


## Article

# Segment-Based Unsupervised Learning Method in Sensor-Based Human Activity Recognition

Koki Takenaka <sup>\*</sup>, Kei Kondo <sup>†</sup> and Tatsuhito Hasegawa 

Graduate School of Engineering, University of Fukui, Fukui 910-8507, Japan; kicont115sub@gmail.com (K.K.); t-hase@u-fukui.ac.jp (T.H.)

<sup>\*</sup> Correspondence: ktakenak@u-fukui.ac.jp

<sup>†</sup> Current address: FLECT Co., Ltd., Tokyo 105-0023, Japan.

**Abstract:** Sensor-based human activity recognition (HAR) is a task to recognize human activities, and HAR has an important role in analyzing human behavior such as in the healthcare field. HAR is typically implemented using traditional machine learning methods. In contrast to traditional machine learning methods, deep learning models can be trained end-to-end with automatic feature extraction from raw sensor data. Therefore, deep learning models can adapt to various situations. However, deep learning models require substantial amounts of training data, and annotating activity labels to construct a training dataset is cost-intensive due to the need for human labor. In this study, we focused on the continuity of activities and propose a segment-based unsupervised deep learning method for HAR using accelerometer sensor data. We define segment data as sensor data measured at one time, and this includes only a single activity. To collect the segment data, we propose a measurement method where the users only need to annotate the starting, changing, and ending points of their activity rather than the activity label. We developed a new segment-based SimCLR, which uses pairs of segment data, and propose a method that combines segment-based SimCLR with SDFD. We investigated the effectiveness of feature representations obtained by training the linear layer with fixed weights obtained by unsupervised learning methods. As a result, we demonstrated that the proposed combined method acquires generalized feature representations. The results of transfer learning on different datasets suggest that the proposed method is robust to the sampling frequency of the sensor data, although it requires more training data than other methods.



**Citation:** Takenaka, K.; Kondo, K.; Hasegawa, T. Segment-Based Unsupervised Learning Method in Sensor-Based Human Activity Recognition. *Sensors* **2023**, *23*, 8449. <https://doi.org/10.3390/s23208449>

Academic Editor: Eui Chul Lee

Received: 24 August 2023

Revised: 22 September 2023

Accepted: 11 October 2023

Published: 13 October 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** human activity recognition; unsupervised representation learning; accelerometer sensor data; segment data

## 1. Introduction

Human activity recognition (HAR) using sensor data is a task that recognizes human activities wherein the data are measured by sensors such as accelerometers and gyroscopes. For example, human activities such as “walking” and “running” were predicted from the measured accelerometer data. This technology is used in various applications from analyzing sports movements [1] to healthcare such as health awareness maintenance and the detection of risky movements by patients [2–4]. HAR is an essential technology because the predicted results have some influence on decision-making.

Recently, neural networks (NNs) have been used for HAR [5–8]. HAR is typically implemented using machine learning methods [9,10] such as support vector machines (SVMs) and hidden Markov models (HMMs). Machine learning models require a training dataset consisting of handcrafted features extracted from sensor data and corresponding activity labels. In contrast to traditional machine learning, deep learning methods such as NNs have the advantage of automatic feature extraction, which enables end-to-end training and prediction from raw sensor data. Therefore, deep learning models can adapt to various situations [11] without requiring domain expertise for feature extraction. NNs

require substantial amounts of training data to acquire generalized feature representations from sensor data compared to traditional methods. In recent years, the widespread use of smartphones and wearable devices has made it easy to measure sensor data. However, the annotation process is time-consuming and labor-intensive to generate a large amount of training data. Thus, there is a need for an unsupervised learning method for NNs.

In the case of sensor-based HAR, we can design how to measure the sensor data for the training dataset. For example, there are two measurement methods for HAR: the sequential measurement of multiple activities and the individual measurement of each activity. When each activity is measured individually without annotation, the accurate activity of the measurement data is unclear (annotator needs to determine the activity label using video, etc.) However, it can be understood that each of the data corresponds to a single activity.

Our purpose was to develop an unsupervised learning method that leverages characteristics implicitly possessed by sensor data for sensor-based HAR using deep learning methods. Existing unsupervised learning techniques for time series data are based on time series characteristics [12,13] or masks [14], but these methods do not focus on the structure of the training data. Xiao et al. [15] proposed a method employing contrastive loss. This approach calculates the loss in the intermediate layers, and it is not entirely independent of the model's structure. Franceschi et al. [16] introduced triplet loss (T-Loss), which selects positive samples from the same subseries as a given time series and randomly selects negative samples from other subseries. When utilizing segmented data, this selection method involves considering the data structure, where positive samples should be chosen from the same segment and negative samples should be chosen from different segments. SimCLR employs the normalized temperature-scaled cross-entropy (NT-Xent) loss [17,18], an expanded loss function that is derived from the triplet loss by incorporating multiple negative samples. As this loss function is compatible with the approach of Franceschi et al. [16], there is room for improvement in utilizing segmented data. We investigated a method that focuses on the data structure for unsupervised learning with discriminators [19]. In this study, we introduced SimCLR, which performs contrastive feature representation based on segments, to segment discrimination and feature decorrelation (SDFD) [19]. We combined these methods to develop a sensor-based unsupervised HAR learning method that does not rely on model structure, but focuses on the structure of segmented data. We refer to segment-based SimCLR as SimCLR (seg) and the combined method segment-based SimCLR with SDFD as SimCLR (seg) + SDFD. In our experiments, in order to evaluate the generalization performance of the proposed method, we conducted a linear evaluation under the two task settings: transfer learning within the same domain and transfer learning across different domains.

In summary, our contributions are as follows:

- We propose a new unsupervised learning method, SimCLR (seg) + SDFD, for sensor-based HAR in an environment where the training dataset consists of segmented data. In the proposed method, segmented data are used only during training and are not used during the evaluation.
- The ablation study showed that SimCLR (seg) and SDFD improved the f1-score compared to methods that do not focus on the data structure. This shows that focusing on the structure of the training dataset is effective.
- We demonstrated that our proposed method acquires generalized feature representations for sensor-based HAR through our experiments.

## 2. Related Works

In sensor-based HAR, models are typically trained using data consisting of sensor data and corresponding activity labels. Liu et al. [10] and Hartmann et al. [20] worked on real-time HAR using HMMs, and Liu et al. [21] demonstrated that HMMs are computationally efficient and fully adaptable to real-time applications. Bento et al. [22] demonstrated that traditional machine learning methods using handcrafted features of sensor data can generalize to data from different domains than the training data. These traditional methods

require manual feature extraction, whereas deep learning methods extract features automatically. Thus, deep learning methods can be trained and predicted end-to-end and have demonstrated their adaptive capabilities [11].

For sensor-based HAR using deep learning methods, various methods using convolutional NNs (CNNs) [5–8] and transformers [23] are proposed. Mahmud et al. [24] proposed a method designed to acquire useful features for HAR from different perspectives by applying multiple transformation methods to sensor data. Sanchez et al. [8] proposed a method that converts sensor data into an image format for HAR. These studies focused on an input format for sensor data to improve recognition accuracy. Approaches to improve the architecture of NNs for HAR have been proposed, such as a method that uses multiple models with different roles [25] and another that uses attention mechanisms [26]. Some studies tackle representation learning with activity labels [27,28]. These studies focus on acquiring feature representations for HAR using datasets with activity labels. Deep learning models require more training data for automatic feature extraction [24,29]. In recent years, the widespread use of smartphones and wearable devices has enabled the collection of large amounts of sensor data. However, annotators need to manually annotate activity labels for each of the measurement data, which requires more-significant human resources and time. On the other hand, this study focuses on learning methods without activity labels, which in turn reduces annotation costs.

In image recognition, several unsupervised learning methods have been proposed [30,31]. Tao et al. [32] proposed a clustering-friendly representation learning method by utilizing instance discrimination (ID) [33], which classifies each of the data as a unique class and removes the redundant correlation of features. Chen et al. [18] proposed SimCLR as a contrastive learning method that acquires feature representations by a straightforward method. Grill et al. [34] proposed bootstrap your own latent (BYOL) as an efficient representation learning method without using negative examples. Since these methods do not depend on a model architecture, they can be applied to unsupervised learning methods with sensor data by changing an encoder architecture into layers for sensor data.

Methods for segmenting unlabeled sensor data have been proposed. Rodrigues et al. [35] proposed a method for detecting the similarity between continuous time series data. Folgado et al. [36] utilized query sequences for data segmentation. These methods require continuous measurement data or query sequences to compute the similarity and features of time series data. Our approach, on the other hand, allows the model trained with our method to compute features from short-term measurement data without the need to retain training data during inference. Furthermore, these methods focus on unsupervised segmentation of time series data. Our approach focuses on training with segmented unsupervised data, and the trained model can be applied to other classification tasks using short-term data.

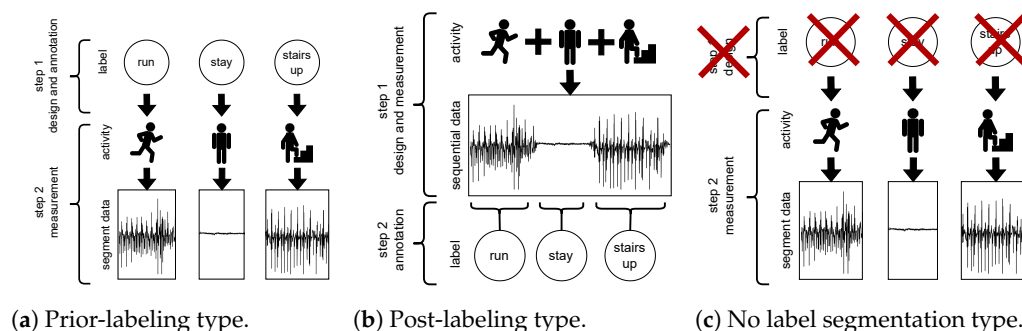
Moreover, unsupervised learning methods for HAR using deep learning methods have been proposed to acquire feature representations from unlabeled sensor data [16,29]. Focusing on time series, Haresamudram et al. [12] proposed a method to use contrastive predictive coding [37] for HAR. The method trains a model by comparing a feature map obtained from sensor data with a predicted feature map. Ma et al. [29] proposed a loss function that uses a reconstruction error, a neighborhood of manually designed features, and a neighborhood of time series data to use an auto-encoder (AE). Tonekaboni et al. [13] introduced the concept of a neighborhood with stationary time series data to propose a neighborhood-based unsupervised learning method for non-stationary multivariate time series data. These methods utilize the neighborhood of time series data. We also used the neighborhood of time series data as the segment data to acquire the feature representations for HAR.

### 3. Proposed Method

#### 3.1. Data-Measurement Method

Figure 1a,b show two typical measurement methods for HAR. The prior-labeling type in Figure 1a shows that an activity is labeled before the subject initiates an activity.

For example, to start the walking measurement, the subject presses the “walk” button. Then, to start the running measurement, the subject presses the “run” button. Therefore, in the prior-labeling type (Figure 1a), subjects need to select the activity label before the measurement each time. The post-labeling type in Figure 1b shows that an activity is labeled after the subject initiates an activity. For example, the subjects move freely with the attached sensor while his/her behavior is recorded by video. After the measurement, the recorded video is visually checked, and each activity is labeled at the given timestamp.



**Figure 1.** Data collection methods used in sensor-based human activity recognition (HAR).

These methods have advantages and disadvantages. Comparing the prior-labeling type (Figure 1a) and the post-labeling type (Figure 1b), the prior-labeling type requires subjects to select an activity label before and after the measurement and is measured for each activity. On the other hand, the post-labeling type does not require the selection of an activity label. In addition, the post-labeling type can continuously measure a series of activities, which reduces the amount of work before and after the measurement, thus reducing the subject’s burden compared to the prior-labeling type. In the prior-labeling type, an annotator does not need to perform any work after the measurement because the activity label is selected before the measurement. In the post-labeling type, the annotator must manually assign the activity label corresponding to the measurement data. As specific examples, Kwapisz et al. [38] and Zhang et al. [39] used the prior-labeling type; Ichino et al. [40,41] and Chavarriaga et al. [42] used the post-labeling type for measurement.

The aim of our method was to acquire feature representations of single activities. The single activity is a motion that consists of a single movement, such as jumping, sitting, or lying down, or a cycle of several consecutive movements, such as walking, running, or climbing stairs [43]. We call data that record one continuous single activity a segment. Our method does not require the annotation of the segment data.

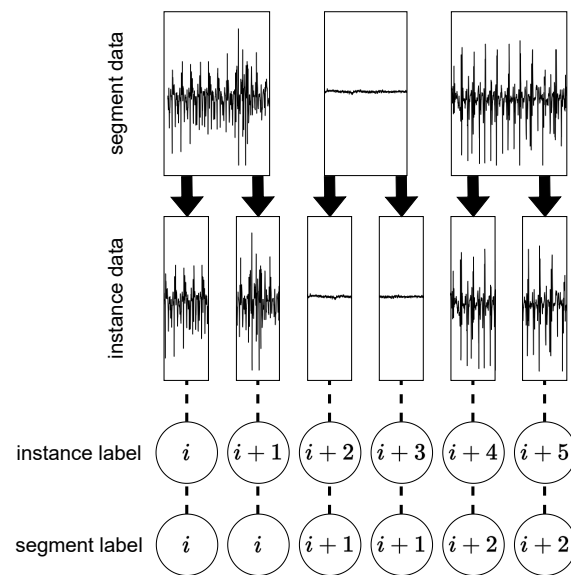
In our proposed method, we used a dataset that can be created with the unlabeled segmented measurement type shown in Figure 1c. This measurement method enables automatic segmentation of measurement data and eliminates the need for annotation work from the measurement methods used in related studies [44,45]. Compared to the two measurement types, the unlabeled segmented measurement type only requires the subject to indicate the start and end of the measurement, which can be any activity in the segmented data.

### 3.2. Preprocessing

We utilized unlabeled segment data for unsupervised learning. We used benchmark datasets in which the sensor data are stored as segment data. In the case of a benchmark dataset that does not have stored segment data, we need to create segment data using segmentation methods [35,36].

A training dataset  $D$  consists of input data created from segment data and pseudo-label pairs. The input data are instances of segment data divided by a sliding window method with a window size  $L$  and a stride width  $S$ . We show the creation method of pseudo-labels in Figure 2. The pseudo-labels are unique values assigned to the segment data and are called segment labels. On the other hand, pseudo-labels assigned to instance data are called

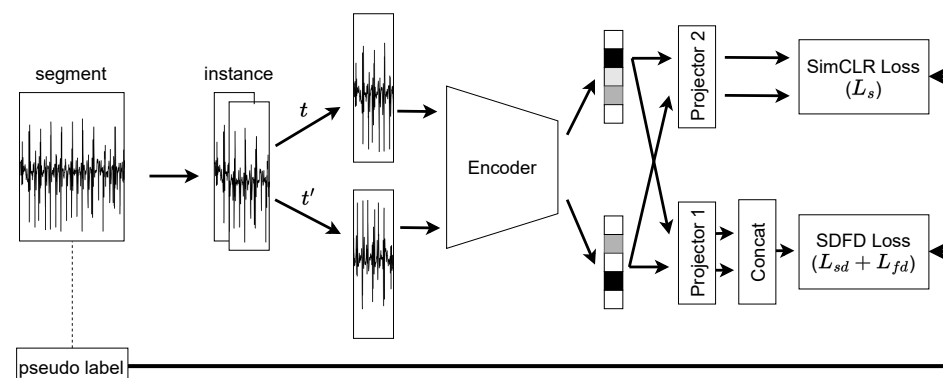
instance labels. A total number of instance labels and segment labels, denoted by  $N$  and  $M$ , respectively, has a relation  $M \leq N$ .



**Figure 2.** The difference in assigning pseudo-labels used in instance discrimination (ID) and segment discrimination (SD). An instance label (upper) is assigned to the input data in ID. A segment label (lower) is assigned to the segment data in the proposed SD. Therefore, the same segment label can be assigned to different input data.

### 3.3. Model Training Method

Figure 3 shows an overview of the proposed method utilizing segment data. In the proposed method, the model consists of an encoder  $f_e$  and two projectors  $f_{p1}, f_{p2}$ . The encoder is used to acquire a feature representation of segment data, and the projectors are used for segment label prediction and contrastive loss calculation. We updated the model parameters by using loss functions in SimCLR and SDFD.



**Figure 3.** Overview of the proposed unsupervised learning method. We refer to the proposed segment-based SimCLR with segment discrimination and feature decorrelation as SimCLR (seg) + SDFD. For the model input, instance data are created from segment data upon which the model is then trained using the proposed loss function.

#### 3.3.1. Segment Discrimination

Segment discrimination (SD) is an ID-based method that differs from ID in the assignment of pseudo-labels. ID is an unsupervised learning method proposed by Wu et al. [33]. The SD method uses pseudo-labels like ID, but it uses segment labels instead of instance labels, as shown in Figure 2. In the SD implementation, the memory bank [33] size is the same as the number of segment labels. The memory bank is a set of normalized feature

representations. Meanwhile, for SD, a model is trained in the same manner as ID using a segment label instead of an instance label. When the number of segments used in SD is  $M$ , the memory bank is  $\{f_1, f_2, \dots, f_M\}$ . When the feature  $f = f_{p2}(f_e(x))$  obtained from the input data  $x$  and the segment label is  $s$ , the probability  $P(s|f)$  is expressed as follows:

$$P(s|f) = \frac{\exp(f_s^\top f)}{\sum_{j=1}^S \exp(f_j^\top f)}. \quad (1)$$

In SD, it is known that the  $i$ -th input data  $x_i$  is the  $s_i$ -th segment. Therefore, the loss function  $L_{sd}$  using the feature in the memory bank is

$$L_{sd} = - \sum_{i=1}^N \log P(s_i|f_i) = - \sum_{i=1}^N \log \frac{\exp(f_{s_i}^\top f)}{\sum_{j=1}^S \exp(f_j^\top f)}. \quad (2)$$

There are two advantages to using SD. The first is that the number of output dimensions can be fixed. The total number of segment labels is less than the total number of instance labels. However, compared to the output size of a typical classification model, it can be extremely large. SD is solved in the same way as ID. Second, it is possible to utilize information implicit in the segments. The assigned segment labels allow input data based on the segment data to output the same features as the same activity. This is expected to be robust to phase differences between the input data.

### 3.3.2. SD and Feature Decorrelation

SD and feature decorrelation (SDFD) is a method using SD for IDFD as proposed by Tao et al. [32]. The loss function  $L_{idfd}$  used in IDFD is a combination of the loss function  $L_{id}$  used in ID and a loss function  $L_{fd}$  called feature-independent softmax (FIS). Since SDFD is based on SD, the loss function of SDFD is  $L_{sdfd} = L_{sd} + L_{fd}$ , which changes the ID loss function part from the IDFD loss function  $L_{idfd} = L_{id} + L_{fd}$ .

FIS is a soft orthogonal constraint on the feature representation that allows non-zeros. Tao et al. [32] demonstrated the learning stability advantage of using FIS over an ordinary orthogonal constraint. Let us denote the feature representation for each batch  $F = [f_1, f_2, \dots, f_b]$  as the dimension of the feature representation  $d$ , the batch size  $b$ , and the  $i$ -th feature representation  $f_i$ .  $V = F^\top = [v_1, v_2, \dots, v_d]$  is used, where the FIS loss function  $L_{fd}$  is as follows:

$$L_{fd} = - \sum_{i=1}^d \log \frac{\exp(v_i^\top v_i)}{\sum_{j=1}^d \exp(v_j^\top v_i)}. \quad (3)$$

We expect that the use of FIS in IDFD and SDFD will make the elements of the feature representation non-correlated.

### 3.3.3. Segment-Based SimCLR

SimCLR (seg) is the segment-based SimCLR, which uses a segment to create positive pairs. Figure 4 shows the difference in how positive pairs are created. When SimCLR is applied to HAR, the input positive pairs are created from instance data divided by the window size. In contrast, SimCLR (seg) positive pairs are created by selecting two instance data from the same segment.

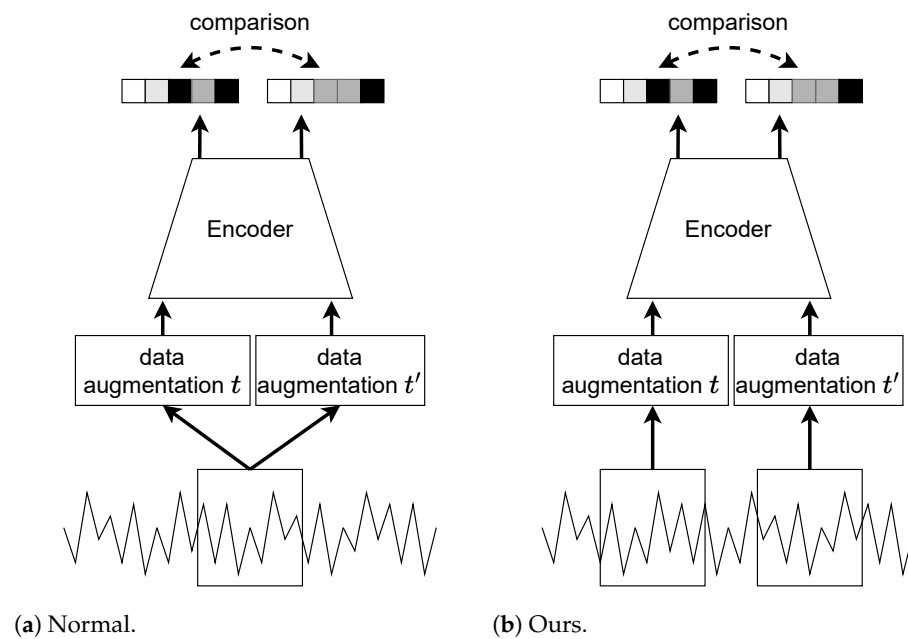
In SimCLR (seg), the loss function is the same as the loss function used in SimCLR. A positive pair consists of two instance data,  $x_i$  and  $x_j$ , which is created from the same segment. Let  $t$  and  $t'$  be two data augmentation methods and  $f(x) = f_{p1}(f_e(x))$  be the model to be trained. The feature representations of the two positive pairs used in the



SimCLR loss function are  $z_i = f(t(x_i))$  and  $z_j = f(t'(x_j))$ . Therefore, the SimCLR loss function  $L_s$  is expressed using the feature representations of the positive pairs  $z_i$  and  $z_j$  as

$$L_s = -\log \frac{\exp\left(\frac{\text{sim}(z_i, z_j)}{\tau}\right)}{\sum_{k=1}^{2b} \mathbb{1}(k \neq i) \exp\left(\frac{\text{sim}(z_i, z_k)}{\tau}\right)}. \quad (4)$$

Here,  $\tau$  is the temperature parameter,  $b$  is the batch size,  $\mathbb{1}$  is the indicator function, and  $\text{sim}$  is the cosine similarity. We used  $\tau = 1$  for simple cross-entropy in this study. Feature representations are expected to be acquired in a measured activity by creating positive pairs based on segments.



**Figure 4.** Difference in how positive pairs are created as follows: (a) Normal SimCLR creates positive pairs on an instance. (b) Segment-based SimCLR creates positive pairs from a segment.

### 3.3.4. Segment-Based SimCLR with SDFD

We introduce a method that combines SDFD and SimCLR (seg). The input data are created in the same manner as SimCLR (seg). Since an SDFD loss function does not require a data pair when passing the data to the loss function, the pair is combined and passed as one batch. When combined, the loss function  $L$  is:

$$L = \lambda_1 L_s + \lambda_2 L_{sd} + \lambda_3 L_{fd}. \quad (5)$$

In this loss function,  $\lambda_1$ ,  $\lambda_2$ , and  $\lambda_3$  denote the weights of each respective loss function. We set  $\lambda_2 = \lambda_3 = 1$  following the work [32] and set  $\lambda_1$  to 1. Since tuning the hyperparameters is time-consuming [46,47], it may be possible to improve accuracy by adjusting these parameters.

We discuss the computational complexity of the loss function of the proposed method. The computational complexity of the loss function for SimCLR (seg) is the same as that of SimCLR, since the only change is in the data-selection method. When changing from IDFD to SDFD, the computational complexity per batch is reduced by  $\mathcal{O}(bd(N - M))$  because the memory bank is treated as a matrix. Since SimCLR outputs twice as many feature maps as the batch size in a batch, the computational complexity of the proposed method is the sum of the computational complexity of SimCLR and twice the computational complexity of SDFD.

## 4. Experiments

### 4.1. Experimental Setup

We evaluated the proposed method on five benchmark datasets for HAR (HASC [40], Wisdm [38], Opportunity [42], UCI-HAR [48], and USC-HAD [39]). We preprocessed these benchmark datasets into datasets composed of input data and corresponding segment labels. Because the HASC [40] dataset contains multiple sensor devices, we used only measurement data with a sampling frequency of 100 Hz from Apple devices in this experiment. UCI-HAR [48] was divided in advance with a window size of 120 and a stride width of 60. In order to apply the proposed method, segment labels are required. Therefore, UCI-HAR [48] segment information was inferred by comparing the data to create segment labels. In this study, it was assumed that the change point ground truth information is known in order to focus on segment utilization effectiveness. We used only accelerometer data since this is common to each dataset.

We compared the proposed method and baseline methods with a shallow CNN and MLP as a backbone model. The baseline methods were ID [33], IDFD [32], SimCLR [18], and BYOL [34]. ID [33] and IDFD [32] are self-supervised learning methods that classify instance feature representations, with IDFD [32] incorporating an approach where feature representations become uncorrelated from the ID [33]. SimCLR [18] and BYOL [34] are contrastive learning methods that perform model training by comparing feature representations. While SimCLR [18] requires positive and negative pairs to train, BYOL [34] is an extension of SimCLR to allow training with positive pairs only. For the backbones of these methods, we used a three-layer CNN as an encoder and a two-layer multi-layer perceptron (MLP) for the projectors. In the case of BYOL [34], the projector and predictor had a Batch-Norm1d in two-layer MLPs. We set the number of output dimensions of the encoder to 128 and the number of input and output dimensions of all projectors and predictors to 128.

We investigated the generalization performance of feature representations acquired in few-shot transfer learning (TL) scenarios. In our experiments, we initially constructed a dataset consisting of pairs of segment labels and instance data for training, validation, and testing from benchmark datasets. We split the training, validation, and testing datasets by randomly selecting subjects for each dataset. Table 1 shows the number of subjects in each dataset. We then trained an encoder using an unsupervised learning method with the training dataset as a pretext task. Finally, we performed TL by selecting a subset of the training dataset. We assessed the generalization performance of the feature representations by comparing the f1-score when classified. During TL, we fixed the weight of the encoder and trained only the linear layer attached to the end of the encoder.

**Table 1.** The number of subjects in each dataset. HASC [40] used a limited number of individuals.

Dataset	Training	Validation	Testing
HASC [40]	80	20	30
WisdM [38]	21	6	9
Opportunity [42]	2	1	1
UCI-HAR [48]	19	5	6
USC-HAD [39]	8	2	4

We used a partially different training setup for the pretext task and TL. For the pretext task, we set a batch size of 64 for Opportunity [42] and Wisdm [38] and a batch size of 256 for the other datasets. We used an optimization function, Adam, with a learning rate of 0.01 and 150 epochs. As data augmentations [49], we applied jitter, scaling, and permutation for UCI-HAR [48] and USC-HAD [39] and jitter, scaling, and rotate for the other datasets. We used the weights that resulted in the lowest learning loss during training. For TL, we used a batch size of 512 for UCI-HAR [48] and HASC [40] and 256 for the other datasets. We trained the linear layer with a learning rate of 0.001 and 300 epochs without data augmentations. We used the weights with the lowest validation loss for the evaluation. We set the other settings the same as for the pretext task.



The window size selected has room for discussion. Based on the selected window size and the sampling frequency of the respective dataset, the duration for the utilized instances ranged from a minimum of 2.56 s to a maximum of 6.4 s. Specifically, for the HASC dataset, each instance had a duration of 2.56 s. In addition, it is highly probable that at least a single movement is included, considering the fact that the single movements are at intervals of 1–2 s [43]. Therefore, the experimental environment was considered to have no lack of activity features in a single instance. We confirmed that the proposed method was able to acquire a feature representation of the activity in at least such an environment.

#### 4.2. Ten-Shot Transfer Learning Verification

We tested the effectiveness of the feature representation acquired in the pretext task by performing transfer learning (TL) with 10-shot data per class. In this experiment, we randomly selected 10-shot data from each activity in the training dataset and trained the linear layer by TL. Table 2 shows the average f1-scores for five trials. The Non-pre in Table 2 shows the results of training the entire model with initial weights on the selected 10-shots without a pretext task. The proposed SimCLR (seg) + SDFD achieved 70.12%, 76.36%, and 91.08% on the HASC, Wisdm, and UCI-HAR, respectively. In particular, the HASC results showed a 7.26% improvement in the f1-score compared to SimCLR, which does not focus on segments. We considered that the reason for the relatively high score of Non-pre in the results of Wisdm and UCI-HAR is that we used a small model and the training dataset was sufficient to train the entire model for classification. These results showed that the proposed method can effectively acquire feature representations in unsupervised learning methods using sensor data.

**Table 2.** Average f1-score of each dataset in the same-domain transfer learning (TL). The encoder weights are fixed therein, and the number of instances is  $N = 10$ . The bold font indicates the highest score, while the underlined font is the second-highest among the datasets used.

	HASC	Wisdm	UCI-HAR
Non-pre	0.3714	0.7286	<b>0.9397</b>
BYOL [34]	0.6072	0.5676	0.7198
SimCLR [18]	<u>0.6287</u>	0.6717	0.8492
ID [33]	0.2354	0.6661	0.8926
IDFD [32]	0.6195	0.6962	0.8986
SimCLR (seg) + SDFD (ours)	<b>0.7012</b>	<b>0.7636</b>	<u>0.9108</u>

Table 3 shows the results of the ablation study. Comparing IDFD and SDFD, the Wisdm results showed an f1-score improvement of 10.18%. The SimCLR (seg) results showed an improvement from the SimCLR results by 6.62%, 9.62%, and 4.94% for HASC, Wisdm, and UCI-HAR, respectively. The results for SDFD and SimCLR (seg) showed that SimCLR (seg) was better for HASC and SDFD was better for Wisdm and UCI-HAR. SimCLR (seg) + SDFD outperformed SimCLR (seg) and SDFD by 0.64% and 0.61% for HASC and UCI-HAR, respectively. SimCLR (seg) + SDFD was 0.43% lower than SimCLR (seg) in Wisdm. The results showed that introducing the segment-based method improved the f1-score. Considering the training dataset size, SimCLR (seg) requires a larger number of subjects than SDFD to acquire feature representations, but its combination with SDFD may have made it possible to acquire feature representations with a relatively small number of subjects. Considering the effectiveness of FIS and segment-based approaches, we considered it important to group similar features in unsupervised HAR in order to acquire more-generalized feature representations.

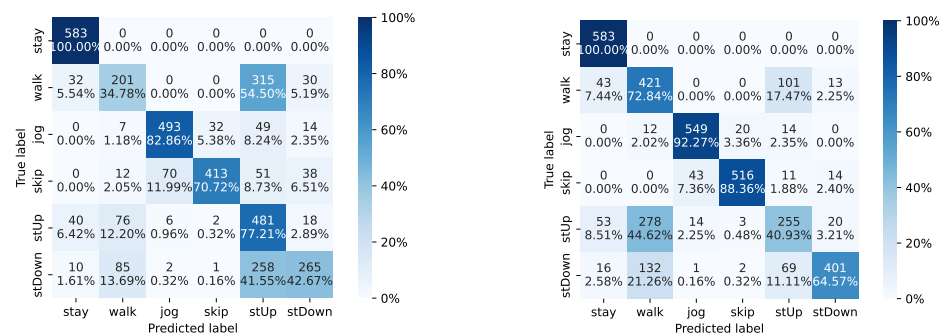
#### 4.3. Confusion Matrix

Figure 5 shows the confusion matrix resulting from TL using 10-shot data for each class in HASC. We presented the confusion matrix using the most-accurate results from the five trials. The HASC dataset contains six activities: “stay”, “walk”, “jog”, “skip”, “stUp”

(stair up), and “stDown” (stair down). The confusion matrix revealed that “walk”, “stUp”, and “stDown” were frequently misclassified. Compared to these results, the proposed method made more errors for “stUp”, but improved the prediction of “walk” and “stDown”. Table 4 shows the average f1-scores for each activity. The results indicate that the proposed method had higher scores for “walk”, “jog”, “skip”, and “stDown”, but lower scores for “stay” and “stUp” compared to other methods. SDFD and SimCLR (seg) showed similar or higher scores for most activities compared to IDFD and SimCLR, respectively. The proposed method achieved the highest f1-scores of 87.38% and 82.68% for “jog” and “skip”, respectively. The proposed method improved the scores of “walk” and “stDown” by up to 16.23% and 11.93%, respectively, and decreased the scores of “stUp” by 1.90% compared to IDFD and SimCLR. From these results, we considered that the proposed method achieved a better separation of the “walk” and “stDown” feature representations, but brought the “walk” and “stUp” feature representations closer together. Furthermore, based on the confusion matrix results, it can be considered that the proposed method mitigated the bias in predicting “stUp”.

**Table 3.** Ablation study. The bold and underlined fonts indicate the highest and second-highest f1-score, respectively.

	HASC	Wisdm	UCI-HAR
IDFD [32]	0.6195	0.6962	0.8986
SDFD	0.6875	<b>0.7980</b>	<u>0.9047</u>
SimCLR [18]	0.6287	0.6717	0.8492
SimCLR (seg)	<u>0.6948</u>	<u>0.7679</u>	0.8986
SimCLR (seg) + SDFD	<b>0.7012</b>	0.7636	<b>0.9108</b>



(a) SimCLR.

(b) SimCLR (seg) + SDFD.

**Figure 5.** Confusion matrix with the number of instances and percentages. The axes represent the activities included in HASC. In particular, “stUp” represents “stair up”, and “stDown” represents “stair down”.

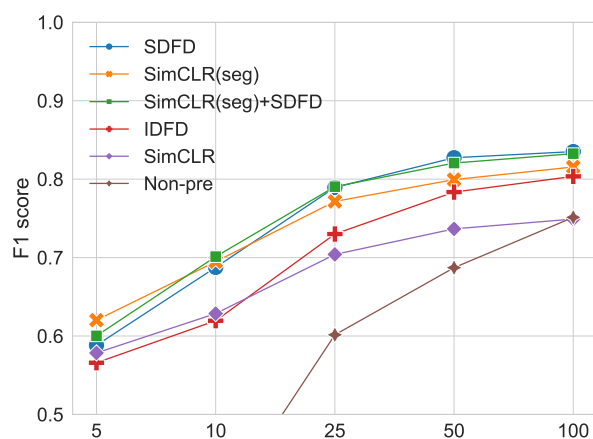
**Table 4.** f1-scores by activity for the HASC when TL with 10-shots per class. The bolded font indicates the most-accurate values.

	Stay	Walk	Jog	Skip	stUp	stDown
IDFD [32]	0.8970	0.3520	0.7752	0.6862	0.5017	0.5048
SimCLR [18]	<b>0.8992</b>	0.3916	0.7682	0.7092	0.4895	0.5147
SDFD	0.8977	0.4470	0.8548	0.8183	0.5170	0.5905
SimCLR (seg)	0.8756	0.4942	0.8525	0.8213	<b>0.5432</b>	0.5819
SimCLR (seg) + SDFD	0.8857	<b>0.5143</b>	<b>0.8738</b>	<b>0.8268</b>	0.4827	<b>0.6241</b>

#### 4.4. Instance Effectiveness for Few-Shot TL

We investigated the effect of the number of instances in few-shot TL. We compared the average f1-scores of five trials by performing TL on randomly selected few-shot data per class. Figure 6 shows the average f1-scores for five trials when only the linear layer was trained. The legend represents methods with pretext tasks, and Non-pre represents a method that trained the entire model without pretext tasks. All methods tended to have higher f1-score as the number of instances increased. When the number of instances was five and ten, Non-pre showed low values at 33.41% and 37.14%, respectively. Compared to Non-pre, the models trained with pre-training showed high f1-scores. In particular, SimCLR (seg) and SimCLR (seg) + SDFD showed relatively high f1-scores at 62.02% and 60.02%, respectively, for five-shot TL. As the number of labels increased, the f1-score of SDFD and SimCLR (seg) + SDFD also increased. SimCLR (seg) and SDFD showed relatively high f1-scores when the number of labeled data was 5 and 100, respectively.

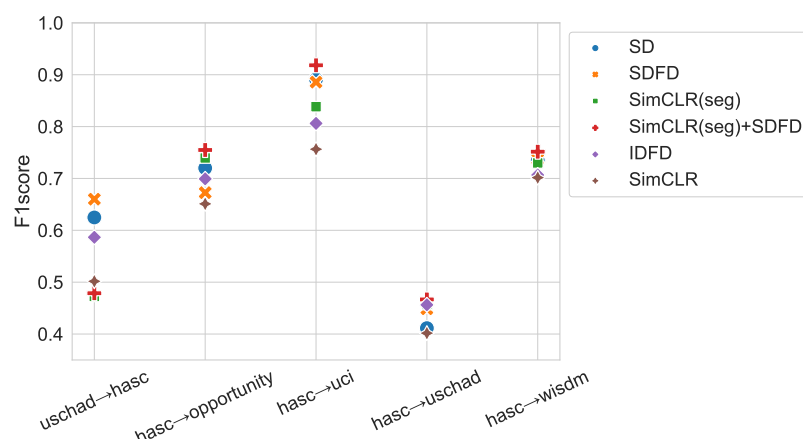
The combination of SimCLR (seg) and SDFD improved the f1-score of SDFD in TL with extremely small numbers of labels.



**Figure 6.** The number of labeled data and the f1-score of each model. The horizontal axis indicates the number of instances.

#### 4.5. Effectiveness for Different Domains

We investigated whether the proposed method can acquire a generalized feature representation to different domains. In this experiment, we performed TL on a training dataset (target dataset) created from a different benchmark dataset than the dataset used in the pretext task (pretext dataset). Figure 7 shows the results of training on the target dataset, where the horizontal axis is labeled “pretext dataset → target dataset”. When transferred from HASC, SimCLR (seg) + SDFD showed a relatively high f1-score. In particular, when transferred to UCI-HAR, SimCLR (seg) + SDFD achieved 91.82%. When transferred from USC-HAD to HASC, SDFD had the highest f1-score of 66.01%. In the results of TL for different domains from USC-HAD to HASC, the SD-based methods showed a higher f1-score, while methods using SimCLR showed a lower score. The number of subjects in the USC-HAD training dataset was small compared to HASC and Wism. Therefore, we considered the lower f1-score of the SimCLR-based method caused by the smaller total number of training data, which prevented the model from obtaining a generalized feature representation. When the size of the dataset was not large enough during the pretext task, we assumed that the SDFD-based method was an effective approach to obtain generalized feature representations. In addition, since HASC, Wism, and UCI-HAR all have different sampling frequencies, we considered SimCLR (seg) + SDFD to be robust to different sampling frequencies when there was a sufficient amount of training data for the pretext task.



**Figure 7.** Transfer performance to different datasets. The x-axis indicates “pretext dataset → target dataset”.

## 5. Conclusions

In this study, we tackled developing a new unsupervised representation learning method using segment information for HAR. We explained the measurement methods for sensor data to construct a HAR dataset. We introduced segment data, which are measurement data containing only one activity without annotating an activity label. The dataset used in our method consisted of instance data and segment labels, which were pseudo-labels assigned to the segment data. We proposed the new unsupervised learning method SimCLR (seg) + SDFD by combining SimCLR (seg) and SDFD. To investigate the effectiveness of the feature representation obtained by the proposed method, we conducted experiments by training only a linear layer attached at the end of an encoder with fixed encoder weights as TL. The experimental results showed that our proposed segment-based SimCLR methods obtained more-generalized feature representations with a higher f1-score than the regular SimCLR in HAR. We applied TL to different domains and found that the proposed SimCLR (seg) + SDFD performed robustly with respect to the sampling frequency of the sensor data. From these experiments, we suggest that the SimCLR-based method requires more training data than SDFD for the pretext task, but is effective when there are sufficient unlabeled data.

However, there are several limitations to consider. Firstly, this study used only accelerometer data for the datasets and did not discuss the quantity or other sensor data such as gyroscope or magnetometer data. Therefore, it can be expected that the accuracy of the proposed method will be improved by multimodalization. Secondly, this study only used a dataset with basic activities. It is expected that more-generalized feature representations could be obtained by using a dataset with a wider range of activities. Third, this study did not use a method to split the accelerometer data containing multiple activities into segment data. As future work, we will introduce multimodalization in the proposed method, using datasets with a wider range of activities and enhancing a framework that includes a method for splitting accelerometer data with multiple activities into segment data. These approaches will lead to more-accurate and -effective HAR systems.

**Author Contributions:** Conceptualization, K.T., K.K. and T.H.; Methodology, K.T., K.K. and T.H.; Software, K.T. and K.K.; Validation, K.T.; Investigation, K.T.; Writing—original draft, K.T.; Writing—review & editing, K.T. and T.H.; Visualization, K.T.; Supervision, T.H.; Project administration, T.H. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported in part by the Japan Society for the Promotion of Science (JSPS) KAKENHI Grant-in-Aid for Early-Career Scientists and Grant-in-Aid for Scientific Research (C) under Grants 19K20420 and 23K11164.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

## References

1. Thiel, D.V.; Sarkar, A.K. Swing Profiles in Sport: An Accelerometer Analysis. *Procedia Eng.* **2014**, *72*, 624–629. [\[CrossRef\]](#)
2. Klein, M.C.; Manzoor, A.; Mollee, J.S. Active2Gether: A Personalized m-Health Intervention to Encourage Physical Activity. *Sensors* **2017**, *17*, 1436. [\[CrossRef\]](#) [\[PubMed\]](#)
3. Plangger, K.; Campbell, C.; Robson, K.; Montecchi, M. Little rewards, big changes: Using exercise analytics to motivate sustainable changes in physical activity. *Inf. Manag.* **2022**, *59*, 103216. [\[CrossRef\]](#)
4. Thomas, B.; Lu, M.L.; Jha, R.; Bertrand, J. Machine Learning for Detection and Risk Assessment of Lifting Action. *IEEE Trans.-Hum.-Mach. Syst.* **2022**, *52*, 1196–1204. [\[CrossRef\]](#)
5. Zeng, M.; Nguyen, L.T.; Yu, B.; Mengshoel, O.J.; Zhu, J.; Wu, P.; Zhang, J. Convolutional Neural Networks for human activity recognition using mobile sensors. In Proceedings of the 6th International Conference on Mobile Computing, Applications and Services, Austin, TX, USA, 6–7 November 2014; pp. 197–205.
6. Ordóñez, F.J.; Roggen, D. Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition. *Sensors* **2016**, *16*, 115. [\[CrossRef\]](#) [\[PubMed\]](#)
7. Murad, A.; Pyun, J.Y. Deep Recurrent Neural Networks for Human Activity Recognition. *Sensors* **2017**, *17*, 2556. [\[CrossRef\]](#) [\[PubMed\]](#)
8. Guinea, A.S.; Sarabchian, M.; Mühlhäuser, M. Improving Wearable-Based Activity Recognition Using Image Representations. *Sensors* **2022**, *22*, 1840. [\[CrossRef\]](#) [\[PubMed\]](#)
9. Jalal, A.; Quaid, M.A.K.; Hasan, A.S. Wearable Sensor-Based Human Behavior Understanding and Recognition in Daily Life for Smart Environments. In Proceedings of the 2018 International Conference on Frontiers of Information Technology (FIT), Islamabad, Pakistan, 17–19 December 2018; pp. 105–110.
10. Liu, H.; Schultz, T. A Wearable Real-time Human Activity Recognition System using Biosensors Integrated into a Knee Bandage. In Proceedings of the 12th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2019)-BIODEVICES, Prague, Czech Republic, 22–24 February 2019; INSTICC; SciTePress: Setúbal, Portugal, 2019; pp. 47–55.
11. Liu, H.; Gamboa, H.; Schultz, T. Sensor-Based Human Activity and Behavior Research: Where Advanced Sensing and Recognition Technologies Meet. *Sensors* **2023**, *23*, 125. [\[CrossRef\]](#)
12. Haresamudram, H.; Essa, I.; Plötz, T. Contrastive Predictive Coding for Human Activity Recognition. *ACM Interact. Mob. Wearable Ubiquitous Technol.* **2021**, *5*, 1–26. [\[CrossRef\]](#)
13. Tonekaboni, S.; Eytan, D.; Goldenberg, A. Unsupervised Representation Learning for Time Series with Temporal Neighborhood Coding. In Proceedings of the ninth International Conference on Learning Representations, Virtual only, 3–7 May 2021.
14. Zerveas, G.; Jayaraman, S.; Patel, D.; Bhamidipaty, A.; Eickhoff, C. A Transformer-Based Framework for Multivariate Time Series Representation Learning. In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, New York, NY, USA, 14–18 August 2021; pp. 2114–2124.
15. Xiao, Z.; Xing, H.; Zhao, B.; Qu, R.; Luo, S.; Dai, P.; Li, K.; Zhu, Z. Deep Contrastive Representation Learning with Self-Distillation. *IEEE Trans. Emerg. Top. Comput. Intell.* **2023**, 1–13. [\[CrossRef\]](#)
16. Franceschi, J.Y.; Dieuleveut, A.; Jaggi, M. Unsupervised scalable representation learning for multivariate time series. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, Canada, 8–14 December 2019; Wallach, H., Larochelle, H., Beygelzimer, A., d’Alché-Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2019; Volume 32.
17. Sohn, K. Improved Deep Metric Learning with Multi-class N-pair Loss Objective. In Proceedings of the Advances in Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2016; Volume 29.
18. Chen, T.; Kornblith, S.; Norouzi, M.; Hinton, G. A Simple Framework for Contrastive Learning of Visual Representations. In Proceedings of the 37th International Conference on Machine Learning, Virtual only, 12–18 July 2020.
19. Takenaka, K.; Hasegawa, T. Unsupervised Representation Learning Method In Sensor Based Human Activity Recognition. In Proceedings of the International Conference on Machine Learning and Cybernetics, Toyama, Japan, 9–11 September 2022; pp. 25–30.
20. Hartmann, Y.; Liu, H.; Schultz, T. Feature Space Reduction for Multimodal Human Activity Recognition. In Proceedings of the 13th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2020)-BIOSIGNALS, Valletta, Malta, 24–26 February 2020; INSTICC; SciTePress: Setúbal, Portugal, 2020; pp. 135–140.
21. Liu, H.; Xue, T.; Schultz, T. On a Real Real-Time Wearable Human Activity Recognition System. In Proceedings of the 16th International Joint Conference on Biomedical Engineering Systems and Technologies-WHC, Lisbon, Portugal, 16–18 February 2023; INSTICC; SciTePress: Setúbal, Portugal, 2023; pp. 711–720.



22. Bento, N.; Rebelo, J.; Barandas, M.; Carreiro, A.V.; Campagner, A.; Cabrita, F.; Gamboa, H. Comparing Handcrafted Features and Deep Neural Representations for Domain Generalization in Human Activity Recognition. *Sensors* **2022**, *22*, 7324. [[CrossRef](#)] [[PubMed](#)]
23. Dirgová Luptáková, I.; Kubovčík, M.; Pospíchal, J. Wearable Sensor-Based Human Activity Recognition with Transformer Model. *Sensors* **2022**, *22*, 1911. [[CrossRef](#)] [[PubMed](#)]
24. Mahmud, T.; Sazzad Sayyed, A.Q.M.; Fattah, S.A.; Kung, S.Y. A Novel Multi-Stage Training Approach for Human Activity Recognition From Multimodal Wearable Sensor Data Using Deep Neural Network. *IEEE Sens. J.* **2021**, *21*, 1715–1726. [[CrossRef](#)]
25. Qian, H.; Pan, S.J.; Da, B.; Miao, C. A Novel Distribution-Embedded Neural Network for Sensor-Based Activity Recognition. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19, Macao SAR, China, 10–16 August 2019; pp. 5614–5620.
26. Gao, W.; Zhang, L.; Teng, Q.; He, J.; Wu, H. DanHAR: Dual Attention Network for multimodal human activity recognition using wearable sensors. *Appl. Soft Comput.* **2021**, *111*, 107728. [[CrossRef](#)]
27. Sozinov, K.; Vlassov, V.; Girdzijauskas, S. Human Activity Recognition Using Federated Learning. In Proceedings of the IEEE Intl Conf on Parallel & Distributed Processing with Applications, Ubiquitous Computing & Communications, Big Data & Cloud Computing, Social Computing & Networking, Sustainable Computing & Communications, Melbourne, VIC, Australia, 11–13 December 2018; pp. 1103–1111.
28. Li, C.; Niu, D.; Jiang, B.; Zuo, X.; Yang, J. Meta-HAR: Federated Representation Learning for Human Activity Recognition. In Proceedings of the Web Conference 2021, New York, NY, USA, 19–23 April 2021; pp. 912–922.
29. Ma, H.; Zhang, Z.; Li, W.; Lu, S. Unsupervised Human Activity Representation Learning with Multi-Task Deep Clustering. *Acm Interact. Mob. Wearable Ubiquitous Technol.* **2021**, *5*, 1–25. [[CrossRef](#)]
30. He, K.; Fan, H.; Wu, Y.; Xie, S.; Girshick, R. Momentum Contrast for Unsupervised Visual Representation Learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 9726–9735.
31. Chen, X.; He, K. Exploring simple siamese representation learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 15750–15758.
32. Tao, Y.; Takagi, K.; Nakata, K. Clustering-friendly Representation Learning via Instance Discrimination and Feature Decorrelation. In Proceedings of the International Conference on Learning Representations, Virtual, 3–7 May 2021.
33. Wu, Z.; Xiong, Y.; Yu, S.X.; Lin, D. Unsupervised Feature Learning via Non-Parametric Instance Discrimination. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 3733–3742.
34. Grill, J.B.; Strub, F.; Altché, F.; Tallec, C.; Richemond, P.H.; Buchatskaya, E.; Doersch, C.; Pires, B.A.; Guo, Z.D.; Azar, M.G.; et al. Bootstrap Your Own Latent a New Approach to Self-Supervised Learning. In Proceedings of the Advances in Neural Information Processing Systems, Virtual only, 6–12 December 2020; Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2020; Volume 33.
35. Rodrigues, J.; Liu, H.; Folgado, D.; Belo, D.; Schultz, T.; Gamboa, H. Feature-Based Information Retrieval of Multimodal Biosignals with a Self-Similarity Matrix: Focus on Automatic Segmentation. *Biosensors* **2022**, *12*, 1182. [[CrossRef](#)] [[PubMed](#)]
36. Folgado, D.; Barandas, M.; Antunes, M.; Nunes, M.L.; Liu, H.; Hartmann, Y.; Schultz, T.; Gamboa, H. TSSEARCH: Time Series Subsequence Search Library. *SoftwareX* **2022**, *18*, 101049. [[CrossRef](#)]
37. Oord, A.v.d.; Li, Y.; Vinyals, O. Representation learning with contrastive predictive coding. *arXiv* **2018**, arXiv:1807.03748.
38. Kwapisz, J.R.; Weiss, G.M.; Moore, S.A. Activity Recognition Using Cell Phone Accelerometers. *Acm Sigkdd Explor. Newsl.* **2011**, *12*, 74–82. [[CrossRef](#)]
39. Zhang, M.; Sawchuk, A.A. USC-HAD: A Daily Activity Dataset for Ubiquitous Activity Recognition Using Wearable Sensors. In Proceedings of the 2012 ACM Conference on Ubiquitous Computing, New York, NY, USA, 5–8 September 2012; pp. 1036–1043.
40. Ichino, H.; Kaji, K.; Sakurada, K.; Hiroi, K.; Kawaguchi, N. HASC-PAC2016: Large Scale Human Pedestrian Activity Corpus and Its Baseline Recognition. In Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct, New York, NY, USA, 12–16 September 2016; pp. 705–714.
41. Kawaguchi, N.; Yang, Y.; Yang, T.; Ogawa, N.; Iwasaki, Y.; Kaji, K.; Terada, T.; Murao, K.; Inoue, S.; Kawahara, Y.; et al. HASC2011corpus: Towards the Common Ground of Human Activity Recognition. In Proceedings of the 13th International Conference on Ubiquitous Computing, New York, NY, USA, 17–21 September 2011; pp. 571–572.
42. Chavarriaga, R.; Sagha, H.; Calatroni, A.; Digumarti, S.T.; Tröster, G.; del R. Millán, J.; Roggen, D. The Opportunity challenge: A benchmark database for on-body sensor-based activity recognition. *Pattern Recognit. Lett.* **2013**, *34*, 2033–2042. [[CrossRef](#)]
43. Liu, H.; Schultz, T. How Long Are Various Types of Daily Activities? Statistical Analysis of a Multimodal Wearable Sensor-Based Human Activity Dataset. In Proceedings of the 15th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2022)-Volume 5: HEALTHINF, Online Streaming, 9–11 February 2022; pp. 680–688.
44. Liu, H.; Schultz, T. ASK: A Framework for Data Acquisition and Activity Recognition. In Proceedings of the International Conference on Bio-inspired Systems and Signal Processing, Funchal, Madeira, Portugal, 19–21 January 2018.
45. Liu, H.; Hartmann, Y.; Schultz, T. CSL-SHARE: A Multimodal Wearable Sensor-Based Human Activity Dataset. *Front. Comput. Sci.* **2021**, *3*, 759136. [[CrossRef](#)]
46. Castro, R.L.; Andrade, D.; Fraguera, B. Reusing Trained Layers of Convolutional Neural Networks to Shorten Hyperparameters Tuning Time. *arXiv* **2020**, arXiv:2006.09083.



47. Oyelade, O.N.; Ezugwu, A.E. A comparative performance study of random-grid model for hyperparameters selection in detection of abnormalities in digital breast images. *Concurr. Comput. Pract. Exp.* **2022**, *34*, e6914. [[CrossRef](#)]
48. Anguita, D.; Ghio, A.; Oneto, L.; Parra, X.; Reyes-Ortiz, J.L. A Public Domain Dataset for Human Activity Recognition Using Smartphones. In Proceedings of the European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, Bruges, Belgium, 24–26 April 2013; pp. 437–442.
49. Um, T.T.; Pfister, F.M.J.; Pichler, D.; Endo, S.; Lang, M.; Hirche, S.; Fietzek, U.; Kulić, D. 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, New York, NY, USA, 13–17 November 2017; pp. 216–220.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.