains of MCD, which is probably due to the different behavioral habits of users. As for Widar 3.0, due to the fact that all the experiments were conducted in a laboratory-like setting and proper pre-processing was applied, the differences in Widar 3.0 is the smallest. It follows that the problem of domain shift widely presents in wireless sensing. Although pre-processing can alleviate this phenomenon to some extent, it does not completely solve the problem. To make machine learning models more applicable and robust, measures should be taken to address the domain shift problem.

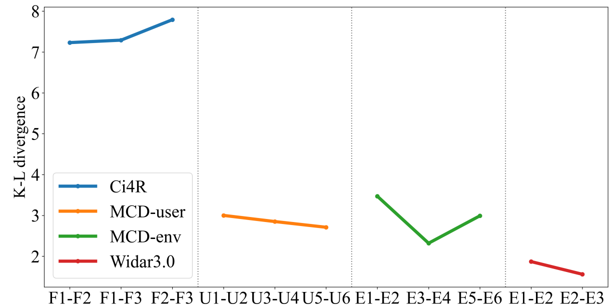

Fig. 1. The K-L divergence between domains in various datasets.

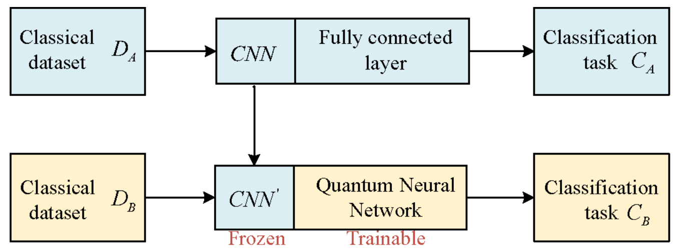

Fig. 2. The classical to quantum (C2Q) transfer learning network architecture.

B. Basic Classical to Quantum Transfer Learning Network

The domain shift problem in wireless sensing can be solved using transfer learning or domain adaptation, which have been remarkably successful in classical machine learning. QML has also attracted considerable attention recently. For instance, QTL has been successful in applications such as image classification. The model used in the prior work, termed the C2Q TL network, is plotted in Fig. 2. The model uses CNN as feature extractor and is trained on dataset D_A to complete the classification task C_A. However, when the trained model is directly used on dataset D_B for task C_B, the model performance will be degraded. To reduce the cost of retraining, the model performs the classification task by *knowledge migration*, which is to freeze the pre-trained CNN feature extractor and ensure the parameters of the QNN are trainable. QNN will first embed classical data into qubits, then use unitary operations to upload parameters, and finally obtain the state of the entire quantum circuit through measurements to help update the parameters during training.

The above C2Q TL model is suitable for small offsets, but it does not work well with data having large domain shifts. Differing from the prior work, we introduce the technique

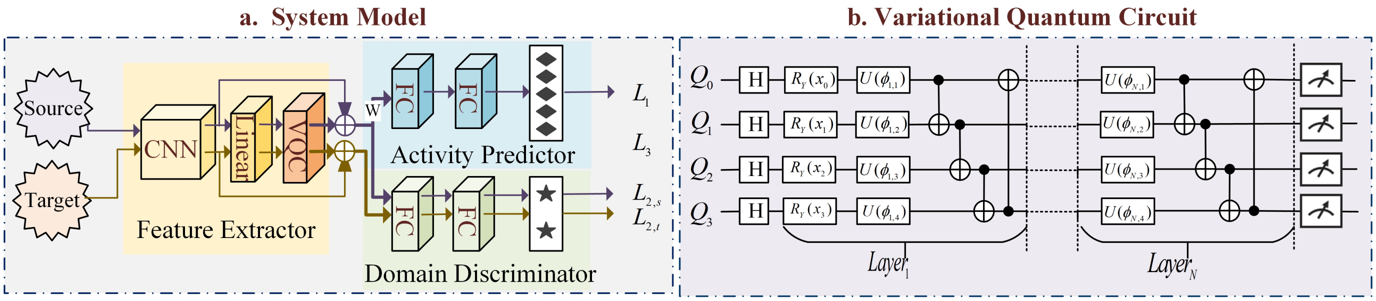

Fig. 3. System model and the structure of the proposed VQC.

of importance weighting and an adversarial strategy into the model to address the domain shift problem in wireless sensing.

III. Proposed C2Q TL Model

An overview of the proposed C2Q TL model is shown in Fig. 3. The input data to our model are divided into source data and target data, which are from the same dataset but were collected in different domains. This problem setting requires that the proposed approach can be capable of learning transferable features for different domains.

Towards this goal, unlike conventional CNNs, our feature extraction module is a CNN where the last layer is replaced by a *variational quantum circuit* (VQC). Through this model, the input data is transformed into a low-dimensional feature, which will be fed into both the activity predictor and domain discriminator. The goal of the activity predictor is to maximize the prediction accuracy and obtain the predictions on input data. To introduce adversarial learning strategies, a classical domain discriminator is incorporated, which contains two fully-connected (FC) layers. Its goal is to label each domain. The model learns domain-independent features by means of a feature extractor. Specifically, the feature extractor generates features that can deceive the domain discriminator as much as possible, while maintaining the recognition performance of the active discriminator. This dual-goal is basically achieved by updating the parameters in the VQC during the training process, and at the same time, freeze the parameters of the CNN. Moreover, we introduce importance weight and design constraints, which can alleviate the offset problems in the dataset and significantly improve the inference performance.

A. Feature Extractor

The feature extractor in our quantum TL model consists of a CNN and VQC. Wireless sensing data is typically classical and high-dimensional. To apply VQC to classical data, the first problem to be solved is uploading data to quantum states, as quantum circuits only act on qubits. Limited by quantum hardware, the number of qubits available today is much fewer than the number of data samples available in classical sensing data. To solve this problem, we use a CNN to compress the high-dimensional wireless sensing data into low-dimensional features. The CNN architecture we use here is Resnet-18 [20], and with this network, a 512-dimensional feature is the ultimate output. A linear layer is then used to map the compressed features to the same size as the qubits. The obtained data vector can be represented as $X = [x_0, x_1, ..., x_n]$, where n is the number of qubits. To successfully upload classical data into quantum state, one of the methods is to introduce specific gates and rotation operation in the beginning of quantum circuit. As shown in Fig. 3(b), each data point in this vector will be embedded in the quantum circuit by applying a Hadamard gate (H) and performing a rotation around the y-axis (R_Y) of the Bloch sphere. Finally, the quantum state of the input data $|X\rangle$ is obtained. A universal parameterized quantum gate $U(\phi_{i,j})$, $i \in [1, n], j \in [1, N]$ is subsequently trained on it, where U are the unitary quantum gates, ϕ is the angle of rotation, which is trainable, and N is the number of layers or the depth of the VQC network. Furthermore, an entangling unitary operation is also executed, which is made of n controlled-NOT (CNOT) gates. Entanglement is proved to be the core of VQC, which can help discovery information between qubits. A two-qubit entangling gate builds entanglement between two qubits but the two qubits are not necessarily neighboring. At the end of the quantum circuit, output states are measured on the classical register using the Pauli-Z matrix, and the concatenation of it and the output features of the CNN is passed to the Activity Predictor and Domain Discriminator.

In order to better transfer the knowledge from the source domain to the target domain, importance weighting [21] is introduced in the training process of feature extractor, which assigns weights to the band values of different training samples to balance the distribution bias. It contains two steps, which are (i) the estimation of the ratio between test and training densities and (ii) the training of a predictive model by weighting the training losses. Specifically, source data will be multiplied with importance weights W, $W = \frac{\exp(KDE^S)}{\exp(KDE^T)}$ to alleviate the data shift problem existing in wireless datasets, where KDE^S and KDE^T respectively represent the Kernel Density Estimation (KDE) of the source data and target data.

B. Activity Predictor and Domain Discriminator

Both the activity predictor and domain discriminator are composed of two fully-connected layers, where the first is used to learn the representation of input and the other is to map the feature representation into a new latent space

$H_i \in \Re^C$, where C is the number of different human activities in the activity predictor or the number of domains in the domain discriminator. In addition, a softmax layer is used to obtain the probability vector of activities or domains as follows $\hat{y}^P = \text{softmax}(W * F + b)$, where F is the output of feature extractor, and W and b are the weight matrix and bias, respectively. Also $\hat{y}^P = [y^{c,P}, y^{d,P}]$, where $y^{c,P}$ denotes the predicted probabilities of activities, and $y^{d,P}$ represents the predicted probabilities of domains.

For the activity predictor, the cross-entropy function is used to calculate the loss between the predictions and the ground truths as follows.

$$L_1 = -\frac{1}{N_a} \sum_X \sum_{i=1}^{M} y_i^{c,T} \log\left(y_i^{c,P}\right), \tag{1}$$

where N_a is the number of training samples in activity predictor, M is the number of activity classes, and $y^{c,T}$ is ground-truth labels of activities. To identify the domain labels of the input activities in the domain discriminator, we define the loss between the domain distributions and true domain labels as follows.

$$L_2 = \frac{1}{N_d} \sum_i -\left[y_i^{d,T} \log\left(y_i^{d,P}\right) + (1 - y_i^{d,T}) \log\left(1 - y_i^{d,P}\right)\right], \tag{2}$$

where N_d is the number of training samples in domain discriminator and $y_i^{d,T}$ is the ground-truth label of the domain.

C. Training Strategies

In the training process, we iteratively update the model parameters. At first, we fix the parameters in the domain discriminator and update the parameters in the VQC and activity predictor according to the Adam algorithm. Next, we fix the parameters in the the parameters in the VQC and activity predictor, and update the rest of the parameters.

At the beginning of training, we use importance weights to prioritize the training set and then train the model using the weighted training set. As can be seen from Section II, the degree of offsets varies cross different wireless sensing datasets. Importance weighting can help to better fit the different datasets and correct the data shifts in them. It uses Gaussian KDE to estimate the probability densities of the test and training sets. And the importance weights for each training instance are calculated, which is the ratio of the test set probability density to the training set probability density.

The main purpose of the adversarial learning process is to extract the domain-independent features by discarding the specific features of each domain, while retaining the common features used for activity recognition. The features extracted by the feature extractor shall confuse the domain discriminator, while the predictor network can effectively tell from which activities the input features are. Our methods adjust parameters in the VQC to better work with adversarial learning network, while the parameters in the CNN is always frozen. Additionally, the distribution of source data and target data are different, and this difference can be quantified by KL divergence. Therefore, we add the KL divergence between domains as constraints in the training process, that is,

$$\begin{aligned} L_3 &= KL\left(P_{X,Y}^{(t\arg et)} || P_{X,Y}^{(source)}\right) \\ &= \int \log\left(\frac{dP_{X,Y}^{(t\arg et)}}{dP_{X,Y}^{(source)}}\right) dP_{X,Y}^{(t\arg et)}, \end{aligned} \tag{3}$$

where $P_{X,Y}^{source}$ and $P_{X,Y}^{target}$ represent the probability distribution of the source domain and target domain, respectively.

With the above loss and constraint, we can finally write the overall loss function as follows.

$$Loss = L_1 + \alpha(L_{2,s} + L_{2,t}) + \beta L_3, \tag{4}$$

where α and β are predefined hyper-parameters, and $L_{2,s}$ and $L_{2,t}$ are domain losses of source data and target data, respectively. In our experiments, α and β are set to 0.1 and 0.0001, respectively.

IV. Experimental Evaluation

In this section, we first describe our experimental setup, including the dataset and performance metrics. Then we present a comparison study of several baselines and our scheme across different domains, including devices, users, and environments. Finally, we evaluate the impact of QNN on recognition ability in the domains and across domains.

A. Experimental Setup

1) Dataset: In our evaluation, three public datasets described in Section II are used. They have different extents of the data shift problem. The data collected from different domains are choose as the source dataset and target dataset, respectively. In particular, we evaluate our proposed methods across three frequencies (F1-F3) of radar sensors in Ci4R, six users (U1-U6) and six environments (E1-E6) in MCD, and three environments (E1-E3) in Widar3.0.

For the QNN, the number of qubits was set to 4. The 512 compressed classical features were encoded into quantum embeddings that go through the VQC model. The depth of VQC was set to 15, leading to 60 adjustable parameters in total. We use a PyTorch open-source library to implement the deep neural networks, and PennyLane cross-platform library to construct the QNNs. Limited by the hardware, the experiments were conducted on a classical computer, which is equipped with 16GB memory, Intel i7-10700 CPU @2.90GHz, and the Nvidia GTX 1660 Graphics Card. QNNs are deemed as a torch layer in our model. Our method is executed for 100 epochs during the experiment. And we update the parameters by using the Adam optimizer with a learning rate of 0.0004 and weight decay rate of 10^{-4}.

2) Performance Metrics: We choose accuracy to quantify the performance of the evaluated schemes. Accuracy represents the percentage of the number of correctly recognized samples over the total number of tested samples in the target domain, which is calculated by $Acc = \frac{N_{correct}}{N_{total}}$.

TABLE I
CROSS-DOMAIN PERFORMANCE OF OUR PROPOSED MODEL AND BASELINE TRANSFER LEARNING METHODS (ACCURACY(%))

	Ci4R			MCD-user			MCD-env			Widar3.0		
	F1-F2	F1-F3	F2-F3	U1-U2	U3-U4	U5-U6	E1-E2	E3-E4	E5-E6	E1-E2	E1-E3	E2-E3
ADDA	48.5	50.8	52.3	58.5	55.7	58.9	59.1	61	58.9	59.9	56.3	57.8
DANN	59.4	60.1	61.7	67.9	65.4	68.2	69.1	71.2	72.3	66.6	70.2	69.7
CTL	19.4	18.76	19.2	56.3	57.8	54.4	55.3	54.6	56	76.5	78.9	77.4
QTL	20.5	17.9	19.8	55.7	56.5	55.6	54.9	55.2	55.4	75.2	76.1	76.8
Proposed	56.7	61.2	58.4	82.5	83.5	79.5	79.3	80.2	78.3	87.8	86.5	86.7

3) Methods Evaluated: We compare our method with other state-of-the-art learning models, including classical models and traditional QTL models, which are presented as follows.

- *ADDA [22]*: ADDA is a domain-adversarial method, which combines the discriminate model, untied weight sharing, and a GAN loss.
- *DANN [23]*: DANN induces the adversarial theory into domain adaptation. It aims at generating features that represent both the source domain and target domain.
- *CTL [14]*: A CNN feature extractor is trained on the source domain, and then the trained CNN is used in the target domain. Parameters in the followed DNN structure are fine-tuned in the training process.
- *QTL [14]*: The QNN model is first trained with labeled data from the source domain and then migrated to learn with few labeled data samples from the target domain with the input/output layers (classical layers) frozen.

B. Ability of Cross-domain Recognition

We evaluate the cross-domain performance of the four baselines and our method on each of the datasets. Table I presents the experimental results. First, the performance of our method substantially better compared to the CTL and QTL methods. For example, in Ci4R, directly migrating data from the source domain to the target domain and performing fine-tuning leads to dramatic performance degradation in the target domain due to the large differences between devices. The recognition accuracy in the source domain can exceed 90%, but the accuracy degradation in the target domain is more than 60%. Our model can achieve a more than 58% improvement over the baselines, which validates that importance weighting and the adversarial strategies are effective.

Compared with the two classical methods for domain adversarial, our method has better cross-domain performance on both the MCD and Widar3.0 datasets, but poorer performance than DANN in Ci4R. The main reason for this is that during the training process, we only set the update parameters of VQC in the feature extractor, while the parameters of CNN are frozen. This indicates that the feature extractor should have a more complex structure to extract domain-independent features from datasets with large domain shifts. In general, the C2Q TL framework we designed provides an effective solutions to the domain shift problem in wireless sensing.

C. Impact of QNN in the Proposed Model

In this experiment, `Ours*` is a new model obtained by replacing the VQC part of our model with the classical fully connected layer. We compare it with our proposed model (Ours) on in-domain and cross-domain recognition. Fig. 4 presents the experiment results of in-domain recognition, meaning that the training data and the test data are from the same domain. In each domain, we select 75% as training data and the remaining 25% as test data. We validate the classical CNN and our method on three datasets. As shown in Fig. 4, both methods achieve more than 95% accuracy in same domain recognition. It validates that our QNN can adequately act as a classical layer. Although it does not outperform the classical, the potential advantage of QNN lies in its fewer trainable parameters than classical layers (over $2,000$ parameters) and its computational efficiency to manipulate 2^n quantum states at once with a small number of quantum gates.

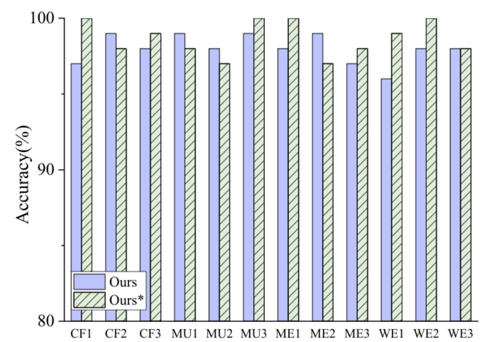

Fig. 4. The in-domain performance of our proposed model and Ours*. The initials of the horizontal labels represent the dataset used, and C, M, and W denote Ci4R, MCD, and Widar3.0, respectively.

Fig. 5 shows the performance of our proposed approach across devices, users, and environments for three different datasets. **Cross devices:** Different radar devices use different bandwidths, frequencies, and modulation methods. It is important to investigate cross-device activity recognition models, which can reduce the cost of data collection. We validate the cross-device feasibility of our proposed method on Ci4R, a dataset collected from three different radar devices. As can be seen from Fig. 5, both methods perform similarly: both having the worst results in F1-F2 and the best results in F1-F3. Furthermore, we can see that the two methods perform the worst on Ci4R, because the source and target data distributions differ the most, compared to the other three datasets. **Cross users:** Due to differences in individual habits such as speed when performing activities, a model that is trained and performs well on data collected from one user may perform less well

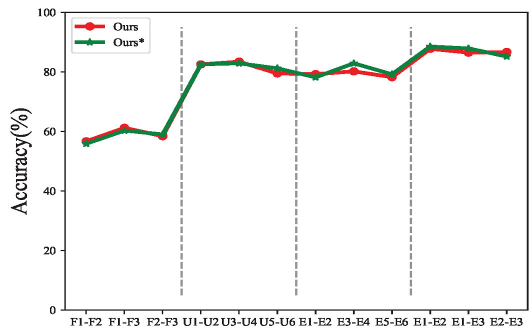

Fig. 5. The cross-domain performance of our proposed model and Ours*. The vertical dashed lines separate each data set and the corresponding selected domain. From left to right are the data from different devices in Ci4R, the data from different users in MCD, the data from different environments in MCD, and the data from different environments in Widar 3.0.

on another user's data. We validate the cross-user feasibility of our proposed approach on MCD. Data collected by the same user in different environments are considered as the same domain. We treat the data from different domains as the source and target domains, respectively. The figure illustrates that our method outperforms Ours* at U3-U4, performs almost the same at U1-U2, and performs slightly worse at U3-U4, by 1.7%. **Cross environments:** Wireless sensing is susceptible to the influence of the surrounding environment, where the number and location of objects within the environment can lead to different multipath effects. We verify the feasibility of our proposed method across environments on MCD and Widar 3.0, where data collected by different users in the same environment are considered as the same domain. It shows the difficulty in comparing the performance of Ours* and our model. Also, although the performance of our method is also evaluated across environments, the performance is not the same on the two different datasets. In Widar's cross-domain evaluation, the best performances of our model and Ours* separately reach 87.8% and 88.5%, while in MCD, they are only 80.2% and 82.9%. The reason for this is explained in Section II, where the KL divergence of Widar3.0 is smaller than that of MCD, implying that the difference in data distribution is smaller for Widar3.0.

V. Conclusions

In this paper, we designed a C2Q TL model to address the domain shift problem in wireless sensing. We analyzed the data shift in several wireless sensing datasets. Importance weighting was proposed to reduce the distribution difference between domains when selecting the training and test sets. We then designed a feature extractor with a hybrid of classical and quantum structures, where the classical CNN was used only to compress high-dimensional features. The inputs and outputs of our VQC connected the classic model by using classical linear layers. Finally, we introduced an adversarial learning strategy. The parameters of the quantum circuit were updated during training. The experimental results demonstrated the feasibility of our proposed framework.

References

[1] J. Liu, H. Liu, Y. Chen, Y. Wang, and C. Wang, "Wireless sensing for human activity: A survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1629–1645, Thirdquarter 2019.

[2] X. Wang, X. Wang, and S. Mao, "RF sensing in the internet of things: A general deep learning framework," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 62–67, 2018.

[3] P. Liao, X. Wang, L. An, S. Mao, T. Zhao, and C. Yang, "TFSemantic: A time-frequency semantic GAN framework for imbalanced classification using radio signals," *ACM Transactions on Sensor Networks*, 2023.

[4] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Springer Journal of Big data*, vol. 3, no. 1, pp. 1–40, May 2016.

[5] C. Yang, L. Wang, X. Wang, and S. Mao, "Environment adaptive RFID based 3D human pose tracking with a meta-learning approach," *IEEE J. Radio Frequency Identification*, vol. 6, no. 1, pp. 413–425, Jan. 2022.

[6] C. Yang, X. Wang, and S. Mao, "TARF: Technology-agnostic RF sensing for human activity recognition," *IEEE Journal of Biomedical and Health Informatics*, vol. 27, no. 2, pp. 636–647, Feb. 2023.

[7] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, "Quantum machine learning," *Nature*, vol. 549, no. 7671, pp. 195–202, Sept. 2017.

[8] S. Jeswal and S. Chakraverty, "Recent developments and applications in quantum neural network: A review," *Springer Archives of Computational Methods in Engineering*, vol. 26, pp. 793–807, May 2019.

[9] J. Landman, N. Mathur, Y. Y. Li, M. Strahm, S. Kazdaghli, A. Prakash, and I. Kerenidis, "Quantum methods for neural networks and application to medical image classification," *Quantum*, vol. 6, p. 881, 2022.

[10] A. Mari, T. R. Bromley, J. Izaac, M. Schuld, and N. Killoran, "Transfer learning in hybrid classical-quantum neural networks," *Quantum*, vol. 4, p. 340, 2020.

[11] J. Preskill, "Quantum computing in the NISQ era and beyond," *Quantum*, vol. 2, p. 79, 2018.

[12] J. Kim, J. Huh, and D. K. Park, "Classical-to-quantum convolutional neural network transfer learning," *arXiv preprint arXiv:2208.14708*, Aug. 2022. [Online]. Available: https://arxiv.org/abs/2208.14708

[13] J. Qi and J. Tejedor, "Classical-to-quantum transfer learning for spoken command recognition based on quantum neural networks," in *Proc. IEEE ICASSP 2022*, Singapore, May 2022, pp. 8627–8631.

[14] T. Koike-Akino, P. Wang, and Y. Wang, "Quantum transfer learning for Wi-Fi sensing," in *Proc. IEEE ICC 2022*, Seoul, South Korea, May 2022, pp. 654–659.

[15] F. M. Polo, R. Izbicki, E. Gomes Lacerda Jr, J. P. Ibieta-Jimenez, and R. Vicente, "A unified framework for dataset shift diagnostics," *arXiv e-prints arXiv:2205.08340*, May 2022. [Online]. Available: https://arxiv.org/abs/2205.08340

[16] S. Z. Gurbuz, M. M. Rahman, E. Kurtoglu, T. Macks, and F. Fioranelli, "Cross-frequency training with adversarial learning for radar micro-Doppler signature classification (rising researcher)," in *Radar Sensor Technology XXIV*, Virtual Conf., Apr./May 2020, pp. 58–68.

[17] Y. Li, D. Zhang, J. Chen, J. Wan, D. Zhang, Y. Hu, Q. Sun, and Y. Chen, "Towards domain-independent and real-time gesture recognition using mmwave signal," *IEEE Trans. Mobile Comput.*, pp. 1–15, 2022.

[18] ——, "DI-Gesture: Domain-independent and real-time gesture recognition with millimeter-wave signals," in *Proc. IEEE GLOBECOM 2022*, Rio de Janeiro, Brazil, Dec. 2022, pp. 5007–5012.

[19] Y. Zheng, Y. Zhang, K. Qian, G. Zhang, Y. Liu, C. Wu, and Z. Yang, "Zero-effort cross-domain gesture recognition with Wi-Fi," in *Proc. ACM MobiSys 2019*, Seoul, South Korea, Oct. 2019, pp. 313–325.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE CVPR 2016*, Las Vegas, NV, June 2016, pp. 770–778.

[21] M. Sugiyama, M. Krauledat, and K.-R. Müller, "Covariate shift adaptation by importance weighted cross validation," *Journal of Machine Learning Research*, vol. 8, no. 5, pp. 985–1005, May 2007.

[22] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. IEEE CVPR 2017*, Honolulu, HI, July 2017, pp. 7167–7176.

[23] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *J. Machine Learning Research*, vol. 17, no. 1, pp. 2096–2030, Apr. 2016.