Knowledge-Based Systems xxx (xxxx) xxx

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Learning hierarchical time series data augmentation invariances via contrastive supervision for human activity recognition

Dongzhou Cheng [a], Lei Zhang [a,*], Can Bu [a], Hao Wu [b,*], Aiguo Song [c]

[a] School of Electrical and Automation Engineering, Nanjing Normal University, 210023, Nanjing, China
[b] School of Information Science and Engineering, Yunnan University, 650500, Kunming, China
[c] School of Instrument Science and Engineering, Southeast University, 210096, Nanjing, China

ARTICLE INFO

Article history:
Received 1 February 2023
Received in revised form 3 July 2023
Accepted 4 July 2023
Available online xxxx

Keywords:
Human activity recognition
Sensors
Data augmentation
Contrastive loss
Deep supervision

ABSTRACT

Human activity recognition (HAR) using wearable sensors is always a research hotspot in ubiquitous computing scenario, in which feature learning has played a crucial role. Recent years have witnessed outstanding success of contrastive learning in image data, which learns invariant representations by adding contrastive loss to the last layer of deep neural networks. However, the advantages of contrastive loss have been rarely leveraged in time series data for activity recognition. A fundamental obstacle to contrastive learning in HAR is that image-based augmentation could not fit well with sensor data, which raises a critical issue: the distortions induced by augmentation might be further enlarged by intermediate layers of a network and thus severely harm semantic structure of original activity instance. In this paper, taking an inspiration from deeply-supervised learning, we propose a novel approach called Contrastive Supervision by considering "where" to contrast, which aims to learn time series augmentation invariances by forcing positive pairs nearby and negative pairs far apart at different depths of neural network. Our approach can be seen as a generalization of contrastive learning in a deeply-supervised setting, where the contrastive loss is used to supervise the intermediate layers instead of only the last layer, allowing us to effectively leverage label information so as to better fuse the multi-level features. Experiments on popular benchmarks demonstrate that our approach can learn better representations and improve classification accuracy without additional inference cost for various HAR tasks in supervised and semi-supervised learning paradigms.

1. Introduction

1.1. Background

During recent years, there has been an exceptional development of Internet of Things and miniaturized sensors, and their prominent advantages such as low manufacturing price, small size, and high accuracy enable a wide range of sensors to be incorporated into smart watches, phones, as well as other portable devices [1]. Due to vast proliferation of sensor devices, human activity recognition (HAR) has received considerable attention in ubiquitous computing scenario, which adopts various machine learning algorithms to analyze and comprehend human activities through input data collected from various embedded sensors attached to different body positions, thus motivating various context-aware applications including healthcare, fitness monitoring, smart homes, elderly fall detection, etc. [2–4]. In particular, thanks to an ability of automatic feature extraction, deep neural networks (DNNs), especially convolutional neural networks (CNNs) have demonstrated remarkable performance in learning multi-level features to mine intrinsic activity characteristics [5,6].

1.2. Current challenges

Along above research line, a common trend is to stack more and more convolutional layers to form deeper network [5–8], among which the subsampling layers and indispensable nonlinear activation layers are interweaved with one another. There has been a great wave of CNN-based researches emphasizing the importance of automatic and hierarchical feature learning, which have become one of the most dominant models for various HAR tasks (also see the survey paper in [5]). Previous researches [2,9–11] have primarily focused on designing different CNN architectures to fit a large variety of HAR applications. Embracing such sophisticated CNNs with innovative connection topology and tens of thousands of parameters, the past decade has witnessed their great success. However, the success of deep CNNs is often accompanied by rapid growth in computational cost,

* Corresponding authors.
E-mail addresses: 211843002@njnu.edu.cn (D. Cheng), leizhang@njnu.edu.cn (L. Zhang), 211843007@njnu.edu.cn (C. Bu), haowu@ynu.edu.cn (H. Wu), a.g.song@seu.edu.cn (A. Song).

https://doi.org/10.1016/j.knosys.2023.110789
0950-7051/

and the resulting computation would inevitably increase latency or inference time, which has an important impact on practical HAR implementation. As a consequence, one could not arbitrarily increase model size due to restricted computing resources on mobile devices [12]. Therefore, it remains a challenging issue whether there still exists room for performance improvement in these CNN-based methods, providing that one does not intend to increase model size. In fact, developing different CNN architectures is not the only way to boost model performance. Instead, we argue that improving the network learning paradigm is another feasible solution to achieve the compelling performance without increasing additional parameters and computation during activity inference.

On the other hand, the widespread popularity of mobile devices such as smartphones has led to a vast amount of sensor data streams, which could be utilized to better analyze and understand human behaviors for various healthcare applications. Recent advancements in deep learning have been laying a groundwork for the development of more accurate HAR systems. These complex deep models are typically trained through supervised learning, which usually depends on a set of large-scale labeled data samples [7,12–15]. However, it is not easy to collect labeled sensor data, which has been one fundamental obstacle in performing such data-hungry deep learning algorithms for activity prediction. In fact, it is impossible for an annotator to precisely label time series sensor data without an aid of corresponding video. Unlike video data, time series sensor data stream such as accelerometer traces recorded from mobile devices is far difficult to interpret by visual inspection, which renders manual labeling of sensor data to be expensive, time-consuming, and tedious [5,16,17]. As a result, labeled data collection is commonly done in a controlled or semi-controlled laboratory setting (mostly involving less than 50 subjects) [12,18,19], which limits the generalizability of these deep models in real-world HAR applications. Therefore, it deserves deeper investigation into how to make an economic use of limited labeled data as effectively as possible.

1.3. Research motivation

The aforementioned issue motivates us to improve the CNN learning strategy so as to enhance model generalization and improve accuracy for activity recognition. Our reasoning primarily comes from two critical standpoints. On the one hand, unlike previous studies targeted at designing different network architectures, one promising research line is to apply deep supervision for CNN-based HAR, which lays an emphasis on the intermediate feature representation and hidden layer supervision, instead of only adding the supervision to the final layer of the whole network [20,21]. It is well known that different convolutional layers tend to learn features at different levels. In general, the last layers learn more task-related high-level features, while the shallow layers learn more common low-level features, where deep supervision could force the shallow layers to learn the task-related knowledge at an earlier time by applying the supervised task loss to optimize the intermediate layers. There has been a consensus that deep supervision could effectively boost model performance, and help the neural networks to learn more discriminative features at different levels [21–23]. To the best of our knowledge, no HAR-related deep supervision research exists so far. On the other hand, contrastive learning has achieved state-of-the-art performance in representation learning across a large range of computer vision applications [24,25]. However, traditional data augmentation techniques used for image data like color distortion generally could not fit well with time-series sensor data [17,26]. The contrastive learning has been rarely leveraged for time-series sensor data in the context of deeply-supervised HAR. From a novel perspective of representation learning, we argue that contrastive learning could provide better supervision for intermediate layers compared to traditional deep supervision, which deserves deeper investigations. By regarding two augmentations from the same activity as a positive pair and different activity samples as negative pairs, the CNN could be trained to maximize the distance of a negative pair while minimizing the distance of a positive pair, which enables the network to learn the invariant representation from various sensor data augmentation like *Jittering* and *Resampling*. Because these data augmentation invariances are generally low-level and task-irrelevant for various HAR tasks, they might provide more beneficial knowledge for supervising the intermediate layers.

1.4. Contribution

Recently, contrastive learning using data augmentation as an alternative source of training data, has been proposed to address the limitations caused by the lack of labels [24,25,27,28], which seeks to learn invariant representations by contrasting positive pairs against negative pairs, being an active study area in computer vision. In this paper, we investigate for the first time, to the best of our knowledge, the effectiveness of contrastive learning in sensor-based HAR data. In particular, we apply deep supervision in different levels [21] to enable contrastive loss in supervising intermediate layers with augmented wearable sensor data. To this end, a set of data augmentation schemes has been introduced for time series sensor data, in place of traditional image data augmentation operators. These data augmentation techniques are evaluated on public HAR benchmarks. A comprehensive and systematic ablation study is provided to analyze the effect of different data transformations, which shows that different combinations of transformations could result in significant differences in performance. Under both fully-supervised and semi-supervised paradigms, the invariant representations learned by the contrastive supervision framework can yield better performance when models are trained with limited labels, which indicates the potential of contrastive supervision framework in HAR due to the modality-agnostic advantage of contrastive learning, paving a way to further studies that fully leverage these data augmentation techniques in limited label information.

To be specific, taking an inspiration from above insight, we propose a novel approach called Contrastive Supervision by combining contrastive learning with naïve deep supervision for activity recognition task, which aims at training state-of-the-art CNNs with improved accuracy and without causing extra computational cost during activity inference. As illustrated in Fig. 1, a few projection heads are attached to the intermediate layers of the whole neural network, in which each of them is trained by contrastive loss. Here we borrow the concept of contrastive learning from a self-supervised learning paradigm [25,27–29], which is used to generate useful feature representations for activity recognition. Due to recent great success, this study has gained popularity. A core design in the majority of these architectures is to append a projection head at the end of the backbone classification network, which is comprised of a Global Average Pooling (GAP) operation, a Rectified Linear Unit (ReLU), and a linear fully-connected (FC) layer [25,27,28]. In this paper, the module's main role is to project the backbone features into a latent low-dimensional space at different depths before applying the contrastive loss. Note that since the projection head contains a ReLU activation function, it is still a nonlinear transformation, but has only one hidden layer. We highlight that the projection head can be essentially viewed as a low-dimensional mapping [27] (here we map the output of each layer into 128-dimensional embedding feature vectors), which is useful for obtaining generalizable activity features from

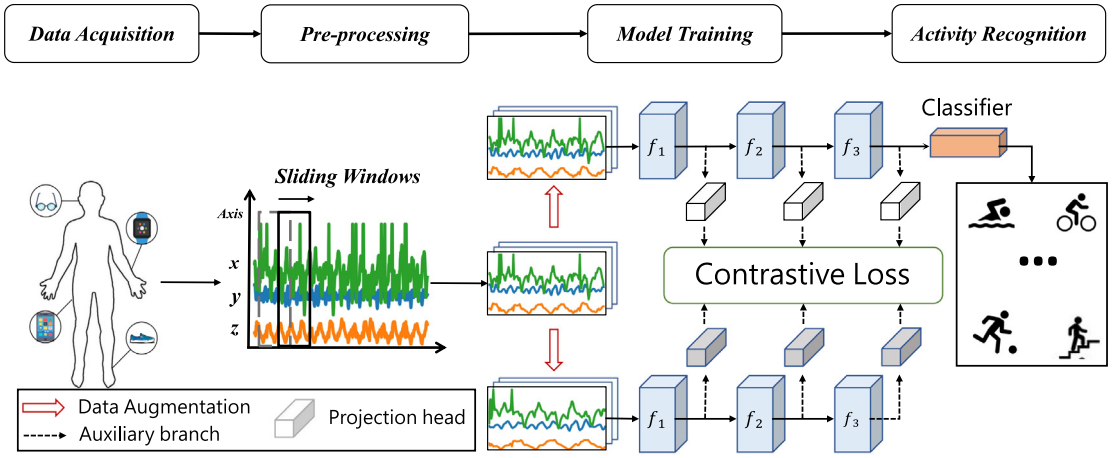


Fig. 1. An overview of the proposed Contrastive Supervision, where these auxiliary branches at different depths would be removed during activity inference stage without additional memory or computational overhead.

a small set of labeled sensor data. Due to its practical importance, the projection head has always played a key role in the contrastive learning. The learned projection head would be discarded once training is completed. This is to say, the proposed approach seeks to optimize the intermediate layers by Contrastive Supervision instead of traditional deep supervision. Such projection heads would be simply discarded during inference stage so as to avoid additional computation and memory overhead. Unlike deep supervision targeted at optimizing every intermediate layer to learn task-related knowledge for a specific activity recognition task, the intermediate layers in our Contrastive Supervision can be trained to learn invariant representation from various time series data augmentations, which makes the network be able to generalize better. In addition, because contrastive learning generally performs well with smaller unlabeled data, the proposed approach could be easily extended to other semi-supervised HAR paradigm. Extensive experiment analyses are conducted on four public benchmark datasets including UCI-HAR [30], WISDM [31], PAMAP2 [32], and UniMib-SHAR [33], which show the effectiveness of Contrastive Supervision on general activity classification as well as semi-supervised HAR. The main contributions of this paper are summarized as follows:

- In this paper, we revisit the concept of deep supervision for activity recognition tasks, and propose a novel approach named Contrastive Supervision targeted at applying contrastive loss rather than traditional supervised loss to optimize the intermediate layers of CNNs with improved accuracy, without causing additional parameters and computational cost comparing to standard activity inference.
- Various time series data augmentations are utilized to generate different views for the same activity instance, which help Contrastive Supervision to better learn invariant representations from augmented sensor data. Experimental results empirically verify the effectiveness of Contrastive Supervision in supervised and semi-supervised HAR scenarios.
- We perform comprehensive and in-depth ablation analyses by studying the effect of several key components such as contrastive loss, scalar temperature, and data augmentation to emphasize their significance in deeply-supervised HAR. In addition, a visualizing analysis is provided, which shows that Contrastive Supervision can effectively enhance the abstraction ability of CNNs to produce better embedding distribution. Actual implementation is evaluated on a mobile platform.

Despite higher performance, it is worth noting that most complicated structures usually slow down the inference. One cannot increase model complexity arbitrarily because of real-world business requirements or hardware limitations. It would be more reasonable to judge the quality of deep models according to the trade-off between recognition performance and inference-time costs, e.g., memory footprint and inference latency [5,12]. Inspired by the fact, this paper primarily seeks to complicate the training-time network structure (i.e., attaching contrastive supervision branches to numerous intermediate layers) so as to boost model performance while maintaining the original inference-time structure the same or unchanged. After training is done, all auxiliary branches associated with contrastive supervision would be removed during inference. That is to say, both the training-time and inference-time structure might be decoupled by only complicating the network structure during training and then transforming it back into the original inference-time structure for deployment. In this way, the network could be trained to reach a higher level of performance and then converted back to its original inference-time structure for inference, without sacrificing the inference-time costs. In the common cases, one can train deep models on powerful workstations equipped with GPU, and then deploy them into resource-sensitive mobile devices. Therefore, it would be acceptable to improve model performance at the expense of more training resources, as long as the deployed model size remains unchanged.

The remaining part of this paper is structured as follows: Section 2 reviews related literatures. Section 3 presents an overall framework of our proposed Contrastive Supervision. Comprehensive experiments, ablation analyses, as well as actual implementation, are conducted in Section 4. Finally, the whole paper is concluded in Section 5.

2. Related works

In this section, we review some relevant researches in previous literature, from which our approach takes an inspiration. In addition, we analyze their main distinctions from our mechanism.

2.1. Deep supervision and contrastive learning

In this part, we categorize the focus of our paper, i.e., Contrastive Supervision into deep supervision and contrastive learning, and review the background of each category. On the one hand, deep supervision has been first proposed in 2015, when Lee et al. has introduced a framework called Deeply Supervised

Net (DSN) [21] for image recognition task. During the past years, there has been an ever-growing number of researches devoted to applying deep supervision for performance improvement in a large variety of computer vision applications such as object detection [34], image super-resolution [35], and semantic segmentation [36]. Li et al. have recently presented a comprehensive and in-depth survey about deep supervision, and summarizes its main applications in various areas [20]. In fact, deep supervision has been never investigated in the context of HAR. On the other hand, contrastive learning has started to play a dominant role in representational learning, which is able to effectively learn invariant representations via data augmentation. For example, He et al. have introduced an approach of contrastive learning named MoCo [25], where a momentum encoder is utilized to learn invariant representations from negative pairs through a memory bank. Chen et al. have presented SimCLR [24], in which the momentum encoder is replaced via utilizing a larger batch of negative pairs. To alleviate the burden by the huge number of negative samples, Grill et al. achieve competitive performance via BYOL [37] even with no need of using negative samples. Similarly, motivated by an idea of neglecting negative samples, Chen and He have proposed an approach called SimSiam [38], which uses a stop-gradient operation to help simple Siamese networks achieve the state-of-the-art performance. Despite great success of contrastive learning in image data, it could not perform well on time series sensor data that has different properties due to temporal dependency. Focusing on time series data augmentation from a perspective of contrastive learning [16,17,39], this paper presents a novel cross-view Contrastive Supervision approach to learn the invariant representations from augmented sensor data. Different from previous works, our study aims to fill the gap between contrastive learning and deep supervision for HAR using time-series sensor data.

2.2. Deep learning for HAR

During recent years, there has been a considerable number of studies dedicated to learning discriminative activity features automatically via feeding sensor data into various deep neural networks such as convolutional networks, residual networks, and Transformers, which demonstrate state-of-the-art performance in a wide range of activity recognition tasks [2,5]. For example, Hammerla et al. [8], Ronao et al. [6], and Yang et al. [40] at the earliest time have applied deep CNNs for activity recognition problem, which have gained wide popularity in learning multi-level and abstracted feature representations from raw sensor data. To improve the comprehensibility of HAR, Zeng et al. [41] and Ma et al. [7] have combined an attention module with convolution networks to highlight more important time steps as well as sensor channels along multimodal sensor data. Guan et al. [14] have proposed to ensemble multiple deep networks so as to improve the performance of activity recognition. In light of obvious advantages of recurrent structures, Ordóñez proposed a novel network called DeepConvLSTM [13], which combines CNN and LSTM to extract both local and global activity features at the same time. Al-qaness et al. [42] have integrated a multilevel residual network with bidirectional gated recurrent unit (GRU) as well as an attention module to implement activity recognition, which achieves an impressive performance. Utilizing deep CNNs for feature embedding, Sharma et al. [43] have shown the effectiveness of transformer-based architecture in feature fusion for HAR tasks. Haresamudram et al. [44] have introduced a framework of Contrastive Predictive Coding (CPC) to model the long-range temporal dependencies of sensor data, which leads to significantly improved performance for HAR. On the whole, current researches primarily focus on designing different network architectures to improve performance but have not given the comprehensive and in-depth considerations involved in how to apply Contrastive Supervision in the best way across a range of HAR applications. To our knowledge, this paper is the first work to address this issue.

3. Model

This section presents an overview of the proposed Contrastive Supervision HAR framework. To be specific, we first introduce a generic layer-wise deep supervision HAR framework, and then present an improved Contrastive Supervision scheme in the deeply-supervised setting.

3.1. A generic deep supervision framework in HAR

We first introduce the formulation of deep supervision in the context of HAR. In essence, sensor-based HAR might be seen as a problem of classifying raw time series data recorded by wearable sensors into a set of well-defined activities. Fig. 1 presents the standard activity recognition workflow consisting of four main steps: data acquisition, data pre-processing, feature extraction, and activity classification, where raw sensor signals need to be first preprocessed by performing noise filters and data standardization, then segmented into fixed-width fragments through sliding window [13,41]. Different from traditional approaches that involve handcrafted feature engineering, deep learning algorithms are able to automatically extract activity features from raw sensor data.

A classification network is often comprised of a feature extractor and a classifier with linear fully-connected (FC) layer [21,23,45]. Without loss of generality, let $\omega = h \circ f$ be a classification network for activity classification, where h stands for the classifier at the final layer that utilizes standard cross entropy loss. Given that $f = f_K \circ f_{K-1} \circ \cdots f_1$ is the feature extractor, K is equal to the number of intermediate convolutional layers stacked within f. Referring to previous literature [6,17], we adopt a three-layer CNN architecture (i.e., $K = 3$) as our backbone feature extractor, where each convolutional block consists of a convolution layer, a Batch Normalization (BN) layer, a ReLU activation function, and a Max-pooling operation. In general, the supervision would be only employed at the final layer of the network during standard training scheme. In such a manner, there are overall $K - 1$ auxiliary classifiers, which can be mathematically formulated as:

$$\begin{aligned} c_1(x) &= g_1 \circ f_1(x), \\ c_2(x) &= g_2 \circ f_2 \circ f_1(x), \\ &\cdots \\ c_{K-1}(x) &= g_{K-1} \circ f_{K-1} \circ \cdots \circ f_1(x), \\ \omega(x) &= h \circ f_K \circ f_{K-1} \circ \cdots \circ f_1(x). \end{aligned} \tag{1}$$

For a given training set of N examples $\mathcal{X} = \{x_i\}_i^N$ with labels $\mathcal{Y} = \{y_i\}_i^N$, the overall loss function $\mathcal{L}_{\text{DS}}$ can be defined in the following form with a combination:

$$\mathcal{L}_{\text{DS}} = \underbrace{\mathcal{L}_{\text{CE}}\left(\omega(\mathcal{X}), \mathcal{Y}\right)}_{\text{from standard training}} + \alpha \cdot \sum_{k=1}^{K-1} \underbrace{\mathcal{L}_{\text{CE}}\left(c_i(\mathcal{X}), \mathcal{Y}\right)}_{\text{from deep supervision}}, \tag{2}$$

which includes two loss items and $\mathcal{L}_{\text{CE}}$ is the standard cross entropy loss. To be specific, the first item denotes the final layer loss while the second item denotes intermediate supervision loss, in which α is a hyper-parameter used to control the balance between both items.

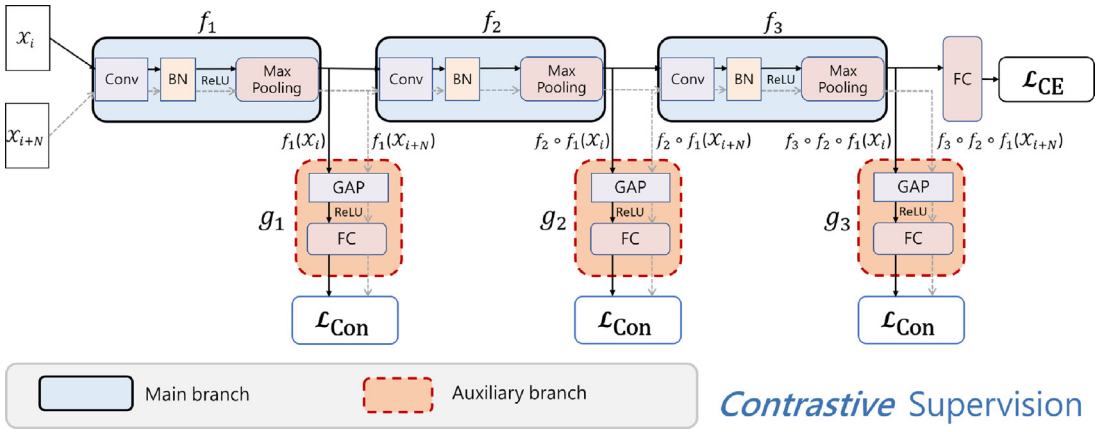


Fig. 2. Schematic digram for proposed Contrastive Supervision pipeline. The detailed structure is shown at the top. The bottom illustrates interior structure of both block types.

Table 1
Details of various data augmentation techniques.

Augmentation	Implementation details
Jittering	Inject a random Gaussian noise with zero mean and standard deviation of 0.8 into raw sensor data to simulate disturbing sensor noise.
Permuting	Divide one sensor window into no more than 5 fragments with the same length, all of which are randomly shuffled and merged into a new window.
Scaling	Multiply each sensor channel by a random scalar with a mean of 2 and standard deviation of 1.1 for amplifying motion signal amplitude.
Flipping	Multiply the value of sensor signal by a factor of -1 to generate a vertical mirror flip for input signal.
Shuffling	Permute all channels of sensor data randomly to simulate different wearing directions of the sensor.
Rotating	Plot a 3D random axis that obeys a uniform distribution and then rotate triaxial sensor readings (e.g., x, y, and z) by a random angle in the 3D space to simulate different sensor locations.
Resampling	Use linear interpolation to up-sample raw sensor data three times along time axis and then recover its original dimensions by random sub-sampling.

3.2. Time series contrastive supervision for HAR

In this subsection, we further present our proposed time series contrastive supervision when applying deep supervision for HAR. Without loss of generally, there are two crucial concerns: (1) How to form the positive pairs and negative pairs for activity data; (2) How to add the contrastive supervised loss. As shown in Fig. 2, in our work, contrastive supervision is employed after each hidden convolutional block f_i (i.e., $k = 1, 2, 3$) by attaching a projection head g_i, which can provide intermediate supervision so as to make the learned features more discriminative. That is to say, the convolutional blocks (i.e., f_1, f_2, f_3) denote the intermediate layers. On the other hand, the core idea of contrastive learning is to pull positive pairs together while pushing negative pairs away in the feature embedding space, as indicated by its name [23,24]. Intuitively, a positive pair may be formed from two views by various data augmentations for the same sample, while negative pairs may be formed by different samples. Nevertheless, prior most data augmentation techniques in contrastive learning primarily focus on image data (e.g., cropping-invariance), where such invariance is potentially suboptimal for activity data since cropping a subset from time-series data would inevitably cause information loss because of crucial temporal dependencies. According to this cue, this paper seeks to explore how to tailor time series data through different data transformation techniques [17, 26,39], as listed in Table 1. In fact, the main goal of the auxiliary branches is to compare x_i and x_{i+N} for contrastive supervision. Specifically, given a minibatch of activity sample $\mathcal{X} = \{x_i\}_i^N$ and its corresponding labels $\mathcal{Y} = \{y_i\}_i^N$, we can perform time series data augmentation over each instance twice, which may result in a minibatch of $2N$ augmented samples. For convenience, we regard x_i and x_{i+N} as a positive pair generated from two augmentations for the same sample, while x_i and other samples in a batch are regarded as a negative pair, as shown in Fig. 3.

As indicated above, the main goal of contrastive learning is to learn such an embedding normalized space where positive pairs stay close to each other while negative pairs are far apart. To this end, several additional projection heads g_k would be added to the intermediate layer k of backbone networks during training stage. In other words, $c_k = g_k \circ f_k(x)$ is in charge of mapping the backbone features (e.g. in the kth intermediate layer) into a lower-dimensional embedding space in the context of Contrastive Supervision. Since the input features used in Contrastive Supervision comes from the intermediate layers rather than the last layer, the design of such projection head inevitably plays a crucial role in model performance. In particular, it is worth emphasizing that these additional branches would be removed during inference time without cause extra memory and computational overhead.

To this concrete, let $z = c(x)$ denote a composite function consisting of the intermediate feature extractor $f_k(x)$ and a normalization projection head g_k, where g_k is comprised of a GAP operation, a ReLU activation function, and a linear FC layer [25,27] (Fig. 2), the contrastive loss for given positive and negative pairs may be formulated as:

$$\mathcal{L}_{\text{Con}} = -\sum_{i=1}^{N} \log \frac{\exp\left(z_i \cdot z_{i+N}\right)/\tau}{\sum_{m=1}^{2N} \mathbb{I}_{[m \neq i]} \exp\left(z_i \cdot z_m\right)/\tau}, \tag{3}$$

where τ is a constant temperature hyperparameter. $\mathbb{I}_{[m \neq i]}$ is an indicator function, whose value is equal to 1 if $m \neq i$ and 0 otherwise. The loss may be utilized to learn powerful representations in a self-supervised scenario. Without loss of generality, in the supervised scenario, label has been always used to guide the definition of positive and negative pairs [23,46]. In this case, Eq. (3) can be rewritten as:

$$\mathcal{L}_{\text{Con}} = \frac{-1}{2N_{\tilde{\boldsymbol{y}}_i} - 1} \sum_{i=1}^{N} \sum_{j=1}^{2N} \mathbb{I}_{i \neq j} \cdot \mathbb{I}_{\tilde{\boldsymbol{y}}_i = \tilde{\boldsymbol{y}}_j} \cdot \log \frac{\exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_j / \tau\right)}{\sum_{m=1}^{2N} \mathbb{I}_{i \neq m} \cdot \exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_m / \tau\right)}, \tag{4}$$

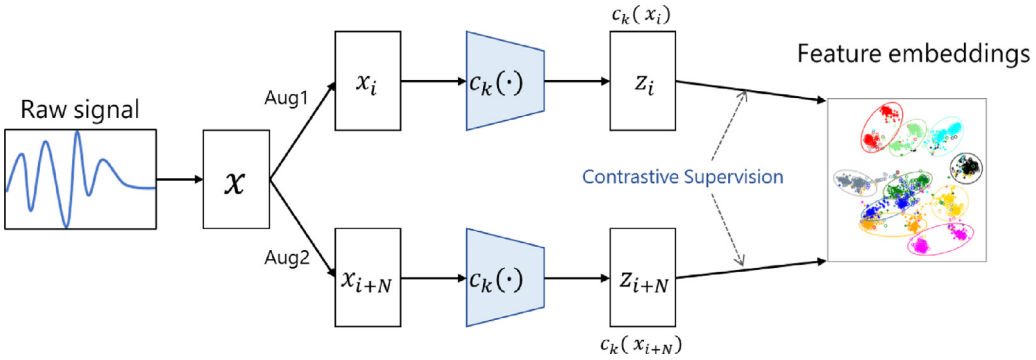


Fig. 3. Contrastive Supervision with time series data augmentation in kth convolutional layer.

where $N_{\tilde{\boldsymbol{y}}}$ denotes the number of total samples with the same label $\tilde{\boldsymbol{y}}$ in one batch. Assuming that $z_i = c_k(x_i)$ and $z_{(i+N)} = c_k(x_{(i+N)})$ are 128-dimensional embedding feature vectors, we further implement the contrastive loss (as represented as the arrows in Fig. 3 from z to feature embedded space) to force positive samples with similar semantics (in the same color) close and negative sample pairs with different semantics (in different colors) far apart. Given the contrastive loss $\mathcal{L}_{\text{Con}}$, we highlight that the main distinction between above deep supervision and our contrastive supervision is that deep supervision trains these attached intermediate classifiers with standard cross-entropy loss while our approach trains them by the contrastive loss with time series data augmentation. To this end, let $\mathcal{L}_{\text{Con}}(\mathcal{X}; c_k)$ denote the contrastive loss at c_k, one can recast Eq. (2) in the following form:

$$\mathcal{L}_{\text{CoS}} = \underbrace{\mathcal{L}_{\text{CE}}(\omega(\mathcal{X}), \mathcal{Y})}_{\text{from standard training}} + \alpha \cdot \sum_{k=1}^{K} \underbrace{\mathcal{L}_{\text{Con}}(\mathcal{X}; c_k)}_{\text{from contrastive supervision}}, \tag{5}$$

where the first item still represents the standard training loss while the second item denotes the contrastive supervision loss employed at the intermediate layers. Since the second item does not use task-related loss $\mathcal{L}_{\text{CE}}$ without yielding redundant optimizations, the last convolutional layer still utilizes Contrastive Supervision to further improve representation quality. Here the top panel in Fig. 2 illustrates a detailed architecture of the proposed contrastive supervision, which consists of a main branch and three attached auxiliary supervision branches. The contrastive supervision (i.e., projection head) is added after each hidden convolutional block along the main branch. The blocks with the same type are presented in the same color: the projection heads in auxiliary supervision branches are marked by dashed boxes, while each convolutional block in main branch is indicated by solid boxes. A legend below the schematic diagram indicates both block types including the projection head in auxiliary branch and convolutional block in main branch. During training stage, the optimized objective function is equivalent to a sum of the cross-entropy loss (final layer) and contrastive losses associated with all auxiliary branches, as indicated in Eq. (5).

On basis of above formulations in supervised learning setting, one can easily extend the proposed approach in a semi-supervised learning paradigm [4,19,23], where there is a labeled dataset $\mathcal{X}_1$ with labels $\mathcal{Y}_1$ and an unlabeled dataset $\mathcal{X}_2$ respectively. In this case, because of the lack of labels, we only apply the Contrastive Supervision to optimize the second item (i.e., $\mathcal{L}_{\text{Con}}$) on the unlabeled data:

$$\mathcal{L}_{\text{CoS}}(\mathcal{X}_1, \mathcal{Y}_1) + \mathcal{L}_{\text{Con}}(\mathcal{X}_2). \tag{6}$$

The relevant results will be introduced in the following sections.

4. Experiment setting, main results, and analyses

In this section, we first introduce the benchmark datasets and implementation details, and then evaluate our proposed Contrastive Supervision approach in the supervised learning scheme and semi-supervised learning scheme. To ensure fair comparisons, we conduct all the experiments by applying different training strategies in exactly the same settings such as data preprocessing, batch size, data splitting, and training epochs. Moreover, we also provide more comprehensive and in-depth ablation analyses for Contrastive Supervision by exploring the effectiveness of several key components. Finally, actual implementation is evaluated on an embedded platform.

4.1. Experimental setting

Benchmark datasets. Following previous standard data preprocessing pipeline [6,13,41], we adopt four popular benchmarks including UCI-HAR, PAMAP2, UniMib-SHAR, and WISDM recorded in different scenarios with various sensors, e.g., accelerometers, gyroscope, which are detailed as follows: **UCI-HAR** [30]: The dataset includes sensor recordings from 30 subjects, who are instructed to perform six activities of daily living (ADLs) comprised of "standing", "lying", "sitting", "walking", "walking upstairs" and "walking downstairs", while wearing a waist-mounted smartphone (Samsung Galaxy S II) with embedded inertial sensors. During data recording, the smartphone is utilized to acquire three-axis linear acceleration and three-axis angular velocity signals at a fixed sampling rate of 50 Hz. For performance evaluation, the data is divided into a 70:10:20 ratio for training, validation, and test. **PAMAP2** [32]: The dataset is composed of sensor recordings from nine participants, who are asked to participate in 18 types of physical activities, including 12 protocol activities such as "cycling", "walking", and "rope jumping" and a few optional activities such as car "driving", "playing soccer", and "watching TV". Each participant wears three wireless Inertial Measurement Units (IMUs) attached to different body positions including ankle,chest, and hand respectively. The original sampling rate of 100 Hz is further subsampled to 33.3 Hz for convenience of analysis. Referring to previous literatures [41,47], the data from Participant 5 and Participant 6 are held out as the validation set and test set, while the other participants' data is used for training. **UniMib-SHAR** [33]: This is a new acceleration dataset collected by the research team from the University of Milano Bica, which is purposely designed to monitor human activities and detect falls. 30 volunteers whose ages range between 18 and 60 years participate in the data collection. All samples are recorded by Android smartphones at a sampling rate of 50 Hz, which are roughly divided into two categories: 8 types of falls and 9 types of activities of daily living (ADLs). In particular, we perform a 30-fold

Table 2
Statistical information of datasets and experiment setup.

Dataset	UCI-HAR	PAMAP2	UniMib-SHAR	WISDM
Sampling rate (Hz)	50	33.3	50	20
Sensors	9	36	3	3
Subjects	30	9	30	29
Classes	6	12	17	6
Window size	128	171	151	200
Overlap (%)	50	78	50	90
Learning rate	1e−2	1e−2	1e−2	1e−2
Batch size	128	512	256	128
Epoch	200	200	200	200
Augmentation	Flipping	Flipping	Rotating	Permuting

leave-one-subject-out cross validation for our evaluation [48].

WISDM [31]: The Wireless Sensor Data Mining (WISDM) research team collects this activity dataset via utilizing various mobile devices such as smartphones, laptop computers, and music players. The whole 29 volunteers participate in data collection, where each of them wears an Android smartphone in front leg pocket and performs 6 different types of activities including "walking", "upstairs", "standing", "jogging", "downstairs" and "sitting". The sampling rate is constantly maintained at 20 Hz, resulting in total 10,981 samples. Referring to previous literatures [31,49], a 10-fold cross validation is utilized for the experiments. The details of data processing are introduced in Table 2.

Implementation details. To show relative performance gain, we apply different training strategies (i.e., traditional supervision v.s. Contrastive Supervision) to our baseline CNN architecture for comprehensive comparisons. To implement Contrastive Supervision, we attach three auxiliary projection heads to all intermediate layers of the CNN backbone, where each auxiliary projection head is respectively added after the corresponding convolutional block containing a sub-sampling layer, as shown in Fig. 2. More specifically, the three convolutional blocks contain 64, 128, and 256 channels, respectively. As indicated above, all auxiliary projection heads have the same structure comprised of a GAP layer, a ReLU activation function, and a linear FC layer, which are in charge of transforming the backbone features into 128-dimensional embedding feature vectors for Contrastive Supervision. All networks are trained by SGD optimizer with a momentum of 0.9. The initial learning rate starts at 0.01, which is gradually decayed by a factor of 10 every 40 epochs through total 200 training epochs. Because Contrastive Supervision derives positive or negative pairs in mini-batch, we apply varying batch sizes (also see Table 2) on different benchmark datasets to ensure global convergence. To comprehensively evaluate the proposed Contrastive Supervision, we apply four popular performance metrics including Accuracy, F1-score, Recall, and Precision, which are formulated as follows:

$$\begin{aligned} \text{Accuracy} &= \frac{TP+TN}{TP+FP+FN+TN}, \\ \text{F1-score} &= 2\times\frac{\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}}, \\ \text{Recall} &= \frac{TP}{TP+FN}, \\ \text{Precision} &= \frac{TP}{TP+FP}, \end{aligned} \tag{7}$$

in which TP, FP, TN, and FN denote true positives, false positives, true negatives, and false negatives respectively [11,15,50]. All models are implemented with PyTorch, and the code is released at https://github.com/cheng-haha/CoS. Each model is repeated 5 times and the average results are reported for reliable comparison.

Table 3
Test results on benchmark datasets.

Dataset	UCI-HAR	PAMAP2	UniMib-SHAR	WISDM
Baseline	A(%)96.33	90.34	75.13	97.21
	F(%)96.39	90.52	74.78	97.21
	R(%)96.41	90.62	75.15	97.21
	P(%)96.62	91.59	76.26	97.42
CoS	A(%)**98.00** (↑**1.67**)	**93.22** (↑**2.88**)	**78.83** (↑**3.70**)	**98.89** (↑**1.68**)
	F(%)**98.00** (↑**1.61**)	**92.97** (↑**2.45**)	**78.30** (↑**3.52**)	**98.88** (↑**1.67**)
	R(%)**98.01** (↑**1.60**)	**93.04** (↑**2.42**)	**78.83** (↑**3.68**)	**98.88** (↑**1.67**)
	P(%)**98.14** (↑**1.52**)	**93.77** (↑**2.18**)	**80.27** (↑**4.01**)	**98.98** (↑**1.56**)
Related works	95.41A [51]	89.96F [41]	75.65A [52]	97.20F [15]
	95.75A [6]	90.40F [47]	77.03A [48]	96.44A [11]
	95.38A [15]	89.30F [7]	76.39F [53]	98.23A [54]

A: Accuracy. F: F1-score. R: Recall. P: Precision.

4.2. Main results

Supervised activity classification. Table 3 summarizes our main results, where the Baseline denotes standard training scheme with traditional supervised loss, while CoS denotes the Contrastive Supervision learning scheme with additional auxiliary projection heads. In general, it could be observed that our approach performs the best in all cases, which can significantly boost model performance compared to the corresponding baseline. More specifically, its gain in UCI-HAR and PAMAP2 is 1.67%/1.61%/1.60%/1.52% and 2.88%/2.45%/2.42%/2.18% in terms of performance metrics, i.e., Accuracy/F1-score/Recall/Precision, respectively. Benefiting from the proposed Contrastive Supervision, CoS achieves better results on UniMib-SHAR and WISDM, which outperforms the standard baselines with a large margin of 3.70%/3.52%/3.68%/4.01% and 1.68%/1.67%/1.67%/1.56% in terms of Accuracy/F1-score/Recall/Precision, respectively. The above experimental results clearly validate the effectiveness of the proposed approach when implementing Contrastive Supervision for activity recognition.

Decoupling contrastive loss and deep supervision on optimization. Because these above results could not clearly disentangle the independent effects of each component on optimization, we compare different training schemes on several benchmark datasets by looking at their respective test error curves so as to better shed light on this. To ensure fair comparisons, we have included test errors for the tested backbone architecture with (or without) deep supervision and contrastive loss, respectively. As shown in Fig. 4, for networks trained with deep supervision alone (DS) [21], the standard cross-entropy is attached to each intermediate layer. Instead, for networks trained with contrastive loss without deep supervision (Con) [46], the contrastive loss is only added to the last layer. In fact, once training is done, all auxiliary branches could be removed during inference stage, and then both DS and CoS would have the same architecture in all the intermediate convolutional layer and final FC layer (i.e., main branch). That is to say, one may discard all attached projection heads at the end of contrastive training. As a consequence, our inference-time model has exactly the same number of parameters and FLOPs as such deeply-supervised model using the same architecture. However, we highlight that there is still a clear distinction between DS and CoS during training time: the former trains all auxiliary branches by the standard cross-entropy loss $\mathcal{L}_{\text{CE}}$, while the latter trains them with the contrastive loss $\mathcal{L}_{\text{Con}}$. All the hyperparameters other than the loss functions are kept the same for a given dataset. As indicated above, the objective of deep supervision is to create a hidden layer representation that can be suitable for classification, independently of the last layer. An interesting observation is that in some cases the sole deep supervision is not able to provide a remarkably good supervised

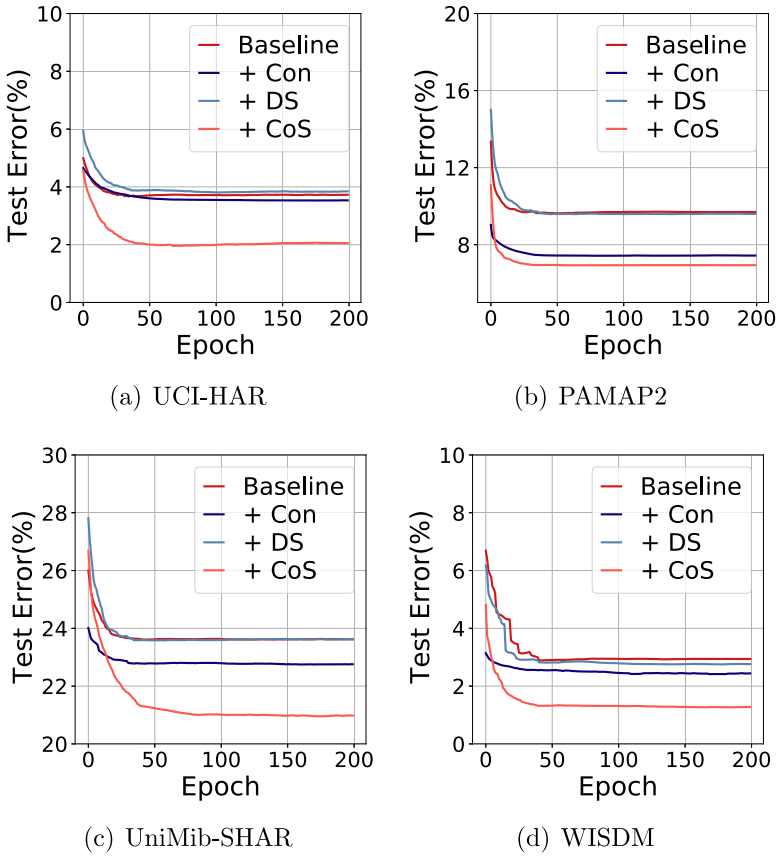
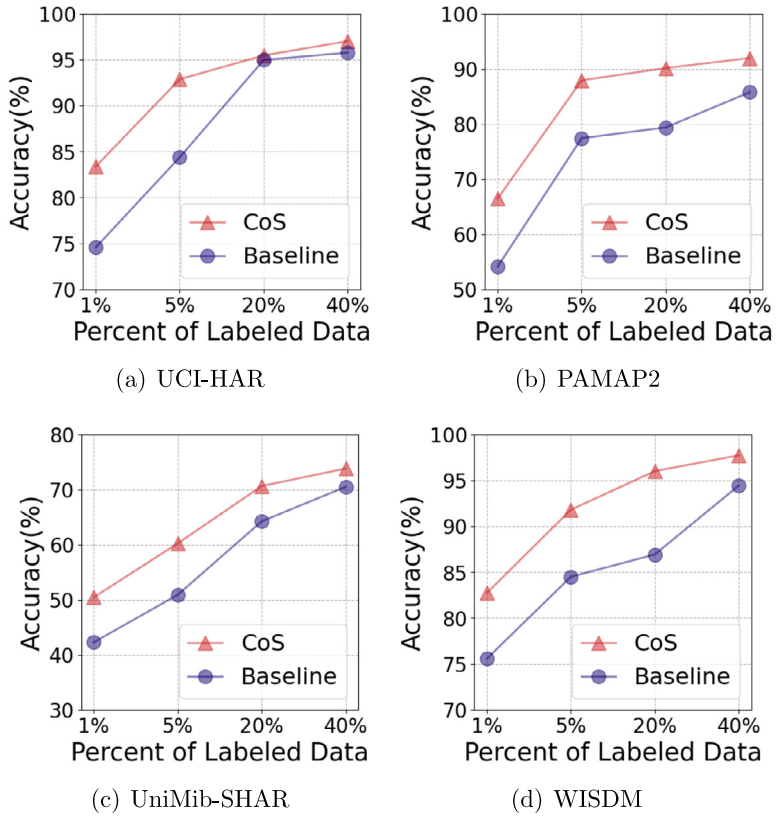


Fig. 4. Compare the test errors of different methods.

signal for intermediate layers. In some cases, the test errors are even lower than those obtained by the standard training scheme with the final layer loss, which indicates that the traditional supervised loss probably could not provide the best supervision for optimizing these intermediate layers. As illustrated by test error curves, it can be clearly observed that using traditional deep supervision (DS) or contrastive loss (Con) alone does not match our contrastive supervised loss (CoS). However, the recognition performance improves significantly when both are combined. On average, the Contrastive Supervision may lead to a considerable drop in test errors compared to the other supervisions, which implies that the invariant representations learned by our method are more beneficial for optimizing the intermediate layers.

Comparing with the state-of-the-arts. We further compare CoS with previous state-of-the-art (SOTA) HAR algorithms including AttenSense [7] (Ma et al. 2019), CNN [6] (Ronao et al. 2016), Deep Belief Networks [54] (Alsheikh et al. 2016), HFBS [51](Dong et al. 2021), Continuous Attention [41] (Zeng et al., 2017), and attention-based CNN [15] (Khan et al. 2021), etc. As shown in Table 3, it could be observed that our CoS achieves the best performance against prior works across various networks. On UCI-HAR, our CoS significantly outperforms Ronao et al. [6]'s approach that uses the same CNN architecture without additional Contrastive Supervision by 2.25%. It also surpasses Dong et al. [51]'s approach utilizing hesitant fuzzy belief framework and Khan et al. [15]'s approach using CNN-induced multi-head attention by 2.59% and 2.62%, respectively. In the case of PAMAP2, the CNN trained with CoS is significantly superior to the AttenSense approach proposed by Ma et al. [7], achieving a 3.67% performance improvement. Zeng et al. [41] have proposed an approach that combines Continuous Temporal Attention and Continuous Sensor Attention within an individual LSTM unit, resulting in an F1-score of 89.96%. Our CoS surpasses the second-best baseline by a margin of 3.01% according to F1-score. Comparing to the strongest baseline reported by Khaertdinov et al. [47] who utilize different triple loss functions, our approach is still able to lead to an absolute performance gain of 2.57%. In addition, our CoS method surpasses the previous SOTA [48] using Codebook Approach with Soft Variants by a margin of 1.80% on UniMib-SHAR and a margin of 1.91% over the second-best method using an Asymmetric Residual Neural Network [53], respectively. Similar gain could also be seen in WISDM. On average, our CoS method leads to an accuracy improvement of 0.66% over Alsheikh et al. [54]'s approach using deep belief networks. It also surpasses the second-best method reported by Zhang et al. [11] utilizing multi-head convolutional attention by a relative performance gain of 2.45%. These results suggest that exploiting contrastive representation may provide an effective way for deeply-supervised learning beyond standard supervised strategy.

Apply Contrastive Supervision to semi-supervised learning. Current popular deep learning approaches for HAR heavily rely on the availability and quantity of labeled data. However, as aforementioned, labeling sensor data is time-consuming and labor-intensive. Time series sensor data annotation is often difficult and costly to curate, and as a result direct application of deep models to human behavior analysis tends to drastically overfit due to the lack of labeled sensor data. Hence, to overcome such limitations, we are motivated to explore semi-supervised learning for improving data efficiency and performance of deep models with limited labeled data. To this end, we propose to extend the fully-supervised contrastive approach to the semi-supervised setting, that combines the power of data-efficient contrastive-supervised feature learning via applying contrastive loss in different depths, hence being more effective in leveraging label information. Following the common practice in semi-supervised learning paradigm, we conduct experiments with a small portion



(a) UCI-HAR (b) PAMAP2

(c) UniMib-SHAR (d) WISDM

Fig. 5. Experiment results in the semi-supervised setting.

of labeled data and a large portion of unlabeled data to evaluate how effective our CoS method improves model performance with limited labeled data. We investigate the effectiveness of our proposed method in a semi-supervised HAR scenario, by training the baseline model with 1%, 5%, 20%, and 40% of the randomly selected activity samples from training set. Fig. 5 presents the comparison results of our CoS along with the semi-supervision under the aforementioned settings, where there are two main observations: (1) Our CoS would lead to a consistent performance gain along all the ratios of labeled data. (2) The performance gap brought by our CoS tends to broaden when there are fewer labeled samples. In particular, it can be clearly observed that our CoS (red curves) achieves significantly higher accuracy when using only 1% of labeled data, while traditional supervised training (purple curves) performs much worse due to limited labeled data, which indicates that such contrastive learning is very effective in applying smaller labeled data to supervise these hidden layers. Overall, excellent performance can be obtained, demonstrating the effectiveness of our CoS in the semi-supervised setting, which consistently surpasses the supervised-only baselines with 1%, 5%, 20%, and 40% labeled data by a large margin on all four benchmarks. For instance, it can be clearly observed that our CoS (red curves) achieves significantly higher accuracy when using only 1% of labeled data (e.g., relative 6.33%, 16.62%, 6.76%, and 7.73% performance gains), while traditional supervised training (purple curves) performs much worse due to limited labeled data, which indicates that such contrastive learning is very effective in applying smaller labeled data to supervise these hidden layers. Our CoS with 5% labeled data obtains comparable accuracy compared to the supervised-only baseline trained with 100% labeled data on PAMAP2 dataset. The results confirm the data-efficiency of CoS in the semi-supervised HAR scenario.

4.3. Ablation study

The effect of varying network depth. In fact, one could find a large number of possible combinations for hyper-parameters in such an experiment setting. Based on an idea of greedy-wise tuning, we increase the number of layers from two to six (i.e., layer-two, layer-three, layer-four, layer-five, layer-six) to access the impact of varying network depth. We first compare our CoS with the corresponding baseline on PAMAP2 and WISDM by analyzing the effect of increasing the number of layers on recognition accuracy. As illustrated in Fig. 6, it can be seen that adding more layers does not always translate into performance improvement, where there is a non-monotonic trend according to recognition accuracy as the number of layers is increased. The performance gain from layer-three to layer-four is almost negligible. In particular, adding the fifth and sixth layer even results in a considerable decrease in accuracy from previous layer, which indicates that an overfitting phenomenon occurs. Comparing to the baseline, as well as the aforementioned Con and DS, one can find that attaching Contrastive Supervision to all intermediate layers could lead to a consistent and significant performance gain at different depths. One can argue that the CoS with time series data augmentation might be playing a regulating role in helping to alleviate overfitting problem caused by limited activity data, which allows the network to learn more discriminative features at larger depths. The figures indicate that there is a steady increase in accuracy as the number of layers is increased, and the accuracy improvement only tends to saturate when the number of layers is greater than 4. Though a larger model capacity is generally more beneficial, our ablations analyses suggest that a proper network capacity should be layer-specific for activity classification in case of limited data, which agrees well with previous observation reported in Ronao et al.'s literature [6].

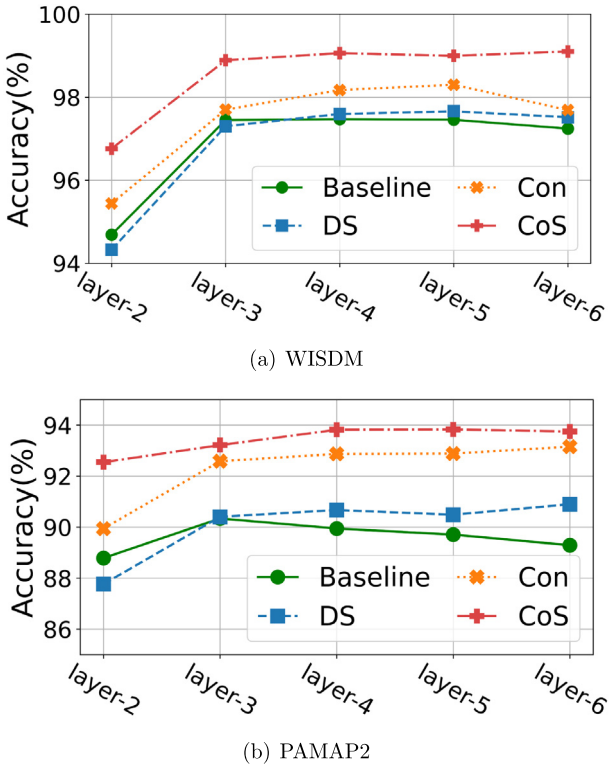


Fig. 6. Performance comparisons at different network depths.

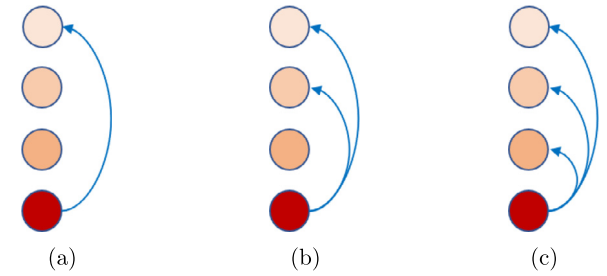


Fig. 7. Visualization for using CoS in the intermediate layers. (a) denotes utilizing CoS in a single layer. Similarly, (b), (c) respectively represent CoS applied to two or three intermediate layers.

Table 4
Apply CoS at different depths. 1st, 2nd, and 3rd represent the first, second, and third layers of the whole network, respectively. The ✓ in the each column indicates that the Contrastive Supervision is applied after the corresponding layer.

	1st	2nd	3rd	Accuracy (%)
Base	–	–	–	75.13
Placement	✓	–	–	75.40
	–	✓	–	76.08
	–	–	✓	77.59
	✓	✓	–	75.48
	✓	–	✓	78.25
	–	✓	✓	77.83
	✓	✓	✓	**78.83**

Table 5
Effect of hyperparameter α.

	UCI-HAR	PAMAP2	UniMib-SHAR	WISDM
1.0	97.33	92.51	78.27	98.88
Base	96.33	90.34	75.13	97.21
5.0	**98.00**	92.98	77.71	98.59
10.0	97.79	93.02	**78.83**	98.64
15.0	97.87	**93.22**	78.64	**98.89**
20.0	97.63	93.14	78.13	98.89

Where to place contrastive loss? Since our CoS can be employed at different layers of a whole neural network, where to add contrastive loss along the network remain a critical problem. To address this issue, we perform extensive experiments on UniMib-HAR dataset to analyze the effect of auxiliary projection heads by attaching them to at most three different intermediate layers and training each model independently, as illustrated in Fig. 7. Table 4 presents the corresponding results, from which there are three main observations: (1) Two or three auxiliary projection heads are able to achieve larger accuracy gain than only one auxiliary projection head; (2) Adding a single auxiliary projection head to a relatively deep layer is better than adding it to a shallow layer, which is in well line with previous observation [22]. One can observe a gradual increase in accuracy along with the increased number of auxiliary projection heads. Intuitively, if too less Contrastive Supervision is applied, it may not fully exploit Contrastive Supervision to learn invariant representations from time series data augmentation. Referring to the above results, we choose to attach auxiliary projection heads to all intermediate layers, where shallow layers are in charge of learning low and local activity features while deep layers are in charge of learning global and high-level activity features. In such a way, our CoS can combine both shallow and deep layer outputs from Contrastive Supervision to improve final activity recognition performance.

The impact of hyper-Parameter α. We perform ablation analysis on several HAR benchmarks to explore the impact of the key hyper-parameter α, that is used to balance the trade-off between supervised and contrastive losses. Table 5 ablates the effect of α on the overall performance, which indicates a non-monotonic variation trend. In fact, α reflects the weights of contrastive loss included in the overall loss function. As α is set to 0, our model will degenerate to the standard CNN architecture trained by the final layer loss. On average, our approach always outperforms the corresponding baseline when tuning the ratio $\alpha > 0$, which shows the necessity of contrastive loss. For example, setting $\alpha =$ 1, one can consider that Contrastive Supervision employed at all intermediate layers and standard supervision employed at the last layer play equal roles in the optimizing process. It can be clearly seen that increasing the percentage of contrastive loss tends to improve model performance, but too large percentage would harm the performance. The optimal selection of α could be dynamically adjusted according to activity recognition performance on validation set. In most cases, the contrastive loss will be allocated higher weights than the traditional loss at the last layer, which ensures that such contrastive loss always plays a dominant role compared to the final layer loss.

Sensitivity analysis on temperature factor τ. We perform sensitivity analysis on UCI-HAR dataset to investigate the effect of a temperature hyper-parameter τ on the classification performance of Contrastive Supervision, which plays a dominant part in controlling the strength of penalties on hard negative samples. As one can see from above Eq. (3), the contrastive loss scales inversely as the temperature τ varies. Because extremely small temperature value (e.g., $\tau = 0.001$) might make the model hard to converge due to numerical instability, we only show the effect of temperature scalar τ that is varied between 0.01 and 1.0 while keeping all other hyperparameters fixed. It can be found that Fig. 8 presents an inverse U shape, which indicates that our model is less sensitive to its value when $\tau \leq 0.5$ while it is more sensitive to the value that is larger than 0.5. On the whole, relatively low temperature benefits Contrastive Supervision more than high ones. Unsurprisingly, the temperature between 0.1 and

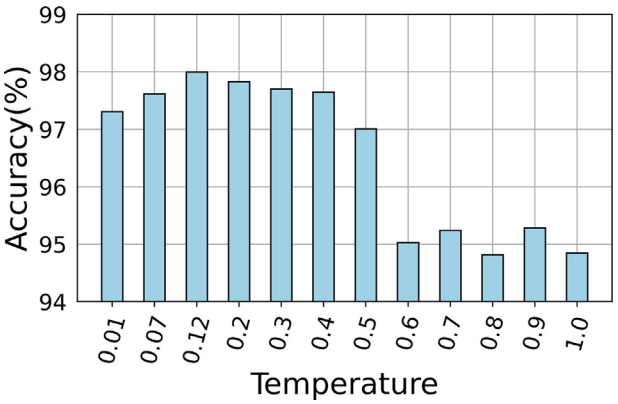
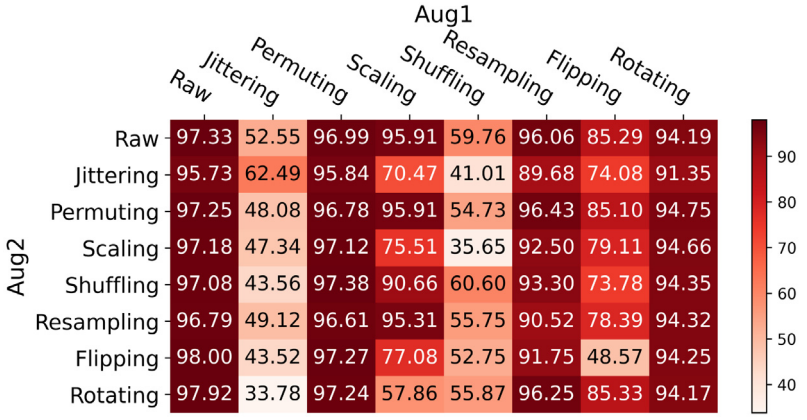


Fig. 8. Effect of temperature factor τ.

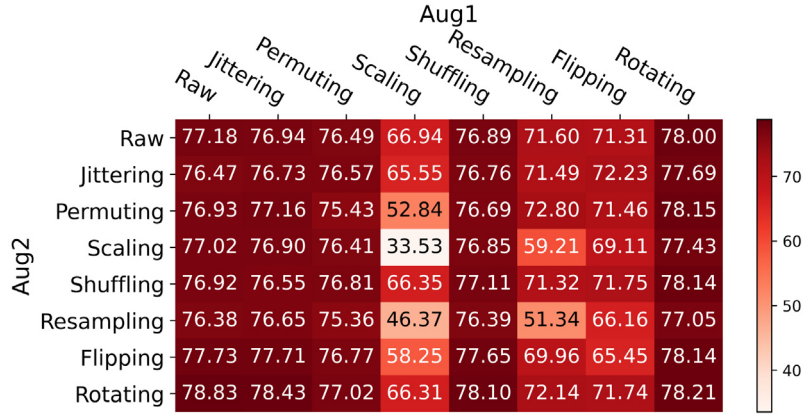
0.5 can achieve better performance, while an extremely high temperature leads to a considerable accuracy drop, i.e., suboptimal performance. This is due to that contrastive loss caused by extremely high temperature tends to be less sensitive to hard negative samples and degenerate model performance, which is in well line with previous observations in contrastive learning [46,55]. Without loss of generality, we keep the temperature value within a reasonable interval (i.e., $\tau = 0.12$) for all our experiments.

The effect of data augmentation. As shown in Fig. 3, to learn invariant representations from different views, the Contrastive Supervision needs to construct positive pairs by applying different data augmentations for the same activity twice in an instance level. We empirically explore how various time series data augmentations (as listed in Table 1) affect the final performance of our Contrastive Supervision approach. Fig. 9 shows the obtained results on UCI-HAR and UniMib-SHAR datasets respectively, where Aug1 denotes the data augmentation applied for the first branch, while Aug2 denotes the data augmentation applied for the second branch. To provide a comprehensive evaluation, we sometimes still use raw sensor samples in one branch while only performing data augmentation in the other branch, or vice versa. As shown in Fig. 9, one can find that there is quite a big variation between performance, where there is no unique data augmentation that consistently outperforms others. In most cases, using a single augmentation performs even better than using both. For example, using *Flipping* and *Rotating* alone can achieve the best performance on UCI-HAR and UniMib-SHAR datasets respectively, which suggests that combining two augmentations at the same time might potentially cause the distortion of an activity instance and fail to preserve its original semantic meaning for activity classification. Unlike image data, it should be cautioned that the variance caused by the time series data augmentation could not be taken for granted. In fact, how to find a proper time series data augmentation automatically still remains a critical challenge in contrastive learning [16,17].

Confusion matrices and visualizing analysis. To demonstrate the discriminative power of Contrastive Supervision, we



(a) UCI-HAR



(b) UniMib-SHAR

Fig. 9. Different data augmentation techniques.

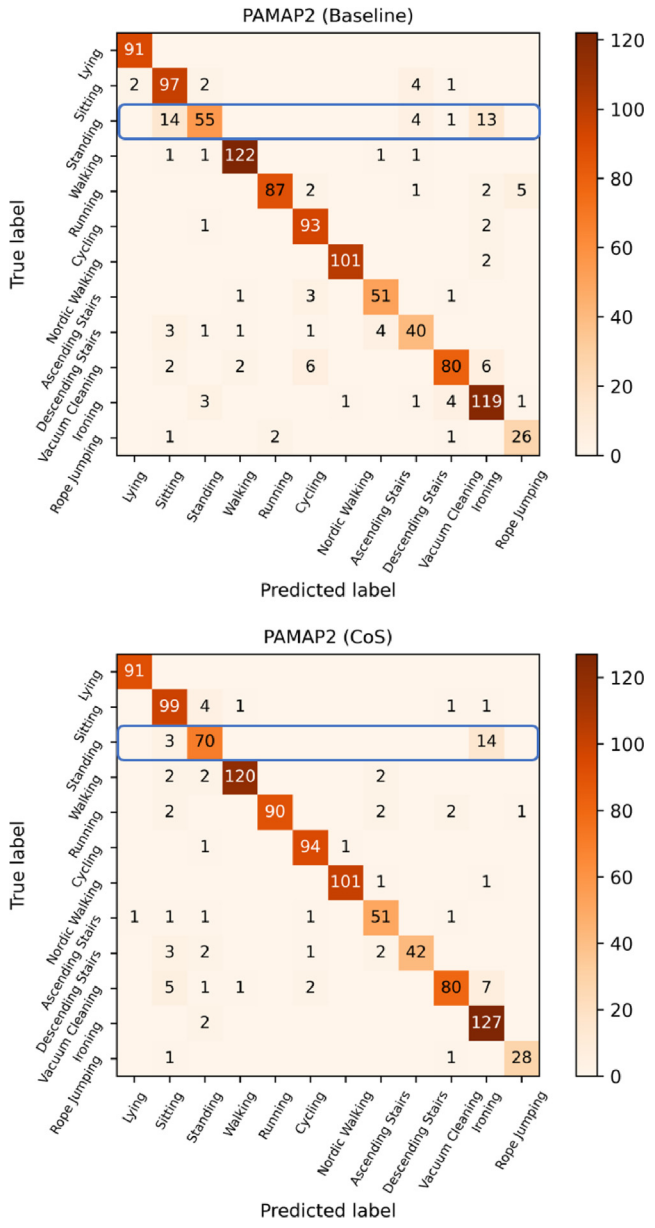


Fig. 10. Confusion matrix for PAMAP2 dataset.

further compute the confusion matrices of 12 studied activities on PAMAP2 dataset, which will be more informative based on a detailed analysis of all caused errors or confusions between different activity classes. Fig. 10 presents the confusion matrices of the CNN trained by Contrastive Supervision (bottom panel) and traditional supervision (top panel), respectively, where each element along the main diagonal line denotes the number of correctly recognized activities (higher is better), while the remaining off-diagonal elements denote the number of misclassifications. By inspecting both diagonal and off-diagonal elements, one can clearly observe that some activity categories are more difficult to distinguish than other ones. For example, as shown in the left panel, since "Standing", "Ironing", and "Sitting" have high confusion with each other, there are only 55 samples belonging to the "Standing" category that have been correctly predicted by traditional supervision. In contrast, our method can further improve the number of correct predictions to reach a higher level, (i.e., 70), as illustrated in the right panel. In order to obtain a better understanding about how the Contrastive Supervision

Table 6
Test F1-score on PAMAP2 Dataset.

Model	Plain baseline	Plain baseline + CoS
LSTM-Baseline [14]	75.60	**81.87**
DeepConvLSTM [13]	74.80	**86.42**
Att. model [57]	87.50	**89.03**
CAE [58]	82.90	**91.48**
Attend-Discriminate [59]	90.80	**91.93**

succeeds, we provide a T-SNE [56] visualization in the embedded feature space, as shown in Fig. 11. Compared to the baseline method, it can be seen that the embedding learned with our Contrastive Supervision indeed shows better activity class separation, which presents a reasonable distribution being locally clustered and globally separated between those similar activities.

Integration with other modern architectures. To evaluate the generalization performance of our approach and whether it is compatible with what were done in prior works, we further test our CoS method with several popular backbone networks on PAMAP2 dataset. Our main intention is to evaluate the benefit when embedding our CoS into the corresponding architectures, so that the relative performance gain could be attributed to invariant feature representations caused by contrastive supervision. Here we apply CoS uniquely on several top-performing published models, and modify them similarly as in our CNN classification experiments. **LSTM-Baseline** [14]: a two-layer LSTM network where we only add the CoS block after the LSTM block/stage of baseline; **DeepConvLSTM** [13]: a hybrid model of CNN and LSTM for activity recognition, that is comprised of four convolutional layers and two LSTM layers to learn both spatial and temporal dependency, where the CoS block is applied after each intermediate convolutional layer, as well as the final LSTM block/stage; **Att. Model** [57]: it can be regarded as an attention-based version of DeepConvLSTM, which embeds an attention mechanism into the LSTM network so as to determine the 'important' time step, where CoS is applied after each convolutional layer and the attention module; **CAE** [58]: an autoencoder network consists of an encoder and a decoder bridged by a bottleneck layer, where the feature vector of the encoder is mapped onto the latent feature space via this bottleneck layer. The encoder includes four convolutional blocks, and the decoder employs four deconvolution blocks in sequence to reconstruct the original input by reversing the encoding process. Here CoS is employed after each convolutional block, which is then applied to the latent representation vectors obtained from the encoder without attaching projection head; **Attend and Discriminate** [59]: a novel cross-channel interaction encoder, which incorporates a self-attention mechanism to learn the latent interactions between multiple sensor channels so as to exploit different capabilities of sensor modalities in capturing and encoding activities. CoS is applied after each convolutional layer and an attention-based GRU encoder to enhance the feature representation. Table 6 reports the results by comparing LSTM-Baseline [14], DeepConvLSTM [13], Att. Model [57], CAE [58], Attend and Discriminate [59] with their CoS counterparts on the PAMAP2 validation set, respectively. The baseline numbers are taken from the referenced papers [13,14, 57–59]. To enable a fair comparison, we also re-implement all models using contrastive supervision in the same setting reproduced from previous works for HAR. It can be seen that incorporating our time series data augmentation strategies into contrastive supervision could consistently improve the results further over the supervised-only counterparts, and our CoS is capable of benefiting other network architectures based on the F1-score metric. It surpasses the previous state-of-the-art models, which leads to relative performance gains of 6.27%, 11.62%, 1.53%,

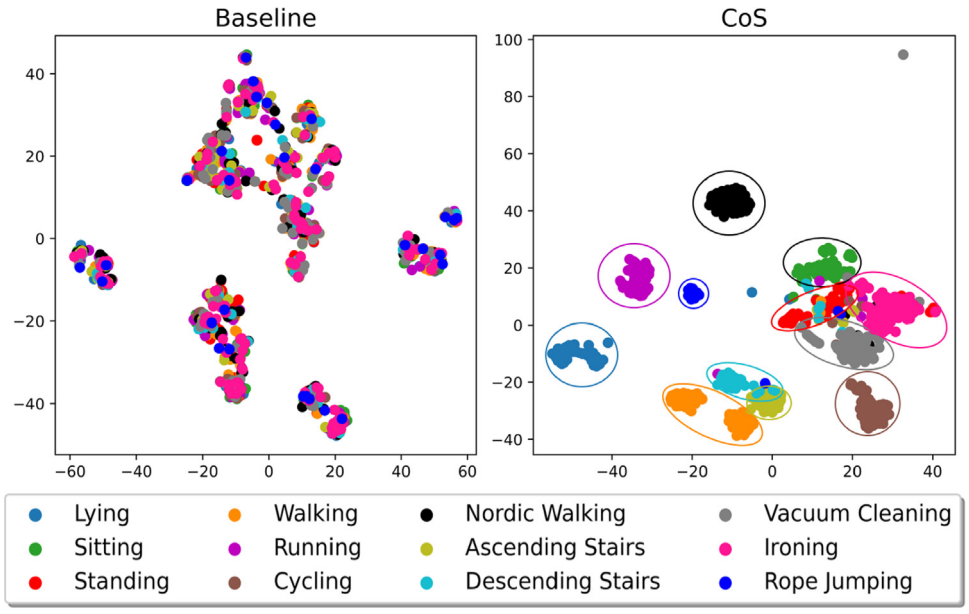


Fig. 11. T-SNE visualization in the feature embedding space. Different colors represent different activity categories.

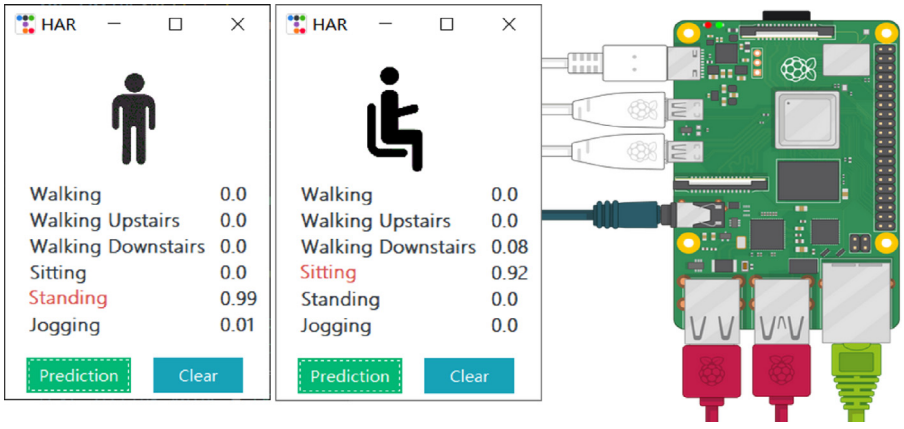


Fig. 12. Model deployment.

8.58%, and 1.13% on this HAR task, providing evidence that our CoS block can perform well on different models.

Actual Implementation. The inference latency would be an important factor in deploying a deep network for activity recognition. To provide an accurate enough estimation, we evaluate actual runtime of activity inference on a resource-restrained mobile device beyond considering an indirect metric, i.e., FLOPs alone. Since PyTorch currently can support deploying deep learning models for Raspberry Pi 4, this work is all tested with Raspberry Pi 4 Model B equipped with Quad-Core Cortex-A72 (ARM v8) 64-bit SoC@1.5 GHz and 4 GB LPDDR4 SDRAM. Following the standard workflow, we first train our backbone network with and without Contrastive Supervision respectively on WISDM dataset, and then load two trained PyTorch models and use them to execute activity inference. For quantifying inference time, an application program is written in Python language and its main user interface is shown in Fig. 12, in which an activity label highlighted in red color indicates the obtained prediction result. The *time* library in Python is used to measure inference time. Because the variance of the runtime can be significant, it is essential to run the network over plenty of activity examples and then average the results (400 examples can be a good number). The mean and variance of the measurements can be seen in Fig. 13, which indicates that our Contrastive Supervision can enhance the abstraction ability of CNNs without incurring extra inference-time cost.

5. Conclution

In this work, we present a generic Contrastive Supervision approach to tackle various time series data augmentations and learn the hierarchical augmentation invariance at different depths of neural network. Experiments on several activity recognition benchmarks demonstrate that the proposed method leads to a consistent and significant performance boost in supervised and semi-supervised settings. Detailed ablation analyses are conducted to study the effectiveness of each component. By computing multiple contrastive losses at different intermediate layers of backbone network, we show that deeply supervised learning can help contrastive loss to realize better fusion of low-level and high-level features in sensor data, which strongly suggests that attaching contrastive loss to intermediated layers could prevent the augmentation induced information loss. In addition, to guide the choice of augmentations, we systematically analyze the effect of data augmentations in our proposed method. Actual implementation is run on an embedded platform, which indicates that our method does not incur extra inference-time cost. We hope

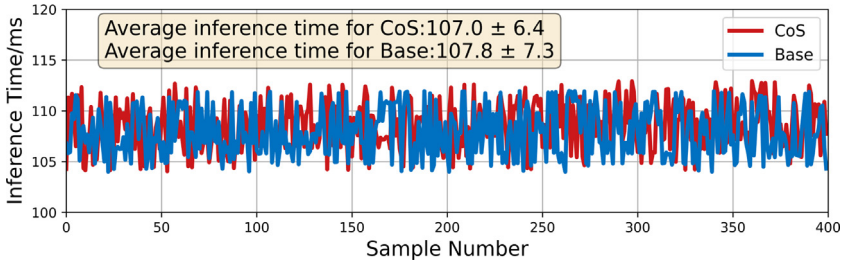


Fig. 13. Inference time.

our work may motivate other researchers to shed light on how such contrastive loss helps deep neural works to improve the feature quality in connection with sensor data for downstream activity recognition tasks.

CRediT authorship contribution statement

Dongzhou Cheng: Software, Methodology, Conceptualization. **Lei Zhang:** Writing – original draft, Data curation. **Can Bu:** Visualization, Validation. **Hao Wu:** Supervision. **Aiguo Song:** Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

The work was supported in part by the National Nature Science Foundation of China under Grant 61962061 and the Industry Academia Cooperation Innovation Fund Projection of Jiangsu Province under Grant BY2016001-02, and in part by the Natural Science Foundation of Jiangsu Province under grant BK20191371.

References

[1] Andreas Bulling, Ulf Blanke, Bernt Schiele, A tutorial on human activity recognition using body-worn inertial sensors, ACM Comput. Surv. 46 (3) (2014) 1–33.
[2] Santosh Kumar Yadav, Kamlesh Tiwari, Hari Mohan Pandey, Shaik Ali Akbar, A review of multimodal human activity recognition with special emphasis on classification, applications, challenges and future directions, Knowl.-Based Syst. 223 (2021) 106970.
[3] Daniel Garcia-Gonzalez, Daniel Rivero, Enrique Fernandez-Blanco, Miguel R Luaces, New machine learning approaches for real-life human activity recognition using smartphone sensor-based data, Knowl.-Based Syst. (2023) 110260.
[4] Claudio Bettini, Gabriele Civitarese, Riccardo Presotto, Caviar: Context-driven active and incremental activity recognition, Knowl.-Based Syst. 196 (2020) 105816.
[5] Jindong Wang, Yiqiang Chen, Shuji Hao, Xiaohui Peng, Lisha Hu, Deep learning for sensor-based activity recognition: A survey, Pattern Recognit. Lett. 119 (2019) 3–11.
[6] Charissa Ann Ronao, Sung-Bae Cho, Human activity recognition with smartphone sensors using deep learning neural networks, Expert Syst. Appl. 59 (2016) 235–244.
[7] Haojie Ma, Wenzhong Li, Xiao Zhang, Songcheng Gao, Sanglu Lu, AttnSense: Multi-level attention mechanism for multimodal human activity recognition., in: IJCAI, 2019, pp. 3109–3115.
[8] Nils Y Hammerla, Shane Halloran, Thomas Plötz, Deep, convolutional, and recurrent models for human activity recognition using wearables, 2016, arXiv preprint arXiv:1604.08880.
[9] Eunji Kim, Interpretable and accurate convolutional neural networks for human activity recognition, IEEE Trans. Ind. Inform. 16 (11) (2020) 7190–7198.
[10] Rui Xi, Mengshu Hou, Mingsheng Fu, Hong Qu, Daibo Liu, Deep dilated convolution on multimodality time series for human activity recognition, in: IJCNN, IEEE, 2018, pp. 1–8.
[11] Haoxi Zhang, Zhiwen Xiao, Juan Wang, Fei Li, Edward Szczerbicki, A novel IoT-perceptive human activity recognition (HAR) approach using multihead convolutional attention, IEEE Internet Things J. 7 (2) (2019) 1072–1080.
[12] Kaixuan Chen, Dalin Zhang, Lina Yao, Bin Guo, Zhiwen Yu, Yunhao Liu, Deep learning for sensor-based human activity recognition: Overview, challenges, and opportunities, ACM Comput. Surv. 54 (4) (2021) 1–40.
[13] Francisco Javier Ordóñez, Daniel Roggen, Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition, Sensors 16 (1) (2016) 115.
[14] Yu Guan, Thomas Plötz, Ensembles of deep lstm learners for activity recognition using wearables, Proc. ACM Interact., Mob., Wearable Ubiquitous Technol. 1 (2) (2017) 1–28.
[15] Zanobya N. Khan, Jamil Ahmad, Attention induced multi-head convolutional neural network for human activity recognition, Appl. Soft Comput. 110 (2021) 107671.
[16] Chi Ian Tang, Ignacio Perez-Pozuelo, Dimitris Spathis, Cecilia Mascolo, Exploring contrastive learning in human activity recognition for healthcare, 2020, arXiv preprint arXiv:2011.11542.
[17] Hangwei Qian, Tian Tian, Chunyan Miao, What makes good contrastive learning on small-scale wearable-based tasks? in: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2022, pp. 3761–3771.
[18] Ming Zeng, Tong Yu, Xiao Wang, Le T. Nguyen, Ole J. Mengshoel, Ian R. Lane, Semi-supervised convolutional neural networks for human activity recognition, in: IEEE BigData, IEEE Computer Society, 2017, pp. 522–529.
[19] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, Geoffrey E Hinton, Big self-supervised models are strong semi-supervised learners, Adv. Neural Inf. Process. Syst. 33 (2020) 22243–22255.
[20] Renjie Li, Xinyi Wang, Guan Huang, Wenli Yang, Kaining Zhang, Xiaotong Gu, Son N Tran, Saurabh Garg, Jane Alty, Quan Bai, A comprehensive review on deep supervision: Theories and applications, 2022, arXiv preprint arXiv:2207.02376.
[21] Chen-Yu Lee, Saining Xie, Patrick Gallagher, Zhengyou Zhang, Zhuowen Tu, Deeply-supervised nets, in: Artificial Intelligence and Statistics, PMLR, 2015, pp. 562–570.
[22] Dawei Sun, Anbang Yao, Aojun Zhou, Hao Zhao, Deeply-supervised knowledge synergy, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 6997–7006.
[23] Linfeng Zhang, Xin Chen, Junbo Zhang, Runpei Dong, Kaisheng Ma, Contrastive deep supervision, in: Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVI, Springer, 2022, pp. 1–19.
[24] Ting Chen, Simon Kornblith, Mohammad Norouzi, Geoffrey Hinton, A simple framework for contrastive learning of visual representations, in: International Conference on Machine Learning, PMLR, 2020, pp. 1597–1607.
[25] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, Ross Girshick, Momentum contrast for unsupervised visual representation learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 9729–9738.
[26] Qingsong Wen, Liang Sun, Fan Yang, Xiaomin Song, Jingkun Gao, Xue Wang, Huan Xu, Time series data augmentation for deep learning: A survey, 2020, arXiv preprint arXiv:2002.12478.
[27] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, Dahua Lin, Unsupervised feature learning via non-parametric instance discrimination, in: CVPR, Computer Vision Foundation / IEEE Computer Society, 2018, pp. 3733–3742.
[28] Mang Ye, Xu Zhang, Pong C. Yuen, Shih-Fu Chang, Unsupervised embedding learning via invariant and spreading instance feature, in: CVPR, Computer Vision Foundation / IEEE, 2019, pp. 6210–6219.

[29] Aäron van den Oord, Yazhe Li, Oriol Vinyals, Representation learning with contrastive predictive coding, CoRR (2018) arXiv:1807.03748.
[30] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, Jorge L Reyes-Ortiz, Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine, in: Ambient Assisted Living and Home Care: 4th International Workshop, IWAAL 2012, Vitoria-Gasteiz, Spain, December 3-5, 2012. Proceedings 4, Springer, 2012, pp. 216–223.
[31] Jennifer R. Kwapisz, Gary M. Weiss, Samuel A. Moore, Activity recognition using cell phone accelerometers, ACM SigKDD Explor. Newslett. 12 (2) (2011) 74–82.
[32] Attila Reiss, Didier Stricker, Introducing a new benchmarked dataset for activity monitoring, in: 2012 16th International Symposium on Wearable Computers, IEEE, 2012, pp. 108–109.
[33] Daniela Micucci, Marco Mobilio, Paolo Napoletano, Unimib shar: A dataset for human activity recognition using acceleration data from smartphones, Appl. Sci. 7 (10) (2017) 1101.
[34] Chi Li, M Zeeshan Zia, Quoc-Huy Tran, Xiang Yu, Gregory D Hager, Manmohan Chandraker, Deep supervision with shape concepts for occlusion-aware 3d object parsing, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5465–5474.
[35] Jiwon Kim, Jung Kwon Lee, Kyoung Mu Lee, Deeply-recursive convolutional network for image super-resolution, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1637–1645.
[36] Yishuo Zhang, Albert C.S. Chung, Deep supervision with additional labels for retinal vessel segmentation task, in: Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part II 11, Springer, 2018, pp. 83–91.
[37] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al., Bootstrap your own latent-a new approach to self-supervised learning, Adv. Neural Inf. Process. Syst. 33 (2020) 21271–21284.
[38] Xinlei Chen, Kaiming He, Exploring simple siamese representation learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 15750–15758.
[39] Jinqiang Wang, Tao Zhu, Jingyuan Gan, Liming Chen, Huansheng Ning, Yaping Wan, Sensor data augmentation by resampling in contrastive learning for human activity recognition, IEEE Sens. J. (2022).
[40] Jianbo Yang, Minh Nhut Nguyen, Phyo Phyo San, Xiaoli Li, Shonali Krishnaswamy, Deep convolutional neural networks on multichannel time series for human activity recognition, in: IJCAI, AAAI Press, 2015, pp. 3995–4001.
[41] Ming Zeng, Haoxiang Gao, Tong Yu, Ole J Mengshoel, Helge Langseth, Ian Lane, Xiaobing Liu, Understanding and improving recurrent networks for human activity recognition by continuous attention, in: Proceedings of the 2018 ACM International Symposium on Wearable Computers, 2018, pp. 56–63.
[42] Mohammed AA Al-qaness, Abdelghani Dahou, Mohamed Abd Elaziz, AM Helmi, Multi-ResAtt: Multilevel residual network with attention for human activity recognition using wearable sensors, IEEE Trans. Ind. Inform. (2022).
[43] Gulshan Sharma, Abhinav Dhall, Ramanathan Subramanian, A transformer based approach for activity detection, in: Proceedings of the 30th ACM International Conference on Multimedia, 2022, pp. 7155–7159.
[44] Harish Haresamudram, Irfan Essa, Thomas Plötz, Contrastive predictive coding for human activity recognition, Proc. ACM Interact., Mob., Wearable Ubiquitous Technol. 5 (2) (2021) 1–26.
[45] Yoshua Bengio, Aaron C. Courville, Pascal Vincent, Representation learning: A review and new perspectives, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1798–1828.
[46] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, Dilip Krishnan, Supervised contrastive learning, Adv. Neural Inf. Process. Syst. 33 (2020) 18661–18673.
[47] Bulat Khaertdinov, Esam Ghaleb, Stylianos Asteriadis, Deep triplet networks with attention for sensor-based human activity recognition, in: 2021 IEEE International Conference on Pervasive Computing and Communications (PerCom), IEEE, 2021, pp. 1–10.
[48] Frédéric Li, Kimiaki Shirahama, Muhammad Adeel Nisar, Lukas Köping, Marcin Grzegorzek, Comparison of feature learning methods for human activity recognition using wearable sensors, Sensors 18 (2) (2018) 679.
[49] Daniele Ravi, Charence Wong, Benny Lo, Guang-Zhong Yang, Deep learning for human activity recognition: A resource efficient implementation on low-power devices, in: 2016 IEEE 13th International Conference on Wearable and Implantable Body Sensor Networks, BSN, IEEE, 2016, pp. 71–76.
[50] Marina Sokolova, Guy Lapalme, A systematic analysis of performance measures for classification tasks, Inf. Process. Manage. 45 (4) (2009) 427–437.
[51] Yilin Dong, Xinde Li, Jean Dezert, Rigui Zhou, Changming Zhu, Lai Wei, Shuzhi Sam Ge, Evidential reasoning with hesitant fuzzy belief structures for human activity recognition, IEEE Trans. Fuzzy Syst. 29 (12) (2021) 3607–3619.
[52] H. Cho, S.M. Yoon, Applying singular value decomposition on accelerometer data for 1d convolutional neural network based fall detection, Electron. Lett. 55 (6) (2019) 320–322.
[53] Jun Long, Wuqing Sun, Zhan Yang, Osolo Ian Raymond, Asymmetric residual neural network for accurate human activity recognition, Information 10 (6) (2019) 203.
[54] Mohammad Abu Alsheikh, Ahmed Selim, Dusit Niyato, Linda Doyle, Shaowei Lin, Hwee-Pink Tan, Deep activity recognition models with triaxial accelerometers, in: AAAI Workshop: Artificial Intelligence Applied To Assistive Technologies and Smart Environments, in: AAAI Technical Report, WS-16-01, AAAI Press, 2016.
[55] Feng Wang, Huaping Liu, Understanding the behaviour of contrastive loss, in: CVPR, Computer Vision Foundation / IEEE, 2021, pp. 2495–2504.
[56] Laurens Van der Maaten, Geoffrey Hinton, Visualizing data using t-sne., J. Mach. Learn. Res. 9 (11) (2008).
[57] Vishvak S. Murahari, Thomas Plötz, On attention models for human activity recognition, in: UbiComp, ACM, 2018, pp. 100–103.
[58] Harish Haresamudram, David V. Anderson, Thomas Plötz, On the role of features in human activity recognition, in: UbiComp, ACM, 2019, pp. 78–88.
[59] Alireza Abedin, Mahsa Ehsanpour, Qinfeng Shi, Hamid Rezatofighi, Damith C. Ranasinghe, Attend and discriminate: Beyond the state-of-the-art for human activity recognition using wearable sensors, Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 5 (1) (2021) 1:1–1:22.