# LOCAL AND GLOBAL ALIGNMENTS FOR GENERALIZABLE SENSOR-BASED HUMAN ACTIVITY RECOGNITION

*Wang Lu*[1,2], *Jindong Wang*[*1,2,3], *Yiqiang Chen*[*1,2]

[1] Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[2] University of Chinese Academy of Sciences, Beijing, China
[3] Microsoft Research Asia, Beijing, China

## ABSTRACT

Sensor-based human activity recognition (HAR) plays an important role in our daily life. Most work on HAR often assumes that training and test samples follow the same data distribution, which is not realistic in practice. For example, activity patterns usually vary from person to person, which will hinder the generalization ability of the model. In this paper, we propose **L**ocal **A**nd **G**lobal alignment (LAG) for generalized sensor-based HAR. Our method is able to alleviate distribution shifts among training and test samples without touching test data. Specially, the proposed method learns domain-invariant features from both the local and global perspectives and utilizes combined features to classify. Comprehensive experimental evaluations are conducted on two benchmarks to demonstrate the superiority of the proposed method over state-of-the-art approaches.

***Index Terms***— Domain generalization, human activity recognition, domain-invariant feature

## 1. INTRODUCTION

Sensor-based human activity recognition (HAR) plays an important role in our daily life. HAR aims to learn high-level knowledge from low-level sensor inputs. It has been applied to many real-world applications such as healthcare, gesture recognition, and smart homes [1]. In recent years, deep learning is widely adopted in HAR and great progress has been made [2]. However, there are mainly two problems in deep learning-based HAR research. First, deep neural networks often require massive labeled data, which is time-consuming and expensive to obtain. Second, there often exist distribution shifts among training data and test samples. For example, even when two persons perform the same activity, the distributions of their signals could be different due to their different body shapes and lifestyles. Therefore, how to learn a good model that can generalize well to the test dataset based on limited training data remains a challenge.

To tackle this challenge, domain adaptation (DA) is proposed [3]. DA learns to maximize the performance on a test dataset (target domain) using the training dataset (source domain) by reducing their distribution divergence. In the past few years, DA received increasing attention and specifically, there is much prior work on DA-based HAR [4, 5, 6, 7]. While DA shows its ability for handling the domain shift problem, it needs access to test datasets, which may not be realistic in many situations. For example, we often want the model to be deployed directly to a new person without collecting or training on his data. Domain generalization (DG) is proposed for this more challenging situation [8]. The goal of DG is to learn a model that can generalize to an unseen test dataset that has different distributions from the training sets. According to [8], DG methods can be grouped into three categories: data manipulation, representation learning, and learning strategy. Many methods have been proposed for DG for computer vision and reinforcement learning [9, 10]. However, little work pays attention to domain generalization for HAR. A recent approach named Generalizable Independent Latent Excitation (GILE) [11] utilized DG for HAR based on a variational auto-encoder, which greatly enhances the cross-person generalization capability of the model. However, GILE is a rather general method and the structure of GILE is complicated, making it not easy to optimize.

In this paper, we propose a novel **L**ocal **A**nd **G**lobal alignment method for domain generalization on HAR, short as **LAG**. LAG learns domain-invariant features by exploiting the *local* and *global* correlations of the sensor signals. Specifically, we are more interested in using Convolutional Neural Net (CNN) as the feature extractor for its computational efficiency. On the one hand, CNN performs convolution operations on a given sequence by computing the correlation between the convolutional filter and local regions, which we call the local correlation. On the other hand, we compute the cross-region correlations which we call the global correlation. Both correlations are important to learn domain-invariant fea-

*Corresponding Authors
Email: {luwang, wangjindong, yqchen}@ict.ac.cn

Acknowledgements: This work is supported by National Key Research and Development Plan of China (No. 2018YFC2000605), Natural Science Foundation of China (No. 61972383, No. 61902377, No. 61902379), Beijing Municipal Science & Technology Commission No.Z211100002121171

tures to perform domain generalization. We propose two implementations of LAG. One is to directly utilize Convolution Neural Networks (CNN) to obtain local and global features. Another is to utilize a distance matrix to represent the front and back correlation of sensor-based time-series activity data. Our experimental results on two HAR benchmarks show that our method can significantly outperform state-of-the-art methods with large margins. The source code will be available at `https://github.com/jindongwang/transferlearning/tree/master/code/DeepDG`.

## 2. PROPOSED METHOD

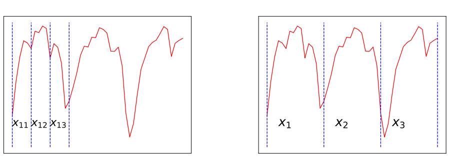
### 2.1. Problem Formulation

Following the definition of generalizable cross-domain activity recognition from existing work [11], we are given $N$ labeled source domains as the training dataset: $\mathcal{D}^{tr} = \{\mathcal{D}^i\}_{i=1}^N$. We use $P^i(\mathbf{x}, y)$ on $\mathcal{X} \times \mathcal{Y}$ to denote the joint distribution of one domain, where $\mathbf{x} \in \mathcal{X}$ denotes the input and $y \in \mathcal{Y} = \{1, \cdots, M\}$ corresponds to output. $C$ denotes the number of classes. Our goal is to learn a generalized model $h$ from $\mathcal{D}^{tr}$ to predict well on an unseen test domain, $\mathcal{D}^T$. In our problem, the training and test domains have the same input and output spaces but different distributions, i.e., $P^i(\mathbf{x}, y) \neq P^j(\mathbf{x}, y), \forall i, j \in \{1, 2, \cdots, N, T\}$. The overall objective is:

$$\min_h \mathbb{E}_{(\mathbf{x},y)\sim P^T}[h(\mathbf{x}) \neq y]. \tag{1}$$

### 2.2. Motivation

To harness the knowledge contained in sensor-based time-series data, we thoroughly analyze the activity signals. Take the walking data as shown in Figure 1 as an example. Figure 1(a) and 1(b) illustrate the local and global correlations, respectively. Using CNN as the feature extractor, Figure 1(b) can be split into three regions: $\{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$. Each region can be split into more fine-grained regions, e.g., $\mathbf{x}_{11}, \mathbf{x}_{12}, \mathbf{x}_{13}$. Obviously, correlation exists in both $(\mathbf{x}_i, \mathbf{x}_j)$ and $(\mathbf{x}_{ki}, \mathbf{x}_{kj})$ for $i, j, k \in \{1, 2, 3\}$. We refer to the correlation between all $(\mathbf{x}_i, \mathbf{x}_j)$ pairs as the *global* correlation and the correlation between all $(\mathbf{x}_{ki}, \mathbf{x}_{kj})$ pairs as the *local* correlations.




(a) Local correlation (b) Global correlation

**Fig. 1**. Data of walking activity to show the main idea of local and global correlation.

These two types of correlations are both important in sensor-based HAR. On the one hand, the regions $\{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$ may represent different stages of walking that are naturally related to each other, e.g., heel strike, support, and swing. On the other hand, inside each region, there still exists temporal correlation in one specific stage. Although Transformer [12] or some other methods can compute correlations for whole sequences, they are often difficult to tune parameters and the models are often large. And few of them take local and global alignments into consideration. To learn generalized models for HAR, we need to align both global and local correlations.

### 2.3. LAG: Local and Global Alignment

In this paper, we propose *LAG: local and global alignment* for generalized sensor-based HAR. We use two domains to illustrate the process of our method in Figure 2. The network mainly consists of two modules: local feature learning module and global feature learning module. Specifically, we adopt CNN as the local feature learning module since CNN mainly extracts features using the local connections of different regions. We propose two alternatives for the global feature learning module, which will be introduced in Section 2.4 as shown in Figure 3.

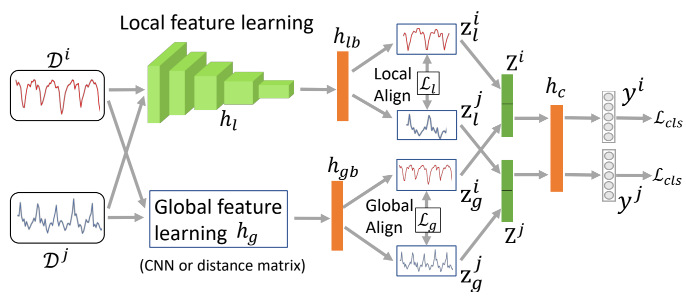



**Fig. 2**. The framework of LAG.

As shown in Figure 2, data from all domains are input in the local and global feature learning modules to extract the local features ($\mathbf{z}_l$) and global features ($\mathbf{z}_g$), respectively. Then, LAG performs the local and global alignment, which leads to two losses: $\mathcal{L}_l$ and $\mathcal{L}_g$. Subsequently, the local and global features are concatenated to form the final features ($\mathbf{z}$), which then goes through the classification layer ($h_c$) to compute the classification loss ($\mathcal{L}_{cls}$). Overall speaking, the learning objective for LAG is formulated as:

$$\mathcal{L} = \mathcal{L}_{cls} + \lambda_1 \mathcal{L}_l + \lambda_2 \mathcal{L}_g, \tag{2}$$

where $\lambda_1, \lambda_2$ are trade-off hyperparameters. Taking domain $\mathcal{D}^i$ as an example, the classification loss is computed as:

$$\mathcal{L}_{cls} = -\log h_c(\mathbf{z}^i). \tag{3}$$

### 2.4. Local Alignment

The local feature learning module directly utilizes a CNN to extract features, then we can perform alignment, which is formulated as:

$$\mathcal{L}_l = \frac{2}{N \times (N-1)} \sum_{i \neq j}^{N} ||\mathbf{C}_l^i - \mathbf{C}_l^j||_F^2, \tag{4}$$

where $||\cdot||_F^2$ denotes the matrix Frobenius norm and $\mathbf{C}_l^i$ is the covariance matrix of local features in the $i$-th domain, computed as:

$$\mathbf{C}_l = \mathrm{Cov}(h_f(\mathbf{x})), \tag{5}$$

where $\mathrm{Cov}(\cdot)$ denotes the covariance operation.

### 2.5. Global Alignment

Similar to local alignment, global alignment is formulated as:

$$\mathcal{L}_g = \frac{2}{N \times (N-1)} \sum_{i \neq j}^{N} ||\mathbf{C}_g^i - \mathbf{C}_g^j||_F^2, \tag{6}$$

where $\mathbf{C}_g$ denotes the covariance matrix for global features.

We propose two alternatives to compute the global features as shown in Figure 3: (1) a simple 2D CNN to extract features in two dimensions and (2) cross-covariance matrix for all regions. We denote LAG learned by these two alternatives as $\mathrm{LAG}_{\mathrm{CNN}}$ and $\mathrm{LAG}_{\mathrm{MAT}}$, respectively.

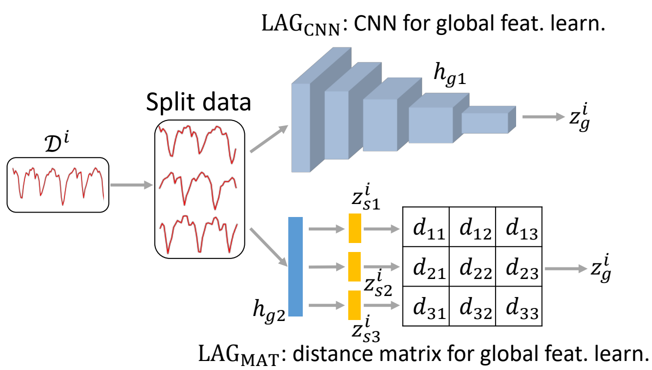



**Fig. 3**. Two alternatives for global feature learning.

$\mathrm{LAG}_{\mathrm{CNN}}$**:** We directly input the reshaped training data to a two-dimensional CNN $h_{g1}$ to get global features since 2-D convolution can find correlations on both rows and columns. The height of one reshaped data equals to the number of splits per sample while the width equals to the length of each split. To get correlations among each split (i.e., correlations among different rows), we simply set the height of the convolution kernel to the number of splits of a sample. Then, we can get global features $z_g$ and global alignment loss $\mathcal{L}_g$ by Eq. (6).

$\mathrm{LAG}_{\mathrm{MAT}}$**:** We send each split of a sequence to a simple layer $h_{g2}$ (Linear or AdaptiveAvgPool). For example, a sequence is split into 3 parts and all parts pass through the global feature net to obtain hidden representations $\mathbf{z}_{s=1}^3$ corresponding to yellow squares in Figure 3. Then, distances between each other are computed to obtain the distances matrix $\mathbf{D}$:

$$D_{s_1 s_2} = d(\mathbf{z}_{s_1}, \mathbf{z}_{s_2}), \tag{7}$$

where $s_1$ and $s_2$ are segment indices and $d(\cdot,\cdot)$ is a distance function which can be cosine distance or $l_1, l_2$-distance. We reshape $\mathbf{D}$ and pass it to the global bottleneck layer $h_{gb}$ to obtain the final global features $\mathbf{z}_g$. Then, the global alignment can be done following Eq. (6).

## 3. EXPERIMENT

We evaluate the proposed method on two publicly-available sensor-based HAR datasets.

### 3.1. Datasets and Implementation Details

UCI daily and sports dataset (**DSADS**) [13] consists of 19 activities with 1,140,000 samples collected from 8 subjects wearing body-worn sensors on 5 body parts. Each subject wears three sensors: accelerometer, gyroscope, and magnetometer. We divide DSADS into four domains and each domain contains data of two persons. USC-SIPI human activity dataset (**USC-HAD**) [14] is composed of 14 subjects (7 male, 7 female, aged from 21 to 49) executing 12 activities with a sensor tied on the front right hip. The data dimension is 6, the sample rate is 100Hz, and the dataset contains 5,441,000 samples. We also divide this dataset into 4 domains.

For DSADS, we directly utilize the dataset and each sequence is split into 5 parts in our methods. For USC-HAD, we adopt the sliding window technique with 50% overlap to construct training samples following common practice in HAR, and each sequence is split into 10 parts in our methods. We use $0, 1, 2, 3$ to denote the four divided domains.

For all benchmarks, we select the best model via validation accuracy. We train our model on the training splits and select the best model on the validation splits of all source domains. We leave 20% of source domain data as validation splits while the rest data are for training. For testing, we evaluate the selected models on all data of the held-out target domain.

We compare our methods with eight state-of-the-art methods, including ERM, DANN [15], CORAL [16], Transformer [12], GroupDRO [17], RSC [18], ANDMask [19], and GILE [11]. We reproduced all other methods with the same network architecture in Pytorch for fairness.

For LAG, the CNN contains two blocks, and each has one convolution layer, one pool layer, and one batch normalization layer. A single fully-connected layer is used as the bottleneck block while another fully-connected layer serves as the classifier. The batch size is 32 and the maximum training epoch is 150. We use the Adam optimizer with a learning rate $10^{-2}$ and weight decay $5 \times 10^{-4}$. We tune the hyperparameters for all methods for the best performance and repeat the experiments three times to report the average results.

### 3.2. Results

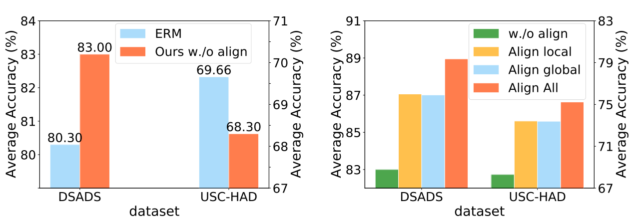
The classification results are shown in Table 1. On average, our proposed $\mathrm{LAG_{CNN}}$ and $\mathrm{LAG_{MAT}}$ substantially outperform the other methods: about 3.5% with $\mathrm{LAG_{CNN}}$ and about 3.7% with $\mathrm{LAG_{MAT}}$. For DSADS, $\mathrm{LAG_{CNN}}$ improves 3.1% while $\mathrm{LAG_{MAT}}$ improves about 3.8%. For USC-HAD, $\mathrm{LAG_{CNN}}$ improves 3.6% while $\mathrm{LAG_{MAT}}$ improves about 3.4%. This indicates that our methods are effective for generalizable cross-domain HAR applications.

**Table 1**. Results on DSADS and USC-HAD datasets. The **bold** and <u>underline</u> items are the best and the second-best.

| Method | DSADS 0 | 1 | 2 | 3 | AVG | USC-HAD 0 | 1 | 2 | 3 | AVG | ALL AVG |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ERM | 83.1 | 79.3 | 87.8 | 71.0 | 80.3 | 81.0 | 57.7 | 74.0 | 65.9 | 69.6 | 75.0 |
| DANN [15] | 89.1 | 84.2 | 85.9 | <u>83.4</u> | 85.6 | 81.2 | 57.9 | 76.7 | <u>70.7</u> | 71.6 | 78.6 |
| Transformer [12] | 81.3 | 82.1 | 79.6 | <u>83.5</u> | 81.6 | 78.2 | 62.6 | <u>78.1</u> | 63.6 | 70.6 | 76.1 |
| CORAL [16] | 91.0 | 85.8 | 86.6 | 78.2 | 85.4 | 78.8 | 58.9 | 75.0 | 53.7 | 66.6 | 76.0 |
| GroupDRO [17] | <u>91.7</u> | 85.9 | 87.6 | 78.3 | 85.9 | 80.1 | 55.5 | 74.7 | 60.0 | 67.6 | 76.7 |
| RSC [18] | 84.9 | 82.3 | 86.7 | 77.7 | 82.9 | 81.9 | 57.9 | 73.4 | 65.1 | 69.6 | 76.3 |
| ANDMask [19] | 85.0 | 75.8 | 87.0 | 77.6 | 81.4 | 79.9 | 55.3 | 74.5 | 65.0 | 68.7 | 75.0 |
| GILE [11] | 81.0 | 75.0 | 77.0 | 66.0 | 74.7 | 78.0 | 62.0 | 77.0 | 63.0 | 70.0 | 72.4 |
| $\mathrm{LAG_{CNN}}$ | 91.2 | <u>88.8</u> | **92.7** | 83.1 | <u>89.0</u> | **84.4** | **68.6** | **79.2** | 68.8 | **75.2** | <u>82.1</u> |
| $\mathrm{LAG_{MAT}}$ | **95.2** | **89.4** | <u>91.7</u> | 82.4 | **89.7** | <u>82.9</u> | <u>67.5</u> | 76.5 | **73.0** | <u>75.0</u> | **82.3** |

We observe more insightful conclusions. (1) Both $\mathrm{LAG_{CNN}}$ and $\mathrm{LAG_{MAT}}$ achieve the best performance while $\mathrm{LAG_{MAT}}$ is slightly better than $\mathrm{LAG_{CNN}}$. This may be caused by that the computation of covariance as the global features can capture the global correlations better than a 2D CNN. However, the computation of distance matrix can certainly introduce more computations that needs a tradeoff for real applications. (2) Other domain generalization methods achieve better performance than simple ERM, indicating that learning from multiple domains is a challenging problem due to the distribution gaps in these domains. (3) While Transformer can also obtain good results by capturing the global relation using self-attention, it still does not outperform our method. The reason could be that it is not enough to only compute the correlations as features, but we need to align their distributions.

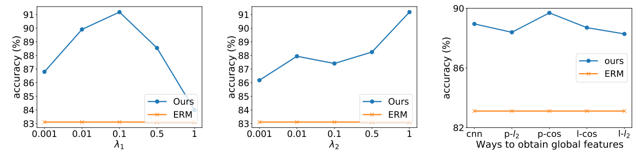
### 3.3. Ablation Study




(a) Without alignments (b) Different alignments

**Fig. 4**. Ablation study of LAG.

We perform ablation study in Figure 4. Firstly, Figure 4(a) shows that LAG without any alignments, but only introduces two feature learning modules has different performance on two datasets (better than ERM for DSADS dataset but worse on USC-HAD), indicating the necessity of performing alignment between domains. Secondly, Figure 4(b) shows that LAG with only local alignment or only global alignment can bring improvements compared with LAG without any alignments. And LAG with both local and global alignments achieves the best performance. These experiments demonstrate that local features and global features combined with alignments can bring stable and remarkable improvements. The experiments are based on $\mathrm{LAG_{CNN}}$, while the conclusions are also the same for $\mathrm{LAG_{MAT}}$.




(a) $\lambda_1$ (b) $\lambda_2$ (c) Feature Extraction

**Fig. 5**. Parameter sensitivity of LAG.

### 3.4. Parameter Sensitivity

We evaluate the parameter sensitivity of LAG in Figure 5. There are mainly three hyperparameters in our method: $\lambda_1$ for local alignment, $\lambda_2$ for global alignment, and different ways to obtain global features. From Figure 5(a) and Figure 5(b), we can see that the results with parameters around the highest points are all better than ERM. Figure 5(c) demonstrates that different ways to obtain global features have different performances and we should choose the right one for better results. p-$l_2$ means LAG with an AdaptiveAvgPool layer and $l_2$ distance while l-cos means LAG with linear layers and cosine distance. p-cos and l-$l_2$ have similar meanings. In a nutshell, the results demonstrate that LAG is effective and robust that can be easily applied to real applications.

## 4. CONCLUSION

In this paper, we proposed LAG for generalizable sensor-based human activity recognition. LAG utilizes CNNs as backbones which are simple and fast and can be easily applied in real applications. For better generalizable performance, LAG introduces both local alignment and global alignment. Extensive experiments on two benchmarks demonstrate the effectiveness of our method.

In the future, we plan to exploit more global futures with simple extensions for HAR. And we also plan to apply our algorithm to more complicated activity recognition and even larger HAR datasets.

## 5. REFERENCES


[1] Kaixuan Chen, Dalin Zhang, Lina Yao, Bin Guo, Zhiwen Yu, and Yunhao Liu, "Deep learning for sensor-based human activity recognition: Overview, challenges, and opportunities," *ACM Computing Surveys (CSUR)*, vol. 54, no. 4, pp. 1–40, 2021.

[2] Jindong Wang, Yiqiang Chen, Shuji Hao, Xiaohui Peng, and Lisha Hu, "Deep learning for sensor-based activity recognition: A survey," *Pattern Recognition Letters*, vol. 119, pp. 3–11, 2019.

[3] Sinno Jialin Pan and Qiang Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2009.

[4] Xin Qin, Yiqiang Chen, Jindong Wang, and Chaohui Yu, "Cross-dataset activity recognition via adaptive spatial-temporal transfer learning," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 3, no. 4, pp. 1–25, 2019.

[5] Diane Cook, Kyle D Feuz, and Narayanan C Krishnan, "Transfer learning for activity recognition: A survey," *Knowledge and information systems*, vol. 36, no. 3, pp. 537–556, 2013.

[6] Wang Lu, Yiqiang Chen, Jindong Wang, and Xin Qin, "Cross-domain activity recognition via substructural optimal transport," *Neurocomputing*, vol. 454, pp. 65–75, 2021.

[7] Jindong Wang, Yiqiang Chen, Lisha Hu, Xiaohui Peng, and S Yu Philip, "Stratified transfer learning for cross-domain activity recognition," in *2018 IEEE international conference on pervasive computing and communications (PerCom)*. IEEE, 2018, pp. 1–10.

[8] Jindong Wang, Cuiling Lan, Chang Liu, Yidong Ouyang, Wenjun Zeng, and Tao Qin, "Generalizing to unseen domains: A survey on domain generalization," in *International Joint Conference on Artificial Intelligence*, 2021.

[9] Fabio M Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi, "Domain generalization by solving jigsaw puzzles," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2229–2238.

[10] Qinwei Xu, Ruipeng Zhang, Ya Zhang, Yanfeng Wang, and Qi Tian, "A fourier-based framework for domain generalization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14383–14392.

[11] Hangwei Qian, Sinno Jialin Pan, Chunyan Miao, H Qian, SJ Pan, and C Miao, "Latent independent excitation for generalizable sensor-based cross-person activity recognition," in *AAAI*, 2021, pp. 11921–11929.

[12] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," in *NeurIPS*, 2017, pp. 5998–6008.

[13] Billur Barshan and Murat Cihan Yüksek, "Recognizing daily and sports activities in two open source machine learning environments using body-worn sensor units," *The Computer Journal*, vol. 57, no. 11, pp. 1649–1667, 2014.

[14] Mi Zhang and Alexander A Sawchuk, "Usc-had: a daily activity dataset for ubiquitous activity recognition using wearable sensors," in *Proceedings of the 2012 ACM conference on ubiquitous computing*, 2012, pp. 1036–1043.

[15] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky, "Domain-adversarial training of neural networks," *The journal of machine learning research*, vol. 17, no. 1, pp. 2096–2030, 2016.

[16] Baochen Sun, Jiashi Feng, and Kate Saenko, "Return of frustratingly easy domain adaptation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2016, vol. 30.

[17] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang, "Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization," in *International Conference on Learning Representations (ICLR)*, 2020.

[18] Zeyi Huang, Haohan Wang, Eric P Xing, and Dong Huang, "Self-challenging improves cross-domain generalization," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*. Springer, 2020, pp. 124–140.

[19] Giambattista Parascandolo, Alexander Neitz, ANTONIO ORVIETO, Luigi Gresele, and Bernhard Schölkopf, "Learning explanations that are hard to vary," in *International Conference on Learning Representations*, 2021.