TU/e EINDHOVEN UNIVERSITY OF TECHNOLOGY

Federated Self-Supervised Learning of Multi-Sensor Representations for Embedded Intelligence

Citation for published version (APA):
Saeed, A., Salim, F. D., Ozcelebi, T., & Lukkien, J. (2021). Federated Self-Supervised Learning of Multi-Sensor Representations for Embedded Intelligence. *IEEE Internet of Things Journal*, *8*(2), 1030-1040. Article 9141293. https://doi.org/10.1109/JIOT.2020.3009358

DOI:
10.1109/JIOT.2020.3009358

Document status and date:
Published: 15/01/2021

Document Version:
Accepted manuscript including changes made at the peer-review stage

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

Download date: 05. Oct. 2023

Federated Self-Supervised Learning of Multi-Sensor Representations for Embedded Intelligence

Aaqib Saeed, Flora D. Salim *Member, IEEE*, Tanir Ozcelebi *Member, IEEE*, and Johan Lukkien *Senior Member, IEEE*

arXiv:2007.13018v1 [cs.LG] 25 Jul 2020

***Abstract*—Smartphones, wearables, and Internet of Things (IoT) devices produce a wealth of data that cannot be accumulated in a centralized repository for learning supervised models due to privacy, bandwidth limitations, and the prohibitive cost of annotations. Federated learning provides a compelling framework for learning models from decentralized data, but conventionally, it assumes the availability of labeled samples, whereas on-device data are generally either unlabeled or cannot be annotated readily through user interaction. To address these issues, we propose a self-supervised approach termed *scalogram-signal correspondence learning* based on wavelet transform to learn useful representations from unlabeled sensor inputs, such as electroencephalography, blood volume pulse, accelerometer, and WiFi channel state information. Our auxiliary task requires a deep temporal neural network to determine if a given pair of a signal and its complementary viewpoint (i.e., a scalogram generated with a wavelet transform) align with each other or not through optimizing a contrastive objective. We extensively assess the quality of learned features with our multi-view strategy on diverse public datasets, achieving strong performance in all domains. We demonstrate the effectiveness of representations learned from an unlabeled input collection on downstream tasks with training a linear classifier over pretrained network, usefulness in low-data regime, transfer learning, and cross-validation. Our methodology achieves competitive performance with fully-supervised networks, and it outperforms pre-training with autoencoders in both central and federated contexts. Notably, it improves the generalization in a semi-supervised setting as it reduces the volume of labeled data required through leveraging self-supervised learning.**

***Index Terms*—self-supervised learning, deep learning, federated learning, embedded intelligence, low-data regime, sensor analytics, learning representations.**

I. INTRODUCTION

LEARNING representations with deep neural networks have made tremendous improvements in the last few years on challenging real-world tasks [1]–[4], thanks to the emergence of massive datasets. In particular, the wealth of sensory data from the Internet of Things (IoT) devices are only recently being leveraged for tackling important problems in understanding context, user monitoring, health, and other predictive analytics tasks, e.g., for emotional well-being [5], [6], sleep tracking [7], and physical activity detection [8]. The success is mainly attributed to the supervised methods that utilize labeled datasets for training models in a central environment. In contrast, learning models from unlabeled decentralized data still presents a major challenge. Obtaining large, well-curated sensory data from edge devices is especially difficult owing to issues like user privacy, the prohibitive cost of labeling, bandwidth limitations, network connectivity, and the diversity of device types [9]. These factors make it significantly challenging to harness abundant data on remote devices for learning semantic features with standard supervised approaches.

To highlight the challenges associated with learning a generalizable model for a particular use case, consider this illustrative example. Let us assume that we aim to develop a robust sleep stage classification model that can be used for a larger population of users. The standard methodology is to learn a supervised model and requires example-label pairs for providing supervision so that a model can differentiate among instances of multiple classes through learning underlying patterns in the input. The procedure begins with the data collection to monitor hundreds of users for electroencephalography (EEG) or other signals as they progress through various stages of sleep and accumulate the multi-sensor data in a centralized (data center) repository for further analysis. The next step is then to get the aggregated inputs annotated by the sleep expert (i.e., generally a professional trained in analyzing physiological signals) for the sleep classes, such as wake, N1, N2, N3, and rapid eye movement. Then, the learning and evaluation phase involves several iterations of improving the model performance. Lastly, the model is deployed in the wild for user monitoring. The process of model development, from data collection to annotation, could be extremely costly owing to the difficulty in setting up an experimental (data collection) protocol. Furthermore, the domain expertise required for the labeling could be severely limited. This problem is exacerbated by the need of supervised deep neural network models for a massive amount of labeled data to learn discriminative features. It becomes painstakingly difficult to inspect and annotate hundreds of thousands of hours of multi-sensor data. Therefore, in practice, limited-sized sensor data are collected and labeled for learning the model, which could further affect its generalization. The important point to note here is that the explained strategy is only applicable when the users

Aaqib Saeed, Tanir Ozcelebi and Johan Lukkien are with the Department of Mathematics and Computer Science, Eindhoven University of Technology, The Netherlands. E-mail: {a.saeed, t.ozcelebi, j.j.lukkien}@tue.nl. Correspondence should be addressed to Aaqib Saeed.

Flora D. Salim is with the School of Science, RMIT University, Melbourne Australia. She co-directs the RMIT Centre for Information Discovery and Data Analytics (CIDDA). E-mail: flora.salim@rmit.edu.au.

This work is funded by SCOTT (www.scott-project.eu) project. It has received funding from the Electronic Component Systems for European Leadership Joint Undertaking under grant agreement No 737422. This Joint Undertaking receives support from the European Union's Horizon 2020 research and innovation programme and Austria, Spain, Finland, Ireland, Sweden, Germany, Poland, Portugal, Netherlands, Belgium, Norway.

agree on sharing their data for learning, which is not ideal in several real-world contexts due to raising privacy issues and misaligned incentives for the user. Likewise, IoT devices produce an astonishing amount of data on a daily basis, and even if the data sharing takes place, its rapidly increasing size limits exploiting for learning models. Therefore, there is a need to develop unsupervised (or self-supervised) methods that can be used to learn general-purpose models from unlabeled data. It is particularly pertinent to on-device learning (such as a smartphone), without the need for data aggregation in a centralized server, and minimal to no human involvement in terms of the annotation process. Consequently, the unsupervised model can be used as a semantic feature extractor or initialization for efficiently adapting to an end-task of interest through fine-tuning with few-labeled instances.

Specifically, the aforementioned challenges motivate the following research questions: Can we train a deep network to extract useful sensory representations in an unsupervised manner without utilizing strong labels for a specific problem, such as activity recognition? Could it also be achieved without aggregating the local data samples from remote devices in a centralized repository, i.e., employing decentralized or on-device learning? Can we improve the network generalization in a low-data regime through fine-tuning it with few-labeled examples that potentially could be easily pooled from a group of users?

Previous approaches to learning representations from time-series of sensory modalities with deep networks can be mainly categorized into three areas: end-to-end training of supervised models with labeled data [4], [6], [7], [10], reconstruction of the actual input for pre-training [11]–[13], and utilizing self-learning with domain-specific transformations or cross-modal learning [14]. Primarily, the focus of these methods is to conduct training of predictive models on a central server in a data center. However, as mentioned earlier, the aggregation of continuously increasing data from distributed devices is practically infeasible, aside from privacy issues. Initially, geo-distributed analytics [15], [16] and distributed learning [17]–[19] in a data center environment is studied to exploit data locality and reducing network costs through pushing code to the data on the edge which generally is a node in the data center which could be across the globe. Nevertheless, these methods do not address the fundamental problem of learning representations with deep networks from unlabeled and highly distributed data that resides on user devices, which can not be aggregated in a central environment for learning.

To address the aforementioned concerns, federated learning [20] is emerging as an effective way of collaboratively training shared models from distributed private data. However, existing exploration in this area is solely focused on learning supervised models for tasks where annotations can be easily acquired based on the user interaction, e.g., mobile keyword prediction [21]. The curation of strongly labeled data becomes infeasible as annotations can not be acquired easily for solving several important problems involving sensory inputs. Because apart from wearables, other sensors could be installed in remote locations, and expert-level domain knowledge could be required to annotate samples. Hence, in such cases, unsupervised approaches provide a compelling substitute to learn from unlabeled data available in huge quantities as they do not require semantic labels.

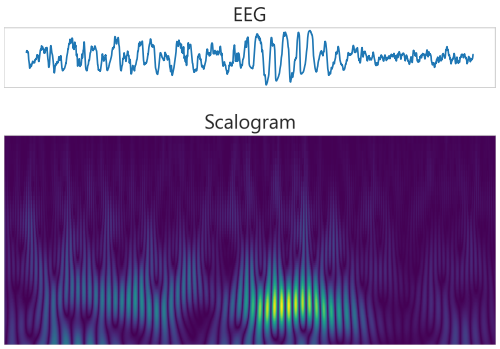

Fig. 1: Illustration of a 30-seconds long electroencephalogram (EEG) signal and a corresponding scalogram extracted with a Morlet wavelet transform.

One of the most rudimentary forms of unsupervised feature discovery has been hand-crafted feature engineering, which turns out to be largely redundant due to its limited discriminative power for building high-performance models [22]. Another area of research that is considerably explored focuses on reconstruction based approaches for extracting low-dimensional embedding through learning from data with deep autoencoders [11]. The main drawback of these methods is that they may waste the network's capacity to model low-level input details through predicting every bit of the signal. This is not needed if the aim is to learn discriminatory features that generalize well to the downstream (or end) tasks, e.g., sleep stage classification with electrical brain activity signals.

A promising substitute is the emerging area of self-supervised learning [23], which enables the learning of representations through solving an auxiliary task for which labels can be acquired from the data without any human intervention. In this case, several techniques are proposed mainly for audio, visual and textual data including estimation of missing input [24], prediction of contextually relevant information [25], recognizing degree of rotation applied on an image [26], contrastive predictive coding [1], synchronization of audio-visual inputs [2], and robotic imitation using multi-view videos [27]. Moreover, cross-modal learning is also utilized by specifying an appropriate loss term between different input modalities for training multimodal networks. However, to the best of our knowledge, previous work did not study self-supervised learning for other sensing modalities (e.g., electroencephalography, accelerometer, blood volume pulse, and others) as produced by a variety of IoT devices.

In this work, we hypothesize that the fusion of self-supervision with federated learning could result in an effective method for learning from unlabeled, private, and diverse types of sensory data, which is crucial for several embedded (personalized) machine learning tasks. To achieve this objective, we develop a novel auxiliary task based on a wavelet transform, which we call *scalogram-signal correspondence learning* (SSCL). A deep temporal convolution network is trained to solve the specified task so as to learn representations

from a variety of sensory inputs (e.g., electroencephalography, inertial measurement unit's sensors (IMUs), and WiFi channel state information). We name it a *scalogram contrastive network* (SCN). Specifically, the self-supervised scheme is designed to contrast between a raw signal (time-series) and its complementary view, which, in our case, is a scalogram, extracted with continuous wavelet transform [28]. We note that other views, such as a spectrogram derived with a fast Fourier transform can also be used for this purpose (or in combination). In this work, we opt for wavelet transformation because it is better at localizing time-frequency properties [29] of the signal.

The core idea behind our pretext task is to determine if a given pair of scalogram-signal inputs are aligned or misaligned, i.e., whether a scalogram is the transformation of a given signal. The presented auxiliary task can formally be seen as a binary classification problem, and we employ a contrastive objective inspired by [30] for optimizing it (see Figure 2 for an overview) in both central and federated settings without involving a human in the data labeling process. Importantly, we would like to highlight that for the model to solve the defined task successfully, it should learn the core semantics in shared input views through possibly relating frequency, scale, and other information present in the signal. The network captures meaningful latent relationships through correlating scalogram-signal inputs in the embedding space. Mainly, the representations that could emerge from the learning process are forms of invariances (such as sensor noise, subject-specific variations), which are essential in several tasks involving sensory data, e.g., stress detection with physiological signals.

The key contributions of this work are three-fold: First, we propose a scalogram-signal correspondence learning framework for self-supervised learning from diverse sensory data. Second, to the best of our knowledge, we, for the first time, propose to unify federated learning with self-supervision to learn from unlabeled and private data on edge devices. Third, we extensively assess the proposed method on several publicly available datasets from different domains with linear classification protocol in central and federated contexts, low-data regime (i.e., semi-supervised setting), and transfer learning including cross-validation. The SCN achieves competitive performance compared with fully-supervised networks that are trained entirely on labeled data and perform significantly better than other approaches. Particularly, SCN fine-tuning with few-labeled instances, e.g., five or ten instances per class, improves the F-score by as much as 5%-6% in comparison to training from scratch. Our approach also works better than transferring supervised features, learned from the source data, between the related tasks.

II. BACKGROUND AND RELATED WORK

We consider learning sensory features from raw unlabeled data with a deep neural network $\mathcal{F}_\theta$ (parameterized by θ), which transforms input from $\mathcal{X}$ into output in $\mathcal{Z}$. Here, we refer to a vector obtained through applying a mapping function $\mathcal{F} : \mathcal{X} \mapsto \mathcal{Z}$ from an arbitrary intermediate or penultimate layer of the network as 'representation' or 'feature.' Our objective is to learn general-purpose representations that can make subsequent tasks of interests easier to solve. To this end, numerous unsupervised methods are developed to leverage a large amount of unlabeled data for achieving better generalization. Moreover, the data required for model development could not only be unannotated but also distributed, without the option to accumulate it in a centralized repository due to privacy concerns and its ever-increasing size. To tackle the issue of learning models from decentralized user data, the field of federated learning [20] is rapidly gaining momentum. Our work is intended to unify self-supervision with federated learning to realize the vision of on-device learning, with a focus on multi-sensor inputs. We describe the details of the essential building blocks of our approach and related work in the following subsections.

A. Self-supervised Learning

The field of unsupervised learning deals with extracting disentangled representations that could be used for solving a wide variety of end-tasks. The most prominent approaches include principal component analysis, Boltzmann machine [31], autoencoders [11], generative adversarial networks [32], and autoregressive models [33]. Another emerging area of research for extracting unsupervised representations is 'self-supervision.' It provides a general and powerful framework for learning with unlabeled inputs through solving pretext tasks. Here, a surrogate objective is specified in such a way that optimizing it would force the network to learn meaningful and usable features for the end-task. Specifically, given an unlabeled dataset $\mathcal{D} = \{x_1, x_2, \ldots, x_\mathcal{M}\}$ with $\mathcal{M}$ instances. A surrogate task is designed that provides pseudo-labels $\{y_1, y_2, \ldots, y_\mathcal{M}\}$ to learn $\mathcal{F}_\theta$ (without the need of any strong class annotations) through minimizing a classification, regression or metric loss $\mathcal{L}$ given by:

$$\min_\theta \frac{1}{\mathcal{M}} \sum_{m=1}^{\mathcal{M}} \mathcal{L}(\mathcal{F}_\theta(x_m), y_m) \tag{1}$$

In the past few years, several self-supervised methods have been developed for vision, audio, language modeling, and other domains. However, little to no attention is paid towards exploring other sensing modalities, such as electroencephalography, IMUs, and blood volume pulse. The prominent approaches for learning from traditional input modalities include, colorization of grayscale images [34], predicting relative location of an image patch [25], audio-visual synchronization [2], temporal alignment in videos through cycle-consistency [35], word2vec (and other variants) [3], signal transformation prediction [8], contrastive predictive coding [1], and robotic imitation learning via time-contrastive networks [27]. These are some of the many strategies proposed for learning from an unlimited amount of unlabeled audio, visual, and textual data.

In this work, we seek to learn representations from data produced by sensors (time-series) on edge as obtaining a large amount of such labeled data is time-consuming and extremely costly. To solve this problem, we utilize a contrastive objective between a raw and complementary view of the data acquired

via wavelet transform. A detailed explanation of the approach is provided in Section III.

B. Wavelet Transform

While the Fourier Transform (FT) sheds light on the frequency properties of the transformed signal, the input signal's time properties are not directly accessible from the Fourier representation. An alternative to this, which provides information about the time properties of the input signal (time locality of signal variations), is the Wavelet Transform (WT) [29]. Like the Short-term Fourier Transform (STFT), the WT divides the input signal into time windows of a certain size and operates on each time window separately. Choosing a larger time window of WT gives better frequency resolution of the WT output signal, while this reduces the time resolution. Precisely, the Wavelet series gives individual coefficients of a set of orthonormal functions (wavelets, e.g., Morlet, Haar, Daubechies). Like its counterparts, this representation effectively decomposes the input signal into combinations of wavelets. Due to these compelling properties, WT has been widely used in a myriad of domains [28]. In particular, continuous WT gained significant popularity compared to discrete counterpart since it is better at localizing time-frequency properties. A wavelet transform of a signal $x(t)$ is defined as follows:

$$T(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} x(t) \cdot \psi \left(\frac{t-b}{a} \right) dt \qquad (2)$$

where ψ represents a wavelet function, a and b denote scaling and translation factors, respectively. It is important to note that although we utilize WT in this work, other approaches like STFT could also be used in conjunction to possibly improve the performance along with segmentation [36].

C. Federated Learning

Autonomous vehicles, wearables, smartphones, and IoT sensors are examples of modern distributed devices producing a wealth of data every second. This massive amount of data offers an excellent opportunity for learning models to solve a diverse range of tasks. The applications of interest include customized fitness plans, personalized language models, and contextual awareness for driving automation. The growing computational power of edge devices allows us to leave the data decentralized and push the network computation to the client, which is also ideal from a privacy aspect. The expanding area of federated learning [20], [37], [38] explores developing methods to achieve the goal of learning from highly distributed and heterogeneous data through aggregating locally trained models on remote devices, such as smartphones and wearables. In this case, the intention is to minimize the following objective [37]:

$$\min_{\theta} \mathcal{F}_{\theta}, \text{ where } \mathcal{F}_{\theta} := \sum_{c}^{\mathcal{C}} \frac{m_c}{m} \mathcal{F}_{\theta_c}. \qquad (3)$$

Here, $\mathcal{C}$ represents the number of participating client devices in a training round, m_c is the total number of instances available for client c with $m = \sum_c m_c$, and lastly θ_c denotes the weights of a local model. To produce a global model, `Federated Averaging` algorithm [20] is typically used to accumulate client updates after every round of local training t as with Equation 3.

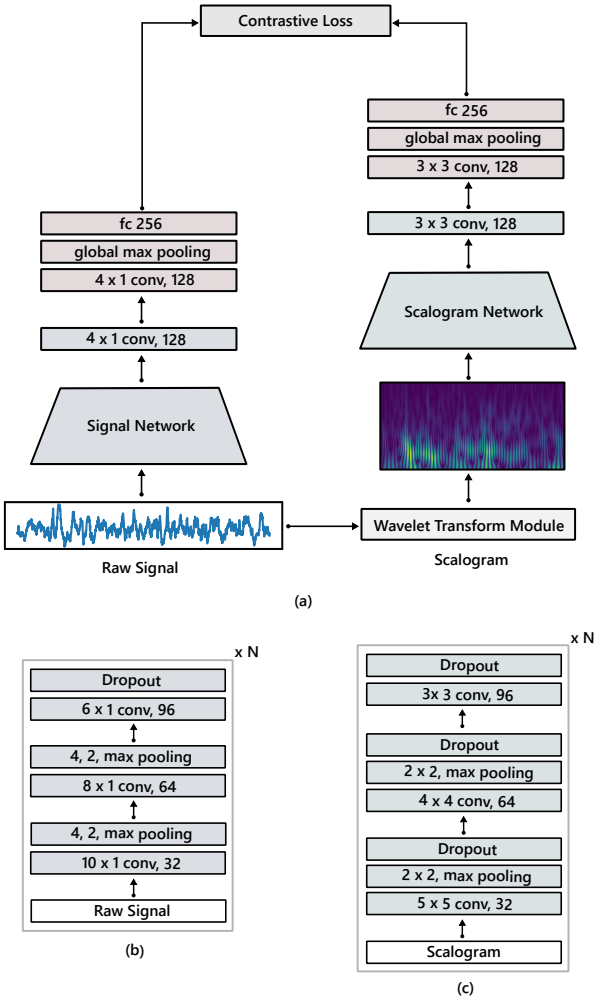

Fig. 2: Scalogram contrastive network. We design a dual-stream architecture to learn from the raw input signal and its complementary view i.e. a scalogram. We map the original signal fragments into another domain and train the network to recognize which pairs belong together. Within this work, we use a wavelet transform. The high-level overview of the method is illustrated in (a) where signal and scalogram networks are also multi-stream networks with a distinct stream for each input modality. The architecture of these modality-specific signal and scalogram networks is shown in (b) and (c), respectively.

The research interest in federated learning revolves around improving communication efficiency [39], personalization [40], fault tolerance [41], privacy preservation [42] as well as looking into the theoretical underpinning of the federated optimization [37]. Specifically, the recent work deals with learning a unified model to solve a single as well as multiple tasks [43]. In addition to improving communication costs, the computational efficiency of federated learning on resource-constrained devices has also been studied [44]. Similarly, the development of frameworks and productionizing of applications built around the idea of decentralized learning are also

surging to address various practical problems, such as next-word and emoji prediction [21], wake word recognition [45], query suggestion [46], and traffic flow forecasting [47]. Deep reinforcement learning has also been investigated in a federated setting for edge caching in IoT to improve the quality of services and dealing with traffic off-loading [48].

Nevertheless, the existing techniques make a strong assumption that labeled training data are always accessible, or annotations can be extracted reliably, e.g., via user interaction with smartphone applications. However, for various problems involving sensory data (such as sleep stage scoring and context recognition), obtaining a large number of annotated examples in a real-world setting to train supervised models is prohibitively expensive and not feasible. This limits the applicability of current methods in learning from unlabeled data available from distributed IoT devices. The approach presented here is a step towards exploring self-supervised representation learning in a federated setting from unannotated multi-sensor data at the edge.

III. APPROACH

Learning multi-sensor representations with deep networks requires a large amount of well-curated data, which is made difficult by the diversity of device types, environmental factors, inter-personal differences, privacy issues, and annotation cost. We propose a self-supervised auxiliary task whose objective at a high level is to contrast or compare raw signals and their corresponding scalograms (which are a visual representation of the wavelet transform) so that a network learns to discriminate between aligned and unaligned scalogram-signal pairs. The rationale of the proposed approach is similar in spirit to cross-view learning in the audio-visual domain [2]. However, it differs in a core way that we obtain aligned and unaligned views[1] from the same modality with wavelet transform. In the absence of the semantic labels, our methodology can be leveraged to generate an endless stream of labeled data. Therefore, it can train the network without any human involvement, which is particularly attractive for on-device learning. In subsequent sections, we describe details of the correspondence learning, sample generation, preference of a loss function, and key network architectural properties.

A. Scalogram-signal Correspondence Learning

The idea behind SSCL is to learn network parameters with a self-supervised objective that determines whether a raw signal and a scalogram correspond (or align) with each other or not. Given a multi-sensor dataset with fixed-length input segments of multiple modalities $\mathcal{D} = \{x_1, x_2, \ldots, x_{\mathcal{M}}\}$ of $\mathcal{M}$ instances, we train a multimodal contrastive network to achieve the objective of synchronizing representations of the raw input with their corresponding scalogram. Specifically, a time-series is segmented into a fixed-sized input with a sliding window having a certain overlap between samples. Afterward, the scalogram s_m of a signal x_m can be generated with a specified wavelet transformation Ψ [29]. This procedure results in synchronized pairs for each x_m and s_m of m-th instance. These co-occurring pairs of inputs are assigned a class label $y_m = 1$, i.e., representing in-sync examples. Likewise, for generating negative samples $y_m = 0$, for a particular x_m, a randomly selected s_m is assigned, which in principle represents that these scalogram-signal pairs do not align with each other. Here, we sample a negative scalogram from the same input modality. However, it can also be selected from a different modality, e.g., for accelerometer, the scalogram of the gyroscope can also be utilized. Importantly, we utilize an equal number of positive and negative instances for training the network. As described earlier, a wavelet transform provides a better multi-resolution analysis of non-stationary signals than Short-Time Fourier Transform (STFT) [28]. Hence, we extract a scalogram, which is an absolute and squared value of a WT operation. It is achieved using a continuous Morlet WT function which is expressed as follows:

$$\psi(t) = \exp^{\frac{-t^2}{2}} \cdot \exp^{-jw_0 t} \tag{4}$$

where w_0 denotes a central frequency of the mother wavelet.

In the broadest sense, the SSCL task requires a semantic understanding of how time-frequency information presented in a scalogram relates to a raw input signal, thus enabling the model to learn general-purpose embedding with a complementary view on the original input. We give a high-level overview of our approach in Figure 2. The aim here is to learn a classifier $\mathcal{H}(.)$ that can minimize an empirical loss, so $\mathcal{H}(x_m, s_m) = y_m$. A natural choice is to cast the specified problem as a binary classification task $p(y|x, s)$ and hence, optimize a cross-entropy loss. Nevertheless, we achieve slightly better convergence through employing a contrastive loss that pulls together embedding of positive pairs and pushes different pairs apart, as it is also shown to be improving generalization in earlier work [30]:

$$\begin{aligned}\mathcal{L} = \frac{1}{\mathcal{M}} \sum_{m=1}^{\mathcal{M}} (y_m)||\mathcal{F}_{\mathcal{X}}(x_m) - \mathcal{F}_{\mathcal{S}}(s_m)||_2^2 \; + \\ (1 - y_m)\max(\alpha - ||\mathcal{F}_{\mathcal{X}}(x_m) - \mathcal{F}_{\mathcal{S}}(s_m)||_2, 0)^2\end{aligned} \tag{5}$$

where α is a margin hyperparameter, which is enforced between positive and negative samples, $\mathcal{F}_{\mathcal{X}}$, and $\mathcal{F}_{\mathcal{S}}$ are signal and scalogram networks, respectively. The contrastive loss optimization solves the proposed self-supervised task through the integration of not just different views of the same underlying signal, but it also aligns samples across multiple sensory modalities. This label-free correspondence learning approach results in rich representations that may be invariant to sensor noise, amplitude (or scale) variations, user-specific differences, and other factors.

B. Network Architecture

To tackle the SSCL task, we design a dual-stream architecture named *scalogram contrastive network*, which is inspired from [2] and it is illustrated in Figure 2. It is composed of two distinct parts: the scalogram network and the signal network, each extracting features from their respective inputs. As the aim here is to learn representations from multiple sensors, each

[1]or in-sync and out-of-sync samples

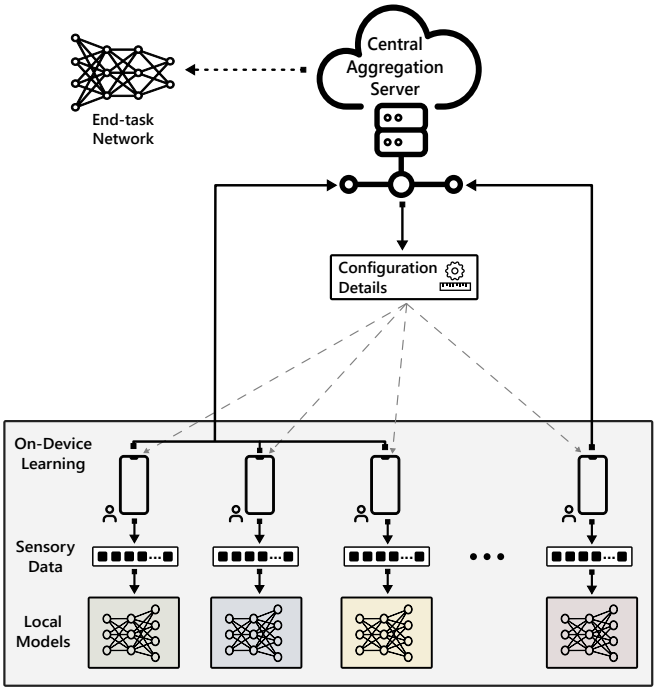

Fig. 3: Overview of federated learning framework. A central server dispatches a randomly initialized model and other training configuration details to the selected clients' devices, as depicted by dashed gray lines. The clients train local models on their private data and send the models back to the server illustrated with solid black lines. The models are aggregated to produce a unified model that is used for the end-task.

network consists of modality-specific and fusion layers to learn specialized and joint embedding, respectively. In particular, we utilize the same network architecture for learning on different datasets unless mentioned otherwise. Likewise, only the features from the signal network are used for evaluation, discarding the scalogram network after pre-training.

The scalogram network consists of three 2D convolution layers with kernel sizes of 5, 4, 3, and 32, 64, 96 feature maps, respectively. Dropout is applied after every layer and max-pooling after the initial two convolutional layers with a pooling size of 2. We use the same design for each input modality, followed by the fusion layer consisting of 128 feature maps with a kernel size of 3. To learn from raw signals, we use a 1D convolutional network with the same structure as the scalogram network but with crucial differences in kernel sizes which are 10, 8, and 6 for sensor-specific layers and 4 in the case of a shared layer with a dropout layer at the end. Moreover, we use additional pre-training related layers for both networks, comprising a convolutional layer with 128 feature maps and a dense layer with 256 hidden units. These layers are discarded after the self-supervised learning phase as we hypothesize that they might learn features relevant to the auxiliary task (i.e., SSCL). We use the Mish [49] activation function in all the layers except the last, which has either linear or softmax activation. Finally, the input to our scalogram network are coefficients of the wavelet transform with a size $(h \times w \times c)$, each representing height, width, and the number of channels, respectively. The signal network directly processes raw input of size $(w \times c)$.

TABLE I: Summary of datasets.

Task	Dataset	#Users	#Outputs
Sleep Stage Scoring	Sleep-EDF	20	5
Activity Recognition	HHAR	9	6
	MobiAct	61	11
Device-Free Sensing	WiFi-CSI	6	7
Stress Detection	WESAD	15	3

C. Implementation Details

For pre-training, we sample the non-corresponding scalogram-signal examples through randomly selecting scalograms from outside the current input batch while keeping the raw input fixed for positives and negatives. We preprocess the signals before computing scalogram or initiating network training as done in the previous works for each considered dataset; further details are provided in Section IV-A. We calculate summary statistics for z-normalization from the training set. We use an Adam optimizer with a fixed learning rate of 0.0001 for pre-training and 0.01 or 0.02 in case of learning a linear classifier, which could also be decayed based on performance on the validation set. The network is trained with a batch size of 24, a dropout rate of 0.1, and L2 regularization rate of 0.0001. For federated learning simulation, we use the Tensorflow federated learning framework[2]. In this case, the networks are trained with a batch size of 12 for 5 local epochs using data of n randomly selected users (typically 10) at each training round with $30 - 50$ rounds in total, depending on the dataset size. Specifically, in our experiments, we randomly divide the training set into multiple subsets (representing each client) that are used to train the models in a federated setting. We opt for this strategy due to fewer users in existing datasets. The availability of bigger datasets with a larger pool of users could be useful to evaluate self-supervised methods in the future. A high-level overview of federated learning is illustrated in Figure 3.

IV. EXPERIMENTS

We evaluate the effectiveness of our approach in multiple ways with several publicly available datasets from different domains. First, we probe the quality of representations with a linear classifier trained on-top of a frozen feature extractor in both central and federated learning settings. Second, we examine whether scalogram-signal correspondence learning could be used to improve the recognition rate in the low-data regime. Finally, we determine the transferability of features on related datasets, followed by an evaluation with cross-validation to determine robustness against subject variations.

[2]https://www.tensorflow.org/federated

A. Datasets and Preprocessing

We experimented with learning models on 5 datasets from the following application areas: sleep stage scoring, human activity recognition, WiFi sensing, and physiological stress detection.

The electroencephalogram (EEG) and electrooculography (EOG) signals are used from the PhysioNet Sleep-EDF dataset [50], [51] for classifying sleep into five stages (i.e., Wake, N1, N2, N3, and Rapid Eye Movement). We preprocess these signals, which are recorded at 100Hz, as done in earlier work [7] and utilize 30-second epochs (segments). For activity classification with smartphones, accelerometer, and gyroscope signals from HHAR [52] and MobiAct [53] datasets are used, which have 6 and 11 output classes, respectively. We segment the raw signals through a sliding window into a segment size of 400 samples with a 50% overlap. For device-free sensing of daily activities, we use the WiFi channel state information data [10] and follow identical preprocessing steps with [10]. Notably, the signals are resampled from 1kHz to 500Hz through uniform temporal downsampling with a rate of 2 for each of the 90 channels (i.e., 30 sub-carriers per antenna) to classify them into 7 classes. The WESAD dataset [5] is used for the detection of stress, normal, and amusement physiological states. Here, we use blood volume pulse, electrodermal activity, and temperature signals collected from a wrist wearable device at 64Hz, 4Hz, and 4Hz, respectively. Following [5], we extract 30-seconds segments and independently normalize each subject's data before the model development phase.

In all the cases, we use a random $70\% - 30\%$ split of the dataset (based on users such that there is no overlap in terms of users' data) for training and evaluation, respectively. We also pick a 20% subset from training split as a validation set for hyperparameter tuning and model selection. Moreover, we also evaluate the performance of our approach with cross-validation based on user split, i.e., leave-one-user-out. Table I summarizes the key characteristics of the datasets used in our evaluation.

B. Quality assessment of the learned features with separability analysis

In Table II and Table III, we provide our key evaluation results in central and federated learning settings. First, we compare the performance of our approach with a) a supervised network trained end-to-end, b) an autoencoder, and c) a randomly initialized network in a central setting, i.e. when the entire data are available for learning on a server. We measure the quality of learned representations through a linear classifier trained on-top of the frozen feature extractor, which is a standard evaluation protocol used in earlier work. In the federated setting (Table III, the supervised network is learned for each user, and the weights are aggregated to create a unified model. For an autoencoder and SCN, the pre-training is performed in a federated setting to learn representations, and a classifier is trained in a standard way i.e., as if the data of end-task are available on the server. In addition, we also assess the performance when unsupervised networks are kept

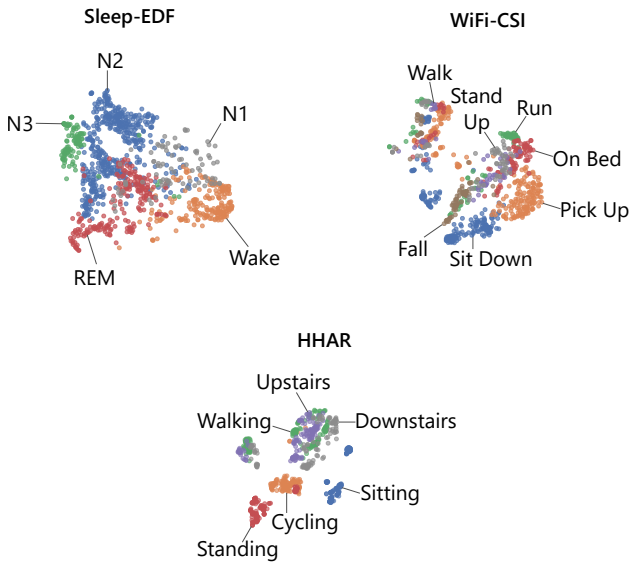

Fig. 4: t-SNE embedding learned with scalogram contrastive network on a random subset of test subjects. Note, t-SNE does not utilize class labels, the colors are added during post-hoc analysis for better interpretability.

frozen, and classifier is also learned in a federated setting. In Table II these entries are represented with FC, which is an abbreviation of a federated classifier.

In particular, we highlight that for federated learning, we utilize random partitioning of the training sets as in [20] to tackle the low number of users in the considered (existing) datasets. This choice might result in a decentralized IID (i.e., independent and identically distributed) dataset that could be unbalanced but does not suffer from extreme heterogeneity in terms of training instances per client as generally, the case is for non-IID data that typically varies heavily based on the users' demographics, device usage, and other factors. However, we would again emphasize that our self-supervised technique does not depend on the user-generated labels for learning representations and could be easily applied to large-scale datasets. However, as the end-task labels are required to evaluate the quality of learned features, the unavailability of massive multi-sensor labeled data is a critical limiting factor towards realizing the goal of assessment in the non-IID setting. We leave the evaluation of self-supervised features on a large pool of users with a greater variety of devices as future work.

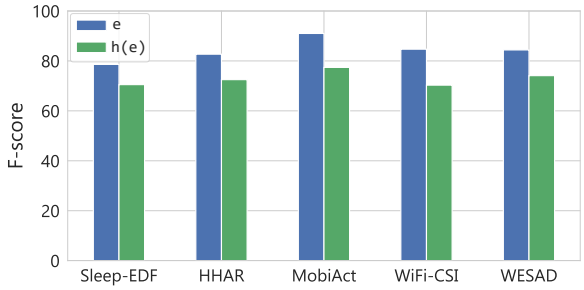

Fig. 5: Performance comparison of linear classifiers trained on-top of representations from encoder (e) and penultimate layer's projection $h(e)$ of SCN denoted with `fc 256` in Figure 2.

TABLE II: Performance evaluation of self-supervised representations learned in a standard central setting with a linear classifier.

	Sleep-EDF		**HHAR**		**MobiAct**		**WiFi-CSI**		**WESAD**	
	F-score	Kappa	F-score	Kappa	F-score	Kappa	F-score	Kappa	F-score	Kappa
Random Init.	0.67	0.54	0.64	0.58	0.65	0.63	0.36	0.24	0.73	0.58
Supervised	0.82	0.76	0.73	0.69	0.95	0.93	0.96	0.95	0.85	0.75
Autoencoder	0.75	0.66	0.69	0.63	0.80	0.78	0.84	0.81	0.83	0.72
SCN	0.78	0.70	0.82	0.79	0.91	0.88	0.84	0.81	0.84	0.73

TABLE III: Assessing performance in a federated learning setting to determine SCN's ability to learn representations from distributed data. The entries marked with FC (federated classifier) denotes metrics when both representations and classifier are learned in a federated context.

	Sleep-EDF		**HHAR**		**MobiAct**		**WiFi-CSI**		**WESAD**	
	F-score	Kappa	F-score	Kappa	F-score	Kappa	F-score	Kappa	F-score	Kappa
Supervised	0.82	0.76	0.77	0.73	0.94	0.92	0.92	0.90	0.85	0.75
Autoencoder	0.76	0.68	0.71	0.66	0.86	0.83	0.85	0.81	0.82	0.70
SCN	0.78	0.70	0.80	0.77	0.90	0.88	0.85	0.82	0.83	0.73
Autoencoder (FC)	0.68	0.56	0.51	0.44	0.54	0.47	0.67	0.60	0.80	0.67
SCN (FC)	0.77	0.69	0.80	0.76	0.82	0.79	0.69	0.63	0.82	0.70

On the evaluated datasets, we observe that the classifiers learned on-top of a fixed randomly initialized network achieve F-score above 60% in most cases. It highlights the representational capacity of our architecture design that, without seeing any samples, the encoder can provide reasonable embedding for a linear classifier. Notably, the SCN surpasses pre-training results with the autoencoder and on HHAR achieves better F-score (82.7) than a supervised baseline (73.0). Particularly, we notice that the results obtained in a federated setting are close to those achieved with learning end-to-end models in a central setting, which hints towards the robustness of our approach in a federated environment. Similarly, when a linear classifier is also trained in a federated setting, the performance of SCN is mainly consistent with the centralized classifier, which is not the case for an autoencoder. Moreover, in Figure 4, we provide the t-SNE embedding of SCN on 1000 randomly selected instances from a test set of Sleep-EDF, WiFi-CSI, and HHAR. The distinct clusters of data points can be seen that are discovered entirely in an unsupervised manner. This further highlights the ability of SCN to learn meaningful representations.

In Figure 5, we compare the performance of downstream task classifiers trained on embedding from two different parts of the network. The representations from the encoder e and the features from the penultimate layer of SCN $h(e)$ are used for this purpose. It can be seen that the classifier trained on the output of e performs significantly better than the one learned using the last layer's features. We think it could be because that layers at the end might learn auxiliary task-specific features that are not useful enough for the end-task.

C. Improving generalization in low-data regime and transfer as evaluation

We explore the effectiveness of the proposed technique for improving performance with few-labeled examples. We pre-train a scalogram-contrastive network with the entire unlabeled data and use the model as initialization for learning a downstream task. We compare the performance with a standard supervised network trained only with certain labeled instances. Specifically, we use 5, 10, 20, and 40 labeled instances per class to learn the end-task model. Figure 6 and Table IV show an average F-score of 100 independent repetitions where different examples are sampled to train the network at each run. In all the cases, the results obtained with utilizing a self-supervised network are better than the baseline, even when limited labeled data are available. This highlights that the SCN efficiently harnesses unlabeled data to learn generalized features.

Similarly, the self-supervised networks are also evaluated in terms of their usefulness in a transfer learning setting. Generally, this is achieved by treating a pre-trained model as a fixed feature extractor, and a linear model is trained on top of it using a different dataset. Here, we assess the performance on activity recognition tasks with HHAR and MobitAct datasets. Table V provides these results and compares with the supervised network, transfer from supervised (Sup.), and SCN trained on the same source instances. In both cases, we see that the recognition improves relatively if the transferred embedding is from SCN compared to a supervised network. Finally, we also assess the performance of SCN when few-labeled instances are available for fine-tuning, but different unlabeled data are available for pre-training, as shown in Table VI. Similar to earlier semi-supervised evaluation, we fine-tune a pre-trained network end-to-end with $5, 10, 20,$ and 40 examples of each class from the target dataset. We notice a $2\% - 3\%$ improvement in F-score over the supervised network when an SCN encoder is utilized.

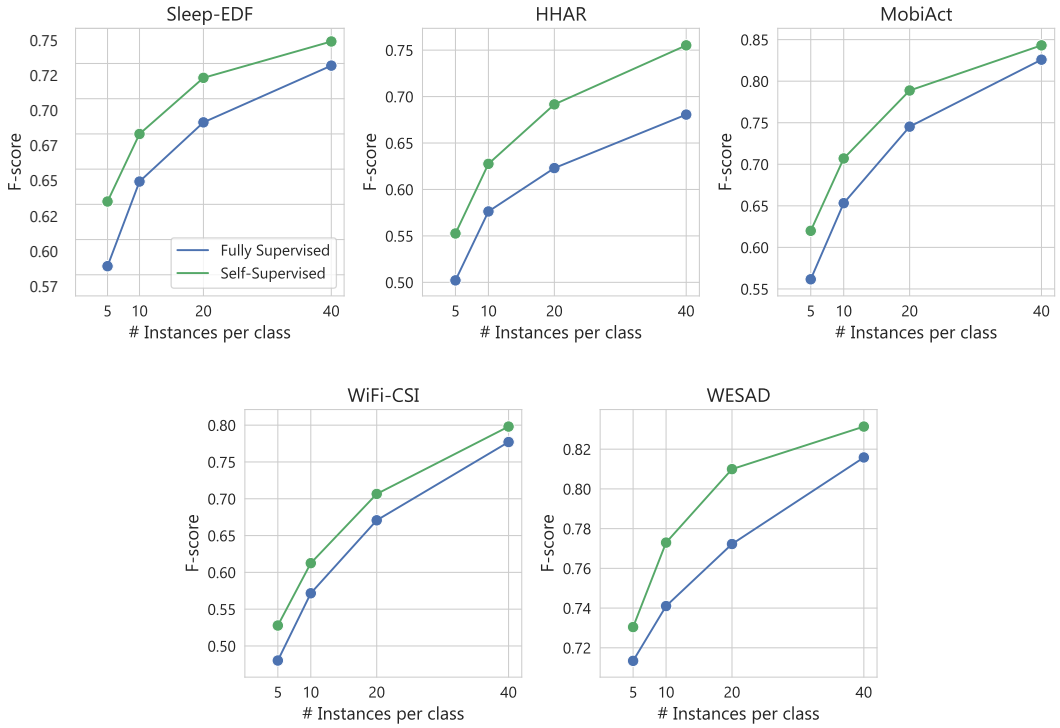

Fig. 6: Effectiveness of self-supervised learning in a low-data regime. The SCN is pre-trained on unlabeled data and fine-tuned end-to-end with few-labeled data points (i.e, $5, 10, 20,$ and 40 instances per class). On all the evaluated datasets, we notice a significant performance improvement over a supervised baseline network, which is trained only with labeled inputs.

TABLE IV: Generalization improvement in semi-supervised setting with self-supervised pre-training.

	Sleep-EDF		**HHAR**		**MobiAct**		**WiFi-CSI**		**WESAD**	
	Supervised	SCN	Supervised	SCN	Supervised	SCN	Supervised	SCN	Supervised	SCN
5	0.58±0.05	0.62±0.05	0.50±0.07	0.55±0.06	0.56±0.06	0.61±0.07	0.48±0.03	0.52±0.03	0.71±0.06	0.73±0.06
10	0.64±0.03	0.67±0.04	0.57±0.06	0.62±0.05	0.65±0.05	0.70±0.05	0.57±0.02	0.61±0.02	0.74±0.03	0.77±0.03
20	0.68±0.05	0.71±0.02	0.62±0.05	0.69±0.04	0.74±0.04	0.78±0.04	0.67±0.02	0.70±0.02	0.77±0.03	0.80±0.03
40	0.72±0.03	0.74±0.02	0.68±0.04	0.75±0.04	0.82±0.02	0.84±0.02	0.77±0.02	0.79±0.02	0.81±0.02	0.83±0.02

TABLE V: Evaluation of self-supervised representation in a standard transfer learning setting.

	HHAR → MobiAct		**MobiAct → HHAR**	
	F-score	Kappa	F-score	Kappa
Supervised	0.95	0.93	0.73	0.69
Source (SCN)	0.91	0.88	0.82	0.79
Transfer (Sup.)	0.86	0.83	0.62	0.54
Transfer (SCN)	0.87	0.84	0.75	0.71

TABLE VI: Fine-tuning transferred model with few-labeled data to improve recognition rate. We report weighted F-score averaged over 100 independent runs. T denotes a transfer learning.

	HHAR → MobiAct		**MobiAct → HHAR**	
	Supervised	SCN (T)	Supervised	SCN (T)
5	0.50	0.51	0.56	0.59
10	0.56	0.59	0.65	0.69
20	0.62	0.65	0.74	0.75
40	0.68	0.70	0.82	0.82

D. Network robustness against subject variation with cross-validation

To determine the robustness of network pre-training with the proposed approach against subject variation, we perform cross-validation (CV) based on user split. For Sleep-EDF, HHAR, and WESAD leave-one-subject-out CV is employed, whereas for MobiAct and WiFi-CSI, a 10-fold stratified CV is used due to a large number of users in the former and unavailability of subject ID's in the latter. We follow the same evaluation strategy as earlier, i.e., training a linear classifier to assess the quality of representations compared to the fully-supervised model and an autoencoder. Table VII summarizes mean and standard deviation of metrics averaged over folds. Overall, we notice that SCN is stable despite the changes

TABLE VII: Comparison of self-supervised representations to a fully-supervised network and pre-training with autoencoder using cross-validation.

	Supervised		**Autoencoder**		**SCN**	
	F-score	Kappa	F-score	Kappa	F-score	Kappa
Sleep-EDF	0.83±0.05	0.77±0.06	0.73±0.08	0.65±0.10	0.82±0.03	0.83±0.03
HHAR	0.82±0.12	0.80±0.13	0.62±0.13	0.59±0.15	0.78±0.11	0.76±0.12
MobiAct	0.94±0.02	0.92±0.03	0.79±0.04	0.75±0.06	0.90±0.02	0.87±0.03
WiFi-CSI	0.97±0.0	0.97±0.0	0.85±0.01	0.82±0.02	0.85±0.01	0.82±0.01
WESAD	0.76±0.11	0.63±0.17	0.71±0.14	0.56±0.25	0.75±0.13	0.63±0.19

of subject data in a training set and achieves significantly better results than an autoencoder. Notably, on Sleep-EDF, our methods achieve a mean kappa score of 0.83 as compared to 0.77 of a supervised network and 0.76 as reported in [7]. Likewise, our self-supervised technique performs better than the hand-designed features from wrist physiological signals on WESAD by achieving an F-score of 75.7 ± 0.13 as compared to 66.33 ± 0.36 [5]. Furthermore, we would like to highlight that a direct comparison of existing approaches on other datasets used in our study is not feasible due to the differences in reported metrics and used sensing modalities. Nevertheless, our results with cross-validation further indicate that self-supervised learning can be effectively utilized for sensor modeling tasks on a large scale and can be combined with active learning methods [54].

V. CONCLUSIONS

In this paper, we propose a self-supervised method for learning representations from unlabeled multi-sensor input data, which is typical in the IoT setting. Our method utilizes wavelet transform to generate a complementary view of the input (i.e., a scalogram) to define an auxiliary task of scalogram-signal correspondence. This procedure is specifically designed to work in federated learning setting to allow training networks with widely distributed and unannotated data as the labels can be readily extracted from the data without human-in-the-loop. We show the efficacy of the developed technique on several publicly available datasets involving diverse sensory streams, such as electroencephalogram, blood volume pulse, and IMUs. Particularly, we evaluate the quality of learned features with a linear classifier on an end-task and compare the performance with a fully-supervised network and pre-training with an autoencoder in both federated and central settings. Furthermore, we demonstrate an improved generalization in the low-data regime with self-supervision, i.e., when few labeled instances are used for fine-tuning network on the desired end-task. Our generic self-supervised approach can be used efficiently to learn general-purpose deep feature extractors entirely on-device without the need to transmit the actual data to the server. In future work, we plan to combine self-supervision with architecture search on larger datasets and evaluate our method in a non-IID setting for federated learning. Another avenue of future research is to explore the effectiveness of self-supervised pre-training for adversarial robustness in a federated setting.

REFERENCES

[1] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[2] A. Owens and A. A. Efros, "Audio-visual scene analysis with self-supervised multisensory features," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 631–648.

[3] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[4] V. Radu, C. Tong, S. Bhattacharya, N. D. Lane, C. Mascolo, M. K. Marina, and F. Kawsar, "Multimodal deep learning for activity and context recognition," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 4, pp. 1–27, 2018.

[5] P. Schmidt, A. Reiss, R. Duerichen, C. Marberger, and K. Van Laerhoven, "Introducing wesad, a multimodal dataset for wearable stress and affect detection," in *Proceedings of the 2018 on International Conference on Multimodal Interaction*. ACM, 2018, pp. 400–408.

[6] B. Ballinger, J. Hsieh, A. Singh, N. Sohoni, J. Wang, G. H. Tison, G. M. Marcus, J. M. Sanchez, C. Maguire, J. E. Olgin *et al.*, "Deepheart: semi-supervised sequence learning for cardiovascular risk prediction," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[7] A. Supratak, H. Dong, C. Wu, and Y. Guo, "Deepsleepnet: A model for automatic sleep stage scoring based on raw single-channel eeg," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 11, pp. 1998–2008, 2017.

[8] A. Saeed, T. Ozcelebi, and J. Lukkien, "Multi-task self-supervised learning for human activity detection," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 3, no. 2, p. 61, 2019.

[9] N. D. Lane, S. Bhattacharya, P. Georgiev, C. Forlivesi, L. Jiao, L. Qendro, and F. Kawsar, "Deepx: A software accelerator for low-power deep learning inference on mobile devices," in *2016 15th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*. IEEE, 2016, pp. 1–12.

[10] S. Yousefi, H. Narui, S. Dayal, S. Ermon, and S. Valaee, "A survey on behavior recognition using wifi channel state information," *IEEE Communications Magazine*, vol. 55, no. 10, pp. 98–104, 2017.

[11] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th international conference on Machine learning*, 2008, pp. 1096–1103.

[12] H. P. Martinez, Y. Bengio, and G. N. Yannakakis, "Learning deep physiological models of affect," *IEEE Computational intelligence magazine*, vol. 8, no. 2, pp. 20–33, 2013.

[13] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: A survey," *Pattern Recognition Letters*, vol. 119, pp. 3–11, 2019.

[14] T. Baltrušaitis, C. Ahuja, and L.-P. Morency, "Multimodal machine learning: A survey and taxonomy," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 2, pp. 423–443, 2018.

[15] A. Vulimiri, C. Curino, B. Godfrey, K. Karanasos, and G. Varghese, "WANalytics: Analytics for a geo-distributed data-intensive world." in *CIDR*, 2015.

[16] J. Dean and S. Ghemawat, "Mapreduce: simplified data processing on large clusters," *Communications of the ACM*, vol. 51, no. 1, pp. 107–113, 2008.

[17] S. Zhang, A. E. Choromanska, and Y. LeCun, "Deep learning with elastic averaging sgd," in *Advances in neural information processing systems*, 2015, pp. 685–693.

[18] M. F. Balcan, A. Blum, S. Fine, and Y. Mansour, "Distributed learning, communication complexity and privacy," in *Conference on Learning Theory*, 2012, pp. 26–1.

[19] O. Shamir and N. Srebro, "Distributed stochastic optimization and learning," in *2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 2014, pp. 850–857.

[20] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2017. [Online]. Available: http://arxiv.org/abs/1602.05629

[21] A. Hard, K. Rao, R. Mathews, S. Ramaswamy, F. Beaufays, S. Augenstein, H. Eichner, C. Kiddon, and D. Ramage, "Federated learning for mobile keyboard prediction," *arXiv preprint arXiv:1811.03604*, 2018.

[22] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[23] V. R. de Sa, "Learning classification with unlabeled data," in *Advances in neural information processing systems*, 1994, pp. 112–119.

[24] R. Zhang, P. Isola, and A. A. Efros, "Split-brain autoencoders: Unsupervised learning by cross-channel prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1058–1067.

[25] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," in *European Conference on Computer Vision*. Springer, 2016, pp. 69–84.

[26] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," *arXiv preprint arXiv:1803.07728*, 2018.

[27] P. Sermanet, C. Lynch, Y. Chebotar, J. Hsu, E. Jang, S. Schaal, S. Levine, and G. Brain, "Time-contrastive networks: Self-supervised learning from video," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1134–1141.

[28] R. Merry, "Wavelet theory and applications: a literature study," *DCT rapporten*, vol. 2005, 2005.

[29] I. Daubechies, "The wavelet transform, time-frequency localization and signal analysis," *IEEE transactions on information theory*, vol. 36, no. 5, pp. 961–1005, 1990.

[30] S. Chopra, R. Hadsell, Y. LeCun *et al.*, "Learning a similarity metric discriminatively, with application to face verification," in *CVPR (1)*, 2005, pp. 539–546.

[31] R. Salakhutdinov and G. Hinton, "Deep boltzmann machines," in *Artificial intelligence and statistics*, 2009, pp. 448–455.

[32] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[33] A. v. d. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[34] G. Larsson, M. Maire, and G. Shakhnarovich, "Colorization as a proxy task for visual understanding," in *CVPR*, vol. 2, 2017, p. 7.

[35] D. Dwibedi, Y. Aytar, J. Tompson, P. Sermanet, and A. Zisserman, "Temporal cycle-consistency learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1801–1810.

[36] A. Sadri, Y. Ren, and F. D. Salim, "Information gain-based metric for recognizing transitions in human activities," *Pervasive and Mobile Computing*, vol. 38, pp. 92–109, 2017.

[37] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *arXiv preprint arXiv:1908.07873*, 2019.

[38] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings *et al.*, "Advances and open problems in federated learning," *arXiv preprint arXiv:1912.04977*, 2019.

[39] J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *arXiv preprint arXiv:1610.05492*, 2016.

[40] K. Wang, R. Mathews, C. Kiddon, H. Eichner, F. Beaufays, and D. Ramage, "Federated evaluation of on-device personalization," *arXiv preprint arXiv:1910.10252*, 2019.

[41] K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konecny, S. Mazzocchi, H. B. McMahan *et al.*, "Towards federated learning at scale: System design," *arXiv preprint arXiv:1902.01046*, 2019.

[42] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth, "Practical secure aggregation for privacy-preserving machine learning," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017, pp. 1175–1191.

[43] V. Smith, C.-K. Chiang, M. Sanjabi, and A. S. Talwalkar, "Federated multi-task learning," in *Advances in Neural Information Processing Systems*, 2017, pp. 4424–4434.

[44] Z. Zhou, S. Yang, L. J. Pu, and S. Yu, "Cefl: Online admission control, data scheduling and accuracy tuning for cost-efficient federated learning across edge nodes," *IEEE Internet of Things Journal*, pp. 1–1, 2020.

[45] D. Leroy, A. Coucke, T. Lavril, T. Gisselbrecht, and J. Dureau, "Federated learning for keyword spotting," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6341–6345.

[46] T. Yang, G. Andrew, H. Eichner, H. Sun, W. Li, N. Kong, D. Ramage, and F. Beaufays, "Applied federated learning: Improving google keyboard query suggestions," *arXiv preprint arXiv:1812.02903*, 2018.

[47] Y. Liu, J. J. Q. Yu, J. Kang, D. Niyato, and S. Zhang, "Privacy-preserving traffic flow prediction: A federated learning approach," *IEEE Internet of Things Journal*, pp. 1–1, 2020.

[48] X. Wang, C. Wang, X. Li, V. C. M. Leung, and T. Taleb, "Federated deep reinforcement learning for internet of things with decentralized cooperative edge caching," *IEEE Internet of Things Journal*, pp. 1–1, 2020.

[49] D. Misra, "Mish: A self regularized non-monotonic neural activation function," *arXiv preprint arXiv:1908.08681*, 2019.

[50] B. Kemp, A. H. Zwinderman, B. Tuk, H. A. Kamphuisen, and J. J. Oberye, "Analysis of a sleep-dependent neuronal feedback loop: the slow-wave microcontinuity of the eeg," *IEEE Transactions on Biomedical Engineering*, vol. 47, no. 9, pp. 1185–1194, 2000.

[51] A. L. Goldberger, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

[52] A. Stisen, H. Blunck, S. Bhattacharya, T. S. Prentow, M. B. Kjærgaard, A. Dey, T. Sonne, and M. M. Jensen, "Smart devices are different: Assessing and mitigatingmobile sensing heterogeneities for activity recognition," in *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*. ACM, 2015, pp. 127–140.

[53] G. Vavoulas, C. Chatzaki, T. Malliotakis, M. Pediaditis, and M. Tsiknakis, "The mobiact dataset: Recognition of activities of daily living using smartphones." in *ICT4AgeingWell*, 2016, pp. 143–151.

[54] J. Liono, F. D. Salim, N. van Berkel, V. Kostakos, and A. K. Qin, "Improving experience sampling with multi-view user-driven annotation prediction," in *2019 IEEE International Conference on Pervasive Computing and Communications (PerCom*. IEEE, 2019, pp. 1–11.