A systematic study of unsupervised domain adaptation for robust human-activity recognition

Chang, Youngjae; Mathur, Akhil; Isopoussu, Anton; Song, Junehwa; Kawsar, Fahim

DOI
10.1145/3380985

Publication date
2020

Document Version
Final published version

Published in
Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies

A Systematic Study of Unsupervised Domain Adaptation for Robust Human-Activity Recognition

YOUNGJAE CHANG*†, KAIST, South Korea
AKHIL MATHUR*, University College London and Nokia Bell Labs, United Kingdom
ANTON ISOPOUSSU, Nokia Bell Labs, United Kingdom
JUNEHWA SONG, KAIST, South Korea
FAHIM KAWSAR, Nokia Bell Labs, United Kingdom and TU Delft, Netherlands

Wearable sensors are increasingly becoming the primary interface for monitoring human activities. However, in order to scale human activity recognition (HAR) using wearable sensors to million of users and devices, it is imperative that HAR computational models are robust against real-world heterogeneity in inertial sensor data. In this paper, we study the problem of wearing diversity which pertains to the placement of the wearable sensor on the human body, and demonstrate that even state-of-the-art deep learning models are not robust against these factors. The core contribution of the paper lies in presenting a first-of-its-kind in-depth study of unsupervised domain adaptation (UDA) algorithms in the context of wearing diversity – we develop and evaluate three adaptation techniques on four HAR datasets to evaluate their relative performance towards addressing the issue of wearing diversity. More importantly, we also do a careful analysis to learn the downsides of each UDA algorithm and uncover several implicit data-related assumptions without which these algorithms suffer a major degradation in accuracy. Taken together, our experimental findings caution against using UDA as a silver bullet for adapting HAR models to new domains, and serve as practical guidelines for HAR practitioners as well as pave the way for future research on domain adaptation in HAR.

CCS Concepts: • **Human-centered computing** → **Ubiquitous and mobile computing**; • **Computing methodologies** → **Machine learning**.

Additional Key Words and Phrases: Human Activity Recognition, Unsupervised Domain Adaptation, Wearing Diversity

ACM Reference Format:
Youngjae Chang, Akhil Mathur, Anton Isopoussu, Junehwa Song, and Fahim Kawsar. 2020. A Systematic Study of Unsupervised Domain Adaptation for Robust Human-Activity Recognition. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 1, Article 39 (March 2020), 30 pages. https://doi.org/10.1145/3380985

*Both authors contributed equally to this research.
†This work was done in part when the author was on an internship at Nokia Bell Labs, UK.

Authors' addresses: Youngjae Chang, yjchang@nclab.kaist.ac.kr, KAIST, South Korea; Akhil Mathur, akhil.mathur@nokia-bell-labs.com, University College London, Nokia Bell Labs, United Kingdom; Anton Isopoussu, anton.isopoussu@gmail.com, Nokia Bell Labs, United Kingdom; Junehwa Song, junesong@nclab.kaist.ac.kr, KAIST, South Korea; Fahim Kawsar, fahim.kawsar@nokia-bell-labs.com, Nokia Bell Labs, United Kingdom and TU Delft, Netherlands.

2474-9567/2020/3-ART39 $15.00
https://doi.org/10.1145/3380985

1 INTRODUCTION

Wearable devices equipped with inertial sensors have become a key driver for sensing user activities and context such as physical activities [17, 24, 35], transportation mode [11, 30], body gestures [47] and even eating episodes [1]. These advancements can be attributed to the years of research in the area of human-activity recognition (HAR) [4, 8, 15, 31] as well as the advances in computational models that process raw inertial data to infer human activities. More recently, deep learning-based approaches [17, 35] have been proposed to infer human activities from inertial sensors, which outperform the performance of traditional shallow classifiers.

Despite these advances, challenges remain to make HAR models robust against a number of diversities exhibited in the real-world. One of the most prominent forms of variability in wearable sensor data is caused by the positioning of the wearable sensors on the human body [24]. For example, inertial sensors can be worn on the wrist (in a smartwatch), on the ear (in an earbud), or be placed inside a user's trouser and shirt pockets (smartphones). More critically, the sensor placement is not always static and may change rapidly during the course of an activity based on users' preference – for instance, a smartphone may move from a pocket to a user's hand, and then to the ear and then go in a handbag – all while the user is engaged in a certain physical activity. As such, it is critical that HAR models are robust to *wearing diversity* and can provide accurate predictions across multiple wearing positions.

From a machine learning perspective, the challenge of wearing diversity in HAR models can be formulated as a *domain shift* problem. A fundamental assumption in supervised learning algorithms that are used to train HAR models is that the distribution of the data remains the same during training and testing stages. However, the variability induced in the data by different wearing positions causes this assumption to be violated, in that they lead to a discrepancy (or shift) between the training and test data distributions, a phenomenon known as *domain shift*. Therefore, if a classifier is trained on data collected from one domain (e.g., a body position such as wrist) and tested on another domain (a new body position), it is likely to perform poorly.

Indeed, a straightforward way to address this challenge is to collect *labeled* training data from all possible body positions in which a wearable device can be worn, and thereafter either train an HAR model per body position [41], or train a generic model which can work across all body positions [5]. However, collecting large amounts of *labeled* human-activity training data for different body positions is both expensive and time-consuming, thereby limiting the practicality of these solutions. Ideally, we desire a solution which can generalize an HAR classifier to a new wearing position using zero or minimal amount of labeled data from that position.

To this end, deep *unsupervised domain adaptation* (UDA) has emerged as a promising technique to adapt deep learning models across domains using only unlabeled data from the target domain. At a high-level, the idea behind UDA is as follows: given a pre-trained classifier for a source domain and an unlabeled dataset from a target domain, how can we adapt the weights of the source model such that it shows better performance in the target domain. While UDA has been an active area of research in the computer vision community, its application to human activity recognition is currently in a nascent stage [3, 21] and to the best of our knowledge, there is no work which systematically studies the applicability of various UDA techniques proposed in the machine learning literature to the problem of wearing diversity in HAR. We argue that as UbiComp and HAR researchers, it is imperative that we carefully analyze the pros, cons, assumptions, and constraints of technique(s) proposed in the machine learning literature, before they are applied to the domain of HAR.

In this paper, we take the first step towards an in-depth study of unsupervised domain adaptation to tackle the problem of wearing diversity in wearables. Specifically, our analysis revolves around four pillars:

- **Choice of UDA algorithms.** Broadly, UDA algorithms that aim to adapt a classifier from a source domain to a target domain can be divided into two categories: i) *Feature Matching* and ii) *Confusion Maximization*. Both algorithms focus on aligning the feature spaces of source and target domains – Feature Matching does it explicitly by minimizing a distance metric in the feature space and *Confusion Maximization* uses

adversarial learning to achieve this objective. While variants of *Feature Matching* have been developed for HAR [21], to the best of our knowledge *Confusion Maximization* has not been applied to the problem of wearing diversity. In this work, we develop two UDA algorithms based on *Feature Matching* and *Confusion Maximization* to address wearing diversity, and compare their performance in the context of HAR tasks. Our key objectives are to understand if a particular class of algorithm is better suited to the problem of wearing diversity, and to uncover the assumptions and downsides of each algorithm. Further, we develop a *Data Augmentation* baseline to compare against the UDA algorithms.

- **Choice of Body Positions.** Prior work has primarily studied the applicability of UDA for two body positions, namely thigh (smartphone) and wrist (smartwatch). However wearable devices are not restricted to just these body positions and can be worn in many, e.g., ears [19], fingers [10], neck [13], etc. We provide the first-ever in-depth analysis of the performance of UDA algorithms across a range of 20 body positions and 5 body position groups from 4 different datasets, by showing results for total 108 adaptation scenarios.
- **Effect of dataset properties.** Indeed, the performance of data-driven UDA algorithms heavily depends on how the underlying datasets are structured. We investigate the effect of class mismatch between source and target domains on UDA performance, as well as the amount of labeled and unlabeled data needed from different body positions to obtain optimal adaptation performance.
- **Evaluation Metrics for UDA.** Finally, our work seeks to uncover the most appropriate metrics to compare the performance of various UDA algorithms in the context of HAR models. To this end, in § 4.2 we propose three evaluation metrics – *adaptability, persistence, and generalizability* for UDA methods.

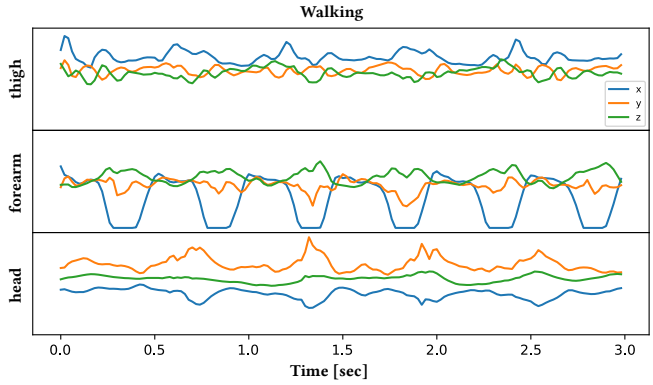
To the best of our knowledge, there is no work which has developed and analyzed UDA algorithms in the context of wearing diversity from the above four lenses. As such, this work makes a novel contribution to the activity recognition literature and aims to provide clear guidelines for ubicomp practitioners or HAR model developers to use UDA techniques in practice. Our key contributions are as follows:

- Borrowing on the theoretical foundations of UDA from machine learning literature, we developed three adaptation solutions in the context of wearing diversity.
- We conduct a systematic study to uncover the performance of state-of-the-art UDA techniques on four HAR datasets collected from various body positions.
- We propose Adaptability, Persistence and Generalizability as performance metrics to evaluate a UDA technique's performance on wearing diversity problem.
- Through a set of more than 100 experiments on various body position pairs, we derive practical guidelines for UbiComp and HAR practitioners on how to train more accurate and robust HAR models.

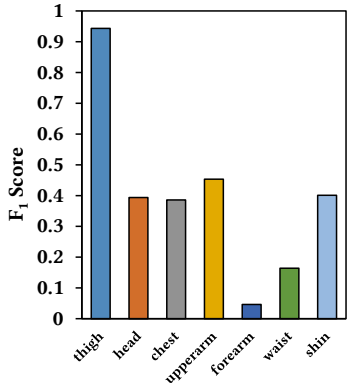
Taken together, our experimental analysis paints a comprehensive picture of using unsupervised domain adaptation to address wearing diversity in HAR. Our findings can serve as practical guidelines for ubicomp practitioners as well as pave the way for future research on domain adaptation in HAR.

1.1 Quantification of Wearing Diversity

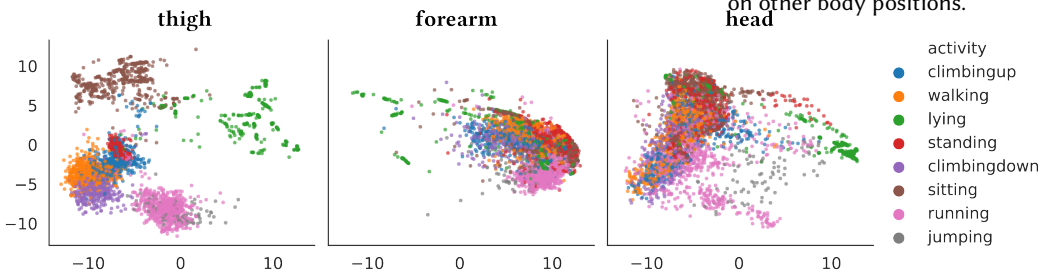
Due to the various form-factors and individual wearing preferences, mobile and wearable devices can be worn or carried by users in diverse ways, for example, a smartphone can be carried in a pocket or held in hand. As shown in Figure 1a, the accelerometer data captured from wearable devices across body positions show significant divergence, which is a strong indication of the *domain shift* caused by wearing diversity. Next, we analyze the impact of this domain shift on the accuracy of a HAR classifier. We trained a HAR model based on a state-of-the-art deep learning algorithm [5] to classify eight human activities based on accelerometer traces obtained from a wearable sensor placed in a thigh pocket. In Figure 1b, we plot the F_1 score obtained by applying the trained HAR model for thigh on test datasets from various body positions. We observe that the HAR model has an F_1 score of 0.94 when it is tested on the same body position on which it was trained (i.e., there is no domain shift).



(a) Accelerometer traces collected from a trouser pocket, a chest pocket and an armband for the same physical activity. Variations across body positions are clear and significant.



(b) The F_1 score of the classification model trained on *thigh* when tested on other body positions.



(c) Feature vectors generated by the classification model trained on *thigh* are projected into 2D space using PCA.

Fig. 1. The wearing diversity is typical in many wearable devices. Such variability results in huge variance in collected accelerometer signals (Figure 1a). Human activity recognition models that do not consider wearing diversity may suffer from significant accuracy drops (Figure 1b). For example, the classification model trained on *thigh* forms clear clusters on traces from *thigh*, but traces from *forearm* and *head* fail to form a cluster (Figure 1c).

However, in the presence of domain shift, the F_1 score of the classifier drastically reduces to as low as 0.05 when the model is tested on IMU data from the forearm.

Finally, in Figure 1c, we visually demonstrate the effect of domain shift on a classification model trained on data from *thigh*. More specifically, we input data from thigh, forearm and head to this classification model and plot the feature vectors obtained from the model. Principal component analysis (PCA) is used to plot the vectors in 2D space; the two principal axes are chosen considering all feature vectors from thigh, forearm and head. As evident, the features are well clustered for the data from 'thigh' (same as the training body position), which allows for higher classification performance. However, for the other two body positions, classes are not easily separable, resulting in worse classification performance.

2 RELATED WORK

While efforts to improve human activity recognition (HAR) have been continued for many years, assuring the robustness of the HAR is still an active area of research. Researchers have examined many different sources

of variability, i.e., device heterogeneity [4], user variance [25], and wearing variability [40]. In this paper, we focus on wearing diversity problem, especially on building an HAR model that works robustly on multiple body positions.

2.1 Enhancing Robustness of Human Activity Recognition on Wearing Diversity

There exist two major approaches in enabling HAR models to perform robustly in diverse body positions. The first approach is to build a body-position-aware model which tries to predict both wearing position and activity, simultaneously. [6] was an early work in this direction, it used support vector machine and hidden markov model to predict wearing position and activity either independently or jointly. [46] suggested a pipelining approach. It first extracted orientation-independent features and passed them to the body position classifier. After body position being identified, the orientation-independent features and the body position information have been used to classify the activity. However, both works focused on a limited number of body positions (~4 positions) and activities (~5 activities). Sztyler [41] extended this approach to work with 8 activities across 7 body positions. The major limitation of this approach exists in its computational cost. Two classification model should run sequentially to detect current activity. Also, it consumes more memory as both wearing position classification model and body-position-specific activity classification models should be loaded into memory on runtime. In addition, it is not scalable. When a new wearing diversity arose, the developers have to collect new labeled dataset and rebuild the body position and activity classification model.

The second approach is to build a body-position-independent model by engineering body-position-independent features. Early works [20, 38] applied decision tree and kernel function over common features that can be extracted from the accelerometer. However, they were limited in the number of activities and body positions. Nguyen et al. [32] selected effective features for each body position and combined them to generate a body-position-independent feature set. The classification model based on the optimized feature set has shown 99.13% accuracy on classifying 13 activities over 4 different body positions. However, while feature engineering is effective on classical models that use hand-crafted features, it is hard to be applied to deep learning models where features are learned from the data.

Recently, Almaslukh et al. [5] trained a deep neural network over a dataset collected from multiple body positions to build a body-position-independent model. Our work significantly differs from it as we utilize an unlabeled dataset rather than a labeled dataset to generalize the model to multiple body positions, thereby reducing the cost of data collection.

2.2 Unsupervised Domain Adaptation Applied to Human Activity Recognition

There exist several studies that applied unsupervised domain adaptation to adapt the existing model for a smartphone to new wearable sensors.

HDCNN [21] used a feature matching approach to adapt the existing model to a smartwatch with an unlabeled dataset collected from the smartwatch. Specifically, they tried to reduce the discrepancy between two datasets after every convolutional and fully connected layer. They used Kullback-Leibler divergence as a distance measure. MotionTransformer [7] used a confusion maximization approach to build a translator that converts a trace from new wearable sensors to resemble the trace collected from smartphones. Akbari et al. [3] explored two adaptation techniques. They first extracted stochastic features by training variational autoencoder [23]. Then, they applied a feature matching approach to adapt the model to a target environment, using an unlabeled dataset collected from the target environment. Kullback-Leibler divergence is used to measure the distance between the labeled and unlabeled datasets.

Our approach differs from the above research in two aspects. First, the goal is different. We target to build models that work robustly on multiple body positions. In contrast, the above studies aim to build a new model

that will only perform on a new wearable sensor attached to a specific body position. Thus, the evaluation of the above research is limited to adaptation performance. Second, our major contribution is on the comparison of adaptation techniques. We concentrate on giving UbiComp practitioners a guideline on applying UDA algorithms to wearing diversity problem.

3 UNSUPERVISED DOMAIN ADAPTATION FOR WEARING DIVERSITY

As mentioned earlier, the goal of this work is to do a systematic evaluation of various adaptation techniques with respect to the problem of wearing diversity in wearable devices. In this section, we present three classes of algorithms for performing unsupervised domain adaptation (UDA) that we will study in this paper.

Please note that we focus our analysis on HAR models trained with deep neural network architectures for two reasons: (a) in recent times, deep learning-based computational models have outperformed shallow models for HAR tasks [17, 35], and (b) ability of neural networks to learn representations from unlabeled data is a key feature of unsupervised domain adaptation. Part (b) is especially important as UDA techniques are built upon this ability to learn representations even in the absence of labeled data. Other machine learning techniques that do not work on the principle of representation learning, e.g., Random Forest or Support Vector Machines, are hence incompatible with these UDA techniques and out of scope of our work.

3.1 Problem Formulation

Let us say we are presented with a source domain (or body position) S with input data X_S and labels Y_S. The input data (X_S, Y_S) here corresponds to the labeled accelerometer or gyroscope data collected from a wearable sensor placed at the source body position. Using supervised learning, we can train a deep neural network on this labeled dataset as has been proposed in prior works on HAR [5, 17]. A key property of deep neural networks is their ability to automatically extract meaningful feature encoding from the raw data – we denote the learned feature encoder for the source domain as $E_S : X_S \longrightarrow F$, and a classifier learned on top of the encoded features as $C_S : F \longrightarrow Y_S$.

Formally, E_S and C_S are trained using supervised learning by solving the optimization problem:

$$\min_{E_S, C_S} \mathcal{L}_{\text{supervised}} = -\mathbb{E}_{(x_s, y_s)\sim(X_S, Y_S)} \sum_{k=1}^{K} 1_{[k=y_s]} [\log(C_S(E_S(x_s))] \tag{1}$$

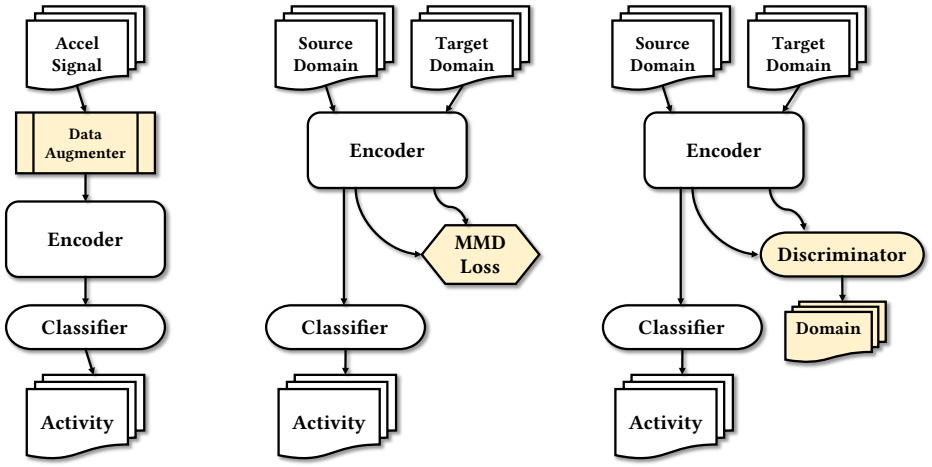
where K denotes the number of activity classes.

Indeed, as the feature encoder E_S and the classifier C_S have been trained solely on data from the source body position (e.g., thigh), it is unlikely that they will provide accurate inferences under domain shift, i.e., when they are tested on a new target body position (e.g., wrist). Therefore, the objective of unsupervised domain adaptation is to adapt the source feature encoder E_S and the source classifier C_S such that they can be applied to an arbitrary target domain T_j for which an input dataset (X_T) is available, but without any labeled observations. Domain adaptation techniques rely on the assumption that there exists a feature representation of the sensor data which is invariant to changes in sensor body position, and is still powerful enough to encode the human activity data.

3.2 Unsupervised Domain Adaptation Techniques

Domain adaptation assumes the availability of two types of datasets:

- The source domain **labeled dataset** $\mathcal{D}_c$, which consists of pairs (x_s, y_s) where x_s is the sensor data sampled from the IMU and y_s is an activity label for x_s.
- The **unlabeled dataset** $\mathcal{D}_d$ which consists of pairs (x_i, d) where d is the body position from which the sensor data x_i was recorded. No labels are available for x_i.



(a) Data augmentation (b) Feature matching (c) Confusion maximization

Fig. 2. High-level diagram depicting operation of three adaptation techniques. Note that data augmentation does not use unsupervised dataset at all and is considered as a domain generalization technique.

We now describe the three adaptation approaches for addressing the wearing diversity problem, namely Data Augmentation, Feature Matching and Confusion Maximization. The three techniques share a common objective: how to learn robust feature representations of inertial sensing data that are invariant to the body position where the sensor is placed. They however differ in how they achieve this objective: as we will describe, Data Augmentation uses the property of deep neural networks to learn robust feature representations from noisy data, Feature Matching achieves feature invariance by explicitly minimizing the distance between features of two domains, and Confusion Maximization achieves this objective using adversarial learning. Therefore, the three techniques are complementary in their approaches and help us in analyzing the performance of UDA techniques from a broad lens.

In the subsequent text, we describe each technique in detail.

Data augmentation. The simplest approach to train neural networks which are robust to domain changes is to perform data augmentation. Data augmentation mocks perturbations that commonly exist in accelerometer and gyroscope traces on a given labeled dataset. This means defining a set of either probabilistic or deterministic mappings $X \longrightarrow X$ on the input space to enlarge the labeled dataset $\mathcal{D}_c$ as depicted in Figure 2a.

Let $x \in \mathbb{R}^{kT}$ be a reasonably smooth k-axis time-varying signal (usually $k = 3$ or $k = 6$). Performing affine similarities in the k axes and continuously deforming the time axes with resampling yields a natural transformation group for wearable inertial sensors. Here, we use three transformations – rescaling, axis rotation, and time-warping – selected from IMU-suitable transformations proposed in Um et al. [44], which have shown a good performance.

- *Rescaling* scales each axis by a random coefficient sampled from $c \sim \mathcal{N}(1, 0.1^2)$, which is related to IMU (Inertial Measurement Unit) scaling factor errors [2].
- *Axis rotation* rotates x with a random roll, pitch, and yaw. This simulates a variability caused by placement of the sensor.

- *Time warping* mocks deformation in the time axes; it adjusts time interval between data points in x and resamples from it. This perturbation is related to sampling rate instability that occurs when operating systems read IMU signals [4].

Overall, the objective of data augmentation based training is to enable a deep neural network to learn robust feature representations of the IMU data, that are invariant to the perturbations introduced in the data.

Feature matching. In this method for training neural networks, we explicitly add a loss term which minimizes a distance measure between the features extracted from different body positions. If $B_d \subset \mathcal{D}_d$ is a batch of data consisting of data B_0 and B_1 coming from body positions 0 and 1, respectively, then we minimize a distance measure between $E(B_0)$ and $E(B_1)$ where E represents a feature extracting neural network such as a CNN ($E\colon X \longrightarrow F$). By minimizing the distance between the features in the training process, we force the model to learn features that are invariant to body positions.

The selection of a proper distance measure is the most important decision when applying feature matching. In our current implementation, we use (and minimize) the maximum mean discrepancy (MMD) [42] using Gaussian kernels as the distance metric between domain-specific feature representations. MMD distance is one of the most widely used distance measures that has been applied and proven its effectiveness in many different domains [18, 21]. Equation 2 describes how the MMD distance is computed across two domains d and d'. A high-level representation of this method can be found in Figure 2b.

For a batch of data $B_d \subset \mathcal{D}_d$ we define the MMD loss as:

$$\mathcal{L}_{\text{MMD}} = \mathop{\mathbb{E}}_{\substack{(x,d),(x',d')\in B_d \\ d=d'}} \left[k\left(E(x),E(x')\right)\right] - \mathop{\mathbb{E}}_{\substack{(x,d),(x',d')\in B_d \\ d\neq d'}} \left[k\left(E(x),E(x')\right)\right], \tag{2}$$

where k is a sum of Gaussian radial basis function kernels

$$k(x,x') = \sum_{c\in C} e^{-\frac{\|x-x'\|^2}{2c}}, \quad C = \{0.0001, 0.001, 0.01, 0.1, 1, 5, 10, 15, 20, 25, 30, 100\} \tag{3}$$

The rationale for using the Gaussian radial basis function kernels is to project the feature representation $E(x)$ into a Reproducing Kernel Hilbert Space (RKHS) [34] which is essential to the computation of the MMD distance [16]. Here C is a hyper-parameter that we borrowed from the official implementation of MMD in TensorFlow.

We now provide intuition on how the MMD loss works. Let us say we are trying to adapt a model trained on data from *Wrist* to work on *Chest*. As such, the goal of the adaptation process is to find a feature representation of the data that is common for both Wrist and Chest. Therefore, we first extract the features for each body position using the feature extractor E and then apply the MMD loss to push the difference of the features to 0. In other words, we aim to align the feature spaces of both body positions.

Confusion Maximization. This technique is built upon the principle of domain adversarial training [14] which is also used for training generative adversarial networks. The key idea is to use an additional neural network called the domain discriminator h_ψ, to make the feature encoder E domain-invariant using adversarial training. The goal of the domain discriminator is to distinguish data from the source domain and target domain – as such, during the training process, it aims to learn a binary classification model that can accurately separate the data from two domains. On the contrary, the goal of the feature encoder is exactly the opposite – it aims to confuse the domain discriminator by generating features that are invariant to source and target body positions, and hence cannot be easily separated. In this sense, the domain discriminator and feature encoder are *adversaries* of each other, tasked with competing objectives. They play this adversarial game of trying to defeat the other, and in the process, both become better at their respective tasks. Importantly from the perspective of wearing diversity, the feature encoder learns to generate features which are invariant to the body positions where the sensor is placed.

Please refer Figure 2c to see how the encoder and the discriminator are connected. In our current implementation, the discriminator is composed of two fully connected layers, with 100 hidden nodes in between.

The discriminator loss in the two domain case $d \in \{0, 1\}$ is given by

$$\mathcal{L}_{\text{discriminator}} = -\frac{1}{|B_d|} \sum_{(x,d) \in B_d} d \log \left(h_\psi \left(E \left(x \right) \right) \right) + (1 - d) \log \left(1 - h_\psi \left(E \left(x \right) \right) \right). \tag{4}$$

Our Contribution. It is important to highlight that from a theoretical viewpoint, the above three techniques are known in the machine learning literature. However, they are not black-box solutions that can simply be applied to any dataset or architecture to enable domain adaptation. As we will discuss in the subsequent sections, there are a number of factors that dictate the performance of UDA, including class mismatch between source and target domains, the choice of metrics used to compute divergence between the domains, and relative proportion of labeled and unlabeled data. As such, one of our contributions is to build upon the theoretical foundation of these techniques, and develop domain adaptation model architectures and data pipelines specific to the problem of wearing diversity.

3.3 Model Training Process

Algorithm 1: Alternating minimization for domain adaptation using Feature Matching and Confusion Maximization. Data Augmentation is not shown in the algorithm as it is typically done apriori on the entire labeled dataset D_c

input : datasets $\mathcal{D}_c$, $\mathcal{D}_d$,
gradient update rule Ω,
method $\in$ {data augmentation, FM, CM}
output: Trained parameters θ, ϕ, ψ
for *number of steps* **do**
  sample batches $B_c \subset \mathcal{D}_c$ and $B_d \subset \mathcal{D}_d$;
  $\theta \leftarrow \theta - \Omega \left(\mathcal{L}_{\text{classification}}, \theta, \phi, B_c \right)$;
  $\phi \leftarrow \phi - \Omega \left(\mathcal{L}_{\text{classification}}, \theta, \phi, B_c \right)$;
  if method *is Feature Matching* **then**
    $\theta \leftarrow \theta - \Omega \left(\mathcal{L}_{\text{MMD}}, \theta, \phi, B_d \right)$;
  end
  if method *is Confusion Maximization* **then**
    $\psi \leftarrow \psi - \Omega \left(\mathcal{L}_{\text{discriminator}}, \theta, \psi, B_d \right)$;
    $\theta \leftarrow \theta + \Omega \left(\mathcal{L}_{\text{discriminator}}, \theta, \psi, B_d \right)$;
  end
end

We now discuss how we leverage different UDA techniques to train robust HAR models. The training of the neural network proceeds by alternating minimizing the classification error and maximizing invariance of features.

We first sample a batch of labeled data $B_c \subset \mathcal{D}_c$ from the source body positions and a batch of unlabeled data $B_d \subset \mathcal{D}_d$ from the target body position.

In the first step, we use the labeled data B_c to minimize the cross-entropy loss as in ordinary supervised training of neural networks, in order to train the parameter θ for the encoder and the parameter ϕ for the classifier.

In the second step, we train the encoder towards producing invariant features. In the case of feature matching, we pass the labeled (B_c) and unlabeled (B_d) batches to the encoder, compute the MMD loss in the feature space based on Equation 2 and minimize it in the training process. Alternatively, for Confusion Maximization, we train the parameter ψ for the discriminator neural network by optimizing the loss in Eq 4 and backpropagate this loss through the encoder with the opposite sign since the encoder has the exact opposite objective as the discriminator. Both the training steps are repeated until the training converges or an early stopping criterion is met. Note that Feature Matching and Confusion Maximization are different methods and are not trained simultaneously. We added them in the same algorithm for ease of explanation. In our evaluation, we will implement domain adaptation separately for both of these techniques and compare their performance.

We will describe the neural network architecture for the encoder and classifier in more detail in § 4.1.

4 STUDY DESIGN

In this section, we present the study design for evaluating the three adaptation techniques from the perspective of wearing diversity.

4.1 Experiment Setup

We begin by describing the datasets and pre-processing operations done on the data, and then proceed to discuss the deep neural network used for the HAR task.

Dataset. We selected the following four datasets for evaluating the impact of different wearing positions on HAR classifiers, with and without domain adaptation.

Table 1. Dataset information

Name	# activity	# bodypos	# train	# eval	sampling rate
RealWorld [40]	8	7	118616	29652	50Hz
Opportunity [37]	5	8	90490	22616	30Hz
HHAR [4]	6	2	30280	5342	50∼200Hz
PAMAP2 [36]	12	3	30726	7680	100Hz

RealWorld. The *RealWorld HAR dataset* [40] includes data recorded from 15 participants performing 7 activities: climbing stairs down and up, jumping, lying, standing, sitting, running/jogging, and walking. Users were instrumented with smartphones and smartwatches placed at 7 different body positions: head, chest, upper arm, waist, forearm, thigh, and shin; and accelerometer and gyroscope data was sampled from the devices simultaneously at a sampling rate of 50 Hz. Each participant performs each activity for 10 minutes, except for jumping (∼1.7 minutes). There are three major reasons why this dataset is ideal to study the wearing diversity problem: a) the physical activities by the participants were performed in naturalistic settings as opposed to controlled experiments, b) the data across male and female participants are equally distributed which reduces the bias in our evaluation, c) to the best of our knowledge, this is the largest dataset of body position variability that is publicly available which makes it easy for other researchers to replicate our results.

Opportunity. The *Opportunity Activity Recognition dataset* [37] includes data recorded from 4 participants performing 5 activities: null, standing, walking, sitting, and lying. Users were instrumented with a custom-designed motion jacket and shoes, collecting accelerometer signals from total 19 body positions. We grouped sensors attached to the same body position, which resulted in an updated dataset of 8 different body positions: UPPERARM, LOWERARM, WRIST, HAND, BACK, HIP, KNEE, and FOOT. The accelerometer data were sampled from the

devices simultaneously at a sampling rate of 30Hz. The users were asked to conduct pre-defined daily home activities for 15-25 minutes. This is a challenging dataset as sensors were placed on the clothes rather than being tied to the specific body position. Moreover, the dataset for each body position consists of traces from multiple sensors which were attached separately. Finally, there was no control in users' activity and the resulting dataset is not balanced.

HHAR. The *Heterogeneity Human Activity Recognition (HHAR) dataset* [4] includes data recorded from 9 participants performing 6 activities: bike, sit, stairsdown, stairsup, stand, and walk. Users were instrumented with 8 smartphones and 4 smartwatches. All 8 smartphones were placed around waist, and smartwatches were worn on each arm. We named the 2 body position groups, phone and watch. Accelerometer data were sampled from the devices simultaneously at a sampling rate of 50 to 200Hz depending on the device. Each user performed each activity for 5 minutes. However, this dataset is noisy compared to the other datasets as timestamps are not contiguous and sampling rates are unstable.

PAMAP2. The *PAMAP2 Physical Activity Monitoring dataset* [36] includes data recorded from 9 subjects performing 18 different activities. As 6 were optional activities, we only used 12 activities among them: ascending stairs, cycling, descending stairs, ironing, lying, nordic walking, rope jumping, running, sitting, standing, vacuum cleaning, and walking. Users were instrumented with 3 wireless inertial measurement units placed at 3 different body positions: head, chest, ankle; and accelerometer and gyroscope data were sampled from the devices simultaneously at a sampling rate of 100 Hz. Each user performed each activity for up to 3 minutes. We particularly selected this dataset to evaluate performance of domain adaptation under more diverse activity classes.

Data characteristics and pre-processing. The accelerometer traces used in our experiment have a shape of (150, 3), i.e., the 3-axis value of an accelerometer logged for 3 seconds with a sampling rate of 50 Hz. This duration was chosen empirically to align with the duration of various human activities in the dataset.

Datasets collected with a different sampling rate are either upsampled and downsampled to match the sampling rate of 50Hz. The accelerometer traces are segmented into time windows of 3 seconds, without any overlap between the samples. If a 3-second-long trace includes an activity transition, timestamp noise, or data points without labels, the trace gets discarded. The whole dataset is normalized to be in the range of -1 and 1.

Every trace in the dataset has two labels: activity and body position that the trace was collected. Other information, i.e., participant ID and sensor type, is discarded.

Classification model. As explained earlier, we focus our analysis on HAR models trained using neural network architectures. As such, we design a convolutional neural network based on the work by Hammerla et al. [17] and Almaslukh et al. [5].

The model consists of two components: a feature extractor (encoder) and a classifier. The *feature extractor* is a 6-layer deep CNN with temporal (1-D) convolutional and pooling layers. We used LeakyReLU activations [27] with $alpha = 0.3$ and instance normalization [43] layers between convolutional layers for faster convergence. We also employed dropout regularization to avoid overfitting. The feature extractor takes as input a 3-second frame of motion data sampled at 50 Hz (150 samples) and outputs a 100-dimensional feature vector. This feature vector is then passed as input to the *classifier* which consists of one fully-connected layer and generates a k dimensional output where k is the number of activity classes (e.g., sitting, walking).

The proposed network architecture (Figure 3) differs from [5, 17] in three aspects. Firstly, all layers in the encoder are convolutional layers. We replaced the traditional max-pooling layers with convolutional layers with a large stride based on recent research [39] which showed that this simple substitution can speed up the training process without any loss in the accuracy. Secondly, kernel size has been engineered to focus on repetitive patterns of human activities that have a frequency of 0.5 to 3Hz, which may be related to whole-body movement [12]. Finally, global average pooling is used instead of a fully-connected layer to encode a feature vector from the

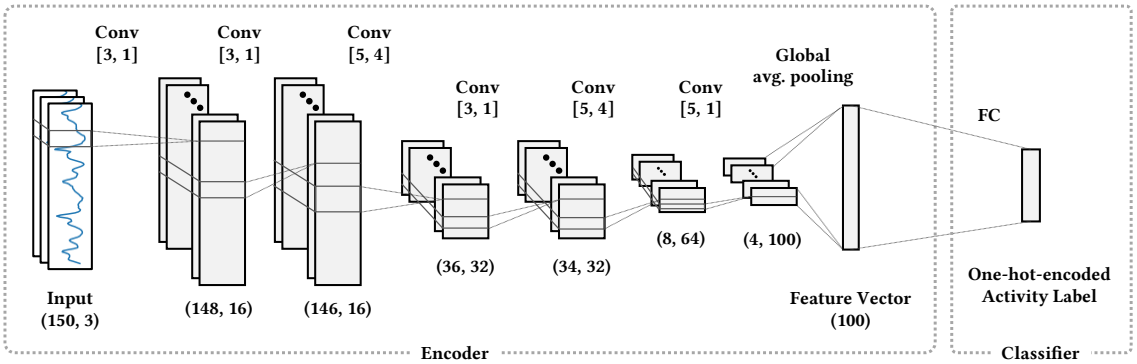


Fig. 3. The architecture of the deep neural network model used in evaluation. Conv stands for *convolutional layer*, and FC stands for *fully connected layer*. The vector under Conv denotes [*kernel size*, *stride*] respectively. The exact size of output vectors are written under each layer.

final output of convolutional layers. This is to reduce the number of weights to be learned. A baseline result on RealWorld dataset (See table 4a) showed that the performance is still on par to [5] while having a smaller model size by using a smaller number of filters and fully-connected layers.

Training process and mini-batch generation for UDA. We trained the network with stochastic gradient descent using the Adam gradient update rule [22] with learning rate 10^{-3}, $\beta_1 = 0.9$ and $\beta_2 = 0.999$. We use a mini-batch size of 125 and follow the adaptation strategies as discussed in Section 3. The mini-batches are not time-aligned, and at every epoch, we shuffle the labeled and unlabeled dataset independently, resulting in an extremely low chance of feeding the same labeled-unlabeled mini-batch pairs to domain adaptation training steps.

4.2 Evaluation Metrics

To holistically address the wearing diversity problem, we propose three key metrics that any domain adaptation solution should improve, namely adaptability, persistence, and generalizability.

- *Adaptability* refers to the performance on target body position(s) supplied as an unlabeled dataset. This is an important metric when models need to be adapted between static body positions – for instance, if a source model was trained on wrist-worn sensors and subsequently need to be deployed for chest-worn sensors, we desire a high adaptation performance on Wrist→Chest experiment.
- *Persistence* measures the performance on source body position(s) supplied as a labeled dataset, after it undergoes the domain adaptation process. It is particularly important because we would like the model to retain its performance in the source domain along with becoming better at doing inferences in the target domain.
- *Generalizability* measures how well a model performs on body positions for which it was neither trained nor adapted. This property is critical because in real-world scenarios, wearing positions do not remain static — i.e., users can wear their device in unexpected positions and orientations. Therefore, a technique which produces more generalizable models is desirable. In this paper, we define *Generalizability* as the aggregate performance on all body positions that are available in a given dataset.

4.2.1 Choice of Macro-averaged F_1 score. To visualize our results, we report a single, macro-averaged F_1 score for each evaluation metric per experimental setting. Yet, we observe that a selection averaging policy can affect the nuances of the results, especially under the existence of a minority class in the data.

For a multi-class classification problem, we can calculate a F_1 score using three different averaging policy: micro-, weighted-macro- and macro-averaging (Equation 5) where $\hat{y}_l$ is true labels for class l.

$$F_{\text{micro}} = F_1\left(y, \hat{y}\right), \quad F_{\text{weighted}} = \frac{1}{\sum_{l \in L} |\hat{y}_l|} \sum_{l \in L} |\hat{y}_l| F_1\left(y_l, \hat{y}_l\right), \quad F_{\text{macro}} = \frac{1}{|L|} \sum_{l \in L} F_1\left(y_l, \hat{y}_l\right). \tag{5}$$

While the conventional wisdom suggests using micro-averaging of F_1 score in cases of class imbalance, we observe that in our experiments, micro-averaging often overestimates the performance of adaptation. We hypothesize that the adaptation process sometimes leads to a class collapse – i.e., one particular activity class may end up with very poor accuracy after adaptation. Especially, if the collapsed class happens to be a minority class, we observe that micro-averaging tends to significantly overestimate the overall performance compared to macro-averaging. Table 2 depicts an exemplary scenario. Jumping (a minority class with 5% of the samples) collapses after adaptation, but the micro-averaged F_1 score does not reflect this issue as the percentage of correct prediction is still high.

Table 2. Exemplary scenario that compares three averaging policies under a minority class collapse. Jumping, a minority class with 5% of the samples, collapses after adaptation, being classified into null class (Table a). Macro-averaging is the only policy that reflect such a collapse in its value (Table b).

(a) Confusion matrix.

True Label	Predicted Label		
	null	walking	jumping
null	45	5	0
walking	5	40	0
jumping	5	0	0

(b) Comparison of three different averaging policies.

Averaging Policy	Metric		
	Precision	Recall	F_1 score
micro	0.85	0.85	0.85
weighted-macro	0.81	0.85	0.83
macro	**0.57**	**0.60**	**0.58**

We believe the choice of F-1 score averaging technique is application-dependent. It is possible that for a certain human activity recognition application, we do not care about the performance of an extreme minority class. But there could be other applications (e.g., fall detection) where the performance of a minority class (e.g., presence of fall) is of primary interest.

In this paper, we primarily present a macro-averaged F_1 score, as we prefer a conservative estimate of the adaptation performance, while accounting for class collapses. However, we also report a comparison of the three F-1 score averaging policies on the Opportunity dataset (See Figure 7); the Opportunity dataset has a minority class – *lying* – that is 3.8% of the samples. Note that according to our empirical analysis, the selection of averaging policy does not change the relative performance among the adaptation techniques, thereby not affecting our conclusions.

4.3 Methodology

Our in-depth empirical analysis revolves around the following aspects of UDA:

- **Comparison of adaptation techniques.** One of the main focus of the paper is to unravel the performance of adaptation techniques under diverse adaptation scenarios related to the wearing diversity problem. By enumerating over body position pairs as a source and target body position, we try to find out (a) the relationship between body positions, e.g., amount of domain shift between two body positions and (b) how does the performance of adaptation techniques vary depending on the body positions.
- **Grouping of multiple body positions.** Collecting labeled and unlabeled datasets from multiple body positions would be an intuitive choice if we aim to build a body-position-independent model. We experiment

by creating intuitive groups of body positions to understand which body positions can lead to more robust HAR models.

- **Dataset size and class distribution.** We further studied the effect of dataset properties on the performance of adaptation algorithms. We systematically altered the size of and the class distribution within the unlabeled dataset and reported the performance of adaptation techniques.

4.3.1 Comparison of Adaptation Techniques. As discussed above, the purpose of this experiment is twofold: (a) to find the relationship between body positions, e.g., amount of domain shift between two body positions and (b) to study pros and cons of adaptation techniques in relation to body positions supplied as labeled and unlabeled datasets.

For each body position, we first train a CNN model for the HAR task and evaluate its performance on other body positions. This helps in quantifying the impact of wearing diversity for different datasets and serves as a key baseline for domain adaptation approaches. Thereafter, we leverage the three adaptation techniques discussed in § 3 and adapt the pre-trained CNN models for a different unlabeled body position. Based on the various evaluation metrics introduced in § 4.2, we compare the efficacy of different adaptation approaches in solving the problem of wearing diversity.

4.3.2 Use of Multiple Body Positions Together. Collecting accelerometer traces from two or more body positions is an intuitive approach if we aim to build truly body-position-independent HAR models. For example, our experiment (Figure 4) showed that if the model is trained with a dataset collected from three body positions, its generalizability performance goes up to 61% compared to 38% when the model is trained with a single body position.

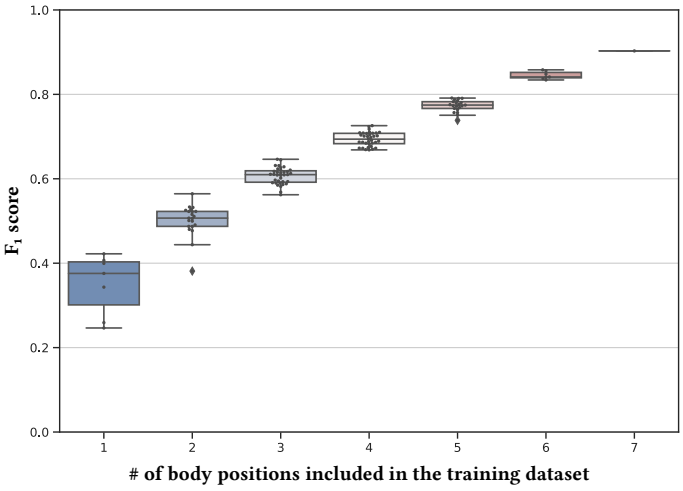


Fig. 4. F_1 score improvement on the evaluation dataset collected from all body positions. Having diverse body positions within the training set clearly improves robustness to wearing diversity. The training and test dataset used in this figure is built from RealWorld dataset [40].

Therefore, we extend our analysis to examine the effect of multiple body positions on unsupervised domain adaptation. More specifically, we ask: How should we pick body positions to get the best performance after applying domain adaptation?

To answer the question, we hand-crafted the mixtures of three body positions (Table 3). We first focus to contrast body positions with high movement and low movement. Body positions have different degrees of

freedom and movement intensity based on their position on the limb. For example, the forearm and ankle have larger degrees of freedom compared to the thigh and chest. Also, the acceleration intensity is bigger on the forearm and ankle as they are placed far from the pivots, i.e., shoulder and pelvis. One intuition is that picking body positions placed on LIMB END as a labeled dataset can help the classification model to learn robust features from more challenging accelerometer signals. However, on the other hand, [41] has reported that *waist* is the best body position to train a body-position-independent model, as the waist is closer to a center of mass of the whole body and is effective in selecting features related to whole-body movement. We thus compared two adaptation scenarios, LIMB END→LIMB MIDDLE and LIMB MIDDLE→LIMB END, wherein LIMB END contains body positions with a higher degree of freedom (e.g., forearm, shin) and LIMB MIDDLE contains positions with a lower degree of freedom (e.g., thigh, chest). We also added body position groups collected from a single limb to test across-the-limb adaptation scenarios.

Table 3. The composition matrix to build the mixture of the body positions. Body position mixtures on the left (Table 3a) is designed to examine the effect of including body positions with high movement within the labeled dataset. The mixtures on the right (Table 3b) is designed to simulate the case that a labeled dataset only comprises traces collected from a single limb, i.e., a training dataset collected for a smartwatch.

(a) Body position mixture selected from multiple limbs

LIMB END	head	forearm	shin
LIMB MIDDLE	chest	upperarm	thigh

(b) Body position mixture selected from a single limb

TORSO	head	chest	waist
ARM	chest	upperarm	forearm
LEG	waist	thigh	shin

4.3.3 Dataset Size and Class Distribution. The performance of data-driven unsupervised domain adaptation algorithms heavily depends on how the underlying datasets are structured. We here examine the effect of size and activity distribution of the unlabeled dataset on the performance of UDA algorithms.

To study the effect of dataset size, we progressively increase the amount of unlabeled data provided to the UDA algorithms in increments of 500 samples, and study the impact of the number of samples on UDA evaluation metrics.

Secondly, we simulate class distribution mismatch between labeled and unlabeled datasets. There would be two approaches in generating class distribution mismatch: (1) changing the ratio between the activity classes and (2) adding an unseen new activity to the unlabeled dataset. Here we only focus on the first option and left the second option as future work. While there are many options to simulate class distribution mismatch, we decide to reduce samples from the more dynamic activity classes, i.e., running, jumping, climbing up, and climbing down. This is done to match in-the-wild data collection scenarios wherein such activities are likely to have a smaller number of samples than the more static activities such as sitting or standing.

5 RESULTS

In this section, we present the experimental results of our investigation into unsupervised domain adaptation approaches for wearing diversity in the context of HAR tasks. The highlights from our experimental results include:

- Overall, Feature Matching outperforms other techniques in adaptation performance and in total, FM succeeded to improve HAR accuracy on target body positions in 93.4% of all experiments conducted. Feature Matching was particularly powerful in learning hidden mappings between domains, thereby boosting the target classifier accuracy by as much as 53% (from 14% to 67%). It also preserves the integrity of the source domain model, with less than a 5% decrease observed in *persistence* of the model in 99% of cases.

- Data Augmentation appears to be a powerful technique when adapting between body positions in the upper torso. However, it suffers a higher (10%) accuracy drop in persistence on average.
- The adaptation performance of Confusion Maximization is weak, but it can be used for special cases when the domain shift between body positions is too high, e.g., adapting from foot to upper torso. Data augmentation and feature matching fail in such challenging scenarios.
- To develop robust body-position-independent HAR models, a collection of labeled and unlabeled datasets from (1) multiple body limbs and (2) body positions with high degrees of freedom is crucial.
- There is a significant impact of dataset size on UDA techniques. While Feature Matching can provide superior adaptation performance with very small amounts of unlabeled data (500 samples), Confusion Maximization requires large amounts of data to reach the same accuracy levels.
- There is a significant adverse impact of Class Distribution Mismatch on the accuracy of domain adaptation when adapting across individual body positions. This impact can be minimized by incorporating heterogeneous body positions in the adaptation process.

Table 4. Baseline F_1 scores for HAR task when the model is trained and tested on the different body positions. This represents the scenario wherein domain shift due to wearing diversity is not considered at all. *BP* stands for *body position*.

(a) RealWorld

	trained BP	head	chest	upperarm	forearm	waist	thigh	shin
head	0.86	-	0.60	0.51	0.26	0.14	0.37	0.23
chest	0.94	0.28	-	0.32	0.25	0.30	0.42	0.24
upperarm	0.89	0.47	0.58	-	0.20	0.18	0.44	0.35
forearm	0.88	0.16	0.12	0.11	-	0.39	0.12	0.09
waist	0.93	0.16	0.22	0.05	0.25	-	0.15	0.02
thigh	0.94	0.39	0.39	0.45	0.05	0.16	-	0.40
shin	0.92	0.06	0.29	0.33	0.03	0.09	0.30	-

(b) Opportunity

	trained BP	upperarm	lowerarm	wrist	hand	back	hip	knee	foot
upperarm	0.75	-	0.40	0.53	0.42	0.38	0.53	0.44	0.24
lowerarm	0.78	0.33	-	0.31	0.28	0.33	0.45	0.23	0.22
wrist	0.64	0.42	0.27	-	0.43	0.21	0.39	0.37	0.23
hand	0.69	0.35	0.15	0.47	-	0.28	0.41	0.37	0.25
back	0.77	0.33	0.26	0.25	0.26	-	0.31	0.40	0.22
hip	0.74	0.32	0.18	0.33	0.30	0.26	-	0.38	0.11
knee	0.80	0.31	0.10	0.32	0.28	0.20	0.24	-	0.08
foot	0.51	0.14	0.13	0.16	0.16	0.21	0.17	0.13	-

(c) HHAR

	trained BP	phone	watch
phone	0.99	-	0.20
watch	0.86	0.22	-

(d) PAMAP2

	trained BP	head	chest	ankle
head	0.86	-	0.23	0.08
chest	0.88	0.16	-	0.02
ankle	0.86	0.01	0.02	-

5.1 Baseline Performance

We first report the performance of HAR models when they are trained and tested on the same body position. This represents the ideal scenario with no presence of domain shift, thereby we expect high classification accuracies. Table 4 aggregates and displays baseline performance of each dataset. For instance, in the RealWorld dataset, a model trained on *head* and *chest* provide F-1 scores of 0.86 and 0.94 when tested on the same body position on which they were trained.

While the classification performance was high on RealWorld, HHAR, and PAMAP2 datasets, we observed that the performance on Opportunity dataset was low, especially on the wrist, hand, and foot. The accuracy drop was due to difficulties in classifying the 'lying' activity from the body positions at the end of body limb where variability within the activity is high. This result is in line with a prior work [41] which also reported that discriminating static activities such as 'lying' is more challenging than dynamic activities.

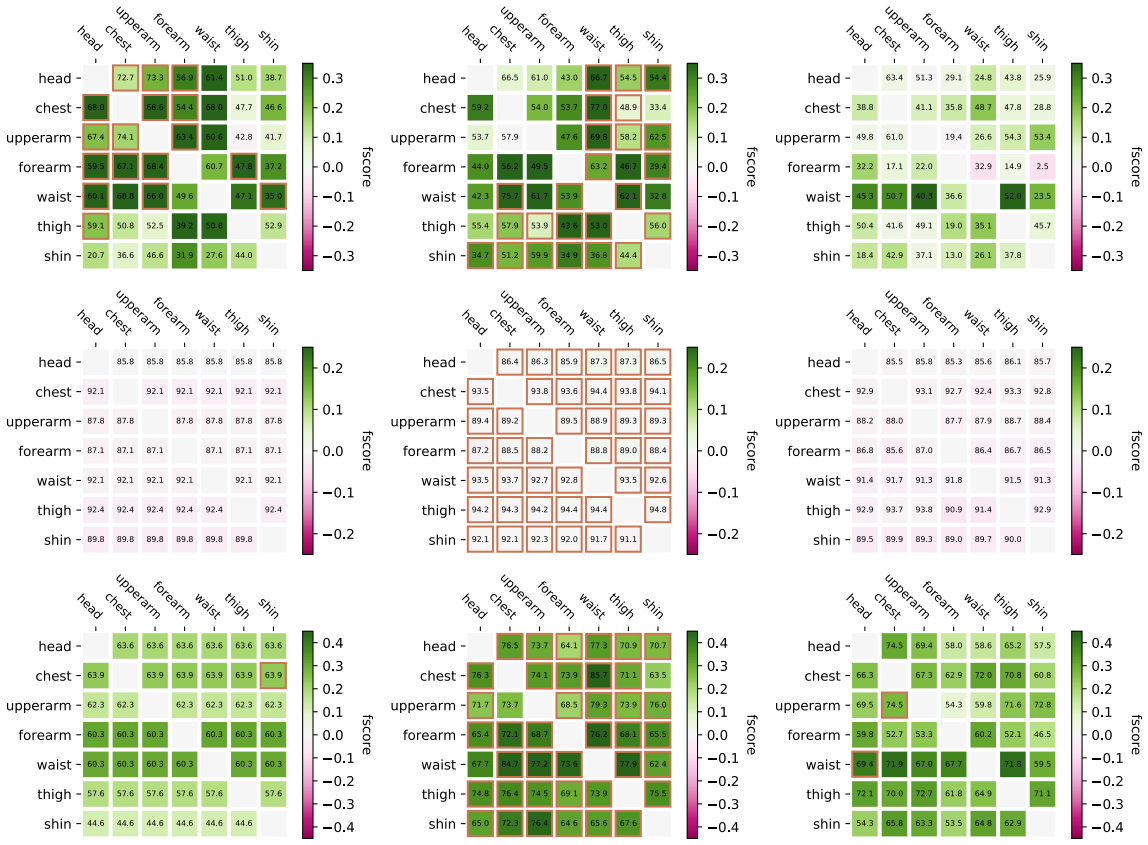
In Table 4 (right part), we further analyze the baseline performance on body positions other than the trained body position. While the drop in accuracy was severe as discussed in Section 1.1, we were able to find body position groups that the accuracy drop was relatively low. In the case of RealWorld dataset, the body positions near the torso – i.e., head, chest, upper arm – showed low accuracy drop when tested on each other. For example, the model was *head* suffered a 72% drop when tested on *waist*, but the drop for *chest* was much less (22%). On Opportunity dataset, body positions from the same limb – i.e., wrist, hand, and upper arm – showed similar behavior. Finally, in all datasets, foot or ankle showed a significant accuracy drop when tested on other body positions. This implies that the domain shift between the foot and other body positions is bigger than other pairs.

5.2 Comparison of the Adaptation Techniques

We now present the results after applying the adaptation techniques in a total of 106 experimental settings, varying the source and target body positions.

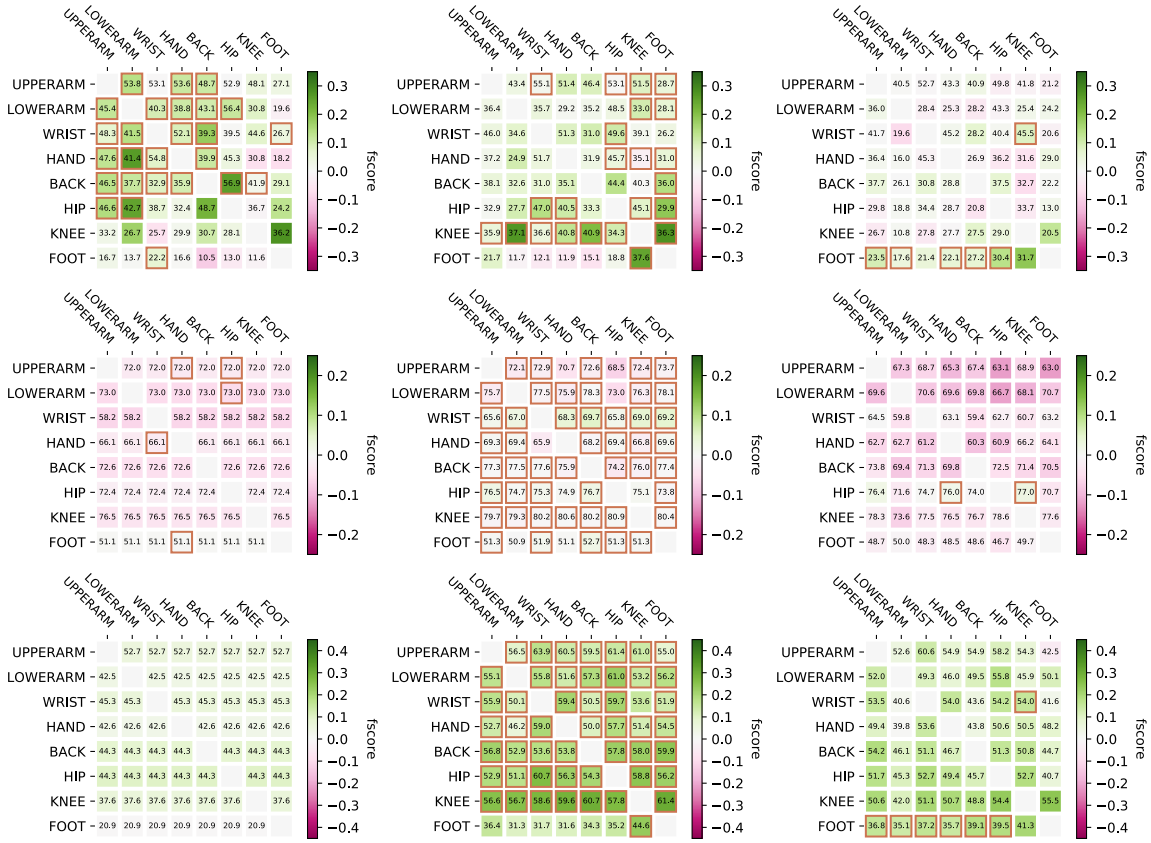
Interpreting the heatmaps. Before presenting our in-depth results in the form of heatmaps, we explain how to read and interpret them. We will present a figure comprising 9 heatmaps for each dataset (Figure 5, 6, 8a, and 8b). Each column corresponds to an adaptation technique; from the left, each column will contain results for the *data augmentation*, *feature matching*, and *confusion maximization*.

A column consists of three heatmaps. The first heatmap depicts results on Adaptability, i.e., accuracy on the target body position supplied as an unlabeled dataset. The higher the adaptability, the better. The second heatmap represents Persistence, accuracy on the source body position after adaptation. As adaptation can potentially degrade the HAR accuracy in the source domain, we would like this accuracy drop to be small. The third heatmap at the bottom reports Generalizability, accuracy on all possible body positions. Each heatmap is composed of cells and each cell denotes one experimental setting. The body position written on the left of the cell denotes the source body position supplied as a labeled dataset. The body position written on the top of the cell denotes the target body position supplied as an unlabeled dataset. By reading the two body positions, we can find out the experiment setting a cell is representing. Delving deeper, the cell has three attributes. First, the color denotes an improvement/decrease compared to the baseline. If the color is green, the F_1 score improved due to adaptation; if the color is pink, the F_1 score decreased (as will be the case for Persistence). Second, the number inside a cell denotes the actual F_1 score for the given metric. Finally, the orange border around a cell shows that the given cell has the best performance on the given experimental setting among three adaptation techniques.



(a) Data augmentation (b) Feature matching (c) Confusion maximization

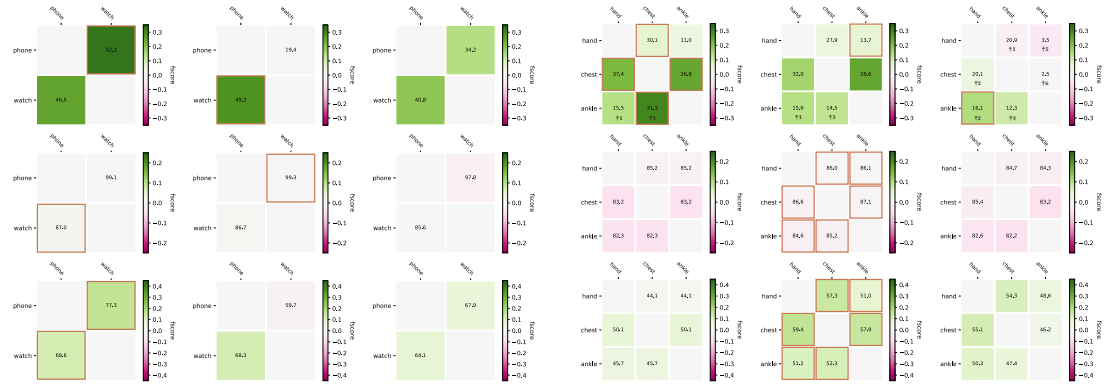
Fig. 5. Comparison of three adaptation techniques on RealWorld dataset. Each column represents the results for the adaptation techniques written below. From the top, each row depicts improvement/decrease in Adaptability, Persistence and Generalizability. Each of the square cell inside heatmaps corresponds to an experimental setting; a body position written on the left is where the labeled dataset is collected, and a body position written on the top is where the unlabeled dataset is collected. The value written inside the cell denotes the absolute F_1 score. The color visualizes the amount of improvement/decrease compared to the baseline. The border around the cell means that the corresponding adaptation technique showed the best performance among three techniques applied under the same experimental setting.
If we focus on the bordered cells, we can find clear patterns; between body position in the upper torso, *data augmentation* outperformed UDA techniques. However, when adapting between the upper torso and lower body, *feature matching* showed the best performance. Nevertheless, when it comes to Persistence and Generalizability *feature matching* championed in most of the experimental settings.



(a) Data augmentation (b) Feature matching (c) Confusion maximization

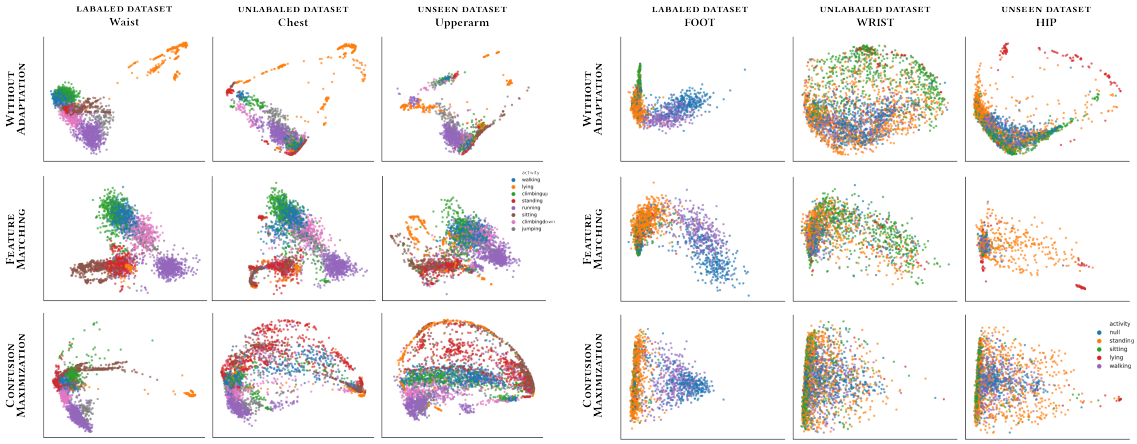
Fig. 6. Comparison of three adaptation techniques on Opportunity dataset. Like Figure 5, each column represents different adaptation techniques and each row denotes improvement/decrease in Adaptability, Persistence and Generalizability (from top to bottom).
Confusion maximization won over other adaptation techniques when adapting from FOOT (See embedding diagram on Figure 9b).

Fig. 7. Comparison of three averaging policies on Opportunity dataset. We pick adaptation performance and Opportunity dataset as they are especially susceptible to a class collapse. The Opportunity dataset has 5 classes – null, standing, walking, sitting, and lying. The *lying* class is a minority class taking only 3.8% of the samples.
The overall pattern of improvement (depicted as colors) is similar; yet, we observe that micro- and weighted-averaged F_1 scores show much higher values compared to macro-averaged F_1 score. The difference deepens as the domain shift increases. Such a difference is mainly caused by low precision/recall on the *lying* class. As such, we select macro-averaged F_1 score as the metric in rest of the paper to best reflect the effect of class collapse.



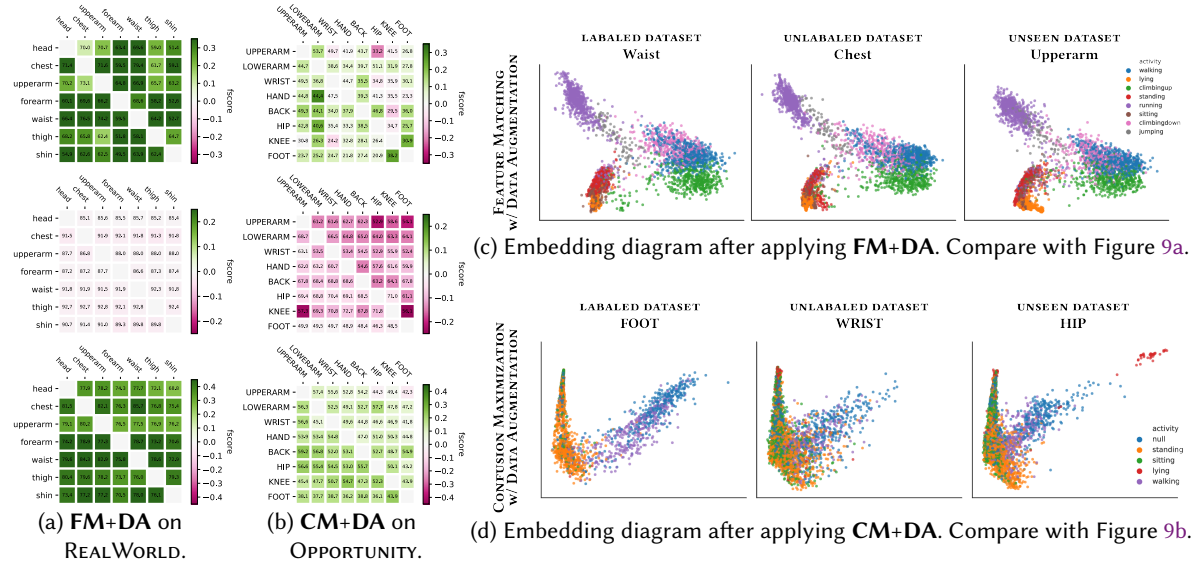
(a) HHAR dataset. (b) PAMAP2 dataset

Fig. 8. Comparison of three adaptation techniques on HHAR and PAMAP2 dataset. From left, each column depicts results for *Data augmentation*, *Feature matching*, and *Confusion maximization*. A dagger with a number besides shows the number of classes that collapsed, i.e. classes that no trace has been classified into. PAMAP2 dataset demonstrated occurrences of collapsing classes.



(a) RealWorld dataset (b) Opportunity dataset

Fig. 9. Embedding diagram demonstrating the effect of UDA techniques. Figure 9a represents the adaptation scenario that *feature matching* has shown a supreme performance. *Confusion maximization* fails to discriminate standing and laying. Figure 9b depicts the scenario that *confusion maximization* performed the best among three adaptation techniques. While all adaptation algorithms failed to formulate clear clusters, *confusion maximization* generated features that are most uniform across body positions.



(a) **FM+DA** on RealWorld.

(b) **CM+DA** on Opportunity.

(c) Embedding diagram after applying **FM+DA**. Compare with Figure 9a.

(d) Embedding diagram after applying **CM+DA**. Compare with Figure 9b.

Fig. 10. Performance improvement/decrease of **FM+DA** on RealWorld (Figure 10a) and **CM+DA** on Opportunity dataset (Figure 10b). Using UDA algorithms with data augmentation clearly outperforms the sole use of adaptation techniques. Nonetheless, the persistence was harmed as if *data augmentation* is applied. Embedding figures are drawn on the right (Figure 10c and 10d) highlights the effectiveness of using *data augmentation* with UDA algorithms.

Data augmentation. When it comes to Adaptability, surprisingly, *data augmentation* shows comparable performance to UDA algorithms. Data augmentation provided the best F_1 score among three adaptation techniques on 50 experimental settings, which is 47% of 106 experimental settings we experimented with. Data augmentation was particularly effective on body positions within the upper torso. This pattern could be found repeatedly on different datasets. In the case of RealWorld dataset, data augmentation outperformed other techniques on the following body positions: head, chest, upper arm, and forearm. Similar patterns can be observed for the Opportunity and PAMAP2 datasets.

Data augmentation, however, performs poorly from the viewpoint of Persistence. In most cases, data augmentation result in 5 to 10% accuracy drop on source body position. This is primarily because the statistical perturbations done to enable data augmentation may not correspond to real human behavior, which in turn degrades the accuracy of the source model. On the other hand, the unsupervised domain adaptation techniques use real traces collected from other body positions – as such, we expect UDA techniques to show better performance in terms of Persistence.

Feature matching. *Feature matching* was overall the most preferable adaptation technique, showing good performance on Adaptability, Persistence, and Generalizability. The application of feature matching technique has improved performance on target body position on 93.4% of all experimental settings. Data augmentation and confusion maximization improved only 90.6% and 74.5% of the pairs, respectively. When it comes to Persistence, the performance was superior; feature matching *improved* performance on the source body position on 54.7% of the pairs. Moreover, if we count the number of experiments that accuracy drop on the source body position was less then 5%, it added up to 105 pairs, which is 99% of total 106 pairs experimented.

Feature matching was able to learn the semantic similarity between body position pairs. When adapting from shin to body positions in upper torso (in RealWorld dataset), or from knee to other body positions (in Opportunity dataset), feature matching showed the best Adaptability on target body positions.

Figure 9a displays the experimental setting, Waist→Chest, in which feature matching shows the best performance across the three adaptation techniques. In the top row, we can observe that the feature embeddings of waist and chest before adaptation are quite different. By learning body-position-independent features using the Feature Matching technique, the embeddings displayed in the second row have a high alignment. Confusion maximization, on the other hand, worsens the feature embeddings in this case.

Confusion maximization. *Confusion maximization* turned out to be the weakest technique when it comes to Adaptability. This is a particularly interesting result because adversarial learning techniques such as Confusion maximization have shown significant improvements in computer vision literature for domain adaptation. However, for the task of wearing diversity in HAR, it was outperformed by other, arguably much simpler, techniques.

That said, we observed a few unique experimental settings where confusion maximization outperforms data augmentation and feature matching. On Opportunity dataset, when source body position is FOOT and we adapt to body positions in the upper torso, confusion maximization performs better than feature matching. As discussed in the baseline results, FOOT and body positions in the upper torso display a big domain shift between them, because the impact from the ground usually makes accelerometer signals collected from foot different from other upper body positions. This can be verified from the embedding in Figure 9b where the classification model trained on FOOT shows jumbled feature embeddings for WRIST (the first row). In this case, feature matching does not perform well, because the distance measure between two domains does not convey any useful information to the training process. While the distribution of the feature vectors resembles each other after applying feature matching techniques (second row), the distribution of activity within the shape is completely different. On the other hand, the embedding diagram from the confusion maximization approach succeeded in placing null activity traces (colored in blue) in a similar region and provided the best adaptation performance.

Combining UDA algorithms with data augmentation. Data augmentation can be used in conjunction with UDA algorithms. Before applying the UDA algorithms, we may enlarge the labeled dataset by applying data augmentation. We observe that combining these two techniques significantly improve the Adaptability of models (Figure 10). Feature matching technique combined with data augmentation (FM+DA) showed the best adaptation performance on 57.5% of body position pairs, winning over data augmentation, feature matching, and confusion maximization techniques applied independently. However, the Persistence was worse than feature matching, showing 5 to 10% of accuracy drop on the source body position. Interestingly, the combination of these techniques enhances the Generalizability potential of the model as can be inferred from the rightmost feature embeddings in Figure 10.

5.3 Use of Multiple Body Positions Together

The choice of body positions is critical for the success of UDA. In this section, we study whether clustering different body positions into groups enhances the performance of UDA techniques. In other words, instead of adapting models from a single source position to a single target position — if we do the adaptation over a group of positions, can it result in better performance? As shown in Table 3, we cluster the body positions based on skeletal adjacency and similarity in their degrees of freedom. For example, TORSO group comprises of samples collected from head, chest and waist.

Our findings reveal a clear pattern that could be of great practical significance for HAR developers. We observe that if the training dataset contains inertial data traces from (1) multiple body limbs and (2) body positions with high degrees of freedom, it provides the highest chance of building a robust body-position-invariant model that can be used with a variety of body positions.

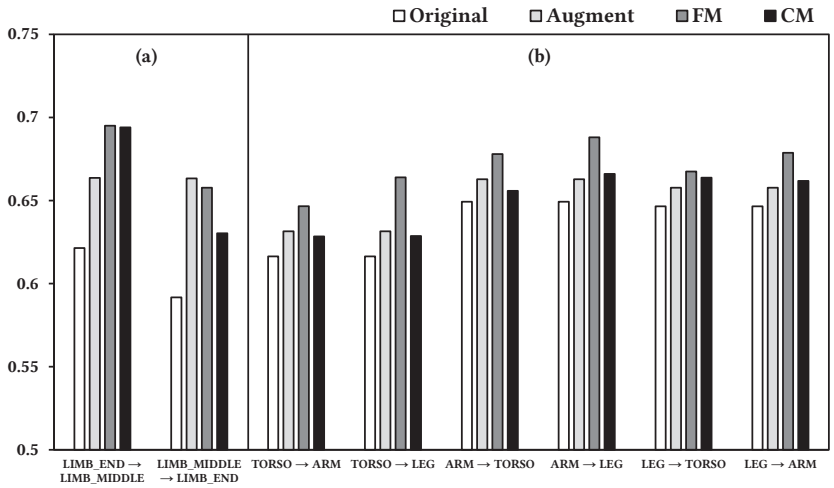
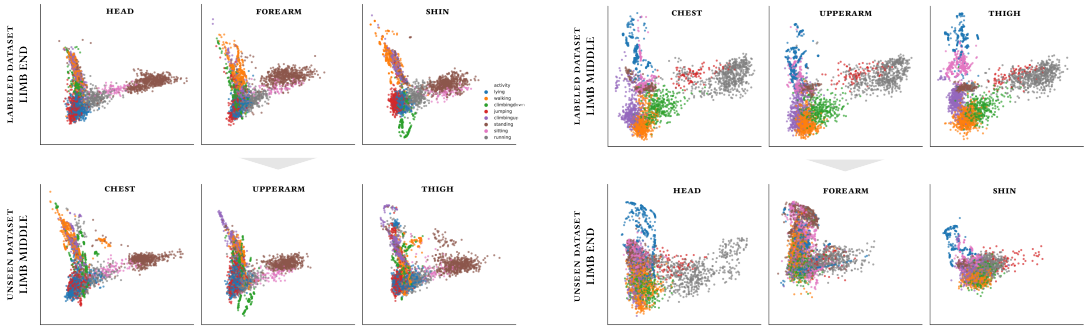


Fig. 11. F_1 score on all body positions (generalization). (a) shows that mixing body positions with high degrees of freedom for a labeled dataset is better for generalization. (b) also demonstrates that the unlabeled dataset collected from limb with high variability result in higher generalization performance.

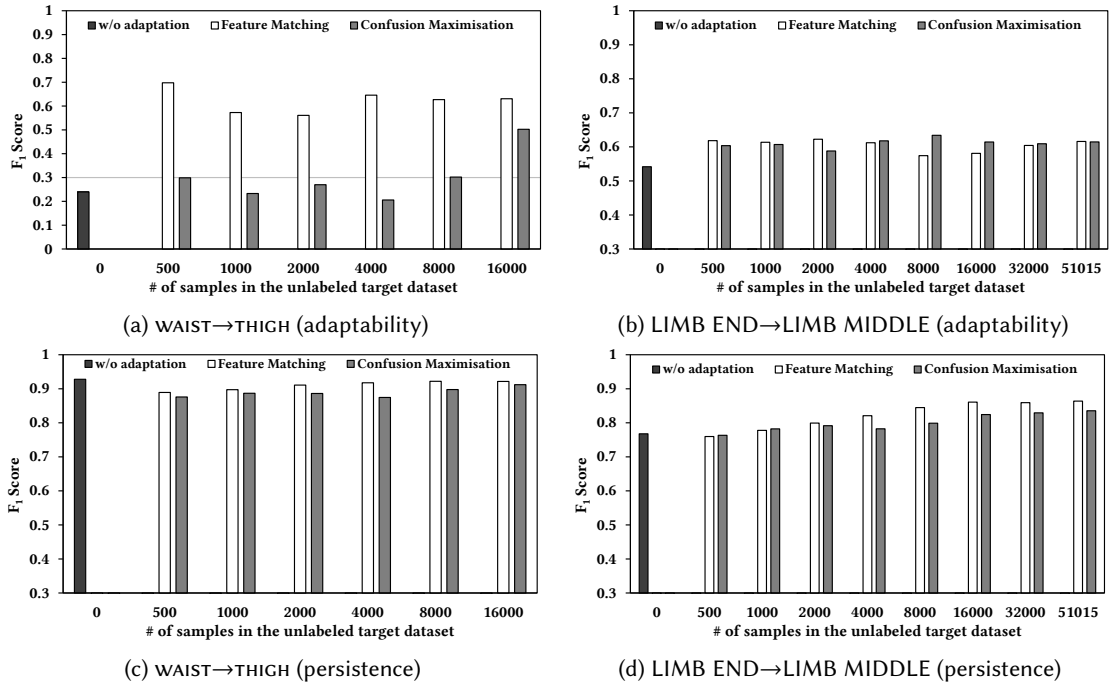


(a) Embedding diagram drawn with an encoder trained with LIMB END body position mixture.

(b) Embedding diagram drawn with an encoder trained with LIMB MIDDLE body position mixture.

Fig. 12. Embedding diagram explains why body position mixture with high variability performs better. While activity clusters are more clean and notable in Figure 12b, the encoder trained on LIMB MIDDLE fails to form clear clusters on *shin* and *forearm* which are body positions with high variability. Formation of activity clusters is essential for feature matching technique to work.

More specifically, on comparing the generalization performance of LIMB END→LIMB MIDDLE and LIMB MIDDLE→LIMB END, we observe that using LIMB END as a labeled dataset is more beneficial than using LIMB MIDDLE as a labeled dataset. Figure 12 explains this result. The upper row shows embeddings of the source domain, while the lower row shows the embedding of the target domain. While the activity clusters for the source domain (LIMB MIDDLE) look more clean and distinctive in Figure 12b, the model trained on LIMB MIDDLE eventually fails on forearm and shin, which are target domains with high degrees of freedom. This is also confirmed by Figure 11a which shows that feature matching fails to improve the adaptability of LIMB



(a) WAIST→THIGH (adaptability)

(b) LIMB END→LIMB MIDDLE (adaptability)

(c) WAIST→THIGH (persistence)

(d) LIMB END→LIMB MIDDLE (persistence)

Fig. 13. F_1 score measured on the source and target body position(s) varying the amount of the unlabeled target dataset.

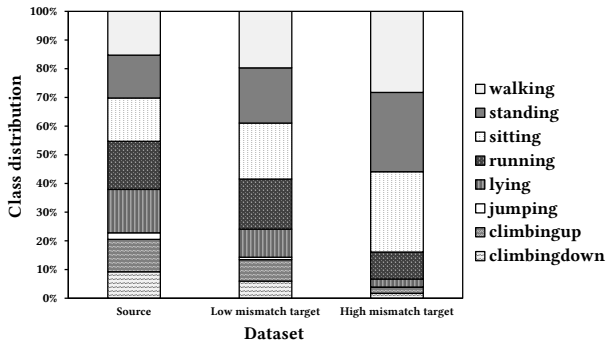
MIDDLE→LIMB END over data augmentation. On the contrary, the embedding diagram Figure 12a) generated by the HAR model trained on LIMB END may look jumbled (top row), but it succeeded in generating clearer activity clusters on LIMB MIDDLE (bottom row).

A similar pattern exists on the body position mixtures constrained to a single limb. If we compare the homogeneity within the body position mixture, TORSO will have the least heterogeneity, and ARM and LEG will have larger heterogeneity compared to TORSO. If we focus on the feature matching technique, we found that the generalizability increased when we select ARM or LEG as a labeled dataset. Also when a labeled dataset is fixed, the use of ARM or LEG as an unlabeled dataset further improves the generalizability.

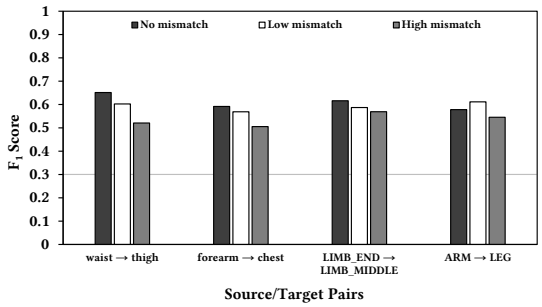
5.4 Effect of Dataset Properties

5.4.1 Size of Unlabeled Dataset. A key consideration in performing domain adaptation is the amount of unlabeled data needed in the target domain. Although the collection of unlabeled data is cheap, it is desirable if we can adapt models with as few data requirements as possible. In this vein, we now study how the size of the unlabeled dataset affects the performance of domain adaptation techniques.

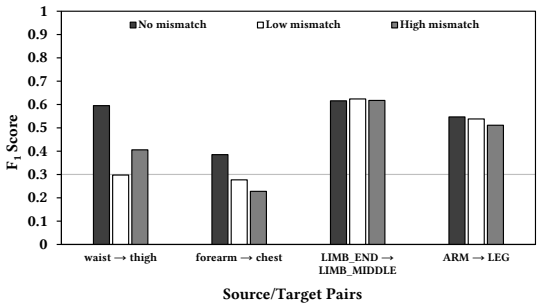
In Figure 13a and 13c, we vary the amount of unlabeled target domain data for the (WAIST→THIGH) adaptation. Our results show that the *feature matching* technique can provide accuracy improvements in the target domain with just 500 samples of unlabeled data, which suggests that explicitly adjusting feature embeddings of the model (i.e., by minimizing the MMD distance) works effectively with small amounts of data. On the other hand, the Confusion Maximization technique performs poorly with a small amount of unlabeled data, but its performance gradually improves as the size of the unlabeled dataset increases. Figure 13c shows the impact of unlabeled data on Persistence – here both techniques show gradual improvements in their *persistence* as the amount of unlabeled



(a) Class distribution of datasets simulating class distribution mismatch.



(b) Feature matching technique



(c) Confusion maximization technique

Fig. 14. F_1 score measured on target body position(s) varying the class distribution of the unlabeled target dataset.

data increases. We also observe similar patterns for *feature matching* and *confusion maximization* techniques for other combinations of body positions.

Next, we study the impact of unlabeled dataset size on adapting HAR classifiers developed using heterogeneous body position data. As illustrated in Figure 13b and 13d, we choose the (LIMB END→LIMB MIDDLE) as an example adaptation task. In contrast to the previous result on adapting for single-body positions (WAIST→THIGH), here we observe that both FM and CM techniques converge with just 500 samples of unlabeled data. The adaptability performance does not show a significant improvement thereafter, however, as more unlabeled data is fed to the model, its persistence improves significantly. In the interest of space, we do not report results of other heterogeneous adaptation tasks, however we observe a similar pattern of both techniques converging much faster for heterogeneous adaptation tasks as compared to single body position adaptation.

5.4.2 Class Distribution Mismatch in Adaptation. Although the collection of the unlabeled dataset from a target domain is cheap, it is important to note that we have little control over the class distribution in the unlabeled data, that is, we do not know a priori the type and proportion of different activity classes in the unlabeled target data. Therefore, it is critical that we evaluate how well do the UDA techniques work under class distribution mismatch between source and target domains.

We simulate three scenarios of class mismatch: a) *no mismatch* wherein the unlabeled target data has the same class distribution as the source domain, b) *low mismatch* wherein there is at least a 5% mismatch for each activity class, and c) *severe mismatch* which has at least a 10% mismatch for each activity class. The three scenarios were

simulated by undersampling from dynamic activities (climbing up/down, jumping, running) and oversampling from static activities (standing, sitting, lying) as illustrated in Figure 14a.

Figures 14b and 14c show the results of class mismatch for Feature Matching and Confusion Maximization technique. We observe that as the severity of class mismatch increases, the performance of both UDA techniques degrade for single position adaptation (i.e., WAIST→THIGH and FOREARM→CHEST). However, heterogeneous position adaptation is much more robust to class mismatch and particularly for the Confusion Maximization technique, there is a minimal impact of class mismatch on the adaptation performance. We surmise that using heterogeneous body positions inherently introduces significant variabilities in the adaptation process and as such, the model is able to learn robust domain-invariant features even under the presence of class distribution mismatch.

There are two main takeaways from our results: firstly, both the domain adaptation techniques are not robust against class mismatch scenarios and further research is needed to account for class mismatch in the adaptation process. Secondly, model developers can consider using heterogeneous body positions in the adaptation process as a first step towards alleviating the class mismatch problem.

5.5 Practical Considerations for Unsupervised Domain Adaptation

Collect training data from diverse body positions. HAR model developers should strive to have position diversity within a labeled dataset. In order to make HAR models generalize better and be body-position invariant, the labeled dataset should be collected from multiple body positions, and preferably, from the body positions with higher degrees of freedom (e.g., forearm). The diversity in the labeled dataset improves the stability and performance of unsupervised domain adaptation, even with a small amount of unlabeled data or class distribution mismatch.

Choice of UDA algorithms. Data augmentation remains a simple yet powerful technique to increase the generalizability of HAR models, and our results suggest that when used in combination with a UDA technique, data augmentation can significantly improve adaptation performance. Further, data augmentation could be useful to address wearing diversity between sessions or users – these diversities are primarily caused by change in sensor orientations [29] and are easy to simulate with data augmentation techniques. Finally, our results show that Feature Matching is a promising UDA technique and often outperforms more complex algorithms such as Confusion Maximization. More importantly, it can increase the performance of the model to a new body position, while maintaining its Persistence in the original position.

Sensitivity to class distribution. UDA is sensitive to class distribution. For a labeled dataset, the model developers should be aware of the existence of a minority class, and in particular watch out for a class collapse of the minority class in the adaptation process. For an unlabeled dataset, the class mismatch with the labeled dataset should be examined before applying a UDA technique, as class mismatch with the labeled dataset degrades the adaptation performance. We also note improving the performance of UDA under class or label space mismatch is an active area of research in the machine learning community [9, 28].

UDA is not a silver bullet. It is critical for model developers to understand the environment (e.g., body position) under which an unlabeled dataset is collected. This will help in choosing an appropriate UDA technique and estimate the effectiveness of domain adaptation. Our results show that UDA is not a "silver bullet" solution that can be applied blindly to any HAR model or an unlabeled dataset. It is most effective when body positions undergoing adaptation either have structural similarity or exhibit similar degrees of freedom, e.g., when the body positions are all placed in the upper torso. If there is a high domain shift between the labeled and unlabeled datasets, the improvement could be low or even negative — e.g., the adaptation from FOOT to HAND.

6 DISCUSSION

In this section, we discuss the limitations of this paper and outlook future research directions.

Alternative interpretations of wearing diversity. Our paper was limited to exploring wearing diversity when a inertial sensing device is placed on multiple body positions. However, wearing diversity can arise from other real-world scenarios as well. For example, Min et al. [29] explored intra-wearing, inter-wearing, and inter-user wearing variabilities, i.e., those that occur during or in between wearing sessions and users. Although an in-depth study of these variabilities was out of scope of our work, we surmise that carefully designed UDA techniques could be effective in these scenarios as well, because the domain shift induced by the body position variability is bigger compared to that from intra-wearing, inter-wearing or inter-user differences [29, 41].

Limited test on model architecture. Our investigation of unsupervised domain adaptation was limited to one model architecture. That said, our model architecture has shown on-par and sometimes better performance compared to the state-of-art CNN model [5] and therefore, it was a reasonable choice for our study. Nevertheless, in the real world, the model structure should be adjusted to match the available computing resources and it is important the understand how UDA performance will be affected by a change in the model structure.

Mobile and Wearable Implementations. In our work, we train the domain adaptation models centrally on the cloud and assume that unlabeled data from mobile and wearable devices will be uploaded to the cloud. Indeed, this raises a number of privacy issues for the users and hence, there is a need to explore distributed privacy-aware solutions for implementing domain adaptation and other transfer learning solutions on edge devices.

Extension to other UDA approaches. Unsupervised Domain Adaptation is a very active area of research for the machine learning community, and new algorithms are being proposed at a rapid pace. It is imperative that other reasonably mature UDA algorithms should be applied to the HAR task and evaluated under the framework of Adaptability, Persistence and Generalizability proposed in this paper.

Comparison with other fully-supervised and domain generalization algorithms. The recent holistic evaluation on semi-supervised learning [33] reported that carefully-tuned fully-supervised baselines which do not use unlabeled dataset at all can achieve similar accuracy with semi-supervised algorithms. Our result also reported cases where data augmentation wins over feature matching and confusion maximization techniques. Future work should expand the comparison to other fully-supervised/domain generalization algorithms, e.g., adversarial data augmentation [45], variational autoencoder [23], episodic training [26].

7 CONCLUSION

While wearable devices offer opportunities for novel interactive applications and ubiquitous monitoring, the quality of inertial sensing remains a bottleneck due to a number of real-world diversities. In this paper, we studied the potential of using unsupervised domain adaptation (UDA) to address the issue of wearing diversity. Our results show that UDA algorithms can improve the accuracy of HAR classifiers on unlabeled body positions by as much as 53%. We also uncovered several limitations and implicit data-related assumptions without which the UDA algorithms suffer a major degradation in accuracy. We hope that our work can serve as practical guidelines to help HAR practitioners and researchers in developing UDA-based HAR models.

REFERENCES

[1] 2017. New technology tracks food intake by monitoring wrist movements. http://gadgetsandwearables.com/2017/03/29/food-tracking/. Accessed: February 10, 2020.

[2] P Aggarwal, Z Syed, X Niu, and N El-Sheimy. 2008. A standard testing and calibration procedure for low cost MEMS inertial sensors and units. *The Journal of Navigation* 61, 2 (2008), 323–336.

[3] Ali Akbari and Roozbeh Jafari. 2019. Transferring activity recognition models for new wearable sensors with deep generative domain adaptation. In *IPSN*.

[4] Allan, Henrik Blunck, Sourav Bhattacharya, Thor Siiger Prentow, Mikkel Baun Kjærgaard, Anind K. Dey, Tobias Sonne, and Mads Møller Jensen. 2015. Smart Devices are Different: Assessing and MitigatingMobile Sensing Heterogeneities for Activity Recognition. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems, SenSys 2015, Seoul, South Korea, November 1-4, 2015*. 127–140. https://doi.org/10.1145/2809695.2809718

[5] Bandar Almaslukh, Abdel Artoli, and Jalal Al-Muhtadi. 2018. A Robust Deep Learning Approach for Position-Independent Smartphone-Based Human Activity Recognition. *Sensors* 18, 11 (2018), 3726.

[6] Stephen A. Antos, Mark V. Albert, and Konrad P. Kording. 2014. Hand, belt, pocket or bag: Practical activity tracking with mobile phones. *Journal of Neuroscience Methods* 231 (2014), 22 – 30. https://doi.org/10.1016/j.jneumeth.2013.09.015 Motion Capture in Animal Models and Humans.

[7] Phil Blunsom, Changhao Chen, Xiaoxuan Lu, Andrew Markham, Yishu Miao, Agathoniki Trigoni, and Linhai Xie. 2019. MotionTransformer: Transferring Neural Inertial Tracking Between Domains.

[8] Andreas Bulling, Ulf Blanke, and Bernt Schiele. 2014. A tutorial on human activity recognition using body-worn inertial sensors. *ACM Computing Surveys (CSUR)* 46, 3 (2014), 33.

[9] Zhangjie Cao, Lijia Ma, Mingsheng Long, and Jianmin Wang. 2018. Partial adversarial domain adaptation. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 135–150.

[10] Liwei Chan, Chien-Ting Weng, Rong-Hao Liang, and Bing-Yu Chen. 2014. AnyButton: Unpowered, Modeless and Highly Available Mobile Input Using Unmodified Clothing Buttons. In *Proceedings of the 5th Augmented Human International Conference* (Kobe, Japan) *(AH '14)*. ACM, New York, NY, USA, Article 24, 2 pages. https://doi.org/10.1145/2582051.2582075

[11] Blunck et al. 2013. On heterogeneity in mobile sensing applications aiming at representative data collection. In *Proceedings of the 2013 ACM Ubicomp*. ACM, 1087–1098.

[12] Davide Figo, Pedro C. Diniz, Diogo R. Ferreira, and João M. P. Cardoso. 2010. Preprocessing techniques for context recognition from accelerometer data. *Personal and Ubiquitous Computing* 14, 7 (01 Oct 2010), 645–662. https://doi.org/10.1007/s00779-010-0293-9

[13] Jérémy Frey, May Grabli, Ronit Slyper, and Jessica R. Cauchard. 2018. Breeze: Sharing Biofeedback Through Wearable Technologies. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. ACM, New York, NY, USA, Article 645, 12 pages. https://doi.org/10.1145/3173574.3174219

[14] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research* 17, 1 (2016), 2096–2030.

[15] Andreas Grammenos, Cecilia Mascolo, and Jon A Crowcroft. 2018. You Are Sensing, but Are You Biased?: A User Unaided Sensor Calibration Approach for Mobile Sensing. *IMWUT* 2 (2018), 11:1–11:26.

[16] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. 2012. A kernel two-sample test. *Journal of Machine Learning Research* 13, Mar (2012), 723–773.

[17] Nils Y. Hammerla, Shane Halloran, and Thomas Plötz. 2016. Deep, Convolutional, and Recurrent Models for Human Activity Recognition using Wearables. In *IJCAI*.

[18] W. Hang, W. Feng, R. Du, S. Liang, Y. Chen, Q. Wang, and X. Liu. 2019. Cross-Subject EEG Signal Recognition Using Deep Domain Adaptation Network. *IEEE Access* 7 (2019), 128273–128282. https://doi.org/10.1109/ACCESS.2019.2939288

[19] F. Kawsar, C. Min, A. Mathur, and A. Montanari. 2018. Earables for Personal-Scale Behavior Analytics. *IEEE Pervasive Computing* 17, 3 (Jul 2018), 83–89. https://doi.org/10.1109/MPRV.2018.03367740

[20] Adil Khan, Muhammad Siddiqi, and Seok-Won Lee. 2013. Exploratory Data Analysis of Acceleration Signals to Select Light-Weight and Accurate Features for Real-Time Activity Recognition on Smartphones. *Sensors* 13, 10 (Sep 2013), 13099âĂŞ13122. https://doi.org/10.3390/s131013099

[21] Md Abdullah Hafiz KHAN, Nirmalya Roy, and Archan Misra. 2018. Scaling human activity recognition via deep learning-based domain adaptation. (2018).

[22] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[23] Diederik P. Kingma and Max Welling. 2013. Auto-Encoding Variational Bayes. *CoRR* abs/1312.6114 (2013).

[24] Nicholas D Lane, Emiliano Miluzzo, Hong Lu, Daniel Peebles, Tanzeem Choudhury, and Andrew T Campbell. 2010. A survey of mobile phone sensing. *IEEE Communications magazine* 48, 9 (2010), 140–150.

[25] Nicholas D. Lane, Ye Xu, Hong Lu, Shaohan Hu, Tanzeem Choudhury, Andrew T. Campbell, and Feng Zhao. 2011. Enabling large-scale human activity inference on smartphones using community similarity networks (csn). In *UbiComp*.

[26] Da Li, Jianshu Zhang, Yongxin Yang, Cong Liu, Yi-Zhe Song, and Timothy M. Hospedales. 2019. Episodic Training for Domain Generalization. arXiv:cs.CV/1902.00113

[27] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. 2013. Rectifier nonlinearities improve neural network acoustic models. In *Proc. icml*, Vol. 30. 3.

[28] Akhil Mathur, Anton Isopoussu, Fahim Kawsar, Nadia Berthouze, and Nicholas Lane. 2019. FlexAdapt: Flexible Cycle-Consistent Domain Adaptation. In *Proceedings of the IEEE International Conference on Machine Learning and Applications*.

[29] Chulhong Min, Akhil Mathur, Alessandro Montanari, and Fahim Kawsar. 2019. An Early Characterisation of Wearing Variability on Motion Signals for Wearables. In *Proceedings of the 23rd International Symposium on Wearable Computers* (London, United Kingdom) *(ISWC '19)*. Association for Computing Machinery, New York, NY, USA, 166–168. https://doi.org/10.1145/3341163.3347716

[30] Prashanth Mohan, Venkata N Padmanabhan, and Ramachandran Ramjee. 2008. Nericell: rich monitoring of road and traffic conditions using mobile smartphones. In *Proceedings of the 6th ACM conference on Embedded network sensor systems*. ACM, 323–336.

[31] Le T. Nguyen, Ming Zeng, Patrick Tague, and Joy Zhang. 2015. I Did Not Smoke 100 Cigarettes Today!: Avoiding False Positives in Real-world Activity Recognition. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (Osaka, Japan) *(UbiComp '15)*. ACM, New York, NY, USA, 1053–1063. https://doi.org/10.1145/2750858.2804256

[32] Nhan Duc Nguyen, Duong Trong Bui, Phuc Huu Truong, and Gu-Min Jeong. 2018. Position-Based Feature Selection for Body Sensors regarding Daily Living Activity Recognition. *J. Sensors* 2018 (2018), 9762098:1–9762098:13.

[33] Avital Oliver, Augustus Odena, Colin A Raffel, Ekin Dogus Cubuk, and Ian Goodfellow. 2018. Realistic Evaluation of Deep Semi-Supervised Learning Algorithms. In *Advances in Neural Information Processing Systems 31*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Eds.). Curran Associates, Inc., 3235–3246. http://papers.nips.cc/paper/7585-realistic-evaluation-of-deep-semi-supervised-learning-algorithms.pdf

[34] Vern I Paulsen and Mrinal Raghupathi. 2016. *An introduction to the theory of reproducing kernel Hilbert spaces*. Vol. 152. Cambridge University Press.

[35] Liangying Peng, Ling Chen, Zhenan Ye, and Yi Zhang. 2018. AROMA: A Deep Multi-Task Learning Based Simple and Complex Human Activity Recognition Method Using Wearable Sensors. *IMWUT* 2 (2018), 74:1–74:16.

[36] Attila Reiss and Didier Stricker. 2012. Introducing a New Benchmarked Dataset for Activity Monitoring. In *Proceedings of the 2012 16th Annual International Symposium on Wearable Computers (ISWC) (ISWC '12)*. IEEE Computer Society, Washington, DC, USA, 108–109. https://doi.org/10.1109/ISWC.2012.13

[37] D. Roggen, A. Calatroni, M. Rossi, T. Holleczek, K. FÃűrster, G. TrÃűster, P. Lukowicz, D. Bannach, G. Pirkl, A. Ferscha, J. Doppler, C. Holzmann, M. Kurz, G. Holl, R. Chavarriaga, H. Sagha, H. Bayati, M. Creatura, and J. d. R. MillÃän. 2010. Collecting complex activity datasets in highly rich networked sensor environments. In *2010 Seventh International Conference on Networked Sensing Systems (INSS)*. 233–240. https://doi.org/10.1109/INSS.2010.5573462

[38] P. Siirtola and J. RÃűning. 2013. Ready-to-use activity recognition for smartphones. In *2013 IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*. 59–64. https://doi.org/10.1109/CIDM.2013.6597218

[39] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. 2014. Striving for Simplicity: The All Convolutional Net. arXiv:cs.LG/1412.6806

[40] Timo Sztyler and Heiner Stuckenschmidt. 2016. On-body Localization of Wearable Devices: An Investigation of Position-Aware Activity Recognition. In *2016 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE Computer Society, 1–9. https://doi.org/10.1109/PERCOM.2016.7456521 http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7456521.

[41] Timo Sztyler, Heiner Stuckenschmidt, and Wolfgang Petrich. 2017. Position-aware activity recognition with wearable devices. *Pervasive and mobile computing* 38 (2017), 281–295.

[42] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. 2014. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474* (2014).

[43] Dmitry Ulyanov, Andrea Vedaldi, and Victor S Lempitsky. 2017. Improved Texture Networks: Maximizing Quality and Diversity in Feed-forward Stylization and Texture Synthesis.. In *CVPR*, Vol. 1. 3.

[44] Terry T. Um, Franz M. J. Pfister, Daniel Pichler, Satoshi Endo, Muriel Lang, Sandra Hirche, Urban Fietzek, and Dana Kulić. 2017. Data Augmentation of Wearable Sensor Data for Parkinson's Disease Monitoring Using Convolutional Neural Networks. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction* (Glasgow, UK) *(ICMI 2017)*. ACM, New York, NY, USA, 216–220. https://doi.org/10.1145/3136755.3136817

[45] Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John Duchi, Vittorio Murino, and Silvio Savarese. 2018. Generalizing to Unseen Domains via Adversarial Data Augmentation. In *Proceedings of the 32Nd International Conference on Neural Information Processing Systems* (Montréal, Canada) *(NIPS'18)*. Curran Associates Inc., USA, 5339–5349. http://dl.acm.org/citation.cfm?id=3327345.3327439

[46] Dian xi Shi, Ran Wang, Yuan Wu, Xiaoyun Mo, and Jing Wei. 2017. A novel orientation- and location-independent activity recognition method. *Personal and Ubiquitous Computing* 21 (2017), 427–441.

[47] Chao Xu, Parth H. Pathak, and Prasant Mohapatra. 2015. Finger-writing with Smartwatch: A Case for Finger and Hand Gesture Recognition Using Smartwatch. In *Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications* (Santa Fe, New Mexico, USA) *(HotMobile '15)*. ACM, New York, NY, USA, 9–14. https://doi.org/10.1145/2699343.2699350