DWLR: Domain Adaptation under Label Shift for Wearable Sensor

Juren Li[1] , **Yang Yang**[1*] , **Youmin Chen**[1] , **Jianfeng Zhang**[2] , **Zeyu Lai**[1] and **Lujia Pan**[2]
[1]College of Computer Science and Technology, Zhejiang University
[2]Huawei Noah's Ark Lab
{jrlee, yangya, youminchen, jerrylai}@zju.edu.cn, {zhangjianfeng3, panlujia}@huawei.com

Abstract

Wearable sensors play a crucial role in real-world scenarios, such as human activity recognition, sleep monitoring and electrocardiogram monitoring. However, deploying classifiers on them is challenged by distribution shifts across users and devices. Unsupervised domain adaptation (UDA) is proposed to address this, yet existing methods mostly focus on feature distribution shift, neglecting the potential misclassification due to label shift. In this paper, we propose Domain adaptation under label shift for Wearable sensor with Learnable Reweighting (DWLR) to handle both feature and label shifts. Specifically, DWLR employs learnable reweighting to align label distributions between source and target domains. It incorporates elements of information gain during the reweighting process to counter potential distribution shift that could emerge from over-reliance on data with high-confidence pseudo labels. Importantly, since wearable sensor data is time-series data, and can be subjected to distribution shifts originating from either the time domain, the frequency domain, or both, DWLR performs reweighting and alignment separately in these two domains to more robustly handle potential feature distribution shifts. Extensive experiments on three distinct wearable sensor datasets demonstrate the effectiveness of DWLR, yielding a remarkable average performance improvement of 5.85%.

1 Introduction

Wearable sensors play a crucial role in various real-world scenarios, such as human activity recognition [Qian *et al.*, 2019; Hu *et al.*, 2023], sleep monitoring [Kwon *et al.*, 2021], electrocardiogram monitoring [Longoni *et al.*, 2018], etc. It is always challenging to obtain high performance for modeling wearable sensors in real applications. Recently, deep learning-based models designed for wearable sensor data have show promising achievements [Qian *et al.*, 2019; McClure *et al.*, 2020]. Additionally, since most wearable sensor data are time series, some time series approaches can also be used to model wearable sensor data [Bai *et al.*, 2018; Chung *et al.*, 2015]. Most deep learning-based works typically require a large amount of labeled data to train their models and achieve satisfactory performance. However, labeled data are scarce in real-world applications due to the laborious labeling process. Furthermore, deep learning-based methods often assume that the training data (i.e., source domain) comes from the same distribution as the test data (i.e., target domain). Nevertheless, this assumption usually does not hold due to variations in the physical environment and behavior patterns of users, which is also known as domain shift.

In order to address the substantial cost of labeling and the inherent domain shift problem, previous works have trained their models on labeled source domains and subsequently transferred them to unlabeled target domains, a method commonly referred to as Unsupervised Domain Adaptation (UDA) [Patel *et al.*, 2015]. Numerous UDA works have demonstrated remarkable success across various applications [Bousmalis *et al.*, 2018; Bousmalis *et al.*, 2017; Saito *et al.*, 2017]. Some discrepancy-based approaches align the feature distributions of target and source domain by minimizing a divergence that measures the distance between the two distributions [Shen *et al.*, 2018; Lee and Raginsky, 2018; Sun *et al.*, 2016; Kang *et al.*, 2019]. Recently, drawing inspiration from Generative Adversarial Aetworks (GANs), several works have employed adversarial learning methods to acquire domain-invariant features for effective knowledge transfer [Long *et al.*, 2015; Tzeng *et al.*, 2017; Liu and Tuzel, 2016]. Moreover, UDA methods for modeling cross-user wearable sensors have exhibited substantial success [Hu *et al.*, 2023; Qian *et al.*, 2019]. Given that the majority of wearable sensor data comprises time series data, UDA methods tailored for such data have made significant progress [Purushotham *et al.*, 2017; Tonutti *et al.*, 2019; Wilson *et al.*, 2020; Ragab *et al.*, 2022; Ozyurt *et al.*, 2023; Liu and Xue, 2021].

While most existing works are based on the assumption that the source and target domains have the same label distribution, this assumption is usually inapplicable in real-world applications, leading to an inevitable problem in application scenarios, namely label shift. For instance, as depicted in Figure 1 (a), in human activity recognition tasks where dif-

*Corresponding author.

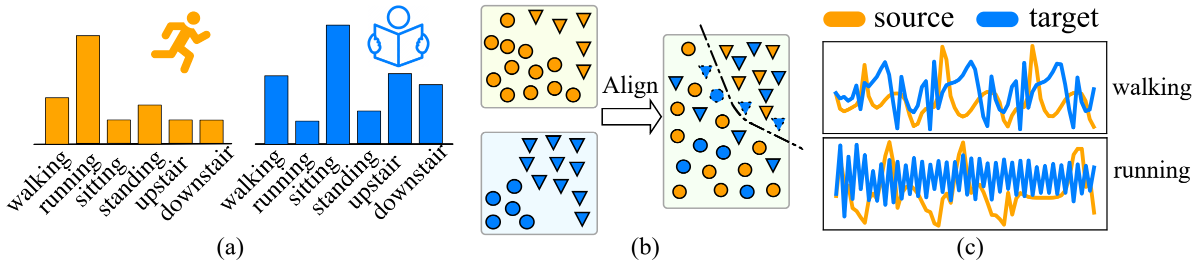

Figure 1: (a)The proportion of behavior categories of different users. (b) Align source and target domains without regard to label shift. (c) Domain shift may occur in either the time domain or the frequency domain.

ferent users are treated as different domains, significant differences in the proportions of certain categories, such as running and sitting for different individuals like sports enthusiasts and avid readers. Despite this, most existing works solely focus on aligning the feature distributions, ignoring label shift, which may degrade the performance of models on the target domain [Zhao *et al.*, 2019]. Figure 1 (b) provides an example where there are numerous circular samples but relatively few triangular samples in the source domain (orange). Conversely, in the target domain (blue), there are relatively few circular samples but many triangular samples. As a result, some triangular samples in the target domain may be incorrectly classified as belonging to the circular category due to the label shift problem. Recently, there are methods aligning feature distributions and label distributions using pseudo labels with high confidence [Tan *et al.*, 2020; Shi *et al.*, 2022]. Nevertheless, as shown in Figure 1 (b), exclusively selecting examples with high-confidence pseudo labels may overlook samples (depicted with dashed borders) at the classification boundary, resulting in a distribution shift problem. Existing studies indicate that data at the classification boundary may contribute more to the model, as they provide more information to the model, a concept referred to as *information gain* [Mukherjee and Awadallah, 2020].

Furthermore, most of these models handle wearable sensor data, which can be viewed as time series data, primarily focusing on the time domain. This focus may potentially overlook valuable information from the frequency domain, which can offer an alternative perspective for modeling time series [Cohen, 1995; He *et al.*, 2023]. In addition, the domain shift may occur in either the time domain or the frequency domain. As shown in Figure 1 (c), the frequency of patterns of source and target data in the walking category is consistent (three similar patterns are repeated in both source and target domains), and domain shift mainly occurs in the time domain. While for the running category, domain shift obviously also occurs in the frequency domain.

To address the above challenges, we propose Domain adaptation under label shift for Wearable sensor with Learnable Reweighting (DWLR). DWLR adopts a learnable reweighting method to address the problem of UDA with both feature shift and label shift for wearable sensor. Specifically, DWLR aligns the label distributions through learnable reweighting combined with maximization of not only confidence but information gain, which select data that contribute more to the model training [Mukherjee and Awadallah, 2020] and avoids the distribution shift problem caused by only select data with high-confidence pseudo labels. Then, DWLR uses adversarial learning to extract domain-invariant features to align source and target domain. Furthermore, considering that the time and frequency domains can provide different perspectives and that domain shift may occur in either domain, DWLR performs reweighting and alignment from the time and frequency domains, respectively. We summarize our contributions as follows:

- We propose a learnable reweighting method DWLR to align label distributions, which adress the problem of label shift that reduces the performance of UDA method.
- We utilize information gain to constrain the process of learnable reweighting, avoiding the distribution shift problem caused by only selecting data with high-confidence pseudo labels.
- We consider the different information and domain shift from the time and frequency domains, and perform reweighting and alignment from each domain respectively.
- Extensive experiments demonstrate the effectiveness of DWLR with an average improvement of 5.85%. [1]

2 Related Work

Unsupervised Domain Adaptation (UDA) can be viewed as a special instance of transfer learning [Patel *et al.*, 2015]. In UDA, the data distributions of the source and target domains are different, while the target task remains the same as that of the source domain, and only the source data is labeled [Bousmalis *et al.*, 2018; Bousmalis *et al.*, 2017]. Some discrepancy-based approaches aim to align the distributions by minimizing a divergence that measures the distance between two feature distributions [Long *et al.*, 2015; Shen *et al.*, 2018; Lee and Raginsky, 2018]. In recent years, Generative Adversarial Networks (GANs) [Goodfellow *et al.*, 2014] have been developed rapidly. Inspired by GANs, some approaches incorporate adversarial training [Long *et al.*, 2015; Tzeng *et al.*, 2017; Liu and Tuzel, 2016], which use a domain discriminator to confuse the features of different domains through adversarial objectives. Some UDA methods

[1]Supplementary material is available at https://github.com/JuRenGithub/DWLR.

use auxiliary self-supervised tasks, such as contrastive learning and reconstruction, to capture the common semantic information of source and target domains [Kang *et al.*, 2019; Tang *et al.*, 2021; Ghifary *et al.*, 2016].

Some UDA methods designed for cross-user wearable sensors modeling has achieved great success. [Hu *et al.*, 2023] proposes SWL-Adapt, which considers the difference among samples by calculating sample weights of each sample. Considering the wearable sensor data are almost time series data, Variational Recurrent Adversarial Deep Domain Adaptation (VRADA) [Purushotham *et al.*, 2017] and Recurrent Domain Adversarial Neural Network (R-DANN) [Tonutti *et al.*, 2019] leverage recurrent neural network as the feature extractor. [Wilson *et al.*, 2020] proposes Convolutional deep Domain Adaptation model for time series sensor data (CoDATS). [Ragab *et al.*, 2022] proposes a SeLf-supervised AutoregRessive Domain Adaptation (SLARDA) framework, which designs an auxiliary task to capture the transferable feature of source domain data. [Cai *et al.*, 2021] proposes Sparse Associative Structure Alignment (SASA) to align the sparse associative structure of the sequential sensor data. Adversarial Spectral Kernel Matching (AdvSKM) [Liu and Xue, 2021] incorporates a hybrid sepectral kernel network to improve domain matching performance. Contrastive Learning for Unsupervised Domain Adaptation (CLUDA) [Ozyurt *et al.*, 2023] introduces contrastive learning for UDA.

Recently, several works [Tan *et al.*, 2020; Jiang *et al.*, 2020; Shi *et al.*, 2022] have emerged focusing on UDA under label shift, a more practical domain adaptation setting, where feature shift and label shift both exist. [Tan *et al.*, 2020] proposes COAL to achieve collaborative alignment of feature and label distributions through pseudo labels. [Jiang *et al.*, 2020] adopts a sampling-based approach to implicitly align feature distributions to deal with UDA under label shift. [Shi *et al.*, 2022] propose a data-augmentation method which can be integrated into existing UDA models to tackle with the label shift problem. [He *et al.*, 2023] proposes alignment and correction strategy to address feature and label shift problem in universal domain adaptation.

3 Method

Problem definition. In the setting of cross-user wearable sensor UDA under label shift, two different data distributions exist, where P is from the source domain (source user) and Q is from the target domain (target user). We are given a source domain $P = \{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^{N^s}$ with N^s labeled data, where $y_i^s \in \{1, \ldots, k\}$ represents the label of data $\mathbf{x}_i^s$, and a target domain $Q = \{\mathbf{x}_i^t\}_{i=1}^{N^t}$ with N^t unlabeled data. For brevity, we omit the subscript i hereafter. And $\mathbf{x} \in \mathbb{R}^{L\times M}$ contains L time points of sensor data with M channels. Our goal is to build an end-to-end deep network that transfers the knowledge learned from P to Q and predict the label $\hat{y}^t$ of each data $\mathbf{x}^t \in Q$.

3.1 Model Overview

The framework of the proposed DWLR is shown in Figure 2. The cornerstone of DWLR is **learnable reweighting** , which aims to reweight the target data to align label distributions. Subsequently, DWLR aligns the feature distributions through adversarial training. Specifically, DWLR first trains the encoder and task classifier with labeled source data. Considering that the wearable sensor data are primarily time series data, DWLR incorporates information from both the time and frequency domains to effectively model the wearable sensor data. Then, a learnable reweighting network $WNet(\cdot)$ conducts a reweighting process on the target data based on the features extracted by the shared encoder from both the time and frequency domains, with the aim of aligning the label distributions. Furthermore, to address the issue of distribution shift resulting from selecting only data with high-confidence pseudo labels, $WNet(\cdot)$ utilizes information gain to constrain the learnable reweighting process. Finally, based on the reweighting result, DWLR utilizes the discriminator to align the feature distributions.

3.2 Backbone and Task Learning

As mentioned in Section 1, the time domain and frequency domain can provide different perspectives on wearable sensor data modeling [Cohen, 1995]. Additionally, domain shift may occur in either the time domain, the frequency domain, or both. Therefore, we have developed two encoders, $F_t(\cdot)$ and $F_f(\cdot)$, which take into account the time domain and frequency domain, respectively.

$$\mathbf{f}_t = F_t(\mathbf{x}), \mathbf{f}_f = F_f(\mathbf{x}) \tag{1}$$

As Eq 1 shows, the representation $\mathbf{f}_t$ in the time domain and the representation $\mathbf{f}_f$ in the frequency domain are obtained from the time-domain encoder $F_t(\cdot)$ and the frequency-domain encoder $F_f(\cdot)$. In this work, $F_t(\cdot)$ is implemented using convolutional neural networks (CNNs) To capture features from the frequency domain, we apply the Discrete Fourier Transform (DFT) [Winograd, 1978] to each channel of $\mathbf{x}$ with a length of L:

$$\mathbf{v}[j]_m = \sum_{l=0}^{L-1} \mathbf{x}[l]_m \cdot \exp(-i2\pi\frac{jl}{L}) \tag{2}$$

where j is the frequency index, m is the channel index, and $\mathbf{v}$ is a complex vector Then we extract the amplitude $\mathbf{a}$ and the phase $\mathbf{p}$ from $\mathbf{v}$:

$$\begin{aligned} \mathbf{a}[j]_m &= \frac{\sqrt{\text{Re}(\mathbf{v}[j]_m)^2 + \text{Im}(\mathbf{v}[j]_m)^2}}{L} \\ \mathbf{p}[j]_m &= \arctan(\frac{\text{Im}(\mathbf{v}[j]_m)}{\text{Re}(\mathbf{v}[j]_m)}) \end{aligned} \tag{3}$$

where $\text{Re}(\cdot)$ and $\text{Im}(\cdot)$ are the imagery and real part. After that we employ CNNs on $\mathbf{a}$ and $\mathbf{p}$:

$$F_f(\mathbf{x}) = \text{Conv}(\text{Conv}(\mathbf{a}) \oplus \text{Conv}(\mathbf{p})) \tag{4}$$

where $\oplus$ is concatenate operation.

To ensure the relevance of the extracted features $\mathbf{f}$ to downstream tasks, we guide $F_t(\cdot)$ and $F_f(\cdot)$ using two task classifiers, $C_t(\cdot)$ and $C_f(\cdot)$. The task classifier outputs probability scores for each class, $\mathbf{h} = C(\mathbf{f})$, where $\mathbf{h} \in \mathbb{R}^k$. Consequently, we predict the label $\hat{y}$ as the class with the highest probability score, i.e., $\hat{y} = \underset{i\in\{1,\cdots,k\}}{\text{argmax}}\ h_i$. We use the source

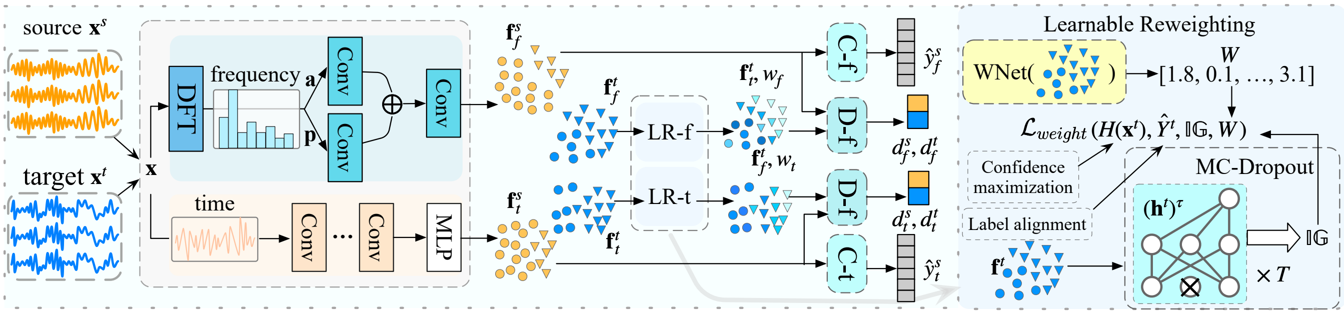

Figure 2: Framework of DWLR. It models sensor from both the time and frequency domains. DWLR aligns the label distributions through learnable reweighting, where the shades of target features indicate the weight. By utilizing information gain, the process of learnable reweighting is constrained. DWLR performs reweighting and alignment in the time domain and frequency domain, respectively.

data to train the classifier and encoder from both the time and frequency domains with a cross-entropy loss function:

$$\mathcal{L}_{task} = \mathbb{E}_{(\mathbf{x}^s, y^s) \in P} \mathcal{L}_{ce}(\mathbf{h}^s, y^s) \tag{5}$$

Note that we omit the subscripts t and f here because this process is the same for the time and frequency domains, and this convention will continue to be followed.

3.3 Learnable Reweighting

In our scenario, both feature shift and label shift exist. As illustrated in Figure 1 (b), if we directly align the feature distributions, ignoring the problem of label shift, some data from the target domain will be misclassified. To address this problem, DWLR chooses to first align the label distributions by reweighting data from the target domain. It is intuitive to assign more weight to the target data with a smaller class proportion compared to the source data and assign less weight to the target data with a larger class proportion than the source data. Since labels are unavailable in the target domain, DWLR addresses this challenge by using pseudo labels $\hat{y}^t$ obtained from the shared classifier $C(\cdot)$.

To ensure the reliability of pseudo labels, many studies have opted to select data with high-confidence pseudo labels [Tan *et al.*, 2020; Ragab *et al.*, 2022], where the predicted probability h^t_i indicates the confidence level for the i-th class. Previous research has demonstrated that high-confidence data tend to cluster around the class centers in the feature space [Liu *et al.*, 2022] However, if we only select data with high-confidence pseudo labels, there is a risk of neglecting the target data that exhibit low confidence and reside near the classification boundary. This ignored data still has feature-shift problem with the source data, which can hinder the classifier's ability to effectively distinguish them. Furthermore, existing studies have shown that data at the classification boundary may contribute more to the model [Mukherjee and Awadallah, 2020]. This data can provide more information to the model and is referred to as information gain, which can measure the reduction of uncertainty of the model when given a random variable [Cover, 1999].

To select appropriate data, we introduce a learnable reweighting network, $WNet(\cdot)$. It assigns a weight w to each target data based on its corresponding feature $\mathbf{f}^t$:

$$[w_1, w_2, \cdots, w_{N^t}] = WNet(Batch(\mathbf{f}^t)) \tag{6}$$

where $WNet(\cdot)$ is implemented by MLP and $Batch(\cdot)$ means to reweight the data one miniBatch containing N^t data at a time, as we need to normalize the weights. Also, $\sum_{i=1}^{N^t} w_i = N^t$, which means the total weight of the data remains unchanged after reweighting. Based on the reweighted results $W = [w_1, w_2, \cdots, w_{N^t}]$ and pseudo labels $\hat{Y}^t = [\hat{y}^t_1, \hat{y}^t_2, \cdots, \hat{y}^t_{N^t}]$, we can obtain the reweighted label distribution of the target domain as $\hat{Q}'_y = \{(w_i, \hat{y}^t_i)\}_{i=1}^{N^t}$. Since the label distribution of the source domain P_y is known to us, we opt to minimize the KL-divergence between the distributions:

$$\mathcal{L}_{label} = \mathbb{KL}(P_y || \hat{Q}'_y) \tag{7}$$

In addition to aligning the label distributions, it is also important to give larger weight to data that provides substantial information gain. However, it is difficult to compute the information gain directly for a given $\mathbf{f}$. [Gal *et al.*, 2017] proposes a method to compute the information gain approximately using Monte-Carlo Dropout (MC-Dropout), which leverages dropout, a regularization technique in deep learning models, to provide an estimation of the information gain:

$$\begin{aligned} \mathbb{IG}(\mathbf{f}) = &- \sum_{i=1}^{k} (\frac{1}{T} \sum_{\tau=1}^{T} h^\tau_i) \log (\frac{1}{T} \sum_{\tau=1}^{T} h^\tau_i) \\ &+ \frac{1}{T} \sum_{i=1}^{k} \sum_{\tau=1}^{T} h^\tau_i \log h^\tau_i \end{aligned} \tag{8}$$

where h^τ_i is the τ-th predicted probability of representation $\mathbf{f}$ for the i-th class out of a total of T predictions. We use the classifier $C(\cdot)$ with dropout to implement MC-Dropout. Therefore, our objective is to maximize the information gain of reweighted data.

$$\mathcal{L}_{IG} = -\frac{1}{N^t} \sum_{i=1}^{N^t} \mathbb{IG}(\mathbf{f}^t_i) \cdot w_i \tag{9}$$

As the data with large information gain tend to have lower confidence, their pseudo labels could be wrong. To address this problem, we give more weight to data with high-confidence pseudo labels by maximizing confidence.

$$\mathcal{L}_{conf} = -\frac{1}{N^t} \sum_{i=1}^{N^t} \hat{h}^t_i \cdot w_i \tag{10}$$

Where we use the predicted probability $\hat{h}_i^t \cdot w_i$ to quantify the confidence of $\mathbf{f}_i^t$. The overall loss function for $WNet(\cdot)$ is

$$\mathcal{L}_{weight} = \mathcal{L}_{label} + \mathcal{L}_{IG} + \mathcal{L}_{conf} \tag{11}$$

3.4 Adversarial Training

We use a discriminator $D(\cdot)$ to align feature distributions by adversarial training, which is commonly used in some existing works [Long *et al.*, 2015; Tzeng *et al.*, 2017; Liu and Tuzel, 2016]. Specifically, the goal of the discriminator is to distinguish whether the feature $\mathbf{f}$ is extracted from the source or target domain. We use cross entropy loss to train the discriminator. For the data $\mathbf{x}^s$ from the source domain, the cross-entropy can be calculated by $\log \mathbf{d}^s$, where $\mathbf{d}^s$ is the output of $D(\mathbf{f}^s)$. While for the target domain, we need to multiply $\log(1 - \mathbf{d}^t)$ by the obtained weight w. Therefore, the objective function of the discriminator is

$$\mathcal{L}_{dis} = -(\frac{1}{N^s}\sum_{i=1}^{N^s} \log \mathbf{d}_i^s + \frac{1}{N^t}\sum_{i=1}^{N^t} w_i \log(1 - \mathbf{d}_i^t)) \tag{12}$$

Meanwhile, it is desirable for the representation $\mathbf{f}$ extracted by the encoder to be as domain-invariant as possible. Therefore, we can employ the opposite objective to $\mathcal{L}_{dis}$ to guide the encoder training, which is called adversarial training:

$$\mathcal{L}_{adv} = -(\frac{1}{N^s}\sum_{i=1}^{N^s} \log(1 - \mathbf{d}_i^s) + \frac{1}{N^t}\sum_{i=1}^{N^t} w_i \log \mathbf{d}_i^t) \tag{13}$$

$\mathcal{L}_{dis}$ and $\mathcal{L}_{adv}$ can be implemented by a gradient reversal layer[Ganin *et al.*, 2015].

3.5 Model Learning

Given wearable sensor data from the source and target domains, we first pretrain the encoder and classifier by $\mathcal{L}_{task}$ with the labeled source data. Then, $WNet(\cdot)$, $F(\cdot)$, $C(\cdot)$, and $D(\cdot)$ are trained iteratively through two stages. In the first stage, the $WNet(\cdot)$, $F(\cdot)$ and $C(\cdot)$ are trained by $\mathcal{L}_{net}$.

$$\mathcal{L}_{net} = \mathcal{L}_{task} + \alpha\mathcal{L}_{adv} + \beta\mathcal{L}_{weight} \tag{14}$$

During learnable reweighting, $C(\cdot)$ is used for obtaining pseudo-labels and calculating information gain, but the gradient of $\mathcal{L}_{weight}$ will not be propagated to $C(\cdot)$. In the second stage, the $D(\cdot)$ is trained by $\mathcal{L}_{dis}$. Note that the above two stages are carried out separately in time and frequency domain, and $F(\cdot)$, $C(\cdot)$, $WNet(\cdot)$, $D(\cdot)$ are not shared in time and frequency domain. Additionally, when making predictions on the target domain, we combine prediction scores from the time and frequency domain by adding them together for the final prediction.

4 Experiment

In this section, we conduct extensive experiments to verify the effectiveness of our proposed model.

4.1 Experiment Setup

Dataset. In our experiments, we use three real-world wearable sensor datasets (*i.e.*, WISDM [Kwapisz *et al.*, 2011], UCIHAR [Anguita *et al.*, 2013] and HHAR [Stisen *et al.*, 2015]). And we also conduct experiment on a human sensor dataset, SleepEDF [Goldberger *et al.*, 2000]. For each dataset, we treat different users as different domains. In addition, we randomly select multiple pairs of users as the source domain and the target domain. More details and category distribution can be found in the supplementary material.

Baseline. In this experiment, we compare our method with three types of baseline methods. The first type of approach is a method designed for wearable sensor modeling without domain adaptation, Distribution-Embedded Deep Neural Network (DDNN) [Qian *et al.*, 2019]. The second type of approach is UDA method suitable for wearable sensor data, including SWL-Adapt [Hu *et al.*, 2023], RDANN [Tonutti *et al.*, 2019], CoDATS [Wilson *et al.*, 2020], VRADA [Purushotham *et al.*, 2017], SLARDA [Ragab *et al.*, 2022], SASA [Cai *et al.*, 2021], AdvSKM [Liu and Xue, 2021] and CLUDA [Ozyurt *et al.*, 2023]. The third type of approach is designed for UDA under label shift including COAL [Tan *et al.*, 2020], PAT [Shi *et al.*, 2022], RAINCOAT [He *et al.*, 2023], and Implicit Class-Conditioned Domain Alignment [Jiang *et al.*, 2020], which is abbreviated as ImMMD as it uses MMD as discrepancy measure.

Implement detail. Considering that the datasets we use are not balanced, we adopt the macro AUC-ROC score to measure the model performance, since it does not take label imbalance into account. For each baseline and our model, we repeat the experiment 10 times and take the average value as the final result. For our model, the α is set to 0.5 and β is set to 2.0. For WISDM, HHAR and UCIHAR the feature extractor and classifier are pretrained for 10 epochs. For SleepEDF, the pretrain epoch is set to 50, and we also adjust the baseline pretrain epoch if pretrain is required. The batch size is set to 256. We employ the AdamW optimizer with a learning rate of 1e-3 and a weight decay of 1e-3.

4.2 Performance Comparison

Table 1 provides a summary of the performance of DWLR with other baseline models on three wearable sensor datasets, WISDM, UCIHAR, and HHAR, and a human sensor dataset, SleepEDF. One of the core aspects of the SASA method is to capture the relationships between variables. Since SleepEDF is a univariate dataset, SASA has not been tested on SleepEDF. Overall, our proposed DWLR outperforms the other baseline models by an average improvement of **5.85%** across the four datasets. It is important to note that DDNN, designed for wearable sensors without domain adaptation, serves as a lower bound for positive transfer. And DDNN performs the worst on most of the experimental results. We attribute this to the possibility of DDNN overfitting the source data, resulting in worse performance in the target domain. This highlights the presence of a domain shift between the source and target domains. However, SLARDA and SASA without considering label shift problem, perform worse than the source-only baseline in some experiments. Additionally,

s→t	DDNN	RDANN	VRADA	CoDATs	SLARDA	SASA	AdvSKM	SWL-Adapt	CLUDA	COAL	PAT	ImMMD	RAINCOAT	DWLR
							WISDM							
20→6	83.02	87.37	95.05	94.29	84.70	84.81	82.31	90.10	86.37	91.11	86.89	83.56	90.55	**95.57**
20→12	73.76	68.52	76.43	86.71	79.42	71.95	90.95	87.57	76.13	85.45	59.15	92.17	95.61	**97.09**
24→21	53.06	83.63	85.55	91.40	86.79	63.05	69.80	90.70	83.16	88.99	68.08	88.75	72.47	**94.36**
5→18	68.00	87.30	86.70	83.03	72.20	85.29	78.11	82.95	79.75	88.69	63.70	79.75	89.36	**96.07**
8→24	74.43	77.87	77.82	78.12	68.32	72.04	65.54	83.94	75.31	82.89	63.64	69.74	84.27	**93.83**
13→6	71.34	87.80	92.51	87.78	82.20	60.73	72.54	81.11	81.18	91.79	79.28	79.17	87.80	**95.52**
							UCIHAR							
3→10	57.17	60.94	68.82	72.06	52.67	57.58	67.07	52.15	61.58	72.09	59.60	65.46	71.81	**79.06**
28→12	63.02	76.04	75.40	83.10	71.33	65.86	83.96	89.26	81.91	79.11	67.41	66.93	76.72	**92.16**
8→20	66.86	70.96	81.73	80.41	71.00	70.71	79.99	87.73	83.78	83.94	75.82	81.96	82.91	**94.28**
4→19	68.03	85.08	87.79	89.96	89.45	69.77	89.32	68.85	88.39	**90.67**	76.69	86.09	88.25	82.68
27→7	63.95	98.04	98.49	93.69	96.30	89.32	82.43	97.54	97.75	99.19	80.71	99.49	83.55	**99.92**
25→16	61.84	79.69	74.48	83.12	65.70	69.86	83.26	80.27	84.92	86.34	61.12	82.01	75.58	**93.20**
16→10	61.52	68.90	77.21	77.17	75.35	59.75	86.72	64.64	76.49	66.19	63.60	71.41	**92.50**	87.15
							HHAR							
a→c	58.85	69.09	70.60	83.26	61.78	70.39	68.36	68.37	86.06	77.95	66.20	66.48	70.00	**89.29**
e→b	55.89	82.21	71.38	68.06	66.04	54.27	73.72	80.44	**91.33**	86.81	74.19	75.88	74.02	90.02
d→g	76.96	74.25	74.24	75.57	88.76	69.61	89.03	93.76	90.97	87.94	79.83	88.91	80.90	**94.33**
e→d	70.47	88.93	85.44	93.01	91.58	57.77	79.19	93.73	95.81	96.16	78.66	89.58	77.76	**97.86**
h→f	83.12	87.30	85.05	**94.58**	65.57	65.73	75.46	84.10	85.56	86.48	68.97	88.49	83.16	87.15
i→g	59.09	71.29	66.80	73.95	87.28	66.64	89.24	88.66	89.69	87.38	78.48	73.87	73.69	**94.19**
h→i	63.99	71.25	64.56	60.86	66.15	71.98	71.44	78.60	80.91	82.67	65.48	70.56	76.63	**89.60**
							SleepEDF							
4→8	70.86	65.73	52.74	78.94	50.85	-	68.02	79.50	58.84	79.50	55.58	73.44	62.28	**81.19**
10→17	67.72	70.30	50.75	77.07	48.73	-	60.19	70.87	53.29	74.17	57.50	66.85	62.43	**82.75**
7→13	75.89	87.26	77.09	**90.50**	52.82	-	83.97	81.48	64.63	86.40	61.36	82.72	68.88	89.64
3→12	69.06	53.96	50.83	77.67	51.48	-	65.65	74.03	57.05	78.02	56.54	73.83	60.79	**79.36**
15→1	75.21	82.42	67.77	**86.38**	47.29	-	75.97	77.32	60.64	81.58	54.44	77.18	62.86	82.73
19→0	71.92	85.14	62.92	87.30	56.86	-	75.14	79.62	61.24	84.23	60.46	81.71	61.42	**88.84**
Average	67.88	77.74	75.31	82.60	70.41	68.85	77.21	81.05	78.18	84.45	67.86	79.00	77.16	**90.30**

Table 1: AUC-ROC on WISDM, UCIHAR, HHAR and SleepEDF. Best in bold, second underlined.

these two baselines are both newer methods and are inferior to some other older baselines in our experiment. We think this is because these newer methods can align the feature distributions better, which yet leads to worse performance when the label shift cannot be ignored. The performance of baseline PAT and ImMMD, designed for UDA under label shift, is relatively poor. This could be attributed to their design for computer vision tasks, while wearable sensor datasets consist of sequential data with temporal dependencies and other characteristics. The RAINCOAT primarily focuses on addressing the label shift problem in universal domain adaptation, albeit our experimental setup differs. Therefore, its performance is not as good. For the experimental results where DWLR does not perform well, particularly on users 3 to 10 in UCIHAR and users 3 to 12 in SleepEDF, we observed inadequate accuracy in pseudo-labeling. The accuracy of pseudo-labeling directly impacts the alignment of label distribution, leading to suboptimal performance, with an AUC-ROC score of only 79%. For the human sensor dataset SleepEDF, despite some declines in the performance of certain baselines such as CLUDA, PAT, and RAINCOAT, the performance of DWLR still remains optimal.

4.3 Ablation Study

Method	WISDM	UCIHAR	HHAR
w/o freq	90.31	87.31	86.73
w/o time	92.14	87.76	86.24
merge	94.49	88.96	87.31
DWLR	95.41	89.78	91.78

Table 2: Ablation study of time and frequency domain.

DWLR performs learnable reweighting and alignment separately in the time domain and frequency domain. To investigate the impact of the time and frequency domain on the performance of DWLR, we removed the time domain part (w/o time) and the frequency domain part (w/o freq). Additionally, we merged the representations of the time domain $\mathbf{f}_t$ frequency domain $\mathbf{f}_f$, and applied learnable reweighting and alignment to the merged representation, exploring the effects of not separating the time and frequency domain. As shown in Table 2, the performance decreases the most when the frequency domain component is removed. Merging the

	Dataset	w/o $\mathcal{L}_{label}$	w/o $\mathcal{L}_{IG}$	w/o $\mathcal{L}_{conf}$	DWLR
	20→6	91.94	92.36	92.49	95.57
W	5→18	89.37	93.45	92.47	96.07
	20→12	92.31	93.08	94.12	97.09
	16→10	79.92	83.34	84.01	87.15
U	3→10	71.12	73.87	72.33	79.06
	28→12	82.59	85.16	84.76	92.16
	e→b	86.37	87.43	88.01	90.02
H	i→g	89.98	91.81	90.45	94.19
	h→i	81.31	84.64	86.83	89.06

Table 3: Ablation study of learable reweighting.

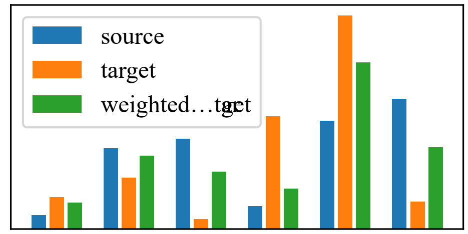

Figure 4: Source, target and weighted target label distributions.

frequency domain and the time domain leads to a certain degree of performance degradation. This indicates that the frequency domain information is crucial for the UDA of wearable sensors. Modeling the time and frequency domain separately enables the capture of information from different perspectives, enhancing the overall understanding of the sensor data.

As mentioned in the Method section, the cornerstone of DWLR is learnable reweighting, of which the objective function $\mathcal{L}_{weight}$ consists of three parts: $\mathcal{L}_{label}$, $\mathcal{L}_{IG}$, and $\mathcal{L}_{conf}$. Thus, we conducted an ablation study to demonstrate the effectiveness of each part on three wearable sensor datasets, where W represents the WISDM dataset, U represents the UCIHAR dataset, and H represents the HHAR dataset. The results in Table 3 show that the alignment of label distribution plays the most important role in DWLR, which is also consistent with our motivation for designing the model. Moreover, maximizing information gain and confidence also indeed help DWLR achieving better performance.

4.4 Analysis of Learnable Reweighting

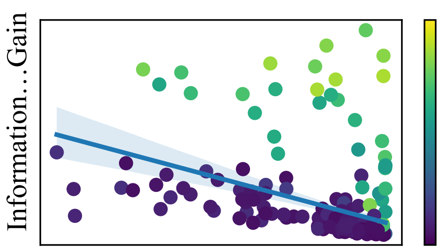

Figure 3: Information gain is negatively correlated with confidence (the blue regression line). Lighter colors indicate larger weight.

In the learnable reweighting component, we constrain the reweighting process with information gain and confidence. To investigate the validity of these constraints, we examine the relationship between the weight and information gain as well as confidence on target data. In the Figure 3, the horizontal axis represents confidence, the vertical axis represents information gain, the color depth indicates the weight on the target data, where lighter colors indicate larger weights. It shows that data with higher information gain and confidence have higher weights, supporting the description in the Method section. Moreover, a negative correlation between information gain and confidence suggests that only considering data with high-confidence pseudo labels may lead to the neglect of informative data.

Furthermore, to verify the effective alignment of learnable reweighting on label distributions, we present in Figure 4 the label distributions of the source domain and the target domain before and after reweighting. These distributions are based on a randomly selected pair of users from the UCIHAR dataset. Figure 4 illustrates that after reweighting, while the label distribution of the target domain does not perfectly match that of the source domain, it is much more similar to the source domain label distribution compared to the original distribution.

4.5 Model Efficiency

Method	WISDM	UCIHAR	HHAR	Avg
w/o $WNet(\cdot)$	4.90	4.51	10.57	6.66
DWLR	6.14	5.37	13.22	8.24

Table 4: Time costs (second) per epoch of with and w/o $WNet(\cdot)$.

We use MC-Dropout to compute the information gain approximately in our learnable reweighting component, which may increase the time complexity of DWLR. To evaluate this, we conduct experiments on three wearable datasets with and without $WNet(\cdot)$, and compare their time costs. As Table 4 shows, without $WNet(\cdot)$, each epoch costs an average of 6.66s. And with $WNet(\cdot)$, the time cost is 8.24s, which just adds 23.8% extra time. This means the learnable sampling module does not significantly increase the training time. More details are in the supplementary material.

5 Conclusion

Our paper proposes a learnable reweighting-based approach DWLR for wearable sensor UDA under label shift. DWLR aligns label distributions and gives large weight to data with high information gain and confidence. Considering that wearable sensor data carries distinct perspectives in both the time and frequency domains, as well as the possibility of domain shifts occurring in either domain, the DWLR conducts reweighting and alignment separately in the time and frequency domains.

Acknowledgments

This work is supported by National Natural Science Foundation of China (No. 62322606, No. 62441605).

We want to thank Jingbo Shang from the University of California, San Diego for his invaluable guidance.

References

[Anguita *et al.*, 2013] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra Perez, and Jorge Luis Reyes Ortiz. A public domain dataset for human activity recognition using smartphones. In *Proceedings of the 21th international European symposium on artificial neural networks, computational intelligence and machine learning*, pages 437–442, 2013.

[Bai *et al.*, 2018] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

[Bousmalis *et al.*, 2017] Konstantinos Bousmalis, Nathan Silberman, David Dohan, Dumitru Erhan, and Dilip Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3722–3731, 2017.

[Bousmalis *et al.*, 2018] Konstantinos Bousmalis, Alex Irpan, Paul Wohlhart, Yunfei Bai, Matthew Kelcey, Mrinal Kalakrishnan, Laura Downs, Julian Ibarz, Peter Pastor, Kurt Konolige, et al. Using simulation and domain adaptation to improve efficiency of deep robotic grasping. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 4243–4250. IEEE, 2018.

[Cai *et al.*, 2021] Ruichu Cai, Jiawei Chen, Zijian Li, Wei Chen, Keli Zhang, Junjian Ye, Zhuozhang Li, Xiaoyan Yang, and Zhenjie Zhang. Time series domain adaptation via sparse associative structure alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 6859–6867, 2021.

[Chung *et al.*, 2015] Junyoung Chung, Kyle Kastner, Laurent Dinh, Kratarth Goel, Aaron C Courville, and Yoshua Bengio. A recurrent latent variable model for sequential data. *Advances in neural information processing systems*, 28, 2015.

[Cohen, 1995] Leon Cohen. *Time-frequency analysis*, volume 778. Prentice hall New Jersey, 1995.

[Cover, 1999] Thomas M Cover. *Elements of information theory*. John Wiley & Sons, 1999.

[Gal *et al.*, 2017] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. In *International Conference on Machine Learning*, pages 1183–1192. PMLR, 2017.

[Ganin *et al.*, 2015] Yaroslav Ganin, E. Ustinova, Hana Ajakan, Pascal Germain, H. Larochelle, François Laviolette, Mario Marchand, and Victor S. Lempitsky. Domain-adversarial training of neural networks. *ArXiv*, abs/1505.07818, 2015.

[Ghifary *et al.*, 2016] Muhammad Ghifary, W Bastiaan Kleijn, Mengjie Zhang, David Balduzzi, and Wen Li. Deep reconstruction-classification networks for unsupervised domain adaptation. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 597–613. Springer, 2016.

[Goldberger *et al.*, 2000] Ary L Goldberger, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *circulation*, 101(23):e215–e220, 2000.

[Goodfellow *et al.*, 2014] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Y. Bengio. Generative adversarial nets. In *Neural Information Processing Systems*, 2014.

[He *et al.*, 2023] Huan He, Owen Queen, Teddy Koker, Consuelo Cuevas, Theodoros Tsiligkaridis, and Marinka Zitnik. Domain adaptation for time series under feature and label shifts. In *International Conference on Machine Learning*, 2023.

[Hu *et al.*, 2023] Rong Hu, Ling Chen, Shenghuan Miao, and Xing Tang. Swl-adapt: An unsupervised domain adaptation model with sample weight learning for cross-user wearable human activity recognition. In *Proceedings of the AAAI Conference on artificial intelligence*, volume 37, pages 6012–6020, 2023.

[Jiang *et al.*, 2020] Xiang Jiang, Qicheng Lao, Stan Matwin, and Mohammad Havaei. Implicit class-conditioned domain alignment for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 4816–4827. PMLR, 2020.

[Kang *et al.*, 2019] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4893–4902, 2019.

[Kwapisz *et al.*, 2011] Jennifer R Kwapisz, Gary M Weiss, and Samuel A Moore. Activity recognition using cell phone accelerometers. *ACM SigKDD Explorations Newsletter*, 12(2):74–82, 2011.

[Kwon *et al.*, 2021] Shinjae Kwon, Hojoong Kim, and Woon-Hong Yeo. Recent advances in wearable sensors and portable electronics for sleep monitoring. *Iscience*, 24(5), 2021.

[Lee and Raginsky, 2018] Jaeho Lee and Maxim Raginsky. Minimax statistical learning with wasserstein distances. *Advances in Neural Information Processing Systems*, 31, 2018.

[Liu and Tuzel, 2016] Ming-Yu Liu and Oncel Tuzel. Coupled generative adversarial networks. *Advances in neural information processing systems*, 29, 2016.

[Liu and Xue, 2021] Qiao Liu and Hui Xue. Adversarial spectral kernel matching for unsupervised time series domain adaptation. In *IJCAI*, pages 2744–2750, 2021.

[Liu *et al.*, 2022] Hongrui Liu, Binbin Hu, Xiao Wang, Chuan Shi, Zhiqiang Zhang, and Jun Zhou. Confidence may cheat: Self-training on graph neural networks under distribution shift. In *Proceedings of the ACM Web Conference 2022*, pages 1248–1258, 2022.

[Long *et al.*, 2015] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In *International conference on machine learning*, pages 97–105. PMLR, 2015.

[Longoni *et al.*, 2018] Marco Longoni, Diego Carrera, Beatrice Rossi, Pasqualina Fragneto, Marco Pessione, Giacomo Boracchi, et al. A wearable device for online and long-term ecg monitoring. In *IJCAI*, pages 5838–5840, 2018.

[McClure *et al.*, 2020] Kristin McClure, Brett Erdreich, Jason HT Bates, Ryan S McGinnis, Axel Masquelin, and Safwan Wshah. Classification and detection of breathing patterns with wearable sensors and deep learning. *Sensors*, 20(22):6481, 2020.

[Mukherjee and Awadallah, 2020] Subhabrata Mukherjee and Ahmed Awadallah. Uncertainty-aware self-training for few-shot text classification. *Advances in Neural Information Processing Systems*, 33:21199–21212, 2020.

[Ozyurt *et al.*, 2023] Yilmazcan Ozyurt, Stefan Feuerriegel, and Ce Zhang. Contrastive learning for unsupervised domain adaptation of time series. *ICLR*, 2023.

[Patel *et al.*, 2015] Vishal M Patel, Raghuraman Gopalan, Ruonan Li, and Rama Chellappa. Visual domain adaptation: A survey of recent advances. *IEEE signal processing magazine*, 32(3):53–69, 2015.

[Purushotham *et al.*, 2017] Sanjay Purushotham, Wilka Carvalho, Tanachat Nilanon, and Yan Liu. Variational recurrent adversarial deep domain adaptation. In *International Conference on Learning Representations*, 2017.

[Qian *et al.*, 2019] Hangwei Qian, Sinno Jialin Pan, Bingshui Da, and Chunyan Miao. A novel distribution-embedded neural network for sensor-based activity recognition. In *IJCAI*, volume 2019, pages 5614–5620, 2019.

[Ragab *et al.*, 2022] Mohamed Ragab, Emadeldeen Eldele, Zhenghua Chen, Min Wu, Chee-Keong Kwoh, and Xiaoli Li. Self-supervised autoregressive domain adaptation for time series data. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[Saito *et al.*, 2017] Kuniaki Saito, Yoshitaka Ushiku, and Tatsuya Harada. Asymmetric tri-training for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 2988–2997. PMLR, 2017.

[Shen *et al.*, 2018] Jian Shen, Yanru Qu, Weinan Zhang, and Yong Yu. Wasserstein distance guided representation learning for domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[Shi *et al.*, 2022] Weili Shi, Ronghang Zhu, and Sheng Li. Pairwise adversarial training for unsupervised class-imbalanced domain adaptation. *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022.

[Stisen *et al.*, 2015] Allan Stisen, Henrik Blunck, Sourav Bhattacharya, Thor Siiger Prentow, Mikkel Baun Kjærgaard, Anind Dey, Tobias Sonne, and Mads Møller Jensen. Smart devices are different: Assessing and mitigatingmobile sensing heterogeneities for activity recognition. In *Proceedings of the 13th ACM conference on embedded networked sensor systems*, pages 127–140, 2015.

[Sun *et al.*, 2016] Baochen Sun, Jiashi Feng, and Kate Saenko. Return of frustratingly easy domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 2016.

[Tan *et al.*, 2020] Shuhan Tan, Xingchao Peng, and Kate Saenko. Class-imbalanced domain adaptation: An empirical odyssey. In *European Conference on Computer Vision*, pages 585–602. Springer, 2020.

[Tang *et al.*, 2021] Shixiang Tang, Peng Su, Dapeng Chen, and Wanli Ouyang. Gradient regularized contrastive learning for continual domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 2665–2673, 2021.

[Tonutti *et al.*, 2019] Michele Tonutti, Emanuele Ruffaldi, Alessandro Cattaneo, and Carlo Alberto Avizzano. Robust and subject-independent driving manoeuvre anticipation through domain-adversarial recurrent neural networks. *Robotics and Autonomous Systems*, 115:162–173, 2019.

[Tzeng *et al.*, 2017] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7167–7176, 2017.

[Wilson *et al.*, 2020] Garrett Wilson, Janardhan Rao Doppa, and Diane J Cook. Multi-source deep domain adaptation with weak supervision for time-series sensor data. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1768–1778, 2020.

[Winograd, 1978] Shmuel Winograd. On computing the discrete fourier transform. *Mathematics of computation*, 32(141):175–199, 1978.

[Zhao *et al.*, 2019] Han Zhao, Remi Tachet Des Combes, Kun Zhang, and Geoffrey Gordon. On learning invariant representations for domain adaptation. In *International Conference on Machine Learning*, pages 7523–7532. PMLR, 2019.