UniTS: Short-Time Fourier Inspired Neural Networks for Sensory Time Series Classification

Shuheng Li

University of California, San Diego
shl060@ucsd.edu

Ranak Roy Chowdhury

University of California, San Diego
rrchowdh@eng.ucsd.edu

Jingbo Shang*

University of California, San Diego
jshang@eng.ucsd.edu

Rajesh K. Gupta*

University of California, San Diego
gupta@eng.ucsd.edu

Dezhi Hong*

University of California, San Diego
dehong@eng.ucsd.edu

ABSTRACT

Discovering patterns in time series data is essential to many key tasks in intelligent sensing systems, such as human activity recognition and event detection. These tasks involve the classification of sensory information from physical measurements such as inertial or temperature change measurements. Due to differences in the underlying physics, existing methods for classification use hand-crafted features combined with traditional learning algorithms, or employ distinct deep neural models to directly learn from raw data.

We propose here a unified neural architecture, UniTS, for sensory time series classification in various tasks, which obviates the need for domain-specific feature, model customization or polished hyper-parameter tuning. This is possible as we believe that discriminative patterns in sensory measurements would manifest when we combine information from both the time and frequency domains.

In particular, to reveal the commonality of sensory signals, we integrate Short-Time Fourier Transform (STFT) into neural networks by initializing convolutional filter weights as the Fourier coefficients. Instead of treating STFT as a static linear transform with fixed coefficients, we make these weights optimizable during network training, which essentially learns to weigh each frequency channel. Recognizing that time-domain signals might represent intuitive physics such as temperature and acceleration, we combine linearly transformed time-domain hidden features with the frequency components within each time chunk. We further extend our model to multiple branches with different time-frequency resolutions to avoid the need of hyper-parameter search.

We conducted experiments on four public datasets containing time-series data from various IoT systems, including motion, WiFi, EEG, and air quality, and compared UniTS with numerous recent models. Results demonstrate that our proposed method achieves an average F1 score of 91.85% with a 2.3-point improvement over the state of the art. We also verified the efficacy of STFT-inspired structures through numerous quantitative studies.

*Corresponding authors

This work is licensed under a Creative Commons Attribution International 4.0 License.

SenSys '21, November 15–17, 2021, Coimbra, Portugal

© 2021 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-9097-2/21/11.

<https://doi.org/10.1145/3485730.3485942>

CCS CONCEPTS

• **Computing methodologies** → **Neural networks**; • **Computer systems organization** → **Sensor networks**.

KEYWORDS

Sensory Time Series, Classification, Time-frequency Analysis, Deep Neural Networks

ACM Reference Format:

Shuheng Li, Ranak Roy Chowdhury, Jingbo Shang, Rajesh K. Gupta, and Dezhi Hong. 2021. UniTS: Short-Time Fourier Inspired Neural Networks for Sensory Time Series Classification. In *The 19th ACM Conference on Embedded Networked Sensor Systems (SenSys '21)*, November 15–17, 2021, Coimbra, Portugal. ACM, New York, NY, USA, 14 pages. <https://doi.org/10.1145/3485730.3485942>

1 INTRODUCTION

Sensor-equipped intelligent systems can greatly assist our lives, ranging from smartphones [32, 43] to smart speakers [5, 33], autonomous vehicles [49, 75] to connected environments [1, 3]. Key to this is their capability to discover patterns in their sensory data that can be classified and put to use. For example, human activity recognition (HAR) is a key task in many assistive services and applications [30, 65], which can be realized either with wearable devices [29] or contactlessly via wireless medium such as radio frequency (RF) [50, 71] and millimeter wave [38, 40].

In the traditional learning paradigm, researchers would develop task-specific algorithms, which often involves designing hand-crafted features, in conjunction with domain knowledge to solve a specific problem such as posture monitoring [36], gesture recognition [10], and epileptic seizure detection [58, 66]. A typical workflow involves converting raw sensory time series into the frequency domain, filtering and extracting representative features. These steps require designing custom, task-specific methods to achieve good performance, which cannot avoid system re-calibration upon environmental changes [36], not to mention the difficulty of generalizing to other new tasks.

In recent years, with massive data being collected by sensor networks, deep learning methods have found great use for its impressive learning ability of hidden patterns. A majority of work replaces complicated data processing and handcrafted feature extraction procedures with deep neural networks, achieving the state-of-the-art in many tasks. Nonetheless, like their predecessors, these solutions target at tasks in one specific domain such as device-free HAR [27] or robotics anomaly detection [46]. Their proposed methods could

be adapted to time series classification problems in other domains. However, this would require nontrivial customization, and their empirical effectiveness remains unclear. We, therefore, propose UniTS, one *Unified* neural networks model for various kinds of sensory Time Series, without need for any task- or dataset-specific tuning.

The underlying difficulties are nonetheless nontrivial. While the sensory time series data collected in different IoT systems can be organized in a multi-dimensional matrix, the underlying physical meaning could be very different. Some measurements like temperature, humidity, light intensity or acceleration, are intuitively understood, others may not mean much to humans, for instance, electroencephalogram (EEG) recording or WiFi channel state information sequence. One potential approach utilizes time-frequency analysis, as the underlying physical processes that drive the phenomena measured by IoT systems depend more on the signal's frequency characteristics as time proceeds. We also follow the idea of time-frequency analysis and equip UniTS with a novel Temporal-Spectral Encoder (TS-Encoder).

However, fine-grained frequency- and time-domain resolutions cannot be guaranteed simultaneously, as Fourier uncertainty suggests [61]. Although STFT can control the trade-off between time and frequency domains, the overall resolution of STFT is still fixed. Besides, the time and frequency domain resolutions required in different IoT systems vary drastically. For applications where the sensor readings measure physical quantities that have intuitive meanings such as temperature and acceleration, often the finest-grained resolution in the time domain is required. In other cases, the sensor readings may be the property of 'invisible' signals such as WiFi, ultrasound, or brain scalp activity, which might not have straightforward meaning in the time domain, whereas the finest frequency-domain information can best describe the signal. As a result, in the TS-Encoder, we partition raw time series into windows and model the data in both the frequency and time domains within each window. We use STFT to compute the frequency components and use a convolutional layer to encode the time-domain information within each partitioned time chunk. Moreover, inspired by the multi-scale strategy in Inception Network [42, 63], we use multiple parallel TS-Encoders of different window sizes to model sensory time series at multiple scales.

Transformed spectral signals fully preserve energy, including noise and non-essential components. Thus, compatible neural network structures for denoising and effective pattern extraction are also required. In deep learning frameworks, simply treating the frequency-domain transform as a static linear transform is not an optimal solution. Instead, we replace static STFT operation in our TS-Encoder with a convolutional layer, whose weights are initialized using STFT weights. The weights of convolution layers are updated as training progresses. From the perspective of deep learning, the insight is that the weights of the derived frequency components by Fourier transform are an orthogonal basis that could efficiently map the input to a latent space. From the perspective of signal processing, enabling Fourier weights to be self-adaptive is to learn the most compatible filtering window function in a data-driven manner.

Therefore, UniTS is capable to capture both temporal and spectral features in multiple resolutions, and adaptively filter noisy and non-essential components, making it unified to all types of sensory

data. We experimented UniTS without any model structure tuning or greedy hyper-parameter search on four sensor-derived time series classification datasets, including a device-based HAR dataset (*Motion*), a device-free HAR dataset (*WiFi*), an epileptic seizure detection dataset (*Seizure*), and an occupancy prediction dataset in building environment (*KETI*). These datasets cover a broad range of sensing phenomena including inertial sensors, WiFi signals, EEG recordings, and air quality sensors. We compared with both deep learning frameworks for IoT applications and for general time series classification. Experimental results demonstrate the effectiveness of our proposed method.

This paper makes the following contributions:

- (1) We propose UniTS, a deep learning framework that enables uniform treatment of time series data from any sensing modalities, obviating the need of task-specific structure customization.
- (2) We design a novel TS-Encoder that can serve as a principled structure for neural networks conducting time-frequency analysis. It is composed of a weight initialization strategy together with a frequency channel selection strategy that integrates STFT into deep learning models as an optimizable module. We conduct experiments to study the behavior of TS-Encoder.
- (3) We experiment on four datasets that cover various sensing modalities. UniTS achieves the best average F1 score of 91.85% with a 2.3-point improvement over the state of the art.
- (4) We study the model's sensitivity and robustness to environments with reduced data volume or degraded data quality. Results show that UniTS can adapt to such environments with minor performance drop.

2 RELATED WORK

This work builds upon recent deep learning approaches for sensory time series classification summarized below with a focus on feature extraction for time domain and frequency domain.

2.1 Traditional Approach to Classification

Sensory time series classification typically consists of feature extraction and a learning algorithm [34], such as instance-based learning (IBL) [72], support vector machine (SVM) [12], k-Nearest Neighbors (KNN) [59], random forest [17], and many more [15, 67]. Feature extraction transforms raw sensory signals into a set of feature vectors by applying specific methods or custom strategies for different sensing modalities.

2.1.1 Time-Domain Feature Extraction. Time-domain methods focus on the numerical and statistical quantities underlying the raw input sensing signals. They are usually applied to human-perceived signals, such as velocity, acceleration, distance, and intensity. For device-based HAR, conventional methods involve computing standard statistical quantities (e.g. mean, standard deviation, quartiles), entropy, or correlation between sensing dimensions [47, 61]. Linear discriminant analysis (LDA) [9] and auto-regressive models [21] are further used for speed-up. In the case of wireless signals described by the received signal strength indicator (RSSI), additional statistical features related to peaks and direction changes are also extracted [55, 59]. For radar data, raw signals are usually mapped to a two-dimensional space of radial distance (or range) and velocity to derive the range-Doppler images with range-Doppler processing.

Low-level features including dispersion, centroid, and instantaneous energy, and data-centric features including range Doppler multi-channel integration (RDMI) and I/Q derivative vector are extracted subsequently [38].

2.1.2 Frequency-Domain Feature Extraction. Frequency-domain [77] methods have found wide-spread use for their explainability. Discrete Fourier Transform (DFT) [4] and Discrete Cosine Transform (DCT) [2] are the two most commonly used methods for frequency-domain conversion. Other methods aiming at noise reduction include Discrete Wave Transform (DWT) and filtering [11, 45]. In sensory data classification tasks, the labels are related to events in the environment, which could cause frequency changes in the monitored phenomena over time. Thus, STFT is another effective technique for generating time-frequency features [51]. Unlike time-domain methods that typically work on a certain type of data, frequency-domain methods mentioned above are generally applicable to all kinds of sensory data. Apart from the methods modeling frequency changes, Wang et al. [69] proposed a novel feature extraction methods from the perspective of phase changes. They proposed to derive the In-phase (I) and Quadrature (Q) components for ultrasound signals and constructed a complex I/Q trace over time as features.

Despite their effectiveness, these feature extraction strategies require domain- or problem-specific treatment, making them difficult to generalize to multiple tasks.

2.2 Deep Learning-based Solutions

Recently, deep learning models have been used for their excellent ability to learn non-linear patterns in data. These solutions typically learn in an end-to-end fashion, where they start directly with raw data or by converting them to the frequency domain, bypassing explicit feature extraction.

2.2.1 Learning in Time and Frequency Domains. For time-domain methods, a wide range of tasks are studied, including human activity and gesture recognition [31, 41], occupancy detection [7, 16], epileptic seizure detection [64], and many more [13]. These works use convolutional neural networks (CNN) [19, 48], recurrent neural networks (RNN) [8], combination of the two for various types of data [37, 56, 68], ensemble of LSTMs [20], and transformer [35]. Dilated CNN layer has also been introduced to expand the receptive field as well as avoid information loss in pooling layers [73]. Recently, attention-based mechanisms that work with time-domain data have shown promising progress [25, 39, 41, 62, 79] while their effectiveness for sensing tasks remains to be evaluated.

For models that learn in the frequency domain, both DFT and STFT can be used for data preprocessing, with 1D-CNN [76] and 2D-CNN [28] modeling the 1D frequency components and 2D time-frequency spectrogram, respectively. Researchers have also sought to develop principled deep neural networks for learning in the frequency domain for sensory time series. Yao et al. proposed DeepSense that encodes frequency-domain input through a hierarchical sensor fusion network [76]. Extending DeepSense,

STFNets introduces a deep learning framework consisting of multi-resolution layers that take as input time-frequency signals for modeling sensory time series [77]. However, such multi-resolution strategy requires input of size of a power of two and the pipeline requires converting data back to the time domain via inverse STFT, making STFNets inflexible.

As device-based tasks often use data collected from different sensing modalities, sensor fusion strategy is also widely adopted to design deep neural network architectures. Modality-specific Architecture (MA) introduces multiple branches of neural layers, where each extracts different modality-specific features [46]. Jeyakumar et al. proposed the Sense-HAR model that fuses sensor information hierarchically and learns an ensemble classifier for each device, enabling the model to recalibrate with ease upon environmental changes [26].

While these techniques have improved performance for various tasks, these methods are mostly focused on information in one domain and need special task-specific processing of input data to fit into their workflow. We shall show later in evaluation that these models can achieve good performance only on some of the datasets. We seek a *unified* architecture that accommodates to all the sensing modalities by exploiting fine-grained information from both time and frequency domains to accommodate various tasks. Nonetheless, lessons and experience discussed in these works are instrumental to our design of UniTS.

2.2.2 Other Learning Paradigms. There have also been many attempts to improve the adaptability and learning efficiency of deep neural networks for IoT tasks. Jiang et al. proposed a CNN-based model containing a domain discriminator to identify the environment activities come from so that activity recognition performance is stable regardless of the scene [27]. Xue et al. proposed a multi-view approach that models data from different transmitter-receiver pairs to derive a common representation shared across multiple different environments [74]. In addition, Ding et al. proposed a meta-learning framework for device-free HAR [14]. They proposed a novel dual-path model that combines frequency domain and time domain inputs and learns temporal hidden features via LSTM with attention and spatial hidden features via ResNet [22]. As they mainly focus on meta-learning, they did not develop specific structures for pattern extraction in the frequency domain.

To relieve the need for large amounts of labeled training data, researchers have also explored various learning paradigms, including weak supervision (e.g. labels from a different task) [57], active learning [24], reinforcement learning [80], domain adaptation or transfer learning [6, 44, 52]. These are different learning paradigms for training any neural network and can also be used with our proposed architecture.

3 METHODOLOGY

We seek an architecture to effectively exploit information in time and frequency domains for sensory time series classification in various tasks. As shown in Figure 1, UniTS takes the raw sensory time series data as input and applies multiple branches of TS-Encoder with varied receptive field, so as to extract multi-scale time-frequency domain features. The TS-Encoder converts readings from each sensing channel to a virtual spectrogram. These

Figure 1: UniTS: A unified model for sensory time series data with multiple branches of temporal-spectral encoder (TS-Encoder) at various scales. In each branch, shared weighted TS-Encoder encodes readings from each sensing channel as a virtual spectrogram. Then a sensor spectrogram fusion module learns the virtual frequency-domain correlation across all the sensors. After further encoding by ResNet, the multi-scale latent features are fused at the end of the model and fed to a dense layer for classification.

Figure 2: TS-Encoder: The spectral path partitions the input time series into time chunks and learns the frequency-domain information by Fourier Initialized Convolution (FIC) and virtual filtering. The temporal path provides time-domain supplementary within the same time chunks by time-domain Resolution Preservation Convolution (RPC). The output of TS-Encoder is a virtual spectrogram that combines time- and frequency-domain formation within each time chunk.

Figure 3: Sensor Spectrogram Fusion: We concatenate the virtual spectrogram from each sensor and apply 1-D convolution with stride 1 to learn the correlation between sensing channels.

spectrograms are then fused hierarchically by sensor-wise fusion modules and a multi-scale fusion module for classification.

We first introduce our problem formulation considering different types of sensory data in Section 3.1. We give an overview of TS-Encoder in Section 3.2 and then explain in detail the three key

strategies used in TS-Encoder, namely, Fourier initialized convolution, virtual filter, and the resolution preservation technique in Section 3.3, 3.4 and 3.5 respectively. Finally, we introduce the hierarchical fusion modules in Section 3.6.

3.1 Problem Formulation

In sensory time series classification tasks, raw measurements from some sensor(s) in a short time window (typically on the order of milliseconds) are observed, and then classification of these readings is performed to determine certain state(s). For instance, whether a room is occupied or not, or a user is walking or sitting. Such state information serves higher-level of decision-making for intelligent sensing systems. Given a raw dataset, we generate examples for classification with a fix-sized sliding window over the entire time-domain sequence. Specifically, for the i^{th} example, the data are organized as a two-dimensional matrix $X^{(i)} \in \mathbb{R}^{C \times T}$ accompanied by its true label $y^{(i)}$, where C denotes the number of unique data channels (e.g. a sensing axis from accelerometer on a phone) and T is the number of temporal samples per channel. In the case of a general IoT sensing scenario consisting of s sensors from C_d devices, we concatenate all the sensor readings from each device separately to obtain $C = C_d \cdot s$ sensing channels. In the case of WiFi sensing, we concatenate measurements from each subcarrier of every transmitter-receiver (tx-rx) pair to obtain C virtual sensing channels. For radar environments, similarly, we stack the fast-time samples of each tx-rx pair together to formulate C virtual input channels. Given an input example, the goal of sensory time series classification is to predict a label $\hat{y}^{(i)}$.

In the rest of this paper, we denote the input to our model as X , its prediction as $\hat{y}$, and the ground truth label as y , omitting the subscript index for simplicity. All the multi-dimensional matrices are denoted as bold upper-case letters (e.g. X and Y) and 1-dimensional vectors are denoted as bold lower-case letters (e.g. x and y). For slicing and indexing, we denote the j^{th} element of vector x as $x_{[j]}$ and the j^{th} slicing along the second dimension of matrix X as $X_{[:,j]}$.

3.2 Overview of TS-Encoder

As illustrated in Figure 2, the TS-Encoder is a shared weighted module that encodes 1D time-series readings from each sensing channel by partitioning it into small time chunks with a sliding window. Within one time chunk, TS-Encoder uses a *spectral path* for frequency-domain feature extraction and uses a *temporal path* for temporal feature extraction as a supplementary. Thus, it formulates a virtual spectrogram in which both time domain and frequency domain features are extracted. For the spectral path, we propose a novel Fourier Initialized Convolution (FIC) layer to imitate STFT operation that segments raw signal to small time chunks and converts them to the frequency domain. We enable weights of FIC to be learnable through standard gradient descent, allowing for model customization for different tasks which may well require different weights on the same frequency component; this is in contrast to the fixed weights derived in standard Fourier transform.

To alleviate the impact of noise in the data on learning, we propose a virtual filtering strategy, which regulates the back-propagation of gradient flow in training and removes noisy frequency channels in testing. Recognizing the need for preserving fine-grained time-domain information within the time chunks, we also introduce a supplementary temporal path with a Resolution Preservation Convolution (RPC) layer. The virtually filtered frequency-domain hidden representation and the time-domain hidden representation generated by RPC are stacked as output of the TS-Encoder, producing a virtual spectrogram.

3.3 Fourier Initialized Convolution

To integrate frequency-domain transform into deep learning pipelines while ensuring flexibility and generalizability for different tasks, we make certain parameters optimizable through standard method such as gradient descent. To this end, we exploit the similarity between STFT and 1D convolution. Given an input from one sensing channel $\mathbf{x} \in \{X_{[1,:]}, X_{[2,:]}, \dots, X_{[C,:]} \}$, an ordinary 1D convolution layer with weights $\mathbf{W} \in \mathbb{R}^{k \times l}$ is formulated as

$$\text{Conv1d}(\mathbf{x}, \mathbf{W})_{[m,n]} = \sum_{i=1}^l \mathbf{x}_{[s \cdot n + i]} \cdot \mathbf{W}_{[m,i]}, \quad (1)$$

where k , l and s denote the number, size, and stride of the convolution kernel, respectively. We omit the bias term for simplicity. For the STFT operation with sliding rectangular window function $w(\cdot)$ of window size l and stride s , the short-time Fourier transform of input $\mathbf{x} \in \mathbb{R}^T$ can be written as

$$\begin{aligned} \text{STFT}(\mathbf{x})_{[m,n]} &= \sum_{i=1}^T \mathbf{x}_{[i]} \cdot w(i - s \cdot n) \cdot e^{-j \frac{2\pi m}{l} (i - s \cdot n)} \\ &= \sum_{i=s \cdot n + 1}^{s \cdot n + l} \mathbf{x}_{[i]} e^{-j \frac{2\pi m}{l} (i - s \cdot n)} \\ &= \sum_{i=1}^l \mathbf{x}_{[s \cdot n + i]} e^{-j \frac{2\pi m}{l} i}. \end{aligned} \quad (2)$$

Combining Eq. (1) and Eq. (2), if we initialize the convolution kernels by $\mathbf{W}_{[m,i]} = e^{-j \frac{2\pi m}{l} i}$, then the convolution layer is identical to

computing STFT of the input time series. To avoid possible implementation difficulty due to neural networks for complex values, we further treat each complex frequency coefficient as two real values, representing the real part and imaginary part. Thus, to represent the m^{th} frequency component, the real-valued convolution kernel is initialized as

$$\begin{aligned} \mathbf{W}_{[2m+1,i]} &= \cos\left(\frac{2\pi m i}{l}\right), \\ \mathbf{W}_{[2m+2,i]} &= \sin\left(\frac{2\pi m i}{l}\right), \end{aligned} \quad (3)$$

where the range of m is $[0, \frac{l}{2}]$, due to the fact that Fourier transform of real number series is conjugate symmetric. As Eq. (3) suggests, we initialize the $2m + 1^{\text{th}}$ and $2m + 2^{\text{th}}$ filter to represent the amplitude of the cosine and sine wave at frequency $\frac{m}{l} \cdot f_{\text{base}}$, respectively, where f_{base} denotes the base sampling frequency in the dataset. From the deep learning perspective, our proposed initialization strategy gives orthogonal weights to each filter, which works as an efficient pre-defined mapping to a latent space. Moreover, compared with other initialization methods, including Xavier initialization and random orthogonal initialization, such Fourier-based initialization is not meaningless. Instead, it extracts local frequency patterns corresponding to the physics underlying the input signals.

3.4 Virtual Filtering

Although Fourier transform reveals important spectral features by viewing signals in the frequency domain, the noise in the raw signal is also preserved during the transform according to Parseval's theorem. Enabling the Fourier weights to be optimizable would emphasize certain frequency components over the others in a data-driven manner, but it could not serve as a filtering mechanism, as no supervision can be provided via the loss function. Thus, we design a compatible filtering mechanism to dampen the noise in the data, enhancing the robustness of our model. The virtual filter is composed of two strategies, low-pass filtering and importance filtering. Low-pass filtering is integrated into FIC while importance filtering regulates the gradient back-propagation flow by selecting important frequency channels.

Traditionally, low pass filters are effective in modeling sensory time series data, as most of high-frequency components in the sensory signals are noise. Additionally, we propose to remove the direct current (DC), i.e. the zero-frequency components. Since the real part of DC is simply the sum of all the temporal readings, it only extracts coarse-grained temporal information and thus can be substituted entirely by the temporal path RPC, which we shall explain in details shortly. In FIC, these strategies can be achieved by directly controlling the number of convolution kernels in the TS-Encoder during initialization. After removing the DC component, the weights for representing the lowest k frequency components in FIC are initialized as follows,

$$\begin{aligned} \mathbf{W}_{[2m-1,i]} &= \cos\left(\frac{2\pi m i}{l}\right), \\ \mathbf{W}_{[2m,i]} &= \sin\left(\frac{2\pi m i}{l}\right), \end{aligned} \quad (4)$$

where $m = 1$ to $\frac{k}{2}$, and k should be an even number such that $k \leq l$.

Commonly, only a few frequency components are decisive factors with large coefficients in the frequency domain of sensory time

series. Even though the low-pass filtering strategy preserves only low-frequency components, some of them could still have small values of little importance to hidden feature extraction. Filtering the frequency channels whose amplitude is below a certain threshold could be a solution. However, the threshold must be chosen specifically for each task and each type of sensor, which can require heavy manual effort. Thus, we directly choose the most important t frequency channels with respect to their absolute value of the amplitude, and concatenate the indices of these frequency channels with the chosen top t values subsequently. The output of the virtual filtered FIC is derived as,

$$\begin{aligned} H_f &= \text{abs}(FIC(x, W_f)) \\ S &= \text{top}_t(H_f, \text{dim} = 1) \\ O_f &= [S, \text{ind}(S)], \end{aligned} \quad (5)$$

where W_f denotes the FIC weights derived by Eq. (4). $H_f \in \mathbb{R}^{2k \times n}$ is the output of FIC, with n time chunks and $2k$ frequency components. $S \in \mathbb{R}^{t \times n}$ is the top $t \leq 2k$ out of total $2k$ frequency channels in n time chunks and ind gives the indices of the chosen frequency channels in S . $[\cdot, \cdot]$ denotes the concatenation operation. Therefore, the output hidden feature of the virtually filtered FIC is $O_f \in \mathbb{R}^{2t \times n}$. Note that in autograd computation graphs, choosing the top t values directly maps the backward gradient flow to the chosen values, while the indices given by ind function are not differentiable. As a result, only those frequency channels that are important will be updated as the FIC weights during training. The importance sampling works by regulating gradient back propagation in training and filtering out less important frequency channels in evaluation.

3.5 Resolution Preservation

The spectral path views small time chunks from the perspective of their constituting frequencies, which can often reveal certain local physics feature in the sensory signal. However, due to the uncertainty principle, the temporal information is lost during the transform. STFT could only balance the frequency and time domain resolution, but it cannot increase the overall resolution in both domains. To preserve the time domain resolution, we propose a temporal path for modeling supplementary time-domain information by a Resolution Preservation Convolution layer that extracts hidden temporal features for the time chunk indexed by the FIC filter. In order to align with the number of time chunks of the output of FIC, we apply an ordinary convolution layer, whose filters have the same size as FIC. Then, the temporal representation can be concatenated with the frequency representation as,

$$\begin{aligned} H_t &= \text{RPC}(x, W_t) + b_t \\ O &= [O_f, H_t], \end{aligned} \quad (6)$$

where O_f is computed following Eq. (5). $W_t \in \mathbb{R}^{t \times n}$ and $b_t \in \mathbb{R}^t$ are the weights and bias of RPC , which are initialized randomly. Since the temporal convolution layer has the same filter size as FIC, the concatenated output is $O \in \mathbb{R}^{(3t) \times n}$.

RPC helps extract the time-domain features and map them to a hidden matrix that has the same size as the output of FIC. From the perspective of signal processing, both H_f and H_t are output of convolution, so it makes sense to concatenate them. From the

deep learning point of view, weights of FIC and RPC are initialized differently. FIC emphasizes certain frequency channels, since only top the t frequency channels can be updated through gradient descent optimization. On the other hand, RPC learns temporal local patterns starting from a randomly initialized convolution weights. With the information learnt from both time and frequency domains, our TS-Encoder generates a virtual spectrogram that preserves the finest temporal and spectral resolutions.

3.6 Hierarchical Fusion Network

As shown in Figure 1, we use multiple TS-Encoder with different FIC window sizes to learn the time-frequency hidden features at multiple temporal resolutions. In each branch, TS-Encoder generates a virtual spectrogram for each input sensing signal. We first fuse sensor-wise information by the fusion module illustrated in Figure 3. For all the virtual spectrograms, the sensor fusion module stacks the virtual frequency dimension and applies a 1D convolution layer with stride 1 and kernel size 1 for learning the correlation across sensors, between the virtual frequency channels. The spectrogram of one sensing modality O has $3t$ virtual frequency channels and there are C different sensing modalities in total. As a result, the 1D convolution for sensor spectrogram fusion takes $3t \times C$ input channels and encode them into a latent space.

We use ResNet's backbone architecture to further extract local temporal patterns in neighboring time chunks. For each TS-Encoder scale, we stack multiple residual blocks with average pooling layers connecting neighboring blocks for time dimension reduction. Each residual block consists of two bottleneck convolution layers with residual connection and batch normalization. Subsequently, these vectors are stacked at the end of the model and fed into a dense layer for prediction [22]. Through the scale fusion mechanism, the best window size for TS-Encoder and the optimal combinations are decided automatically.

4 EVALUATION

We now experimentally verify the effectiveness of UniTS. We focus on four typical sensory time series classification datasets that contain a variety of sensing modalities. We compare UniTS with relevant state-of-the-art approaches. We further examine our proposed strategies through ablation study and demonstrate the robustness of UniTS via sensitivity analyses.

4.1 Datasets

Our choice of datasets follows commonly used datasets in previous works [23, 25, 26, 35]. We use four datasets in our evaluation, namely, *Motion*, *Seizure*, *WiFi* and *KETI*. For the *Motion* dataset, we choose 20 recording files for generating training examples and use the rest 4 files for generating testing examples. For the other datasets, we adopt a 9 : 1 training and testing split for evaluation. We use accuracy and macro-F1 score as the evaluation metrics.

Motion: We conduct human locomotion recognition using wearable motion sensors with inertial measurement units (IMU). The data used in the experiment is a subset of the OPPORTUNITY Activity Recognition Dataset [54]. Following the experimental setting of a recent device-based HAR study [26], we consider five sensors worn at five different locations on human body: left lower arm, left

Table 1: Statistics of datasets used in evaluation

Dataset	Sampling Rate	#Instances	#Classes	Models
Motion	30 Hz	11,073	4	[26][70][53][77]
Seizure	256 Hz	21,436	2	[25][53]
WiFi	1000 Hz	8,099	7	[35][14][77]
KETI	0.1 Hz	16,921	2	-

upper arm, right lower arm, right upper arm and the back. Each device contains an accelerometer, a gyroscope, and a magnetometer, and all three sensors generate three-axis readings. We focus on a 4-class prediction consisting of high-level locomotion activities (sit, stand, walk and lie). Examples for training and evaluation are segmented into windows of 256 timestamps with a sampling rate of 30 Hz. Therefore, each data example is a 45×256 matrix.

Seizure: In this task, we predict epileptic seizure events using EEG recordings collected from pediatric subjects at the Children’s Hospital Boston [18]. The data were collected at a sampling rate of 256 Hz with periods of seizure attack labeled. Among 23 locations on brain scalp used for EEG recording in the dataset, we select 18 locations that were used by all the subjects for the experiment. We segment the data into examples containing 256 timestamps and obtain a 18×256 matrix for each example. We downsample negative examples to match the number of positive examples for label balance.

WiFi: WiFi dataset was collected using Linux 802.11n CSI Tool by Yousefi et al [78]. This dataset contains the amplitude and phase of wireless signals sent by three antennas. Each antenna transmits at 30 subcarriers, and the receiver base sampling frequency is 1000 Hz. The dataset contains seven activities including lie down, fall, pick up, run, sit down, stand up and walk. We also use a sliding window of 256 timestamps to get the segmented examples. Thus, each input example is a matrix of 180×256 .

KETI: Since building data is little studied in related works, we expand the scope of our study by introducing KETI. This dataset was collected from 51 rooms in a large university office building [23]. Each room is instrumented with 4 sensors monitoring CO₂, temperature, humidity and light intensity, with occupancy monitored by an additional PIR sensor in the room. Readings are recorded every 10 seconds, and the dataset contains one-week worth of data. In this experiment, we target at human occupation prediction using the readings of these sensors. Specifically, we split the raw data into examples containing 256 timestamps and label each example as occupied only when any occupant is present at all the timestamps, and vice versa. As a result, the example in this experiment is a 4×256 matrix.

We summarize the statistics of datasets in Table 1. We list all the compared baseline models that use the same type of data, and embolden those that use exactly the same dataset.

4.2 Compared Methods

We compare UniTS with a collection of deep learning models for sensory time series classification.

- **DWT:** We combine discrete wavelet transform (DWT) with neural networks by adopting CNN to model the output of DWT. Following previous study [60], we conduct four-level db-8 wavelet

transform.

- **ResNet:** ResNet is one of the most common deep learning architectures used in computer vision [22]. Wang et al. [70] extend ResNet to time series data, and it has achieved competitive performance in many sensory time series classification tasks. In our experiment, we stack 18 residual blocks and build a deeper architecture to fully leverage the ability of residual networks.
- **MaCNN:** MaCNN is the CNN version of the multi-modality fusion structure proposed by Radu et al. [53]. MaCNN models each sensing modality with a separate CNN and then fuses the multi-modality information for prediction.
- **SenseHAR:** SenseHAR introduces a novel hierarchical CNN-based sensor fusion network that is flexible upon the change of devices/sensors in IoT environments. It uses LSTM for prediction, in which the weights of LSTM are shared across all the sensing modalities [26].
- **STFNets:** STFNets integrates multi-resolution STFT into deep learning frameworks and constructs a time-frequency hologram by learning self-attention for frequency channels. It further proposes three time-frequency domain operations for feature extraction [77].
- **RFNet-base:** Ding et al. propose RFNet, a meta-learning framework for radio-frequency signals [14]. Their dual-path base model that models time and frequency domain input with a spatial and a temporal module is a competitive baseline.
- **THAT:** THAT models sensory time series by two convolution-augmented transformer models for learning both temporal and sensor-wise information [35].
- **LaxCat:** LaxCat introduces an explainable time series classification framework by using CNN for feature extraction and two consecutive attention layers for learning sensor-wise and temporal information [25].

Among all the compared models, ResNet is a conventional model that achieves strong performance in many scenarios. MaCNN, SenseHAR, THAT and LaxCat learn the sensor-wise correlation via certain fusion mechanisms. DWT is a traditional time-frequency analysis approach for signal processing. STFNets and RFNet-base explore deep learning in both time and frequency domains.

4.3 Experimental Setup

We implemented our proposed model with nine branches. Specifically, the FIC window sizes are {7, 16, 32, 48, 64, 80, 96, 112, 128}, and the stride of FIC window is half of the window size, namely, {3, 8, 16, 24, 32, 40, 48, 56, 64}. Since we have pre-processed all the datasets so that the input examples have the same number of timestamps, we firstly use evenly spaced numbers, i.e. from 16 to 128 as FIC window sizes. In this way, local temporal features can be approximately extracted by these multi-scale branches. We additionally add an FIC branch with window size 7. As a smaller prime number, it helps increase the flexibility of the combinations of FIC window sizes. The strides are set to half of the window size, which enables overlapping for finer-grained temporal feature extraction and avoid generating huge hidden matrices.

For each FIC branch, the number of FIC kernels is the same as the window size, i.e., the first half of frequency channels are preserved by virtual low-pass filtering. And $t = \{3, 8, 16, 24, 24, 32, 32, 40, 40\}$

important frequency channels are further selected in the 9 branches. The hidden dimension for 1D sensor spectrogram fusion convolution is 48. The convolution kernel size in ResNet is 3, and the pooling layer size is 2. We stack residual blocks until the rest of temporal dimension is smaller than 3, the output hidden feature is then fed to the multi-scale fusion layer. The batch size for WiFi dataset is 16 for that there are fewer training examples, while we set the batch size as 64 for the other datasets. The model is trained for 50 epochs and optimized by the Adam optimizer with learning rate $1e-3$. We use the above hyper-parameters for all the datasets, without any additional specific tuning.

4.4 Main Results

Table 2 summarizes the experimental results on four datasets. We emphasize the best results and the second best with bold number and underlined number correspondingly.

4.4.1 Motion. For the Motion dataset, UniTS achieves the first place and outperforms the second best model THAT by 1.2%. The improvement is critical, as the error rate is reduced by more than 10%. Among the four baseline models employing sensor-wise fusion mechanism, LaxCat does not perform very well in this task, which might be because their proposed CNN module is too shallow to extract sophisticated features. Since the Motion dataset is the one with the most diverse set of sensors, THAT performs the second best for it is capable to learn very complicated sensor-wise correlations with self-attention mechanism. On the other hand, the good performance of DWT, ResNet and RFNet-base suggests that effective feature extraction modules can obviate the need for explicit sensor fusion.

4.4.2 Seizure. On the Seizure dataset, UniTS and RFNet-base perform similarly. Although UniTS is the second best model, the gap between our model and RFNet-base is less than 0.4%. DWT ranks the third and is the only one that achieves 90% accuracy among the remaining models. It shows that modeling frequency-domain and time-domain information for the Seizure dataset is important, as the temporal intensity information and the frequency changing pattern are both important for seizure detection. It is noteworthy that STFNets also learns in the frequency domain through STFT, and the fact that their model does not exploit time-domain information might explain its poor performance. THAT and LaxCat performs good on the Seizure dataset, which shows attention mechanism can learn the correlation between EEG measurements from different locations on brain scalp.

4.4.3 WiFi. There are two challenges underlying the WiFi dataset. Firstly, there are 180 input sensing channels, thus requiring effective sensor-wise fusion strategies. This difficulty degrades the performance of DWT, RFNet-base and STFNet significantly, as they lack strategies for sensor-wise correlation learning. Secondly, WiFi radio environment is very noisy, deep models with robust neural networks tend to perform better. SenseHAR performs quite well on the WiFi dataset and achieves the second best, which shows their hierarchical fusion strategy is compatible with the structure of WiFi signals, though it was originally designed for device-based HAR problems. UniTS combines the hierarchical fusion methods with a robust structure that learns in time-frequency domain, thus

achieving the first place. We improve upon SenseHAR by 1.3% in accuracy and 2.5% in F1 score.

4.4.4 KETI. Since we do not re-sample examples to balance the classes for the KETI dataset, F1 score is a crucial metric for evaluation. UniTS and THAT are the best models on this dataset. DWT, ResNet, SenseHAR and RFNet-base are also performing well with F1 scores more than 80%. For the KETI dataset, although there are only four input sensing channels, they are collected from completely different types of sensors, and thus vary in terms of range and the underlying physics. Moreover, the ranges of data from different rooms are also distinct due to varied ambient environment. These difficulties require the model to be robust in modeling non-linear numerical hidden patterns, and this explains why MaCNN, STFNets and LaxCat do not perform well.

In general, with the same amount of training data, our model can accommodate all the testing IoT environments and achieve the best or competitive performance. UniTS is the best model on the Motion, WiFi and KETI datasets and performs the second best on the Seizure dataset with performance gap less than 0.4%. THAT is also a strong model that works well on all the datasets. However, as THAT does not consider the frequency domain, it is worse than UniTS. SenseHAR, RFNet-base and ResNet perform well in general. But they did not model sensory time series comprehensively, limiting their performance upper-bound. The other compared baselines degrade greatly in some datasets. Thus, these models are not optimal solutions to sensory time series classification.

4.5 Ablation Study

We conducted ablation study on the four datasets to examine the effect of FIC and RPC in TS-Encoder. Three variants of UniTS are studied, namely, **w/o FIC** that only has the temporal path; **w/o FIC** that only has the spectral path and **w/o trainable**, in which FIC weights are not trainable; in this variant, the FIC layer is then reduced to a static STFT layer. We summarize the results in Table 3.

The effect of FIC, RPC and the trainable mechanism are both evident on all the datasets. Compared with RPC, FIC is more helpful in improving the performance, especially with respect to F1 score. This observation shows that frequency-domain features can uncover the hidden patterns underlying relatively less frequent labels. It would be otherwise difficult to capture such patterns for they are mainly non-linear and do not appear very frequently. Static STFT can capture basic time-frequency patterns in the Motion, Seizure and KETI datasets, which corresponds to the result of DWT in Table 2. Trainable mechanism helps the most on WiFi dataset, which is also the case for FIC and RPC. The fact that high frequency WiFi signals require enhancement for its low frequency parts might explain why trainable FIC is crucial in extracting nonlinear feature.

4.6 Sensitivity Analysis

We further investigate sensitivity of the models to the change of input size and sampling frequency. For each dataset, we compare with the best two baseline models, namely, we compare with RFNet-base and THAT on the Motion and the Seizure datasets and compare with THAT and SenseHAR on the WiFi and the KETI datasets. The following experiment results demonstrate that UniTS can adapt to

Table 2: Performance (%) comparisons of our method against state-of-the-art models on the Motion, Seizure, WiFi and KETI datasets. The best and the second best performance values are marked.

Datasets		DWT	ResNet	MaCNN	SenseHAR	STFNets	RFNet-base	THAT	LaxCat	UniTS
Motion	Accuracy	87.43	89.54	88.32	88.90	88.98	90.22	<u>90.35</u>	60.19	91.58
	Macro-F1	89.01	91.36	90.32	88.60	88.52	<u>92.28</u>	92.12	41.47	93.33
Seizure	Accuracy	90.48	88.98	80.92	85.10	82.82	92.71	89.77	87.03	<u>92.38</u>
	Macro-F1	90.44	88.92	80.77	85.10	82.75	92.70	89.76	86.95	<u>92.37</u>
WiFi	Accuracy	55.00	88.00	84.09	94.90	79.75	83.88	93.00	74.63	96.25
	Macro-F1	47.79	86.30	84.05	<u>92.80</u>	74.29	79.31	90.00	71.09	95.38
KETI	Accuracy	95.56	96.05	93.83	96.50	89.18	95.63	<u>96.61</u>	91.28	96.80
	Macro-F1	82.14	83.12	73.54	84.90	69.32	82.97	<u>86.21</u>	71.13	86.32
average	Accuracy	82.12	90.64	86.79	91.35	85.18	90.61	<u>92.43</u>	78.28	94.25
	Macro-F1	77.35	87.43	82.17	87.85	78.72	86.82	<u>89.52</u>	67.66	91.85

Figure 4: Sensitivity Analysis of Window Size

Figure 5: Sensitivity Analysis of Sampling Frequency: We down sample the original data by a factor of 2 and 4.

datasets with fewer training data.

4.6.1 Input Size. The default input window to all baseline models and ours contains 256 timestamps. Here we vary the input window size by selecting a subsequence of the original input data that contains {128, 64} timestamps to examine the effect of input window size on the performance. Figure 4 shows the results.

THAT is the most sensitive model to the change of input size. The relative performance drop of THAT is the highest among all the compared models. Even though THAT performs the second best in the previous experiment, it strictly demands data to be the desired

shape. RFNet-base is also greatly affected by the reduced input size. For example, on the Seizure dataset, RFNet-base is gradually outperformed by our model as the window size decreasing. SenseHAR is robust to window size change, ascribing to its special design that pools “redundant” information from different sensor channels. UniTS is also relatively unaffected by the window size change. The overall performance drop of our model is minor, which shows that our proposed TS-Encoder can effectively extract time-frequency features. Since the change of window size does not affect the base sampling frequency, the representation ability of TS-Encoder is

Figure 6: Robustness to Noise-Injection

Figure 7: Robustness to Missing Values

Table 3: Ablation study on the effect of FIC and RPC. We report on relative accuracy and F1 score changes.

Datasets		UniTS	w/o FIC	w/o RPC	w/o trainable
Motion	Accuracy	91.58	-0.21	-0.41	-0.87
	Macro-F1	93.33	-0.52	-0.54	-0.70
Seizure	Accuracy	92.38	-1.24	-0.77	-0.49
	Macro-F1	92.37	-1.24	-0.77	-0.47
WiFi	Accuracy	96.25	-7.00	-4.00	-14.80
	Macro-F1	95.38	-8.56	-5.82	-19.06
KETI	Accuracy	96.80	-0.18	-0.51	-0.70
	Macro-F1	86.32	-2.08	-1.09	-2.52

hardly affected.

4.6.2 Sampling Frequency. We also examine how the data sampling frequency affects model performance. Thus, we down-sample each input example by a factor of $\{2, 4\}$ and compare with results from the original setting. We summarize the results in Figure 4.

THAT and SenseHAR are very sensitive to the change of sampling frequency. The impact of sampling frequency to these two models can be positive and negative. Lower sampling frequency improves the performance of the two models on the WiFi dataset, but harms the performance on the KETI dataset. It is because the optimal base frequency for representing local features in different types of sensory data is usually different. In order to achieve the best performance of time-domain deep learning models like

THAT and SenseHAR, one needs to carefully experiment with the base frequency for different datasets. On the contrary, RFNet-base and UniTS model are robust to sampling frequency changes, which shows the advantage of combining the time and frequency domains. Compared with RFNet-base, our model is slightly better because we adopt a virtual filtering mechanism for further masking out noisy frequency channels. We shall note that, on the KETI dataset, the performance of UniTS does degrade slightly due to its original relatively low sample frequency (1 reading every 10 seconds)—when we down sample the data by 4x, which is equivalent to having one reading every 40 seconds, that might well fall out the required Nyquist frequency for occupancy prediction, hence the lowered prediction accuracy.

4.7 Robustness Study

We here study the overall robustness of the models via a noise-injection experiment for simulating noisy environments and a masking experiment for simulating scenarios with random missing data. Similar to the sensitivity study in Section 4.6, we compare UniTS with the top two baselines in each dataset. Results show that UniTS is robust to environments where data quality is largely degraded.

4.7.1 Noise. For simulating noisy environment, we generate random Gaussian noise on the training and testing examples. Specifically, for each sensing channel i , we compute the mean absolute reading values μ_i , and the noise for each reading of the i^{th} sensing

channel is drawn from a Gaussian distribution $\mathcal{N}(0, \alpha \cdot \mu_i)$, where we use α for controlling the intensity of the generated noise. We gradually increase α from 0.1 to 0.5 with a step size of 0.1, and summarize the results in Figure 6.

All the compared models exhibit some level of robustness to noise in some environments while they would experience significant performance drop in the other datasets. For instance, THAT, RFNet-base and SenseHAR can tolerate noise in the WiFi dataset, the KETI dataset and the Seizure dataset, respectively. However, their performance drops significantly on the KETI dataset, the Motion dataset and the KETI dataset correspondingly. Compared with the three baseline models, UniTS can relatively withstand noise in all testing environments. Our model performs the best across all the datasets, which verifies the effectiveness of our proposed virtual filtering mechanism.

4.7.2 Missing Data. For simulating the effect of missing data in IoT environments, we randomly mask each reading in the input window with a probability that increases from 0.1 to 0.5 with a step size of 0.1. Figure 7 summarizes the results.

Similar to the observations in the noise experiments, RFNet-base and THAT perform well on some datasets but degrade on other datasets. SenseHAR and UniTS can best adapt to missing value environments. On two experimented datasets, SenseHAR shows strong robustness, mainly due to their proposed hierarchical fusion strategy that can help compensate for missing values by utilizing information in other sensing channels. Our model is the best model or achieves comparable performance on all the datasets, which implies that our model can be directly used in most sensory time series classification tasks, without the need for data imputation.

5 MODEL BEHAVIOR ANALYSIS

We study the behavior of the proposed TS-Encoder and compare the effectiveness of FIC layer with standard STFT operation. Thus, we conducted experiments and visualization to demonstrate how FIC layers learn representative frequency-domain features with the help of multi-scale branches and the virtual filtering strategy.

5.1 Behavior Study on FIC Layers

In the TS-Encoder, we integrate STFT into deep learning models as a learnable module and calibrate the back propagation of gradient flow by a virtual filter that selects the most important frequency channels. Only the FIC kernel weights from the selected channels can be updated via gradient descent. The optimized FIC kernels can effectively capture frequency-domain features. We conjecture that the learnt FIC kernels would work as an integration of STFT operation with filters that mask out noisy frequency channels and amplifiers that emphasize important frequency components.

To study the behavior of FIC layers, we generate synthetic signals as input to the trained FIC layers, and compare the output of FIC layers and standard STFT operation. For FIC layer with kernel size l , we generate a synthetic signal by summing up the sine wave and cosine wave of frequencies evenly spanning from $\frac{f_{base}}{l}$ to $\frac{f_{base}}{2}$ with the same amplitude. The synthetic signal is computed as

$$\tau_l(n) = \sum_{i=1}^{l/2} \cos\left(\frac{i}{l} 2\pi n\right) + \sum_{i=1}^{l/2} \sin\left(\frac{i}{l} 2\pi n\right),$$

where $n = 1$ to l . We use $\tau_l(n)$ as input to FIC layer and STFT.

In each branch, we compute the coefficient difference for each frequency channel between FIC output and STFT output. Then we interpolate every frequency dimension to align with the branch with the highest frequency-domain resolution, i.e. largest window size. We visualize these differences in frequency domain for each branch and then stack the visualized results together for analysis. As a comparison, we introduce Orthogonal Initialized Convolution (OIC) layer as a variant of FIC. Following the process mentioned above, we compute the difference between the output of the trained OIC layer and its initial value for visualization. Due to space limit, we choose the WiFi dataset as an example to demonstrate the learning behavior of FIC and OIC layers. Results are shown in Figure 8. From branch 1 to branch 8, the window size of FIC and OIC increases from 16 to 128. Red means the frequency channel is emphasized by FIC and OIC compared to the initial weights, and blue denotes the opposite. The intensity of color indicates the magnitude of difference.

All the FIC branches focus on feature extraction for low-frequency components, which aligns with our intuition that energy of the input signals is concentrated around low-frequency bands. Only branch 8 extracts some high-frequency details. Such a pattern is in accord with the inherent feature that WiFi signals have high sampling frequency, and thus are highly subject to high-frequency noise. On the contrary, the behavior of OIC layers is not explainable, because OIC layers are not corresponding between branches and OIC filters within each branch do not have locality. This shortcoming increases the difficulty of subsequent feature extraction, resulting in accuracy drop.

5.2 FIC Feature Extraction Quality

With the back-propagation procedure being calibrated by virtual filters, we conjecture that the learnt FIC weights can be used as a feature extractor independently. We thus conducted experiments to compare the quality of extracted features by trained FIC layers and the STFT operation. In this experiment, we use the trained FIC layer of branch 4, with kernel size 64 and stride 32 for extracting features for 32 frequency channels. We choose this intermediate FIC layer as it can best balance time-frequency domain resolutions for the output spectrogram. For fair comparison, we use STFT with the same window size and stride to also encode input signals by the same 32 frequency channels.

We use the extracted features as input to a tiny ResNet model with only two residual blocks as the learning algorithm for classification. Table 4 presents the results on the Motion and the WiFi datasets. In the Motion dataset, ResNet performs reasonably with the STFT features. FIC feature improves the performance significantly for it highlights low-frequency features and weakens the other frequency channels with noise or unrelated information. For the WiFi dataset, ResNet can hardly learn from the STFT extracted features, with F1 score less than 50%. The features extracted by FIC boost the performance by 40%, which is mainly because FIC filters out a large proportion of the high-frequency noise in WiFi data. It is also noteworthy that, compared with the performance of the ResNet in Table 2 with 18 residual blocks, this shallow two-block ResNet here has better performance on the two datasets with the

Figure 8: Case study on WiFi dataset of visualized FIC and OIC layers behavior for eight main branches. We use red colors to represent layers highlighting and use blue colors to represent the under-emphasizing. The intensity is proportional to color chroma.

Table 4: We view STFT and FIC layer as two feature extraction methods. Raw examples of Motion and WiFi dataset are processed by these two methods and then fed to the same ResNet model for classification.

Extraction Method	Motion		WiFi	
	Accuracy	Macro-F1	Accuracy	Macro-F1
STFT	80.50	78.72	55.13	45.34
FIC Layer	89.40	91.53	90.88	88.96

help of FIC extracted features. These experiment results verify the feature extraction ability of FIC, which can serve as another usage of our model.

5.3 Number of Multi-Scale Branches

Lastly, we study the effect of our proposed multi-scale structure by experimenting on variants of UniTS with $\{1, 3, 5, 7\}$ branches. Following the principle of deciding the kernel sizes in TS-Encoder introduced in Section 4.3, in each variant of UniTS, we use a small prime number as the kernel size for one branch and span the kernel sizes evenly in $[8, 128]$ for the rest of branches. Figure 9 shows the experiment results on the four datasets.

In general, all the variants show competitive performance on all the datasets. Increasing the number of branches helps improve the performance on all the datasets, with the exception of the WiFi dataset. This is possibly because of its noisy environment. The use

Figure 9: Sensitivity study on the number of multi-scale branches in UniTS.

of more branches is more likely to suffer from overfitting to noise in training data.

This experiment result further suggests a principled way of applying our model in a new sensory time series classification task. One can use more multi-scale branches to achieve a satisfying performance quickly. More hyper-parameters introduced by the multi-branch structure would not harm the ease of use of UniTS. In addition, fine-tuned kernel sizes and number of branches can further boost the performance. Our parallel multi-scale structure can improve the efficiency of hyper-parameter tuning, by supporting experimenting on different number of kernel sizes simultaneously.

6 CONCLUSIONS

Inspired by Short-Time Fourier Transform, we developed a novel unified deep neural network model for sensory time series classification from any sensing modalities. Our proposed UniTS constructs parallel-connected multiple scales of TS-Encoder and then fuses the output sensor-wise spectrogram from TS-Encoder in each branch to integrate multi-scale hidden features for prediction hierarchically. The key component of UniTS is the TS-Encoder that uses a temporal path and a spectral path for modeling readings from each sensing modality as a virtual spectrogram. We conducted extensive experiments on four public sensory classification datasets from different IoT systems. These experiments demonstrate the efficacy and robustness of UniTS and the key module TS-Encoder, promising general use for sensory time series classification without the need for model customization, greedy hyper-parameter tuning or even data cleaning. The only limitations are the requirement of a certain amount of training data like usual neural networks, and a slightly longer running time. To gain a better insight into the mechanism of UniTS and TS-Encoder, we further explored the feature extraction ability of TS-Encoder by comparing with static STFT, and experimentally verified the generalizability of multi-branch structure in UniTS.

For future directions, we plan to expand our UniTS to other intelligent sensing tasks for IoT systems, including forecasting and smart control. We also plan to make use of the feature extraction ability of TS-Encoder and explore an alternative usage of TS-Encoder from the perspective of representation learning for sensory time series.

ACKNOWLEDGMENTS

This research is supported by Computing On Network Infrastructure for Pervasive Perception, Cognition and Action Research Center (CONIX) and Hacıoğlu Data Science Institute.

And it is sponsored in part by National Science Foundation Convergence Accelerator under award OIA-2040727 as well as generous gifts from Google, Adobe, and Teradata. Any opinions, findings, and conclusions or recommendations expressed herein are those of the authors and should not be interpreted as necessarily representing the views, either expressed or implied, of the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for government purposes not withstanding any copyright annotation hereon.

REFERENCES

- [1] Fadel Adib, Hongzi Mao, Zachary Kabelac, Dina Katabi, and Robert C Miller. 2015. Smart homes that monitor breathing and heart rate. In *Proceedings of the 33rd annual ACM conference on human factors in computing systems*. 837–846.
- [2] Kerem Altun and Billur Barshan. 2010. Human activity recognition using inertial/magnetic sensor units. In *International workshop on human behavior understanding*. Springer, 38–51.
- [3] Bharathan Balaji, Jian Xu, Anthony Nwokafor, Rajesh Gupta, and Yuvraj Agarwal. 2013. Sentinel: occupancy based HVAC actuation using existing WiFi infrastructure within commercial buildings. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*. 1–14.
- [4] Ling Bao and Stephen S Intille. 2004. Activity recognition from user-annotated acceleration data. In *International conference on pervasive computing*. Springer, 1–17.
- [5] Frank Bentley, Chris Luvogt, Max Silverman, Rushani Wirasinghe, Brooke White, and Danielle Lottridge. 2018. Understanding the long-term use of smart speaker assistants. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 3 (2018), 1–24.
- [6] Youngjae Chang, Akhil Mathur, Anton Isopoussu, Juneha Song, and Fahim Kawsar. 2020. A systematic study of unsupervised domain adaptation for robust human-activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 1 (2020), 1–30.
- [7] Kaixuan Chen, Dalin Zhang, Lina Yao, Bin Guo, Zhiwen Yu, and Yunhao Liu. 2020. Deep learning for sensor-based human activity recognition: overview, challenges and opportunities. *arXiv preprint arXiv:2001.07416* (2020).
- [8] Yuwen Chen, Kunhua Zhong, Ju Zhang, Qilong Sun, and Xueliang Zhao. 2016. LSTM networks for mobile human activity recognition. In *2016 International conference on artificial intelligence: technologies and applications*. Atlantis Press, 50–53.
- [9] Yen-Ping Chen, Jhun-Ying Yang, Shun-Nan Liou, Gwo-Yun Lee, and Jeen-Shing Wang. 2008. Online classifier construction algorithm for human activity detection using a tri-axial accelerometer. *Appl. Math. Comput.* 205, 2 (2008), 849–860.
- [10] Zicheng Chi, Yao Yao, Tiantian Xie, Xin Liu, Zhichuan Huang, Wei Wang, and Ting Zhu. 2018. EAR: Exploiting uncontrollable ambient RF signals in heterogeneous networks for gesture recognition. In *Proceedings of the 16th ACM conference on embedded networked sensor systems*. 237–249.
- [11] Tahmid Z Chowdhury. 2018. *Using Wi-Fi channel state information (CSI) for human activity recognition and fall detection*. Ph.D. Dissertation. University of British Columbia.
- [12] Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning* 20, 3 (1995), 273–297.
- [13] Shohreh Deldari, Daniel V Smith, Amin Sadri, and Flora Salim. 2020. ESPRESSO: Entropy and ShaPe aware time-Series SegmentatiOn for processing heterogeneous sensor data. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 3 (2020), 1–24.
- [14] Shuya Ding, Zhe Chen, Tianyue Zheng, and Jun Luo. 2020. RF-net: a unified meta-learning framework for RF-enabled one-shot human activity recognition. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*. 517–530.
- [15] Bilal Esmail, Arghad Arnaout, Rudolf K Fruhwirth, and Gerhard Thonhauser. 2012. Improving time series classification using Hidden Markov Models. In *2012 12th International Conference on Hybrid Intelligent Systems (HIS)*. IEEE, 502–507.
- [16] Cong Feng, Ali Mehmani, and Jie Zhang. 2020. Deep learning-based real-time building occupancy detection using ami data. *IEEE Transactions on Smart Grid* 11, 5 (2020), 4490–4501.
- [17] Zengtao Feng, Lingfei Mo, and Meng Li. 2015. A Random Forest-based ensemble method for activity recognition. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 5074–5077.
- [18] Ary L Goldberger, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley. 2000. PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals. *circulation* 101, 23 (2000), e215–e220.
- [19] Rene Grzeszick, Jan Marius Lenk, Fernando Moya Rueda, Gernot A Fink, Sascha Feldhorst, and Michael ten Hompel. 2017. Deep neural network based human activity recognition for the order picking process. In *Proceedings of the 4th international Workshop on Sensor-based Activity Recognition and Interaction*. 1–6.
- [20] Yu Guan and Thomas Plötz. 2017. Ensembles of deep lstm learners for activity recognition using wearables. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 2 (2017), 1–28.
- [21] Yuya Hanai, Jun Nishimura, and Tadahiro Kuroda. 2009. Haar-like filtering for human activity recognition using 3d accelerometer. In *2009 IEEE 13th digital signal processing workshop and 5th IEEE signal processing education workshop*. IEEE, 675–678.
- [22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.
- [23] Dezhi Hong, Quanquan Gu, and Kamin Whitehouse. 2017. High-dimensional time series clustering via cross-predictability. In *Artificial Intelligence and Statistics*. PMLR, 642–651.
- [24] HM Sajjad Hossain, MD Abdullah Al Haiz Khan, and Nirmalya Roy. 2018. De-Active: scaling activity recognition with active deep learning. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 2 (2018), 1–23.
- [25] Tsung-Yu Hsieh, Suhang Wang, Yiwei Sun, and Vasant Honavar. 2021. Explainable Multivariate Time Series Classification: A Deep Neural Network Which Learns to Attend to Important Variables As Well As Time Intervals. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. 607–615.
- [26] Jeya Vikranth Jeyakumar, Liangzhen Lai, Naveen Suda, and Mani Srivastava. 2019. SenseHAR: a robust virtual activity sensor for smartphones and wearables. In *Proceedings of the 17th Conference on Embedded Networked Sensor Systems*. 15–28.
- [27] Wenjun Jiang, Chenglin Miao, Fenglong Ma, Shuochao Yao, Yaqing Wang, Ye Yuan, Hongfei Xue, Chen Song, Xin Ma, Dimitrios Koutsonikolas, et al. 2018. Towards environment independent device free human activity recognition. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*. 289–304.
- [28] Wenchao Jiang and Zhaozheng Yin. 2015. Human activity recognition using wearable sensors by deep convolutional neural networks. In *Proceedings of the 23rd ACM international conference on Multimedia*. 1307–1310.
- [29] Matthew Keally, Gang Zhou, Guoliang Xing, Jianxin Wu, and Andrew Pyles. 2011. Pbn: towards practical activity recognition using smartphone-based body sensor networks. In *Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems*. 246–259.
- [30] Eunju Kim, Sumi Helal, and Diane Cook. 2009. Human activity recognition and pattern discovery. *IEEE pervasive computing* 9, 1 (2009), 48–53.
- [31] Youngwook Kim and Brian Toomajian. 2016. Hand gesture recognition using micro-Doppler signatures with convolutional neural network. *IEEE Access* 4 (2016), 7125–7130.
- [32] Jennifer R Kwapisz, Gary M Weiss, and Samuel A Moore. 2011. Activity recognition using cell phone accelerometers. *ACM SigKDD Explorations Newsletter* 12, 2 (2011), 74–82.
- [33] Nicholas D Lane, Petko Georgiev, and Lorena Qendro. 2015. Deeppear: robust smartphone audio sensing in unconstrained acoustic environments using deep learning. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 283–294.
- [34] Oscar D Lara and Miguel A Labrador. 2012. A survey on human activity recognition using wearable sensors. *IEEE communications surveys & tutorials* 15, 3 (2012), 1192–1209.
- [35] Bing Li, Wei Cui, Wei Wang, Le Zhang, Zhenghua Chen, and Min Wu. 2021. Two-Stream Convolution Augmented Transformer for Human Activity Recognition. (2021).
- [36] Tianxing Li, Qiang Liu, and Xia Zhou. 2016. Practical human sensing in the light. In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*. 71–84.
- [37] Xinyu Li, Yanyi Zhang, Jianyu Zhang, Shuhong Chen, Ivan Marsic, Richard A Farneth, and Randall S Burd. 2017. Concurrent activity recognition with multimodal CNN-LSTM structure. *arXiv preprint arXiv:1702.01638* (2017).
- [38] Jaime Lien, Nicholas Gillian, M Emre Karagozler, Patrick Amihoud, Carsten Schwesig, Erik Olson, Hakim Raja, and Ivan Poupyrev. 2016. Soli: Ubiquitous gesture sensing with millimeter wave radar. *ACM Transactions on Graphics (TOG)* 35, 4 (2016), 1–19.
- [39] Shengzhong Liu, Shuochao Yao, Jinyang Li, Dongxin Liu, Tianshi Wang, Huajie Shao, and Tarek Abdelzaher. 2020. GlobalFusion: A global attentional deep learning framework for multisensor information fusion. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 1 (2020), 1–27.

- [40] Chris Xiaoxuan Lu, Muhamad Risqi U Saputra, Pejjun Zhao, Yasin Almaliglu, Pedro PB de Gusmao, Changhao Chen, Ke Sun, Niki Trigoni, and Andrew Markham. 2020. MilliEgo: single-chip mmWave radar aided egomotion estimation via deep sensor fusion. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*. 109–122.
- [41] Haojie Ma, Wenzhong Li, Xiao Zhang, Songcheng Gao, and Sanglu Lu. 2019. AttnSense: Multi-level attention mechanism for multimodal human activity recognition. In *IJCAI*. 3109–3115.
- [42] Wenchi Ma, Yuanwei Wu, Zongbo Wang, and Guanghui Wang. 2018. Mdcn: Multi-scale, deep inception convolutional neural networks for efficient object detection. In *2018 24th International Conference on Pattern Recognition (ICPR)*. IEEE, 2510–2515.
- [43] Akhil Mathur, Tianlin Zhang, Sourav Bhattacharya, Petar Velickovic, Leonid Joffe, Nicholas D Lane, Fahim Kawsar, and Pietro Lió. 2018. Using deep data augmentation training to address software and hardware heterogeneities in wearable and smartphone sensing devices. In *2018 17th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*. IEEE, 200–211.
- [44] Alan Mazankiewicz, Klemens Böhm, and Mario Bergés. 2020. Incremental Real-Time Personalization in Human Activity Recognition Using Domain Adaptive Batch Normalization. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 4 (2020), 1–20.
- [45] Sameera Palipana, David Rojas, Piyush Agrawal, and Dirk Pesch. 2018. FallDeFi: Ubiquitous fall detection using commodity Wi-Fi devices. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 4 (2018), 1–25.
- [46] Daehyung Park, Yuuna Hoshi, and Charles C Kemp. 2018. A multimodal anomaly detector for robot-assisted feeding using an lstm-based variational autoencoder. *IEEE Robotics and Automation Letters* 3, 3 (2018), 1544–1551.
- [47] Juha Parkka, Miikka Ermes, Panu Korpipää, Jani Mantyjarvi, Johannes Peltola, and Ilkka Korhonen. 2006. Activity classification using realistic data from wearable sensors. *IEEE Transactions on information technology in biomedicine* 10, 1 (2006), 119–128.
- [48] Liangying Peng, Ling Chen, Zhenan Ye, and Yi Zhang. 2018. Aroma: A deep multi-task learning based simple and complex human activity recognition method using wearable sensors. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 2 (2018), 1–16.
- [49] Akarsh Prabhakara, Vaibhav Singh, Swarun Kumar, and Anthony Rowe. 2020. Osprey: A mmWave approach to tire wear sensing. In *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services*. 28–41.
- [50] Qifan Pu, Sidhant Gupta, Shyamnath Gollakota, and Shwetak Patel. 2013. Whole-home gesture recognition using wireless signals. In *Proceedings of the 19th annual international conference on Mobile computing & networking*. 27–38.
- [51] Yang Qifan, Tang Hao, Zhao Xuebing, Li Yin, and Zhang Sanfeng. 2014. Dolphin: Ultrasonic-based gesture recognition on smartphone platform. In *2014 IEEE 17th International Conference on Computational Science and Engineering*. IEEE, 1461–1468.
- [52] Xin Qin, Yiqiang Chen, Jindong Wang, and Chaohui Yu. 2019. Cross-dataset activity recognition via adaptive spatial-temporal transfer learning. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 4 (2019), 1–25.
- [53] Valentin Radu, Catherine Tong, Sourav Bhattacharya, Nicholas D Lane, Cecilia Mascolo, Mahesh K Marina, and Fahim Kawsar. 2018. Multimodal deep learning for activity and context recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 4 (2018), 1–27.
- [54] Daniel Roggen, Alberto Calatroni, Mirco Rossi, Thomas Holleczek, Kilian Förster, Gerhard Tröster, Paul Lukowicz, David Bannach, Gerald Pirk, Alois Ferscha, et al. 2010. Collecting complex activity datasets in highly rich networked sensor environments. In *2010 Seventh international conference on networked sensing systems (INSS)*. IEEE, 233–240.
- [55] H Rubiani, E Samsoleh, and S Fitri. 2021. Activity recognition (AR) to detect dementia using wi-fi based wireless sensor network with receive signal strength indicator (RSSI) method. In *IOP Conference Series: Materials Science and Engineering*, Vol. 1115. IOP Publishing, 012079.
- [56] Shunuo Shang, Qingyao Luo, Jinjin Zhao, Ran Xue, Weihao Sun, and Nan Bao. 2021. LSTM-CNN network for human activity recognition using WiFi CSI data. In *Journal of Physics: Conference Series*, Vol. 1883. IOP Publishing, 012139.
- [57] Taoran Sheng and Manfred Huber. 2020. Weakly supervised multi-task representation learning for human activity analysis using wearables. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 2 (2020), 1–18.
- [58] Ali H Shoeb and John V Guttag. 2010. Application of machine learning to epileptic seizure detection. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*. 975–982.
- [59] Stephan Sigg, Shuyu Shi, Felix Buesching, Yusheng Ji, and Lars Wolf. 2013. Leveraging RF-channel fluctuation for activity recognition: Active and passive systems, continuous and RSSI-based signal features. In *Proceedings of International Conference on Advances in Mobile Computing & Multimedia*. 43–52.
- [60] Brij N Singh and Arvind K Tiwari. 2006. Optimal selection of wavelet basis function applied to ECG signal denoising. *Digital signal processing* 16, 3 (2006), 275–287.
- [61] Smith and O. Julius. 2007. Mathematics of the Discrete Fourier Transform (DFT) with Audio Applications.
- [62] Huan Song, Deepta Rajan, Jayaraman Thiagarajan, and Andreas Spanias. 2018. Attend and diagnose: Clinical time series analysis using attention models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.
- [63] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander Alemi. 2017. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 31.
- [64] Pierre Thodoroff, Joelle Pineau, and Andrew Lim. 2016. Learning robust features using deep learning for automatic seizure detection. In *Machine learning for healthcare conference*. PMLR, 178–190.
- [65] Yonglong Tian, Guang-He Lee, Hao He, Chen-Yu Hsu, and Dina Katabi. 2018. RF-based fall monitoring using convolutional neural networks. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 3 (2018), 1–24.
- [66] Alexandros T Tzallas, Markos G Tsipouras, and Dimitrios I Fotiadis. 2009. Epileptic seizure detection in EEGs using time–frequency analysis. *IEEE transactions on information technology in biomedicine* 13, 5 (2009), 703–710.
- [67] Kristof Van Laerhoven. 2001. Combining the self-organizing map and k-means clustering for on-line classification of sensor data. In *International Conference on Artificial Neural Networks*. Springer, 464–469.
- [68] Saiwen Wang, Jie Song, Jaime Lien, Ivan Poupyrev, and Otmar Hilliges. 2016. Interacting with soli: Exploring fine-grained dynamic gesture recognition in the radio-frequency spectrum. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. 851–860.
- [69] Wei Wang, Alex X Liu, and Ke Sun. 2016. Device-free gesture tracking using acoustic signals. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*. 82–94.
- [70] Zhiguang Wang, Weizhong Yan, and Tim Oates. 2017. Time series classification from scratch with deep neural networks: A strong baseline. In *2017 International joint conference on neural networks (IJCNN)*. IEEE, 1578–1585.
- [71] Bo Wei, Wen Hu, Mingrui Yang, and Chun Tung Chou. 2015. Radio-based device-free activity recognition with radio frequency interference. In *Proceedings of the 14th International Conference on Information Processing in Sensor Networks*. 154–165.
- [72] Ian H Witten and Eibe Frank. 2002. Data mining: practical machine learning tools and techniques with Java implementations. *Acm Sigmod Record* 31, 1 (2002), 76–77.
- [73] Rui Xi, Mengshu Hou, Mingsheng Fu, Hong Qu, and Daibo Liu. 2018. Deep dilated convolution on multimodality time series for human activity recognition. In *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–8.
- [74] Hongfei Xue, Wenjun Jiang, Chenglin Miao, Fenglong Ma, Shiyang Wang, Ye Yuan, Shuochao Yao, Aidong Zhang, and Lu Su. 2020. DeepMV: Multi-view deep learning for device-free human activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 1 (2020), 1–26.
- [75] Jie Yang, Simon Sidhom, Gayathri Chandrasekaran, Tam Vu, Hongbo Liu, Nicolae Ceca, Yingying Chen, Marco Gruteser, and Richard P Martin. 2011. Detecting driver phone use leveraging car speakers. In *Proceedings of the 17th annual international conference on Mobile computing and networking*. 97–108.
- [76] Shuochao Yao, Shaohan Hu, Yiran Zhao, Aston Zhang, and Tarek Abdelzaher. 2017. Deepsense: A unified deep learning framework for time-series mobile sensing data processing. In *Proceedings of the 26th International Conference on World Wide Web*. 351–360.
- [77] Shuochao Yao, Ailing Piao, Wenjun Jiang, Yiran Zhao, Huajie Shao, Shengzhong Liu, Dongxin Liu, Jinyang Li, Tianshi Wang, Shaohan Hu, et al. 2019. Sfnet: Learning sensing signals from the time-frequency perspective with short-time fourier neural networks. In *The World Wide Web Conference*. 2192–2202.
- [78] Siamak Yousefi, Hirokazu Narui, Sankalp Dayal, Stefano Ermon, and Shahrokh Valae. 2017. A survey on behavior recognition using WiFi channel state information. *IEEE Communications Magazine* 55, 10 (2017), 98–104.
- [79] Xuchao Zhang, Yifeng Gao, Jessica Lin, and Chang-Tien Lu. 2020. Tapnet: Multivariate time series classification with attentional prototypical network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 6845–6852.
- [80] Xiaokang Zhou, Wei Liang, I Kevin, Kai Wang, Hao Wang, Laurence T Yang, and Qun Jin. 2020. Deep-learning-enhanced human activity recognition for Internet of healthcare things. *IEEE Internet of Things Journal* 7, 7 (2020), 6429–6438.