Information Fusion 80 (2022) 1–22

Contents lists available at ScienceDirect

Information Fusion

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

Meta-learning meets the Internet of Things: Graph prototypical models for sensor-based human activity recognition

Wenbo Zheng [a,b], Lan Yan [a,c], Chao Gou [d,1], Fei-Yue Wang [a,*,2]

[a] State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
[b] School of Software Engineering, Xi'an Jiaotong University, Xi'an 710049, China
[c] School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100190, China
[d] School of Intelligent Systems Engineering, Sun Yat-sen University, Guangzhou 510275, China

ARTICLE INFO

Keywords:
Meta-learning
Internet of Things
Graph model
Attention mechanisms

ABSTRACT

With the rapid growth of the Internet of Things (IoT), smart systems and applications are equipped with an increasing number of wearable sensors and mobile devices. These sensors are used not only to collect data but, more importantly, to assist in tracking and analyzing the daily human activities. Sensor-based human activity recognition is a hotspot and starts to employ deep learning approaches to supersede traditional shallow learning that rely on hand-crafted features. Although many successful methods have been proposed, there are three challenges to overcome: (1) deep model's performance overly depends on the data size; (2) deep model cannot explicitly capture abundant sample distribution characteristics; (3) deep model cannot jointly consider sample features, sample distribution characteristics, and the relationship between the two. To address these issues, we propose a meta-learning-based graph prototypical model with priority attention mechanism for sensor-based human activity recognition. This approach learns not only sample features and sample distribution characteristics via meta-learning-based graph prototypical model, but also the embeddings derived from priority attention mechanism that mines and utilizes relations between sample features and sample distribution characteristics. What is more, the knowledge learned through our approach can be seen as a priori applicable to improve the performance for other general reasoning tasks. Experimental results on fourteen datasets demonstrate that the proposed approach significantly outperforms other state-of-the-art methods. On the other hand, experiments of applying our model to two other tasks show that our model effectively supports other recognition tasks related to human activity and improves performance on the datasets of these tasks.

1. Introduction

Over the last decade, with the Internet of Things (IoT) advances, universal sensing for the purpose of extracting knowledge from data obtained from ubiquitous sensors has become a very active area of research [1,2]. In particular, human activity recognition (HAR) in IoT has attracted a lot of attention in recent years, due to the use of wearable devices such as smart wristbands and smartphones, as well as their capability of real-time information capturing [3]. **Sensor-Based Human Activity Recognition** focuses on predicting participants' activities from the low-level sensor inputs. An example constructed from the OPPORTUNITY dataset [4] is illustrated in Fig. 1, where participants wear sensors on their right and left feet as well as on their left hand. Using the signal information collected from these sensors, our model predicts the participant's activity (i.e., walking or standing?). In general, HAR can be considered as a classical pattern recognition (PR) problem.

Existing HAR methods on PR can be divided into two categories: feature-engineering-based and deep learning-based [5]. The former type of methods aim to extract aspects of the information from each sensor-reading segment, such as statistical information (e.g., the overall shape and spatial information). These methods often require domain knowledge to manually design the appropriate functionality for a particular application, which is a time-consuming and laborious work [5]. The latter type of methods have continued to provide excellent performance in many fields (e.g., computer vision and natural

* Corresponding author.
E-mail addresses: zwb2017@stu.xjtu.edu.cn (W. Zheng), Yanlan2017@ia.ac.cn (L. Yan), gouchao@mail.sysu.edu.cn (C. Gou), feiyue.wang@ia.ac.cn (F. Wang).
[1] IEEE Member.
[2] IEEE Fellow.

https://doi.org/10.1016/j.inffus.2021.10.009
Received 27 November 2020; Received in revised form 16 September 2021; Accepted 31 October 2021
Available online 13 November 2021

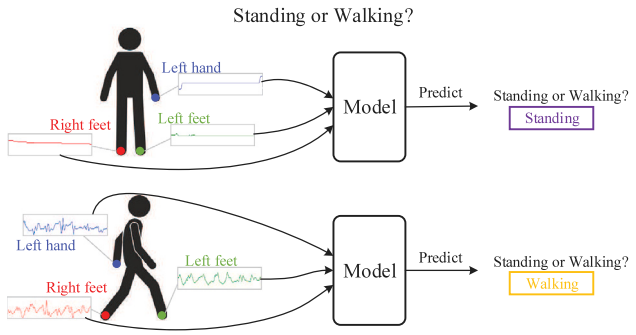

Fig. 1. Illustration of sensor-based human activity recognition task. Sensors are used to collect the data of left hand, left feet, right feet in participants. The answers are the output of our model when inputting corresponding data.

language processing). Recently, their ability to automatically extract advanced features makes deep learning-based methods largely mitigate the shortcomings of feature-engineering-based methods. Different neural networks have been proposed to gain different kinds of information. Notably, the deep feed network (DFN) is used to extract high-level features [6], regardless of time or space information. Convolutional neural network (CNN) [7–9] is used to extract the exact position or local translational invariant features relative to specific patterns within the data segment. Recursive neural network (RNN) [10–12] is suitable for utilizing time dependence within an activity sequence. The state-of-the-art technology based on HAR is a combination of these three basic models. Although deep learning-based approaches can learn powerful features to represent time and/or spatial information under sensor data, they are not able to [13,14] explicitly consider plentiful sample distribution characteristics [15–19]. For example, since data from some emergency or unexpected activities (e.g., accidental falls) are particularly difficult to obtain, these unexpected activities can easily be mistaken by deep models, as regular activities. On the other hand, existing methods make the assumption that training and test data are independent and equally distributed and do not model the distribution of the sample. However, this is impractical since the sensor data used for human activity recognition is heterogeneous. In real-world scenarios in which sensor devices are installed without restrictions, it is easy to observe the difference in distribution between training and test data, and the sudden drop in recognition accuracy raises concerns. In other words, various sensor devices are opportunistically configured in the human body or environment, and the composition and layout of the sensors greatly affect the data collected. Even when data is collected in the same setting, and only the sensor instances are different, for example, a person replaces his smartphone with a new one, the recognition accuracy still declines soon [20]. Besides, the training of deep learning-based approaches depends on the large amounts of sensor-based [21], and this kind of approaches may cause over-fit issues [22] as well as poor generalization ability [23]. To this end, some researchers [24] recommend that the samples may be taken into account along with the relations about their distribution characteristics.

In general, there are three main challenges in the sensor-based human activity recognition tasks:

(1) Deep-learning-based approaches are overly dependent on data size and difficult to be generalized to other human activity tasks [5,25].

(2) Deep-learning-based approaches cannot [13,14] explicitly take into account luxuriant sample distribution characteristics [15,19].

(3) Sample features, sample distribution characteristics, and the relationship between the two must be jointly considered to predict the human activity accurately [5,25].

An ability of learning quickly and generalizing from only a few samples is a key character [26] of human intelligence, as humans can make use of a priori knowledge gained from previous learning experiences. Many deep-learning-based methods have been successful prominently on many tasks, including computer vision and natural language processing. However, these methods still require a lot of data with labels, as these methods assume that tasks are mutually independent, and they are trained without any prior knowledge specific to the task. *How can we extract prior knowledge and apply it on future unknown tasks with limited data?* To this end, meta-learning is presented and has emerged as a promising approach [27]. Generally speaking, the basic meta-learning framework consists of two components: meta-level learners and basic-level learners. Basic-level learners are usually intended for particular tasks, for instance, regressions and classifications. The meta-learner, the key to meta-learning, is geared towards learning a priori knowledge across different tasks. It allows the transfer of prior knowledge to basic-level learners to aid fast adaptation in similar unknown tasks. *Therefore, in this paper, we focus on meta-learning that aims to learn to classify unseen categories in samples. By the strategy of meta-learning or few-shot learning, we are able to address the first challenge, i.e., the performance highly depends on the size of data.*

Graph neural network, a powerful model, can implicitly cover many data structures (e.g., lists, trees) and contain prior combinatorial data. Meta-learning based GNNs are used to build complete graphs where each node feature is concatenated with a corresponding class label, and then the node features are updated to propagate the label information through the attention mechanism of the graph network. But this method explicitly focuses on pairwise characteristics such as node labels or edge labels. To this end, the design of GNNs [28,29] has thought about sufficient distribution characteristics of a large number of samples. It offers a possible solution to *overcome the second challenge of taking account of sample distribution characteristics*. Therefore, *how to construct a graph model to explicitly capture the sample luxuriant distribution characteristics?*

The attention model has become an essential concept in neural networks and has been studied in many application areas. The human biological system can best explain the intuition behind the attention. For example, our visual processing system tends to selectively focus on certain parts of an image and ignore other irrelevant information that can help us perceive. Similarly, in some problems involving language, speech, or vision, many parts of the input may be more relevant than others. The attention model introduces this notion of relevance by allowing the model to focus dynamically on only those parts of the input that contribute to the effective performance of the task at hand. Inspired by priority map in human attention [30,31], which combines top-down and bottom-up activity as well as task relevance, we intend to investigate attention strategies that could learn relationship correspond very well to human intuition [32,33], to *overcome the third challenge of considering sample features, sample distribution characteristics, and the relation between the two*. Thus, *how to design priority attention in a model like the priority map in human attention?*

To tackle all the aforementioned problems and challenges, *motivated by these observations*, we propose a novel, robust, and effective meta-learning-based graph prototypical model with priority attention mechanism for sensor-based human activity recognition, which mines sample and its distribution as well as their relevance. Specifically, we first design a dual complete meta-learning-based graph network, called graph prototypical model, including a sample graph and a distribution graph. Then, through utilizing the theory of positive pointwise mutual information and optimal transport, our model with dual graph architecture propagates label information from labeled examples to unlabeled examples within several update generations. Next, upon dual graphs, we design a novel attention mechanism called priority attention mechanism, similar to priority map in human attention, to extract the sample-level, distribution-level, relation embeddings from sample features, sample distribution characteristics, and their combinations simultaneously, and use the attention mechanism to learn adaptive weights of the embeddings. Finally, our schema uses the embeddings of attention representations with dual graph information

to achieve the prediction of human activity. Experimental results that our approach achieves significantly higher performance than the state-of-the-art methods in sensor-based human activity recognition, using fourteen datasets: the UTD-MHAD dataset [34], the OPPORTUNITY dataset [4], the DSADS dataset [35], the MHealth dataset [36], the WISDM Activity Prediction dataset [37], the PAMAP2 dataset [38], the UCI HAR dataset [39], the Daphnet Freezing of Gait dataset [40], the HHAR dataset [41], the UniMiB-SHAR dataset [42], the MobiAct dataset [43], the MotionSense dataset [44], the UCI HAPT dataset [45], and the USC-HAD dataset [46]. Importantly, we apply our model to two other human activity recognition tasks, which are the WIFI gesture recognition and the cross-modal human action understanding. We report our and other state-of-the-art approaches' results on the Widar 3.0 dataset [47] for the WIFI gesture recognition task, and report our and other state-of-the-art approaches' results on the MMAct dataset [48] for the cross-modal human action understanding task.

Our contributions can be briefly summarized as follows:

☼ We propose a novel, robust, and effective meta-learning-based graph prototypical model with priority attention mechanism for sensor-based human activity recognition, which mines and exploits the features and relations in the whole process, concerning sample and its distribution characteristic.

☼ We present a novel meta-learning-based graph prototypical model to learn the discriminative features cross different datasets. The qualitative discussion demonstrates that this strategy achieves competitive performance over other meta-based methods.

☼ We design a novel attention mechanism called priority attention mechanism, which significantly helps the proposed approach to improve the performance of the task of sensor-based human activity recognition. Through joint learning with the graph prototypical model, our model is able to generalize to other human activity recognition tasks.

☼ The experimental results show that the method is robust and superior to the existing sensor-based human activity recognition method on fourteen datasets.

The rest of this paper is organized as follows: In Section 3, we start with the details of the proposed approach. In Section 4, we conduct experiments to verify the validity of the proposed approach. Finally, the conclusions of this work are presented in Section 5.

2. Related work

Human activity recognition problem is a topic of continuous interest in the Internet of Things. There are many previous works that use machine learning methods. Most of these works are based on feature extraction techniques, such as time–frequency transformations, statistical methods and symbolic representations. For example, the HDS-SP descriptor is proposed [49] consists of both spatial and temporal information from specific viewpoint for HAR. Wu et al. [10] propose dynamic pose images (DPIs) as a compact pose feature for HAR. Mian et al. [50] propose an atomic visual unit called Skepxel to improve CNN architectures without any re-sampling. Jamel et al. [51] propose a parallel approximated clustering approach to handle the computational cost of big data from HAR. Yang et al. [52] propose a linear model for activity recognition based on the state-space method. Kim et al. [53] propose an efficient noise cancellation approach that could enhance conventional spectral subtraction, using an advanced voice activity detection (VAD). Cho et al. [54] propose a method that uses a clustering algorithm for learning data and assign labels to each cluster according to the maximum likelihood for HAR. Obviously, these extracted features are carefully designed and heuristic. In other words, these features are not generic and do not address the problem of distinguishable features for human activities in different scenarios.

In recent years, deep learning has achieved significant success in advanced modeling of abstractions of complex data in many areas of artificial intelligence such as computer vision and natural language processing. There has been a mushrooming of research related to deep learning approaches to the task of human activity recognition (HAR). Due to the strong ability of Convolutional Neural Networks (CNNs) for learning hierarchical representation, some methods, e.g. JSR [55], JMR [55], SR-TSL [56] and etc. [9,34,57–59], exploit CNNs to recognize human actions. Tan et al. [11] propose a novel model called HSR-TSL, which uses hierarchical spatial reasoning and temporal stack learning network to obtain features. Ji et al. [60] propose a solution called BGCLSTM that combines graph convolution and LSTM to model spatio-temporal dynamics. Domnic et al. [8] propose a deep ensemble framework called HNet that combines both CNN and LSTM. Miao et al. [61] propose DDNN is a unified end-to-end trainable deep learning model, which is able to learn different types of powerful features for activity recognition in an automated fashion. Zhang et al. [62] propose a novel dual attention method called DanHAR, which introduces the framework of blending channel attention and temporal attention on a CNN. Zhang et al. [63] propose a lightweight CNN called and Lego-CNN using re-designed Lego filters. Lee et al. [64] propose a novel deep learning architecture called EmbraceNet, whose main components are docking layers and an embracement layer with the representations of multiple modalities. Ali et al. [65] propose a self-attention based neural network model that foregoes recurrent architectures and utilizes different types of attention mechanisms to generate higher dimensional feature representation. Wang et al. [66] propose a deep neural network that combines convolutional layers with long short-term memory. Le et al. [67] propose a method based on a capsule network named SensCapsNet, which is designed to be suitable for spatial–temporal data coming from wearable sensors. Zhang et al. [68] propose a layer-wise convolutional neural networks (CNN) with local loss. Chen et al. [69] propose a novel feature incremental learning method, namely the Feature Incremental Random Forest (FIRF), to improve the performance of an existing model with a small amount of data on newly appeared features. Chen et al. [70] propose an effective class incremental learning method, named Class Incremental Random Forests (CIRF), to enable existing activity recognition models to identify new activities. Yu et al. [71] propose a discriminative adversarial multi-view network (DAMUN) which includes a novel multi-view representation module and a Siamese adversarial training framework. Gumaei et al. [72] propose a new deep learning-based human activity recognition (DL-HAR) framework for improving the primary care of patients in an uncritical stage.

In summary, there are different drawbacks of existing approaches: firstly, feature engineering-based approaches, while being able to extract meaningful features, e.g., statistical or structural information from segments, usually require manually designed features for the characteristics of different datasets; secondly, deep learning models, while being able to automatically learn temporal and/or spatial features from sensor data, require a large number of samples and do not capture statistical information (e.g., distribution characteristics). To address these challenges, we propose to design a dual graph-based graph prototype model, where one graph is used to represent the features of the samples and the other graph is employed to represent the statistical information about the sample features. In addition, we employ a meta-learning strategy to reduce the rely of the deep model on large-scale samples as much as possible.

3. Methodology

In this section, we provide the background of our task, followed by introducing the proposed algorithm in detail. Following the strategy of classical meta-learning [27], a novel, robust, and effective model, consisting of **Graph Prototypical Model** and **Priority Attention Mechanism**, is proposed. Considering to the luxuriant samples and their distribution characteristics, we build the graph prototypical model to play a role as the representation of samples and their distribution. Further, upon the proposed graph prototypical model, we

Table 1
Notations and explanations.

Explanation	Notation
The number of generations	gen
Sample graph	G^S_{gen}
Sample feature	V^S_{gen}
Sample-level similarities	E^S_{gen}
An element of V^S_{gen}	$v^S_{gen,i}$
An element of E^S_{gen}	$e^S_{gen,ij}$
An embedding of G^S_{gen}	$Z^{(gen)}_{sample}$
The final sample-level embedding	Z_S
Distribution graph	G^D_{gen}
Distribution-level feature	V^D_{gen}
Distribution-level similarities	E^D_{gen}
An element of V^D_{gen}	$v^D_{gen,i}$
An element of E^D_{gen}	$e^D_{gen,ij}$
An embedding of G^D_{gen}	$Z^{(gen)}_{distribution}$
The final distribution-level embedding	Z_D
A sample-related embedding from G^D_{gen}	$Z^{(gen)}_{d\to s}$
A distribution-related embedding from G^S_{gen}	$Z^{(gen)}_{s\to d}$
The relation embedding	Z_{Relation}
The final priority attention embedding	Z

constitute the priority attention mechanism to work on mining the sample features, sample distribution characteristics, and the relationship between the two. The overall framework of the proposed algorithm is shown in Fig. 2. In addition, the notations of the proposed model are shown in Table 1.

3.1. Problem setup

In this subsection, we introduce the settings of our model (i.e., few-shot learning). Then, we present the optimal transport problem used to form the similarity between the distribution characteristics of samples in our model.

Few-Shot Problem We consider the task of sensor-based human activity recognition as the *few-shot* classification. Generally speaking, few-shot learning-based meta-learning typically uses episodic training strategies [27,29,73]. In each episode, the model based on meta-learning is trained on a meta-task, which can be viewed as a classification task [29,73,74]. During training, the tasks were randomly selected from the training dataset in the episodes. During the model evaluation, the tasks were selected from a separate test dataset consisting of new classes not included in the training dataset. There are three datasets: a training set, a support set, and a testing set. The support set and testing set share the same label space, but the training set has its own label space that is disjoint with the support/testing set. If the support set contains K labeled examples for each of C unique classes, the target few-shot problem is known as the *C-way K-shot*. In C-way-K-shot few-shot learning, the model based on meta-learning is trained on some tasks sampled from the training dataset, and each task contains a query set and a support set. The task includes C unique class labels, and the support set consists of K labeled data per class. Utilizing the support set, the model learns to predict the labels in the query set. After training, the model based on meta-learning is then evaluated on new tasks sampled from the test set. Similar to the training tasks, each new task consists of C unique class labels with K images (in the support set) each. However, to assess how well the meta-learner performs on new tasks, the classes in the test dataset are not overlapping with the classes in the training set.

The whole model consists of two phases: meta-training and meta-testing. In meta-training, our training data $D_{\text{meta-train}} = \{(x_i, y_i)\}_{i=1}^{\mathcal{N}}$ from a set of classes C_{train} are used for training a classifier, where x_i is a data point, $y_i \in C_{train}$ is the corresponding label, and $\mathcal{N}$ is the number of training samples. In meta-testing, given training set $D_{\text{meta-train}}$, a support set of $\mathcal{K}$ (i.e., $\mathcal{K} = C_{test} \times \mathcal{K}_{sup}$) labeled examples $D_{\text{support}} = \{(x_j, y_j)\}_{j=1}^{\mathcal{K}}$ from a set of new classes C_{test} with $\mathcal{K}_{sup}$ samples for each class is given (i.e., the C_{test}-way $\mathcal{K}_{sup}$-shot setting), where x_j is a data point for testing, and $y_j \in C_{test}$ is the corresponding label. The goal is to predict the labels of a query set $D_{\text{query}} = \{(x_j)\}_{j=\mathcal{K}+1}^{\mathcal{K}+\mathcal{Q}}$, where $\mathcal{Q}$ is the number of queries. This split strategy of training and support set aims to simulate the support and query set encountered at test time. Further, we use the meta-learning on the training set to transfer the extracted knowledge to the support set. It aims to perform the model's learning on the support set better and classify the query set more successfully.

Optimal Transport Problem The optimal mass transport problem is to find a map that minimizes the inter-domain transportation cost [75]. On the basis of the statistical view of machine learning, we can transform one space for all possible input data to another space. We assume the two sets of samples from two domains. For presentation simplicity, we denote these two domains as $\mathbb{D}_\mu$ and $\mathbb{D}_\tau$. Given two probability vectors μ (from $\mathbb{D}_\mu$) and τ (from $\mathbb{D}_\tau$), we denote the transport polytope $U(\mu, \tau)$ as follows:

$$U(\mu, \tau) := \{\pi | \pi \mathbf{1} = \mu, \pi^T \mathbf{1} = \tau\} \tag{1}$$

where $U(\mu, \tau)$ is a convex, closed and bounded set containing joint probability distributions with μ and τ as marginals. The space where μ and τ are located is π.

Assume π has an equal total measure as

$$\int_\pi d\mu = \int_\pi d\tau \tag{2}$$

We aim to find a region to its own form mapping (diffeomorphism), $\mathcal{T} : (\pi, d\mu) \to (\pi, d\tau)$. According to the theory of Gu et al. [76,77], $\mathcal{T} : (\pi, \mu) \to (\pi, \tau)$ is an unique optimal transport mapping. We transform initial probability distribution μ into target probability distribution τ, $\mathcal{T} \otimes \mu = \tau$. At the same time, minimizing the transport cost,

$$\text{Cost}(\mu, \tau) := \min_{\mathcal{T}\otimes\mu=\tau} \int_\pi |x - \mathcal{T}(x)|^2 d\mu \tag{3}$$

where the cost function $\text{Cost}(\mu, \tau)$ measures the cost of moving a unit mass from μ to τ.

We define optimal transport (OT) distance $OT(\cdot)$ as follows:

$$OT(\mu, \tau) := \min_{\pi \in U(\mu,\tau)} \langle \pi, \text{Cost}(\mu, \tau) \rangle \tag{4}$$

where the Frobenius dot product $\langle \cdot, \cdot \rangle_F$, for matrix A and B, $\langle A, B \rangle_F = Tr(A^T B)$, where $Tr(\cdot)$ the function of the trace of the matrix $\cdot$.

However, it is equivalent to solve a linear programming problem to compute OT distances, for which a special algorithm exists with a time complexity of $O(n^3 \log n)$ between two n-bin, normalized histograms. Despite this, it is still too expensive in large-scale settings. To this end, we consider the **Regularized Optimal Transport** (ROT) distance $ROT(\cdot)$ as follows:

$$ROT_\lambda(\mu, \tau) := \min_{\pi \in U(\mu,\tau)} \{\langle \pi, \text{Cost}(\mu, \tau) \rangle - H(\pi)/\lambda\} \tag{5}$$

where $\lambda > 0$ is the regularization parameter controlling the trade-off between sparsity and uniformity of π and $H(\pi)$ is the discrete entropy denoted as:

$$H(\pi) = -\sum \pi(\log \pi - 1) \tag{6}$$

Finally, we can solve the function Eq. (5) using POT library [3][78]. Besides, following the advice of Alaya et al.'s method [79], we have set $\lambda = 1$ unless otherwise specified.

[3] https://github.com/mzalaya/screenkhorn.

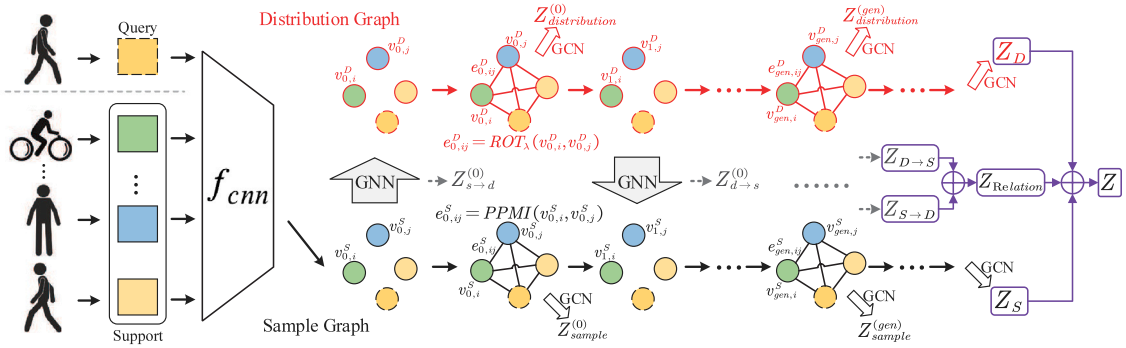

Fig. 2. The overall framework of our proposed approach. In this illustration, we take a 3-way-1-shot task as an example. We design the **Graph Prototypical Model** to represent the sample feature and the sample distribution characteristic. The support and query embeddings obtained from convolutional neural networks (CNN)-based feature extractor are handled to the dual graph (a sample graph and a distribution graph). We use two graph neural networks (GNNs) to represent an interconversion of sample graph to distribution graph. During the construction of dual graph, we design the **Priority Attention Mechanism** that permits sample-level embeddings to propagate not only in sample space, but also in distribution space, and the most correlated information with sample label should be extracted from both of these two spaces.

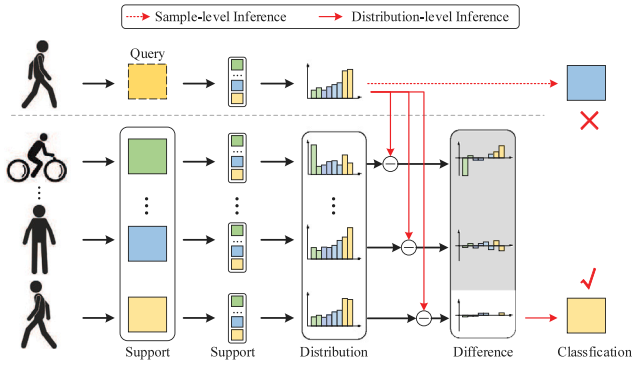

Fig. 3. The illustration of graph prototypical models. Our proposed graph prototypical model adopts contrastive comparisons between each sample with support samples to produce distribution characteristic representation. Then it incorporates distribution-level comparisons with sample-level comparisons when classifying the query sample.

3.2. Graph prototypical models

We design two graphs (i.e., **Sample Graph** and **Distribution Graph**) to explicitly represent the sample feature and the sample distribution characteristic. The whole model shown in Fig. 2, contains the sample graph $G_{gen}^{S} = (V_{gen}^{S}, E_{gen}^{S})$ and the distribution graph $G_{gen}^{D} = (V_{gen}^{D}, E_{gen}^{D})$, where gen means the number of generations. As indicated in Fig. 3, we firstly use a convolutional backbone network [80] to extract the feature of samples and have access to these features for computing the sample-level similarities E_{gen}^{S}. Then, we transfer the sample-level similarities E_{gen}^{S} to build the distribution graph $G_{gen}^{D} = (V_{gen}^{D}, E_{gen}^{D})$. Thirdly, we aggregate the G_{gen}^{D}'s node features to initialize the V_{gen}^{D}, following the position order in G_{gen}^{S}. We use the edge feature E_{gen}^{D} in G_{gen}^{D} to represent the distribution-level similarities between the node features. At last, we transfer the distribution-level similarities E_{gen}^{D} to build more discriminative representations of node features V_{gen}^{S}. Repeat the above process gen times.

All in all, the update process of our model is as follows: $E_{gen}^{S} \to V_{gen}^{D} \to E_{gen}^{D} \to V_{gen}^{S} \to E_{gen+1}^{S}$, in which gen means gen-th generation. E_{gen}^{S}, V_{gen}^{S}, V_{gen}^{D}, E_{gen}^{D} are formulated as follows: $E_{gen}^{S} = \{e_{gen,ij}^{S}\}_{i=1,j=1}^{\mathcal{K}+\mathcal{Q}}$, $V_{gen}^{S} = \{v_{gen,i}^{S}\}_{i=1}^{\mathcal{K}+\mathcal{Q}}$, $V_{gen}^{D} = \{v_{gen,i}^{D}\}_{i=1}^{\mathcal{K}+\mathcal{Q}}$, $E_{gen}^{D} = \{e_{gen,ij}^{D}\}_{i=1,j=1}^{\mathcal{K}+\mathcal{Q}}$.

❶ Sample Graph

We design the sample-level graph $G_{gen}^{S} = (V_{gen}^{S}, E_{gen}^{S})$. Firstly, we use convolutional neural networks (CNN) to extract the feature of samples. We define these sample features as nodes in G_{gen}^{S}. Considering that positive pointwise mutual information (PPMI) requires prior knowledge of the distribution of samples and samples transformed maintain the entropy of the information, we then use PPMI to get the sample-level similarities between nodes. We define these sample-level similarities as the edges in G_{gen}^{S}. Finally, we update G_{gen}^{S} according to the above rules.

① **Initialization** We use the feature extraction function $f_{cnn}(\cdot)$ to initialize/output $v_{0,i}^{S}$, and for each sample x_i. The above operation can be expressed mathematically as follows:

$$v_{0,i}^{S} = f_{cnn}(x_i) \tag{7}$$

where $f_{cnn}(\cdot)$ means the function of the convolutional neural networks. In this paper, we use the EfficientNet [81] as the feature extractor.

② **Sample-Level Similarity** Sample-level similarity is denoted as each edge in G_{gen}^{S}. We consider the matrix of positive pointwise mutual information (PPMI) [82] between sample features in G_{gen}^{S} as $e_{0,ij}^{S}$ initialized as follows :

$$e_{0,ij}^{S} = PPMI(v_{0,i}^{S}, v_{0,j}^{S}) \tag{8}$$

where $PPMI(\cdot)$ means the function of the positive pointwise mutual information; $v_{0,i}^{S}$ and $v_{0,j}^{S}$ mean the sample features of sample x_i and x_j, respectively.

Following the definition of PPMI [82], we can use the matrix of pointwise mutual information (PMI) [83] to compute it as follows:

$$PPMI(v_{0,i}^{S}, v_{0,j}^{S}) = \max(0, PPI(v_{0,i}^{S}, v_{0,j}^{S})) \tag{9}$$

where $PPI(\cdot)$ means the function of the pointwise mutual information. In particular, the above operation can be mathematical as follows:

$$PPI(v_{0,i}^{S}, v_{0,j}^{S}) = \log \frac{p(v_{0,i}^{S}, v_{0,j}^{S})}{p(v_{0,i}^{S})p(v_{0,j}^{S})} \tag{10}$$

where we denote the empirical probability distribution of sample features $v_{0,i}^S$ as $p(v_{0,i}^S)$, the probability of $v_{0,j}^S$ as $p(v_{0,j}^S)$, and the joint distribution by $p(v_{0,i}^S, v_{0,j}^S)$.

The PMI matrix is symmetric definite positive and can be expressed via singular value decomposition as:

$$PPI(v_{0,i}^S, v_{0,j}^S) = U \cdot \sum \cdot U^T \tag{11}$$

where $\sum$ is a diagonal matrix with the eigenvalues in the diagonal. $\mathbb{E} = U\sqrt{\sum}$ is considered as a transformation matrix so that the $PPMI(v_{0,i}^S, v_{0,j}^S) = \mathbb{E} \cdot \mathbb{E}^T$.

③ **Update Procedure** For gen ($gen > 0$) generations, the update process of edge in G_{gen}^S is as follows:

$$e_{gen,ij}^S = PPMI(v_{gen,i}^S, v_{gen,j}^S) \times e_{gen-1,ij}^S \tag{12}$$

where $v_{gen,i}^S$ and $v_{gen,j}^S$ mean the features of samples x_i and x_j, with gen-th generation, respectively; $e_{gen,ij}$ means the edge with gen-th generation in G_{gen}^S; $e_{gen-1,ij}$ means the edge with $gen-1$-th generation in G_{gen}^S.

We do the normalization operation for edge information $e_{gen,ij}^S$ with a holistic view of the graph G_{gen}^S.

❷ From Sample Graph to Distribution Graph

Given the sample graph G_{gen}^S, we design the distribution graph $G_{gen}^D = (V_{gen}^D, E_{gen}^D)$, in order to integrate the relations between samples from G_{gen}^S. Besides, the G_{gen}^D processes the distribution-level relations. In G_{gen}^D, each feature is a $\mathscr{K}$-dimension (-D) vector, where $\mathscr{K}$ means the number of samples from support set and $v_{gen,i}^D$ stands for the relations between sample x_i and sample x_j. Specifically, we use a graph neural network (GNN) backbone [84] to achieve it, in which GNN follows the neighborhood aggregation scheme, and computes the node representations by recursively aggregating and compressing node features from local neighborhoods.

① **Initialization** For first initialization, a GNN layer can be defined as:

$$v_{0,i}^D = combine(x_i) = \begin{cases} \Big\Vert_{j=1}^{\mathscr{K}} \sigma(y_i, y_j) & if\ x_i\ is\ labeled \\ [\underbrace{\frac{1}{\mathscr{K}}, \frac{1}{\mathscr{K}}, \frac{1}{\mathscr{K}}, \ldots, \frac{1}{\mathscr{K}}}_{\mathscr{K}-D}] & otherwise \end{cases} \tag{13}$$

where y_i and y_j are the label of sample x_i and x_j, respectively; $\|$ is the concatenation operator; $\sigma(\cdot)$ means the Kronecker delta function (i.e., $\sigma(y_i, y_j) = \begin{cases} 1 & y_i = y_j \\ 0 & y_i \neq y_j \end{cases}$).

② **Update Procedure** For gen ($gen > 0$) generations, the update process of node $v_{gen,i}^D$ in G_{gen}^D as follows:

$$v_{gen,i}^D = Aggregate(e_{gen,ij}^S, v_{gen-1,i}^D) \tag{14}$$

where $Aggregate(\cdot)$ is the aggregation network for distribution graph G_{gen}^D; in this paper, we use long short-term memory network (LSTM) [85] as the aggregation network as follows:

$$v_{gen,i}^D = LSTM(\Big\Vert_{j=1}^{\mathscr{K}} e_{gen,ij}^S, v_{gen-1,i}^D) \tag{15}$$

in which $LSTM(\cdot)$ is the LSTM network for distribution graph G_{gen}^D; $\|$ is the concatenation operator.

❸ Distribution Graph

Considering that optimal transport theory allows measurement and comparison of different probability distributions, we use optimal transport (i.e., Eq. (5)) to get the distribution-level similarities between nodes. Then, we follow a similar strategy to the sample graph to update G_{gen}^D.

① **Distribution-Level Similarity** The distribution-level similarity represents the similarity between the distribution characteristics of different samples, denoted as E_{gen}^D in G_{gen}^D. For $gen = 0$, we initialize the distribution similarity $e_{0,ij}^D$ as follows:

$$e_{0,ij}^D = ROT_\lambda(v_{0,i}^D, v_{0,j}^D) \tag{16}$$

where $ROT_\lambda(\cdot)$ means the function of optimal transport (i.e., Eq. (5)); $v_{0,i}^D$ and $v_{0,j}^D$ are nodes in G_{gen}^D following Eq. (15).

② **Update Procedure** For $gen > 0$, we define the update rule for $e_{gen,ij}^D$ in G_{gen}^D, as follows:

$$e_{gen,ij}^D = ROT_\lambda(v_{gen,i}^D, v_{gen,j}^D) \times e_{gen-1,ij}^D \tag{17}$$

where $ROT_\lambda(\cdot)$ means the function of optimal transport (i.e., Eq. (5)); $v_{gen,i}^D$ and $v_{gen,j}^D$ are node in G_{gen}^D following Eq. (19).

❹ From Distribution Graph to Sample Graph

We need to send the information from the distribution graph to the sample graph to complete a closed loop. At the end of each generation, we use the distribution characteristic $e_{gen,ij}^D$ in G_{gen}^D to update node features $v_{gen,i}^S$ in G_{gen}^S. We aggregate all node features $v_{gen-1,i}^S$ and edge features $e_{gen,ij}^D$ to gain the $v_{gen,i}^S$, which can capture the distribution characteristic relations, as follows:

$$v_{gen,i}^S = Aggregate(e_{gen,ij}^D, v_{gen-1,i}^S) \tag{18}$$

where $Aggregate(\cdot)$ is the aggregation network for sample graph G_{gen}^S; in this paper, we use LSTM [85] as the aggregation network as follows:

$$v_{gen,i}^S = LSTM(\sum_{j=1}^{\mathscr{K}+\mathbb{Q}} (e_{gen,ij}^D \cdot v_{gen-1,i}^S), v_{gen-1,i}^S) \tag{19}$$

in which $LSTM(\cdot)$ is the LSTM network for distribution graph G_{gen}^S. Note that this LSTM shares parameters with the LSTM in Eq. (19).

3.3. Priority attention mechanisms

Our proposed graph prototypical model in Section 3.2 permits node features to propagate not only in sample space, but also in distribution space, and the most correlated information with sample label should be extracted from both of these two spaces [30,31]. Inspired by priority maps [32,33] in human brain attention that focuses on the relevance of human activity [31,32], we design the **Priority Attention Mechanism** shown in Fig. 2. We use our proposed graph prototypical model to get two graphs: sample graph and distribution graph. The sample features can propagate over both of sample graph and distribution graph to learn **Sample-Level Embedding** and **Distribution-Level Embedding**, denoted as Z_S and Z_D respectively. Then, considering the interaction of sample graph and distribution graph, we design the networks with parameter sharing strategy for two LSTMs with shared parameters (mentioned in Eqs. (15) and (19)) to learn the **Relation Embedding**. The significance of the sample graph to the distribution graph is denoted as $Z_{S\to D}$. The significance of the distribution graph to the sample graph is denoted as $Z_{D\to S}$. Finally, taking into account that sample labels may be related to its distribution or its feature or both, we design an attention mechanism to adaptively fuse these embeddings with the learned weights to gain the most relevant embedding Z for the final classification task.

❶ Sample-Level Embedding

Given the sample graph $G_{gen}^S = (V_{gen}^S, E_{gen}^S)$, we use a graph convolutional network (GCN) [86] with the same hidden layer dimension to get the feature embedding. The gen-th layer output $Z_{sample}^{(gen)}$ can be written as:

$$Z_{sample}^{(gen)} = \mathrm{ReLU}(\widetilde{D}_{sample}^{-\frac{1}{2}} \widetilde{E}_{gen,sample}^S \widetilde{D}_{sample}^{-\frac{1}{2}} Z_{sample}^{(gen-1)} W_{sample}^{(gen)}) \tag{20}$$

in which $\widetilde{D}_{sample}^{-\frac{1}{2}}$ is the diagonal node degree matrix of $\widetilde{E}_{gen,sample}^S$ and $\widetilde{E}_{gen,sample}^S = E_{gen}^S + I_{gen}^S$, where I_{gen}^S is the identify matrix; $W_{sample}^{(gen)}$ is the weight matrix of the gen-th layer in GCN; $\mathrm{ReLU}(\cdot)$ is the ReLU activation

function [87]; at the initialization, $Z^{(0)}_{sample} = V^{S}_{gen}$. At the last layer, the output is denoted as Z_S. With this way, we are able to learn sample feature embeddings that capture feature information Z_S in the sample space.

❷ Distribution-Level Embedding

Similar to the process of sample-level embedding, we use the graph convolutional network (GCN) to gain the distribution-level embedding. Given the distribution graph $G^{D}_{gen} = (V^{D}_{gen}, E^{D}_{gen})$, the gen-th layer output $Z^{(gen)}_{distribution}$ can be represented as:

$$Z^{(gen)}_{distribution} = \mathrm{ReLU}(\widetilde{D}^{-\frac{1}{2}}_{distribution}\widetilde{E}^{D}_{gen,distribution} \widetilde{D}^{-\frac{1}{2}}_{distribution}Z^{(gen-1)}_{distribution}W^{(gen)}_{distribution}) \tag{21}$$

in which $\widetilde{D}^{-\frac{1}{2}}_{distribution}$ is the diagonal node degree matrix of $\widetilde{E}^{D}_{gen,distribution}$ and $\widetilde{E}^{D}_{gen,distribution} = E^{D}_{gen} + I^{D}_{gen}$, where I^{D}_{gen} is the identify matrix; $W^{(gen)}_{distribution}$ is the weight matrix of the gen-th layer in GCN; $\mathrm{ReLU}(\cdot)$ is the ReLU activation function [87]; at the initialization, $Z^{(0)}_{distribution} = V^{D}_{gen}$. At the last layer, the output is denoted as Z_D. With this way, we are able to learn sample distribution characteristic embeddings that capture the distribution characteristic Z_D in the distribution space.

❸ Relation Embedding

In fact, sample space and distribution space are not completely unrelated. In essence, the classification task can be associated with the information in either the sample space or the distribution space or both, which is hard to know in advance. Thus, we need to extract not only characteristic embeddings in both spaces but also related information shared by both spaces. We use the LSTM (mentioned in Eqs. (15) and (19)) with a parameter-sharing strategy to get relation embeddings in both spaces.

First, we utilize the LSTM (mentioned in Eq. (19)) to extract the sample-related embedding from the distribution graph G^{D}_{gen}. The gen-th layer output $Z^{(gen)}_{d\to s}$ can be written as:

$$Z^{(gen)}_{d\to s} = \mathrm{ReLU}(\widetilde{D}^{-\frac{1}{2}}_{distribution}\widetilde{E}^{D}_{gen,distribution} \widetilde{D}^{-\frac{1}{2}}_{distribution}Z^{(gen-1)}_{d\to s}W^{(gen)}_{LSTM}) \tag{22}$$

where $\widetilde{D}^{-\frac{1}{2}}_{distribution}$ and $\widetilde{E}^{D}_{gen,distribution}$ are same to Eq. (21); $W^{(gen)}_{LSTM}$ is the weight matrix of the gen-th layer in LSTM; $\mathrm{ReLU}(\cdot)$ is the ReLU activation function [87]; at the initialization, $Z^{(0)}_{d\to s} = V^{D}_{gen}$. At the last layer, the output is denoted as $Z_{D\to S}$.

At the same time, we utilize the LSTM (mentioned in Eq. (15)) that is shared with another LSTM (mentioned in Eq. (19)) parameters to extract the distribution-related embedding from the sample graph G^{S}_{gen}. The gen-th layer output $Z^{(gen)}_{s\to d}$ can be written as:

$$Z^{(gen)}_{s\to d} = \mathrm{ReLU}(\widetilde{D}^{-\frac{1}{2}}_{sample}\widetilde{E}^{S}_{gen,sample}\widetilde{D}^{-\frac{1}{2}}_{sample}Z^{(gen-1)}_{s\to d}W^{(gen)}_{LSTM}) \tag{23}$$

where $\widetilde{D}^{-\frac{1}{2}}_{sample}$ and $\widetilde{E}^{D}_{gen,sample}$ are same to Eq. (20); $W^{(gen)}_{LSTM}$ is the weight matrix of the gen-th layer in LSTM, and is same to Eq. (22); $\mathrm{ReLU}(\cdot)$ is the ReLU activation function [87]; at the initialization, $Z^{(0)}_{s\to d} = V^{S}_{gen}$. At the last layer, the output is denoted as $Z_{S\to D}$.

Due to different graphs (i.e., sample graph and distribution graph), we can get two output embedding $Z_{S\to D}$ and $Z_{D\to S}$ and the relation embedding Z_{Relation} of the two spaces is:

$$Z_{\mathrm{Relation}} = \frac{Z_{S\to D} + Z_{D\to S}}{2} \tag{24}$$

❹ Priority Attention

Now we get the sample-level embedding Z_S, the distribution-level embedding Z_D, and the relation embedding Z_{Relation}. Considering that the classification task can be associated with one of them or its combinations, we design a novel attention mechanism named **Priority Attention**, which focuses on the relevance of different level embedding and may be closer to the same mechanism as the attentional map (i.e., priority map [31,32]) of the human brain.

Following the idea of graph attention layer [88] and transformer [89], we firstly use a nonlinear transformation to transform the embedding as follows:

$$\begin{aligned}\omega_S &= q^T \cdot \tanh(W_{sample} \cdot {Z_S}^T + b_S)\\ \omega_{\mathrm{Relation}} &= q^T \cdot \tanh(W_{\mathrm{Relation}} \cdot {Z_{\mathrm{Relation}}}^T + b_{\mathrm{Relation}})\\ \omega_D &= q^T \cdot \tanh(W_{distribution} \cdot {Z_D}^T + b_D)\end{aligned} \tag{25}$$

where ω_S, $\omega_{\mathrm{Relation}}$ and ω_D are the attention value for sample-level embedding, the distribution-level embedding, and its relation respectively; q is one shared attention vector; W_{sample}, $W_{distribution}$, W_{Relation} are the weight matrix for sample-level embedding, the distribution-level embedding, and its relation respectively; b_S, b_{Relation} and b_D are the bias vector for sample-level embedding, the distribution-level embedding, and its relation respectively.

Then, we normalize the attention values ω_S, $\omega_{\mathrm{Relation}}$ and ω_D with softmax function $softmax(\cdot)$ to get the final weight as follows:

$$\begin{aligned}\alpha_S &= softmax(\omega_S) = \frac{\exp(\omega_S)}{\exp(\omega_S) + \exp(\omega_{\mathrm{Relation}}) + \exp(\omega_D)}\\ \alpha_R &= softmax(\omega_{\mathrm{Relation}}) = \frac{\exp(\omega_{\mathrm{Relation}})}{\exp(\omega_S) + \exp(\omega_{\mathrm{Relation}}) + \exp(\omega_D)}\\ \alpha_D &= softmax(\omega_D) = \frac{\exp(\omega_D)}{\exp(\omega_S) + \exp(\omega_{\mathrm{Relation}}) + \exp(\omega_D)}\end{aligned} \tag{26}$$

where α_S, α_D, and α_R imply the importance of corresponding embedding.

Finally, we combine these three embeddings to obtain the final embedding Z as follows:

$$Z = \alpha_S \times Z_S + \alpha_R \times Z_{\mathrm{Relation}} + \alpha_D \times Z_D \tag{27}$$

3.4. Objective function

Now we get the final embedding Z in Eq. (27). We use a linear transformation [90] and a softmax function [91] to get the prediction label $\hat{Y}$ in the following way:

$$\hat{Y} = softmax(W \cdot Z + b) \tag{28}$$

where W is the weight matrix for the softmax function $softmax(\cdot)$, b is the bias vector for embedding matrix Z.

Depending on the final embedding of a particular task, we can design different loss functions. In this paper, to address the problem of much sensor data with noisy, we can choose to minimize the active passive loss [92] of all labeled samples between ground-truth Y and prediction $\hat{Y}$:

$$\mathcal{L} = \varepsilon_1 \times \mathcal{L}_{NCE} + \varepsilon_2 \times \mathcal{L}_{RCE} \tag{29}$$

where $\varepsilon_1 > 0$ and $\varepsilon_2 > 0$ are parameters to balance the two terms; $\mathcal{L}_{NCE}$ means the Normalized Cross Entropy (NCE) loss [92] and details about $\mathcal{L}_{NCE}$ are shown in Ref. [92]; $\mathcal{L}_{RCE}$ means the Reverse Cross Entropy (RCE) loss [93] and details about $\mathcal{L}_{RCE}$ are depicted in Ref. [93]. Following the idea of Ma et al. [92], we set ε_1 to 10, and set ε_2 to 1.

4. Experimental results

In this section, we conduct experiments on fourteen kind datasets to evaluate the performance of the proposed approach. Compared with the state-of-the-art models on these datasets, our approach yields better performance in terms of Accuracy (%), Precision (%), Recall (%), F_1-Measure (%).

4.1. Dataset description

In this section, we introduce the description of fourteen kind datasets.

❶ The UTD-MHAD Dataset

The UTD-MHAD dataset [34] is collected using a Microsoft Kinect sensor and a wearable inertial sensor in an indoor environment. The dataset contains twenty-seven actions performed by eight subjects, in which every subject repeats each action four times. It provides 861 data sequences, which are 4 modalities: RGB videos, depth videos, skeleton joint positions, and the inertial sensor signals. These four modalities are recorded in three channels: one is used for simultaneous capture of depth videos and skeleton positions, the second one for RGB videos, and the third one for the inertial sensor signals (3-axis acceleration and 3-axis rotation signals).

❷ The OPPORTUNITY Dataset

The OPPORTUNITY dataset [4] contains data gained from heterogeneous sensors, which includes body-worn, ambient, and object sensors in daily life activities of 12 subjects. In these subjects, seven inertial measurement units (IMU) sensors with five sensors on the upper body and two sensors on the user's shoes provide ninety-six attributes; 12 3D acceleration sensors (placed on the upper body, hip, and leg) support thirty-six attributes and four tags for an ultra-wide-band localization system mounted on the left/right front/back side of the shoulder which give 12 attributes. More detail is listed in the paper Ref. [4]. Following the advice of Cao et al. [94], we classify four activities: stand, walk, sit, and lie.

❸ The DSADS Dataset

The DSADS dataset [35] records daily and sports activities. It contains different motion sensor data of nineteen daily and sports activities, such as walking on a treadmill, exercising on a stepper, and rowing. Eight subjects performed each activity in their own style without restriction for five minutes. Five units and four limbs on the torso were calibrated, and data were acquired at a sampling frequency of 25 Hz. Each unit contains nine sensors: 3-axis accelerators, 3-axis gyroscopes, and 3-axis magnetometers.

❹ The MHealth Dataset

The MHealth dataset [36] contains twelve daily activities from ten subjects. Activity is recorded at a sampling frequency of 50 Hz from four different types of sensors: three 3-axis accelerometers (placed on the chest, right wrist, and left ankle), two 3-axis gyroscope sensors (placed on the right wrist and left ankle), three 3-axis magnetometer sensors (placed on the chest, right wrist, and left ankle), and a 2-lead Electrocardiograph (ECG) sensor (placed on the chest). In our experiments, following the work of Ha et al. [95], we utilize three accelerometers (chest, right wrist, left ankle) and two gyroscopic (right wrist, left ankle) sensors.

❺ The WISDM Activity Prediction Dataset

The WISDM dataset [37] is a public benchmark HAR dataset provided by the Wireless Sensor Data Mining (WISDM) Lab with 6 data attributes: user, activity, timestamp, x-acceleration, y-acceleration, and z-acceleration. Twenty-nine volunteers for particular activities were recruited. Under supervision, these participants are asked to have an Android smartphone in their front pant leg pocket and to walk, jog, climb stairs, descend stairs, and sit and stand for a specific period of time. Data collection is controlled by an application with a simple graphical user interface that is executed on the phone. Time-series sensor data is generated using an embedded triaxial accelerometer, which is collected every 50 milliseconds, or 20 samples per second.

❻ The PAMAP2 Dataset

The PAMAP2 dataset [38] is designed to benchmark daily physical activity. It contains data collected from nine subjects related to 18 daily activities (e.g., vacuum cleaning, ironing, and jumping rope). Similar to the MHEALTH dataset, the data was collected by placing three IMUs on the subject's chest, dominant wrist, and significant ankle at a sampling frequency of 100 Hz.

❼ The UCI HAR Dataset

The UCI HAR dataset [39] is obtained from 30 volunteers using a waist-hung Samsung Galaxy S2 smartphone. Accelerometer and gyroscope signals are acquired at 50 Hz while the subjects perform the following six activities: standing, sitting, lying down, walking, downstairs, and upstairs.

❽ The Daphnet Freezing of Gait Dataset

The Daphnet freezing of gait dataset [40] uses three wearable accelerometers placed on the ankles, thighs, and torsos of eight patients with Parkinson's disease to detect freezing of gait (FOG). FOG is a condition that causes a sudden impairment in walking that leads to a risk of falling. Daphnet data were recorded during a variety of walking tasks in ten different participants with three different annotations: transient activities (i.e., discarded here), gait freeze, and regular movement.

❾ The HHAR Dataset

The Heterogeneous Human Activity Recognition (HHAR) dataset [41] contains signals from two sensors (accelerometer and gyroscope) of a smart-phone and a smart-watch to perform 6 different activities, i.e., bicycling, sitting, standing, walking, walking up and downstairs. Nine participants performed script-written activities lasting 5 minutes so that the class was evenly distributed. Subjects carried 8 smart-phones in a tight belt pouch, carried 4 smart-watches around their waist, and wore 2 on each arm. They used a total of 36 different smart devices from 13 models from 4 manufacturers to cover a variety of tools for sampling rate heterogeneity analysis. The sampling rate of the signals varied widely between phones, with values between 50 and 200 Hz.

❿ The UniMiB-SHAR Dataset

The UniMiB-SHAR dataset [42] is a dataset containing triaxial accelerometer signals collected at 50 Hz frequency from a Samsung Galaxy Nexus smartphone. Thirty subjects participated in the data collection process, forming a population sample of different heights, weights, ages, and genders. For the remainder of the experiment, subjects kept the device in the left front pocket of their pants for a period of time and in the right pocket.

⓫ The MobiAct Dataset

The MobiAct dataset [43] contains signals from the smartphone's inertial sensors (accelerometer, gyroscope, and orientation) for 11 daily activities and 4 falls. It was collected from 66 participants from 3200 different genders, age groups, and weights using a Samsung Galaxy S3 smartphone. The device was placed in a pocket on the pants, which was freely selected by the object in any random orientation to capture the daily use of the phone.

⓬ The MotionSense Dataset

The MotionSense dataset [44] contains an accelerometer, gyroscope, and altitude data from 24 participants of varying ages, genders, weights, and heights. It was collected using an iPhone 6s, which was placed in the user's front pocket. Subjects performed six different activities in 15 trials under similar environments and conditions. The purpose of the study was to infer physical and demographic attributes from time-series data, as well as to discover activities.

⓭ The UCI HAPT Dataset

The UCI HART dataset [45] contains motion sensor data of 6 daily activities collected from 30 subjects, and the average size of individual data collected from one subject is 347.

⓮ The USC-HAD Dataset

The USC-HAD dataset [46] combines six readings from body-worn 3-axis accelerometers and gyroscopic sensors via a motion node device. Data sets have been created in which body size and age are defined with an even distribution (7 positions each) of 14 male and female subjects. Sensor data were sampled at a rate of 100 Hz and contained one of 12 activity category labels for each time step in the dataset.

4.2. Experiments setup

In this subsection, we describe the implementation details.

❶ Network Architecture

In this paper, we use different architectures for the different models.

① **Graph Prototypical Models**

In the sample graph, we use the EfficientNet [81] as the feature extractor. In the process of "From Sample Graph to Distribution Graph", we use the MoNet [96] as the GNN backbone. In this process and the process of "From Distribution Graph to Sample Graph", we use two same original LSTM [85] with shared parameters to pass the information to graphs. We sample twenty-eight meta-tasks in each iteration for meta-training. In all experiments, we utilize the Adam optimizer [97] and set the initial learning rate to 10^{-3}. The learning rate is decayed by 0.1 per 1.5×10^4 iterations with 10^{-5} weight decay.

② **Priority Attention Mechanisms**

We train 3 two-layer GCNs with the same hidden layer dimension and the same output dimension simultaneously. In this process, we use Adam optimizer and set the initial learning rate to 0.0005. We set the dropout rate to 0.5 and weight decay to $5e^{-4}$.

❷ Data Pre-Processing

We do the data pre-processing in order to feed our approach with a certain data dimension:

(1) Considering that these datasets contain the lost data (i.e., NaN or 0), we use the linear interpolation algorithm [98] to fill the lost values.

(2) Due to the different ranges of sensors, we have to normalize [99] the input data to gain the range of 0 to 1.

(3) The input to the model consists of a data sequence. The sequence is a short time sequence extracted from the raw sensor data. The data are recorded continuously during the data collection process. In order to preserve the temporal relationship between data points in the activity, the data which is collected from the motion sensors are segmented using a sliding window with 50% overlap.

❸ Evaluation Protocols

We evaluate meta-learning algorithms, including our proposed approach, in the 5-way-5-shot setting. We follow the evaluation procedure of previous methods [28,29,100,101]. We randomly sample 10,000 tasks and then report the average accuracy (%), precision (%), recall (%), F_1-Measure (%) as well as the 95% confidence interval. We divide each dataset into a meta-training set, a support set, and a meta-test set, where its ratio is $6:2:2$.

4.3. Comparison with state-of-the-art approaches

We compare our results with those from the state-of-the-art approaches on these fourteen datasets.

4.3.1. Comparison on the UTD-MHAD dataset

In this sub-subsection, we use the average accuracy (%), precision (%), recall (%), and F_1-Measure (%) to evaluate our proposed method and other state-of-the-art methods on the UTD-MHAD dataset.

Baselines on The UTD-MHAD Dataset We conduct experiments on the UTD-MHAD dataset with the following baselines to compare them with our proposed model, including Cov3DJ [57], Kinect [34], Inertial [34], Kinect&Inertial [34], JTM [9], Optical Spectra [58], 3DHOT-MBC [59], JDM [102], BNN [12], JSR [55], JMR [55], BSR [55], BMR [55], JSR + JMR + BSR + BMR [55], SR-TSL [56], HSR-TSL [11], BGCLSTM [60], HNet [8], HDS-SP [49], DTIs [10], HCM + CNN + Spherical coordinates [7], TS+MSSFN [103], and Skepxel [50].

Effect of Our Approach According to the results in Table 2, it is obviously that our method is better than these methods. Particularly, ours is 14.21%, 33.71%, 32.61%, 20.71%, 14.01%, 12.81%, 15.41%, 11.71%, 7.71%, 7.91%, 7.51%, 7.01%, 6.51%, 1.41%, 6.71%, 5.41%, 7.71%, 3.11%, 5.86%, 11.44%, 5.00%, 7.48%, and 2.61% higher than Cov3DJ, Kinect, Inertial, Kinect&Inertial, JTM, Optical Spectra, 3DHOT-MBC, JDM, BNN, JSR, JMR, BSR, BMR, JSR + JMR + BSR + BMR, SR-TSL, HSR-TSL, BGCLSTM, HNet, HDS-SP, DTIs, HCM + CNN + Spherical coordinates, TS+MSSFN, and Skepxel, in terms of accuracy, respectively; ours is 14.29%, 33.77%, 32.71%, 20.72%, 14.10%, 12.89%, 15.42%, 11.73%, 7.78%, 7.94%, 7.55%, 7.09%, 6.60%, 1.45%, 6.73%, 5.49%, 7.81%, 3.14%, 5.94%, 11.48%, 5.09%, 7.52%, and 2.63% higher than Cov3DJ, Kinect, Inertial, Kinect&Inertial, JTM, Optical Spectra, 3DHOT-MBC, JDM, BNN, JSR, JMR, BSR, BMR, JSR + JMR + BSR + BMR, SR-TSL, HSR-TSL, BGCLSTM, HNet, HDS-SP, DTIs, HCM + CNN + Spherical coordinates, TS+MSSFN, and Skepxel, in terms of precision, respectively; ours is 14.30%, 33.80%, 32.65%, 20.80%, 14.05%, 12.92%, 15.44%, 11.76%, 7.82%, 8.01%, 7.57%, 7.10%, 6.62%, 1.47%, 6.78%, 5.53%, 7.82%, 3.15%, 5.91%, 11.49%, 5.04%, 7.60%, and 2.65% higher than Cov3DJ, Kinect, Inertial, Kinect&Inertial, JTM, Optical Spectra, 3DHOT-MBC, JDM, BNN, JSR, JMR, BSR, BMR, JSR + JMR + BSR + BMR, SR-TSL, HSR-TSL, BGCLSTM, HNet, HDS-SP, DTIs, HCM + CNN + Spherical coordinates, TS+MSSFN, and Skepxel, in terms of recall, respectively; ours is 14.29%, 33.78%, 32.68%, 20.76%, 14.07%, 12.90%, 15.43%, 11.74%, 7.80%, 7.97%, 7.55%, 7.09%, 6.61%, 1.46%, 6.75%, 5.50%, 7.81%, 3.14%, 5.92%, 11.48%, 5.06%, 7.55%, and 2.64% higher than Cov3DJ, Kinect, Inertial, Kinect&Inertial, JTM, Optical Spectra, 3DHOT-MBC, JDM, BNN, JSR, JMR, BSR, BMR, JSR + JMR + BSR + BMR, SR-TSL, HSR-TSL, BGCLSTM, HNet, HDS-SP, DTIs, HCM + CNN + Spherical coordinates, TS+MSSFN, and Skepxel, in terms of F_1-Measure, respectively. *From the above results, it is obvious that ours is more robust and effective than the state-of-the-arts methods on the UTD-MHAD dataset.*

Table 2
The compared results on the UTD-MHAD dataset.

Methods	Accuracy (%)	Precision (%)	Recall (%)	F_1-measure (%)
Cov3DJ [57]	85.60	85.53	85.53	85.53
Kinect [34]	66.10	66.05	66.03	66.04
Inertial [34]	67.20	67.11	67.18	67.14
Kinect&Inertial [34]	79.10	79.10	79.03	79.06
JTM [9]	85.80	85.72	85.78	85.75
Optical Spectra [58]	87.00	86.93	86.91	86.92
3DHOT-MBC [59]	84.40	84.40	84.39	84.39
JDM [102]	88.10	88.09	88.07	88.08
BNN [12]	92.10	92.04	92.01	92.02
JSR [55]	91.90	91.88	91.82	91.85
JMR [55]	92.30	92.27	92.26	92.27
BSR [55]	92.80	92.73	92.73	92.73
BMR [55]	93.30	93.22	93.21	93.21
JSR + JMR + BSR + BMR [55]	98.40	98.37	98.36	98.36
SR-TSL [56]	93.10	93.09	93.05	93.07
HSR-TSL [11]	94.40	94.33	94.30	94.32
BGCLSTM [60]	92.10	92.01	92.01	92.01
HNet [8]	96.70	96.68	96.68	96.68
HDS-SP [49]	93.95	93.88	93.92	93.90
DTIs [10]	88.37	88.34	88.34	88.34
HCM + CNN + Spherical coordinates [7]	94.81	94.73	94.79	94.76
TS+MSSFN [103]	92.33	92.30	92.23	92.27
Skepxel [50]	97.20	97.19	97.18	97.18
Ours	**99.81**	**99.82**	**99.83**	**99.82**

4.3.2. Comparison on the opportunity dataset

In this sub-subsection, we use the average accuracy (%), precision (%), recall (%), and F_1-Measure (%) to evaluate our proposed method and other state-of-the-art methods on the OPPORTUNITY dataset.

Baselines on The OPPORTUNITY Dataset We compare against various state-of-the-art baselines on the OPPORTUNITY dataset, including DDNN [61], ACED [104], DanHAR [62], DSmT [105], Lego-CNN [63], EmbraceNet [64], ELSTM [65], FSHAR-NGD [106], MVAN [107],

Table 3
Comparison results on the OPPORTUNITY dataset.

Methods	Accuracy (%)	Precision (%)	Recall (%)	F_1-measure (%)
DDNN [61]	86.01	86.50	86.40	86.45
ACED [104]	97.27	97.30	96.45	96.87
DanHAR [62]	93.16	93.10	93.30	93.20
DSmT [105]	97.14	97.20	97.40	97.30
Lego-CNN [63]	86.10	88.20	88.70	88.45
EmbraceNet [64]	87.30	87.45	87.56	87.50
ELSTM [65]	85.40	86.30	86.70	86.50
FSHAR-NGD [106]	85.60	85.70	85.90	85.80
MVAN [107]	96.80	96.50	96.90	96.70
M-U-Net [108]	94.70	94.90	94.50	94.70
SAWSD [109]	66.90	66.70	67.40	67.00
InnoHAR [110]	67.65	67.56	67.67	67.61
LSTM-CNN [66]	92.71	92.71	92.71	92.71
OCL [111]	96.70	96.40	97.10	96.74
OMSD [94]	92.01	92.30	91.23	91.76
SensCapsNet [67]	70.90	71.10	69.90	70.50
LWTCNN [68]	81.00	80.18	81.02	80.58
Ours	**98.99**	**98.97**	**99.23**	**99.10**

M-U-Net [108], SAWSD [109], InnoHAR [110], LSTM-CNN [66], OCL [111], OMSD [94], SensCapsNet [67], and LWTCNN [68].

Effect of Our Approach According to the results in Table 3, it is evident that our method is better than these methods. Particularly, ours is 12.98%, 1.72%, 5.83%, 1.85%, 12.89%, 11.69%, 13.59%, 13.39%, 2.19%, 4.29%, 32.09%, 31.34%, 6.28%, 2.29%, 6.98%, 28.09%, and 17.99% higher than DDNN, ACED, DanHAR, DSmT, Lego-CNN, EmbraceNet, ELSTM, FSHAR-NGD, MVAN, M-U-Net, SAWSD, InnoHAR, LSTM-CNN, OCL, OMSD, SensCapsNet, and LWTCNN, in terms of accuracy, respectively; ours is 12.47%, 1.67%, 5.87%, 1.77%, 10.77%, 11.52%, 12.67%, 13.27%, 2.47%, 4.07%, 32.27%, 31.41%, 6.26%, 2.57%, 6.67%, 27.87%, and 18.7% higher than DDNN, ACED, DanHAR, DSmT, Lego-CNN, EmbraceNet, ELSTM, FSHAR-NGD, MVAN, M-U-Net, SAWSD, InnoHAR, LSTM-CNN, OCL, OMSD, SensCapsNet, and LWTCNN, in terms of precision, respectively; ours is 12.83%, 2.78%, 5.93%, 1.83%, 10.53%, 11.67%, 12.53%, 13.33%, 2.33%, 4.73%, 31.83%, 31.56%, 6.52%, 2.13%, 8.00%, 29.33%, and 18.21% higher than DDNN, ACED, DanHAR, DSmT, Lego-CNN, EmbraceNet, ELSTM, FSHAR-NGD, MVAN, M-U-Net, SAWSD, InnoHAR, LSTM-CNN, OCL, OMSD, SensCapsNet, and LWTCNN, in terms of recall, respectively; ours is 12.65%, 2.23%, 5.90%, 1.80%, 10.65%, 11.60%, 12.60%, 13.30%, 2.40%, 4.40%, 32.10%, 31.49%, 6.39%, 2.36%, 7.34%, 28.60%, and 18.52% higher than DDNN, ACED, DanHAR, DSmT, Lego-CNN, Embrace-Net, ELSTM, FSHAR-NGD, MVAN, M-U-Net, SAWSD, InnoHAR, LSTM-CNN, OCL, OMSD, SensCapsNet, and LWTCNN, in terms of F_1-Measure, respectively. *From the above results, it is obvious that ours is more robust and effective than the state-of-the-arts methods on the OPPORTUNITY dataset.*

4.3.3. Comparison on the DSADS dataset

In this sub-subsection, we use the average accuracy (%), precision (%), recall (%), and F_1-Measure (%) to evaluate our proposed method and other state-of-the-art methods on the DSADS dataset.

Baselines on The DSADS Dataset We compare against various state-of-the-art baselines on the DSADS dataset, including FIRF [69], CIRF [70], DAMUN [71], DSMT [105], DWFS [112], and MSF-EP [111].

Effect of Our Approach According to the results in Table 4, it is evident that our method is better than these methods. Particularly, ours is 4.37%, 1.31%, 7.68%, 4.67%, 6.37%, and 21.34% higher than FIRF, CIRF, DAMUN, DSMT, DWFS, and MSF-EP, in terms of accuracy, respectively; ours is 5.41%, 3.21%, 9.67%, 5.31%, 4.31%, and 12.21% higher than FIRF, CIRF, DAMUN, DSMT, DWFS, and MSF-EP, in terms of precision, respectively; ours is 4.58%, 3.18%, 6.99%, 3.28%, 3.68%, and 13.48% higher than FIRF, CIRF, DAMUN, DSMT, DWFS, and MSF-EP, in terms of recall, respectively; ours is 4.99%, 3.19%, 8.35%, 4.31%, 3.99%, and 12.94% higher than FIRF, CIRF, DAMUN, DSMT, DWFS, and MSF-EP, in terms of F_1-Measure, respectively. *From the above results, it is obvious that ours is more robust and effective than the state-of-the-arts methods on the DSADS dataset.*

Table 4
Comparison results on the DSADS dataset.

Methods	Accuracy (%)	Precision (%)	Recall (%)	F_1-measure (%)
FIRF [69]	95.45	94.50	95.40	94.95
CIRF [70]	98.51	96.70	96.80	96.75
DAMUN [71]	92.14	90.24	92.99	91.59
DSMT [105]	95.15	94.60	96.70	95.63
DWFS [112]	93.45	95.60	96.30	95.95
MSF-EP [111]	78.48	87.70	86.50	87.00
Ours	**99.82**	**99.91**	**99.98**	**99.94**

Table 5
Comparison results on the MHealth dataset.

Methods	Accuracy (%)	Precision (%)	Recall (%)	F_1-measure (%)
DRNNBR [113]	99.00	99.00	99.00	99.00
SSRCAN [114]	94.25	98.23	98.45	98.34
DAMUN [71]	96.07	95.63	96.52	96.07
CNN-pff [95]	91.94	92.04	93.23	92.63
MVAN [107]	95.10	94.80	95.20	95.00
GADF [115]	98.50	98.67	98.76	98.71
TC-CR [116]	92.30	92.40	93.20	92.80
SRRS [117]	98.30	91.80	90.80	90.40
Ours	**99.99**	**99.99**	**99.99**	**99.99**

4.3.4. Comparison on the MHealth dataset

In this sub-subsection, we use the average accuracy (%), precision (%), recall (%), and F_1-Measure (%) to evaluate our proposed method and other state-of-the-art methods on the MHealth dataset.

Baselines on The MHealth Dataset We compare against various state-of-the-art baselines on the MHealth dataset, including DRNNBR [113], SSRCAN [114], DAMUN [71], CNN-pff [95], MVAN [107], GADF [115], TC-CR [116], and SRRS [117].

Effect of Our Approach According to the results in Table 5, it is evident that our method is better than these methods. Particularly, ours is 0.99%, 5.74%, 3.92%, 8.05%, 4.89%, 1.49%, 7.69%, and 1.69% higher than DRNNBR, SSRCAN, DAMUN, CNN-pff, MVAN, GADF, TC-CR, and SRRS, in terms of accuracy, respectively; ours is 0.99%, 1.76%, 4.36%, 7.95%, 5.19%, 1.32%, 7.59%, and 8.19% higher than DRNNBR, SSRCAN, DAMUN, CNN-pff, MVAN, GADF, TC-CR, and SRRS, in terms of precision, respectively; ours is 0.99%, 1.54%, 3.47%, 6.76%, 4.79%, 1.23%, 6.79%, and 9.19% higher than DRNNBR, SSRCAN, DAMUN, CNN-pff, MVAN, GADF, TC-CR, and SRRS, in terms of recall, respectively; ours is 0.99%, 1.65%, 3.92%, 7.36%, 4.99%, 1.28%, 7.19%, 9.59% higher than DRNNBR, SSRCAN, DAMUN, CNN-pff, MVAN, GADF, TC-CR, and SRRS, in terms of F_1-Measure, respectively. *From the above results, it is obvious that ours is more robust and effective than the state-of-the-arts methods on the MHealth dataset.*

4.3.5. Comparison on the WISDM activity prediction dataset

In this sub-subsection, we use the average accuracy (%), precision (%), recall (%), and F_1-Measure (%) to evaluate our proposed method and other state-of-the-art methods on the WISDM dataset.

Baselines on The WISDM Activity Prediction Dataset We compare against various state-of-the-art baselines on the WISDM activity prediction dataset, including KNN-LS-SVM [118], MHCA [119], DanHAR [62], DARMTA [120], DSSL [14], Lego-CNN [63], MVAN [107], M-U-Net [108], SSFT [121], OCL [111], LWTCNN [68], and SRRS [117].

Effect of Our Approach According to the results in Table 6, it is evident that our method is better than these methods. Particularly, ours is 3.72%, 3.62%, 0.27%, 0.89%, 42.42%, 1.61%, 6.02%, 2.72%, 9.11%,

Table 6
Comparison results on the WISDM activity prediction dataset.

Methods	Accuracy (%)	Precision (%)	Recall (%)	F_1-measure (%)
KNN-LS-SVM [118]	95.40	95.00	96.00	95.50
MHCA [119]	95.50	95.60	95.20	95.40
DanHAR [62]	98.85	98.34	98.35	98.34
DARMTA [120]	98.23	98.32	98.34	98.33
DSSL [14]	56.70	56.70	56.30	56.50
Lego-CNN [63]	97.51	97.60	97.56	97.58
MVAN [107]	93.10	92.90	93.10	93.00
M-U-Net [108]	96.40	96.20	96.40	96.30
SSFT [121]	90.01	89.99	85.68	86.86
OCL [111]	81.30	82.32	81.23	81.77
LWTCNN [68]	98.82	98.79	98.83	98.81
SRRS [117]	93.50	93.40	92.60	92.30
Ours	**99.12**	**99.23**	**99.21**	**99.22**

Table 7
Comparison results on the PAMAP2 dataset.

Methods	Accuracy (%)	Precision (%)	Recall (%)	F_1-measure (%)
HCNN [122]	84.56	83.56	84.12	83.84
SSRCAN [114]	83.42	92.90	92.85	92.87
DAMUN [71]	83.21	80.73	83.57	82.13
Lego-CNN [63]	91.40	91.50	91.60	91.55
ELSTM [65]	85.40	86.20	85.94	86.07
FSHAR-NGD [106]	58.98	59.34	60.23	59.78
MVAN [107]	93.20	93.10	93.30	93.20
SAWSD [109]	96.00	96.23	95.95	96.09
InnoHAR [110]	81.40	82.50	83.40	82.95
LWTCNN [68]	92.97	93.00	93.06	93.03
Ours	**98.97**	**99.12**	**99.42**	**99.27**

17.82%, 0.30%, and 5.62% higher than KNN-LS-SVM, MHCA, DanHAR, DARMTA, DSSL, Lego-CNN, MVAN, M-U-Net, SSFT, OCL, LWTCNN, and SRRS, in terms of accuracy, respectively; ours is 4.23%, 3.63%, 0.89%, 0.91%, 42.53%, 1.63%, 6.33%, 3.03%, 9.24%, 16.91%, 0.44%, and 5.83% higher than KNN-LS-SVM, MHCA, DanHAR, DARMTA, DSSL, Lego-CNN, MVAN, M-U-Net, SSFT, OCL, LWTCNN, and SRRS, in terms of precision, respectively; ours is 3.21%, 4.01%, 0.86%, 0.87%, 42.91%, 1.65%, 6.11%, 2.81%, 13.53%, 17.98%, 0.38%, and 6.61% higher than KNN-LS-SVM, MHCA, DanHAR, DARMTA, DSSL, Lego-CNN, MVAN, M-U-Net, SSFT, OCL, LWTCNN, and SRRS, in terms of recall, respectively; ours is 3.72%, 3.82%, 0.88%, 0.89%, 42.72%, 1.64%, 6.22%, 2.92%, 12.36%, 17.45%, 0.41%, 6.92% higher than KNN-LS-SVM, MHCA, DanHAR, DARMTA, DSSL, Lego-CNN, MVAN, M-U-Net, SSFT, OCL, LWTCNN, and SRRS, in terms of F_1-Measure, respectively. *From the above results, it is obvious that ours is more robust and effective than the state-of-the-arts methods on the WISDM activity prediction dataset.*

4.3.6. Comparison on the PAMAP2 dataset

In this sub-subsection, we use the average accuracy (%), precision (%), recall (%), and F_1-Measure (%) to evaluate our proposed method and other state-of-the-art methods on the PAMAP2 dataset.

Baselines on The PAMAP2 Dataset We compare against various state-of-the-art baselines on the PAMAP2 dataset, including HCNN [122], SSRCAN [114], DAMUN [71], Lego-CNN [63], ELSTM [65], FSHAR-NGD [106], MVAN [107], SAWSD [109], InnoHAR [110], and LWTCNN [68].

Effect of Our Approach According to the results in Table 7, it is evident that our method is better than these methods. Particularly, ours is 14.41%, 15.55%, 15.76%, 7.57%, 13.57%, 39.99%, 5.77%, 2.97%, 17.57%, and 6.00% higher than HCNN, SSRCAN, DAMUN, Lego-CNN, ELSTM, FSHAR-NGD, MVAN, SAWSD, InnoHAR, and LWTCNN, in terms of accuracy, respectively; ours is 15.56%, 6.22%, 18.39%, 7.62%, 12.92%, 39.78%, 6.02%, 2.89%, 16.62%, and 6.12% higher than HCNN, SSRCAN, DAMUN, Lego-CNN, ELSTM, FSHAR-NGD, MVAN, SAWSD, InnoHAR, and LWTCNN, in terms of precision, respectively; ours is 15.30%, 6.57%, 15.85%, 7.82%, 13.48%, 39.19%, 6.12%, 3.47%, 16.02%, and 6.36% higher than HCNN, SSRCAN, DAMUN, Lego-CNN, ELSTM, FSHAR-NGD, MVAN, SAWSD, InnoHAR, and LWTCNN, in terms of recall, respectively; ours is 15.43%, 6.40%, 17.14%, 7.72%, 13.20%, 39.49%, 6.07%, 3.18%, 16.32%, and 6.24% higher than HCNN, SSRCAN, DAMUN, Lego-CNN, ELSTM, FSHAR-NGD, MVAN, SAWSD, InnoHAR, and LWTCNN , in terms of F_1-Measure, respectively. *From the above results, it is obvious that ours is more robust and effective than the state-of-the-arts methods on the PAMAP2 dataset.*

Table 8
Comparison results on the UCI HAR dataset.

Methods	Accuracy (%)	Precision (%)	Recall (%)	F_1-measure (%)
CELearning [123]	95.93	95.98	96.12	96.05
SSRCAN [114]	83.12	71.98	72.26	72.12
ELSTM [65]	96.27	96.67	96.45	96.56
AIA [124]	87.65	88.23	88.21	88.22
LabelForest [125]	91.23	91.34	91.21	91.27
LSTM-CNN [66]	90.34	91.34	91.32	91.33
SSFT [121]	91.23	90.97	90.73	90.65
LWTCNN [68]	96.68	96.71	96.64	96.67
Ours	**98.92**	**99.21**	**99.24**	**99.22**

4.3.7. Comparison on the UCI HAR dataset

In this sub-subsection, we use the average accuracy (%), precision (%), recall (%), and F_1-Measure (%) to evaluate our proposed method and other state-of-the-art methods on the UCI HAR dataset.

Baselines on The UCI HAR Dataset We compare against various state-of-the-art baselines on the UCI HAR dataset, including CELearning [123], SSRCAN [114], ELSTM [65], AIA [124], LabelForest [125], LSTM-CNN [66], SSFT [121], and LWTCNN [68] .

Effect of Our Approach According to the results in Table 8, it is evident that our method is better than these methods. Particularly, ours is 2.99%, 15.80%, 2.65%, 11.27%, 7.69%, 8.58%, 7.69%, and 2.24% higher than CELearning, SSRCAN, ELSTM, AIA, LabelForest, LSTM-CNN, SSFT, and LWTCNN, in terms of accuracy, respectively; ours is 3.23%, 27.23%, 2.54%, 10.98%, 7.87%, 7.87%, 8.24%, and 2.50% higher than CELearning, SSRCAN, ELSTM, AIA, LabelForest, LSTM-CNN, SSFT, and LWTCNN, in terms of precision, respectively; ours is 3.12%, 26.98%, 2.79%, 11.03%, 8.03%, 7.92%, 8.51%, and 2.60% higher than CELearning, SSRCAN, ELSTM, AIA, LabelForest, LSTM-CNN, SSFT, and LWTCNN, in terms of recall, respectively; ours is 3.17%, 27.10%, 2.66%, 11.00%, 7.95%, 7.89%, 8.57%, and 2.55% higher than CELearning, SSRCAN, ELSTM, AIA, LabelForest, LSTM-CNN, SSFT, and LWTCNN, in terms of F_1-Measure, respectively. *From the above results, it is obvious that ours is more robust and effective than the state-of-the-arts methods on the UCI HAR dataset.*

4.3.8. Comparison on the daphnet freezing of gait dataset

In this sub-subsection, we use the average accuracy (%), precision (%), recall (%), and F_1-Measure (%) to evaluate our proposed method and other state-of-the-art methods on the Daphnet Freezing of Gait dataset.

Baselines on The Daphnet Freezing of Gait Dataset We compare against various state-of-the-art baselines on the Daphnet freezing of gait dataset, including KNN-LS-SVM [118], DDNN [61], DARMTA [120], MVAN [107], and InnoHAR [110].

Effect of Our Approach According to the results in Table 9, it is evident that our method is better than these methods. Particularly, ours is 1.42%, 6.62%, 7.73%, 2.63%, 5.83% higher than KNN-LS-SVM, DDNN, DARMTA, MVAN, and InnoHAR, in terms of accuracy, respectively; ours is 1.33%, 6.56%, 7.61%, 2.65%, and 5.62% higher than KNN-LS-SVM, DDNN, DARMTA, MVAN, and InnoHAR, in terms of precision, respectively; ours is 2.25%, 6.81%, 7.82%, 2.81%, and 4.93%

Table 9
Comparison results on the daphnet freezing of gait dataset.

Methods	Accuracy (%)	Precision (%)	Recall (%)	F_1-measure (%)
KNN-LS-SVM [118]	97.81	97.79	97.18	97.48
DDNN [61]	92.61	92.56	92.62	92.59
DARMTA [120]	91.50	91.51	91.61	91.56
MVAN [107]	96.60	96.47	96.62	96.50
InnoHAR [110]	93.40	93.50	94.50	94.00
Ours	**99.23**	**99.12**	**99.43**	**99.27**

Table 10
Comparison results on The HHAR dataset.

Methods	Accuracy (%)	Precision (%)	Recall (%)	F_1-measure (%)
UDAWD [126]	87.90	87.80	88.20	88.00
MVAN [107]	90.50	90.30	90.50	90.40
GADF [115]	92.78	92.75	92.77	92.76
Motion2Vector [127]	89.98	88.98	90.02	89.50
TARM [128]	80.80	81.20	81.40	81.30
Vision2Sensor [129]	81.30	83.20	84.30	83.74
Ours	**97.23**	**97.76**	**97.75**	**97.76**

higher than KNN-LS-SVM, DDNN, DARMTA, MVAN, and InnoHAR, in terms of recall, respectively; ours is 1.79%, 6.68%, 7.71%, 2.77%, and 5.27% higher than KNN-LS-SVM, DDNN, DARMTA, MVAN, and InnoHAR, in terms of F_1-Measure, respectively. *From the above results, it is obvious that ours is more robust and effective than the state-of-the-arts methods on the Daphnet freezing of gait dataset.*

4.3.9. Comparison on the HHAR dataset

In this sub-subsection, we use the average accuracy (%), precision (%), recall (%), and F_1-Measure (%) to evaluate our proposed method and other state-of-the-art methods on the HHAR dataset.

Baselines on The HHAR Dataset We compare against various state-of-the-art baselines on the HHAR dataset, including UDAWD [126], MVAN [107], GADF [115], Motion-2Vector [127], TARM [128], and Vision2Sensor [129].

Effect of Our Approach According to the results in Table 10, it is evident that our method is better than these methods. Particularly, ours is 9.33%, 6.73%, 4.45%, 7.25%, 16.43%, and 15.93% higher than UDAWD, MVAN, GADF, Motion2Vector, TARM, and Vision2Sensor, in terms of accuracy, respectively; ours is 9.96%, 7.46%, 5.01%, 8.78%, 16.56%, and 14.56% higher than UDAWD, MVAN, GADF, Motion2Vector, TARM, and Vision2Sensor, in terms of precision, respectively; ours is 9.55%, 7.25%, 4.98%, 7.73%, 16.35%, and 13.45% higher than UDAWD, MVAN, GADF, Motion2Vector, TARM, and Vision2Sensor, in terms of recall, respectively; ours is 9.76%, 7.35%, 4.99%, 8.26%, 16.46%, 14.02% higher than UDAWD, MVAN, GADF, Motion2Vector, TARM, and Vision2Sensor , in terms of F_1-Measure, respectively. *From the above results, it is obvious that ours is more robust and effective than the state-of-the-arts methods on the HHAR dataset.*

4.3.10. Comparison on the UniMiB-SHAR dataset

In this sub-subsection, we use the average accuracy (%), precision (%), recall (%), and F_1-Measure (%) to evaluate our proposed method and other state-of-the-art methods on the UniMiB-SHAR dataset.

Baselines on The UniMiB-SHAR Dataset We compare against various state-of-the-art baselines on the UniMiB-SHAR dataset, including DanHAR [62], Lego-CNN [63], SSFT [121], and LWTCNN [68] .

Effect of Our Approach From Fig. 4, it is evident that our method is better than these methods. Particularly, ours is 19.42%, 23.99%, 13.11%, and 20.38% higher than DanHAR, Lego-CNN, SSFT, and LWTCNN, in terms of accuracy, respectively; ours is 18.42%, 23.92%, 13.37%, and 20.62% higher than DanHAR, Lego-CNN, SSFT, and LWTCNN, in terms of precision, respectively; ours is 19.45%, 23.96%, 14.10%, and 20.59% higher than DanHAR, Lego-CNN, SSFT, and LWTCNN, in terms of recall, respectively; ours is 18.94%, 23.95%, 14.18%, and 20.61% higher than DanHAR, Lego-CNN, SSFT, and LWTCNN, in terms of F_1-Measure, respectively. *From the above results, it is obvious that ours is more robust and effective than the state-of-the-arts methods on the UniMiB-SHAR dataset.*

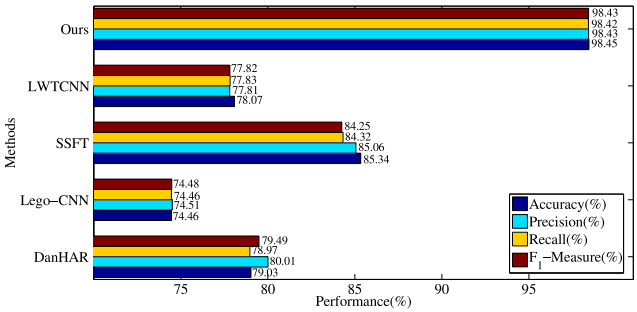

Fig. 4. Comparison results on the UniMiB-SHAR dataset.

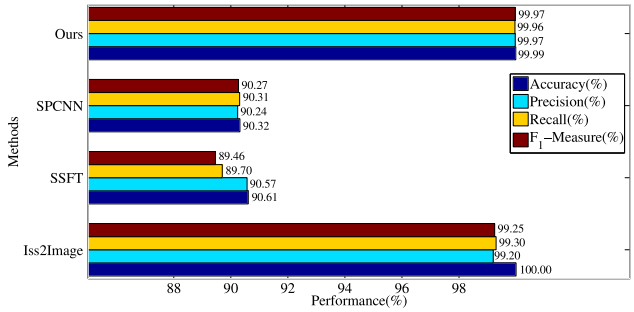

Fig. 5. Comparison results on the MobiAct dataset.

4.3.11. Comparison on the MobiAct dataset

In this sub-subsection, we use the average accuracy (%), precision (%), recall (%), and F_1-Measure (%) to evaluate our proposed method and other state-of-the-art methods on the MobiAct dataset.

Baselines on The MobiAct Dataset We compare against various state-of-the-art baselines on the MobiAct dataset, including Iss2Image [130], SSFT [121], and SPCNN [131] .

Effect of Our Approach From Fig. 5, it is evident that our method is better than these methods. Particularly, ours is 9.38% and 9.67% higher than SSFT and SPCNN, respectively; however, ours is only 0.01% away from Iss2Image, in terms of accuracy; ours is 0.77%, 9.40%, and 9.73% higher than Iss2Image, SSFT, and SPCNN, in terms of precision, respectively; ours is 0.66%, 10.26%, 9.65% higher than Iss2Image, SSFT, and SPCNN, in terms of recall, respectively; ours is 0.72%, 10.51%, and 9.70% higher than Iss2Image, SSFT, and SPCNN, in terms of F_1-Measure, respectively. *From the above results, it is obvious that ours is more robust and effective than the state-of-the-arts methods on the MobiAct dataset.*

4.3.12. Comparison on the MotionSense dataset

In this sub-subsection, we use the average accuracy (%), precision (%), recall (%), and F_1-Measure (%) to evaluate our proposed method and other state-of-the-art methods on the MotionSense dataset.

Baselines on The MotionSense Dataset We compare against various state-of-the-art baselines on the MotionSense dataset, including SSFT [121].

Effect of Our Approach From Fig. 6, it is evident that our method is better than these methods. Particularly, ours is 7.00%, 7.70%, 9.55%, 9.23% higher than SSFT, in terms of accuracy, precision, recall, and F_1-Measure, respectively. *From the above results, it is obvious that ours is more robust and effective than the state-of-the-arts methods on the MotionSense dataset.*

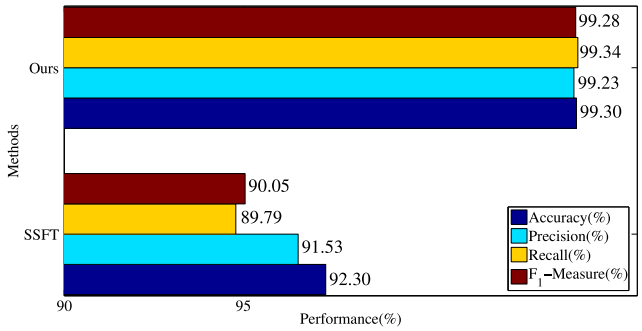

Fig. 6. Comparison results on the MotionSense dataset.

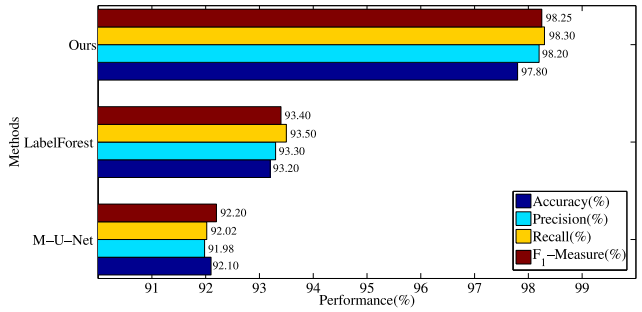

Fig. 7. Comparison results on the UCI HAPT dataset.

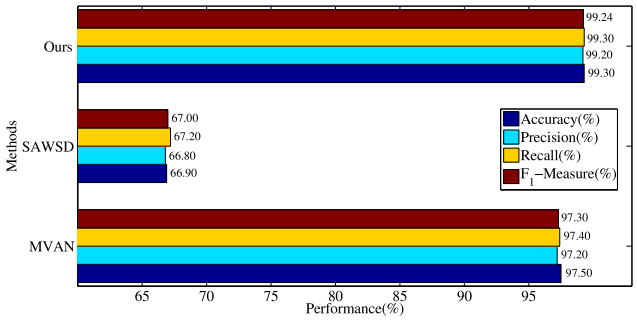

Fig. 8. Comparison results on the USC-HAD dataset.

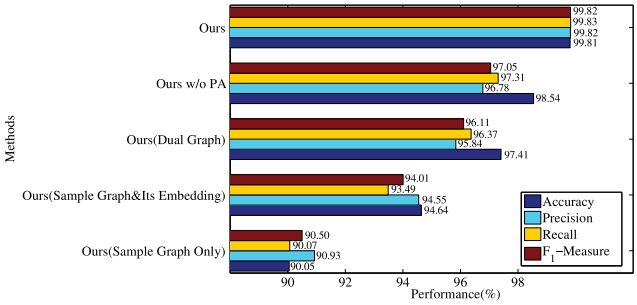

Fig. 9. The results of ablation study on the UTD-MHAD dataset.

4.3.13. Comparison on the UCI HAPT dataset

In this sub-subsection, we use the average accuracy (%), precision (%), recall (%), and F_1-Measure (%) to evaluate our proposed method and other state-of-the-art methods on the UCI HAPT dataset.

Baselines on The UCI HAPT Dataset We compare against various state-of-the-art baselines on the UCI HAPT dataset, including M-U-Net [108] and LabelForest [125].

Effect of Our Approach From Fig. 7, it is evident that our method is better than these methods. Particularly, ours is 5.70% and 4.60% higher than M-U-Net and LabelForest, in terms of accuracy, respectively. ours is 6.22% and 4.90% higher than M-U-Net and LabelForest, in terms of precision, respectively; ours is 6.28% and 4.80% higher than M-U-Net and LabelForest, in terms of recall, respectively; ours is 6.05% and 4.85% higher than M-U-Net and LabelForest, in terms of F_1-Measure, respectively. *From the above results, it is obvious that ours is more robust and effective than the state-of-the-arts methods on the UCI HAPT dataset.*

4.3.14. Comparison on the USC-HAD dataset

In this sub-subsection, we use the average accuracy (%), precision (%), recall (%), and F_1-Measure (%) to evaluate our proposed method and other state-of-the-art methods on the USC-HAD dataset.

Baselines on The USC-HAD Dataset We compare against various state-of-the-art baselines on the USC-HAD dataset, including MVAN [107] and SAWSD [109] .

Effect of Our Approach From Fig. 8, it is evident that our method is better than these methods. Particularly, ours is 1.80% and 32.40% higher than MVAN and SAWSD, in terms of accuracy, respectively. Ours is 2.00% and 32.40% higher than MVAN and SAWSD, in terms of precision, respectively; ours is 1.90% and 32.10% higher than MVAN and SAWSD, in terms of recall, respectively; ours is 1.94% and 32.24% higher than MVAN and SAWSD, in terms of F_1-Measure, respectively. *From the above results, it is obvious that ours is more robust and effective than the state-of-the-arts methods on the USC-HAD dataset.*

4.4. Ablation study

In order to validate the effectiveness and reasonableness of each component of our dual graph model and our graph priority attention mechanism, we design the ablation experiment on the UTD-MHAD dataset. In Fig. 9, "**Ours(Sample Graph Only)** " means a variant of Ours, which only constitutes the sample graph; "**Ours(Sample Graph&Its Embedding)** " means a variant of Ours, which represents the sample graph and utilizes sample-level embedding; "**Ours(Dual Graph)**" means a variant of Ours, which constitutes the sample graph and distribution graph without any attentions; "**Ours w/o PA**" represents a variant of Ours, which takes priority attention mechanisms away, and utilizes the sample graph and distribution graph with sample-level embedding and distribution-level embedding. We analyze the following three aspects:

Compared with "Dual Graph" From Fig. 9, "Ours(Dual Graph)" is 7.35%, 4.91%, 6.30%, and 5.61% higher than "Ours(Sample Graph Only)", in terms of accuracy, precision, recall, F_1-Measure, respectively. *As we can see, "Ours(Dual Graph)" is better than "Ours(Sample Graph Only)". These suggest the model of dual graph helps us to improve the task of human activity recognition.*

Compared with Graph and Its Embedding From Fig. 9, "Ours(Sample Graph&Its Embedding)" is 4.59%, 3.62%, 3.42%, and 3.52% higher than "Ours(Sample Graph Only)", in terms of accuracy, precision, recall, F_1-Measure, respectively. "Ours w/o PA" is 1.13%, 0.94%, 0.94%, and 0.94% higher than "Ours(Dual Graph)", in terms of accuracy, precision, recall, F_1-Measure, respectively. *As we can see, the variant of "Ours" with embedding is better than these without embedding. These suggest the role of the embedding of the graph is vital for the task of human activity recognition.*

Compared with "Ours" From Fig. 9, "Ours" is 9.76%, 5.17%, 2.40%, and 1.27% higher than "Ours(Sample Graph Only)", "Ours(Sample Graph&Its Embedding)", "Ours(Dual Graph)", and "Ours w/o PA", in terms of accuracy, respectively. *As we can see, "Ours" is better than others. These suggest the importance of graph priority attention mechanisms.*

From the above, we get the conclusion in the following two aspects:

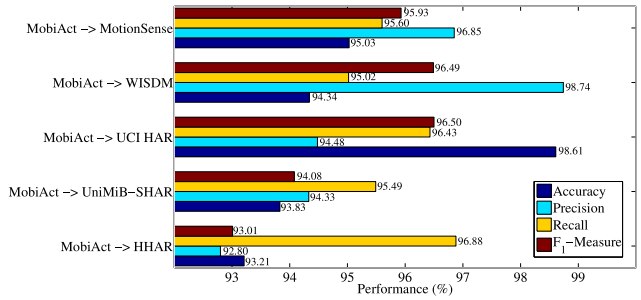

Fig. 10. Cross-dataset recognition performance of the proposed approach. We pre-train our proposed approach on MobiAct dataset. The classifier is trained in an end-to-end fashion on a particular activity recognition dataset. We chose MobiAct for cross-dataset evaluation because of the large number of users and activity classes it covers. The reported results (i.e., Accuracy (%), Precision (%), Recall (%) and F_1-Measure (%)) are averaged over 10 independent runs.

(1) *It is apparent that the design of the dual graph improves human activity recognition.*

(2) *It is distinct that the drawing of the graph priority attention mechanism is effective.*

4.5. Discussion on the generalization ability

Here, we discuss the generalization capabilities of our model. Besides, the protocol of the dataset we used requires that the researcher is not able to modify the dataset, therefore, we have to temporarily abandon the option of exploring further by modifying the dataset. Further, considering that there are different modalities and sensor data in different scenarios, we design the following generalization experiment.

To probe into the generalization ability of ours, we train our model on the MobiAct dataset. Then, we use this trained model to test the other five datasets. Fig. 10 shows the results with respect to four evaluation metrics (namely, accuracy, precision, recall, F_1-Measure) for ten-independent runs on the six datasets (i.e., MobiAct dataset, MotionSense dataset, WISDM dataset, UCI HAR dataset, UniMiB-SHAR dataset, and HHAR dataset) described earlier. *This experiment indicates that the proposed method could achieve excellent recognition performance in such a challenging scenario.*

4.6. Discussion about different meta learning

In this subsection, we compare with state-of-the-art meta-learning approaches to verify the effectiveness of ours. We evaluate these methods on the UTD-MHAD dataset. The Table 11 shows the results of performance comparison with different meta-learnings.

We can divide meta-learning methods into three categories [132]:

(1) Metric learning methods (i.e., MatchingNets [133], ProtoNets [134], RelationNets [73], Graph neural network (GraphNN) [135], Ridge regression [136], TransductiveProp [137], Fine-tuning Baseline [138], URT [139], DSN-MR [140], CDFS [141], DeepEMD [142], EPNet [143], ACC + Amphibian [144], FEAT [145], MsSoSN+SS+SD+DD [146], RFS [147], RFS+ CRAT [148], IDA [149], LR + ICI [150], FEAT+MLMT [151], BOHB [152], CSPN [153], SUR [154], SKD [155], TAFSSL [156], TRPN [157], TransMatch [158]) learn a similarity space in which learning is particularly efficient for few-shot examples.

(2) Memory network methods (i.e., Meta Networks [159], TADAM [160], MCFS [161], MRN [162]) learn to store "experience" when learning seen tasks and then generalize it to unseen tasks.

(3) Gradient descent based meta-learning methods (i.e., MAML [163], Meta-LSTM [164], MetaGAN [165], LEO [166], LGM-Net [167], CTM [168], MetaOptNet [169],

SIB+E3BM [170], and LSBC [171]) intend for adjusting the optimization algorithm so that the model can converge within a small number of optimization steps (with a few examples).

From above and Table 11, we can get the following three points:

First, metric-based methods propose that samples of the same class are close to each other, and samples of the different classes are far away from each other by simulating the metric distribution among samples. Generally speaking, the neural network is used to construct the embedding space (feature space) of samples, and some measure is used to calculate the similarity between samples. The sensory data used for human activity recognition are heterogeneous. The existing metric-based methods assume that the training and test data are independent and distributed equally, and does not model the distribution of the samples. Therefore, in the feature space of heterogeneous data, most data of the features are unserviceable, and similarity among these data cannot be performed effectively.

Second, gradient descent-based methods mostly directly optimize an initial feature representation. Based on this feature representation, the model can be efficiently adjusted using gradient updating based on a few images. However, in the human activity recognition task of this paper, even though the gradient descent-based approach may initially understand the resultant features of some data, most of the heterogeneous data with widely varying (or differing) distributions would still prevent this approach from enabling the adjustment of the gradient in subsequent training. Besides, this kind of model cannot model the distributions of data. As a result, the accuracy and the generalization ability of this kind method decrease.

Third, the memory network methods have an architecture that enhances memory capacity, which provides the ability to encode and retrieve new information quickly. In other words, this kind method focuses on what is in memory capacity (memory network). Most of the storage capacity is a useless function due to the varying distribution of samples in heterogeneous sensory data. As a result, the accuracy and generalizability of this method are reduced in human activity recognition tasks.

All in all, from Table 11, ours is better than others. *From the above discussion, it is clear that our approach is more effective than state-of-the-art meta-learning approaches.*

4.7. Discussion about different graph meta-learnings

In this subsection, we compare with some state-of-the-art graph meta-learnings on the UTD-MHAD dataset to verify the effectiveness of the proposed graph meta-Learning. Firstly, we introduce state-of-the-art graph meta-learnings. Then, we analyze discussion experiment results.

Graph-based approaches follow from the graph neural nets frameworks, aiming to solve the few-shot learning problems by the supervised message-passing networks. We review the theory and implementation of state-of-the-art graph meta-learnings, like the following:

★ AS-MAML [175]: a novel GNNs based graph meta-learner, which captures the features efficiently of sub-structures by fast adaptation mechanism.

★ AdarGCN [176]: a convolutional graph network-based few-shot learning method, which can perform adaptive aggregation based on a multi-head multi-level aggregation module.

★ APNet [177]: a neural nets model, propagating the attributes of every class on the category graph to its neighbors, aims to generate attribute vectors, followed by the nearest neighbor classifier with learnable similarity metric.

★ LRNN [178]: a neural nets model, which can use simple relational logic programs to capture advanced convolutional neural architectures.

★ DPGN [29]: this model builds a bridge connection between labeled and unlabeled samples in the form of similarity distribution.

★ EGNN [28]: this model uses the similarity/dissimilarity between samples and dynamically update both node and edge features for complicated interactions.

★ G-META [179]: this model uses local subgraphs to transfer subgraph-specific information and make the model learn the essential knowledge faster via meta gradients.

Table 11
Comparison with different meta-learnings on the UTD-MHAD dataset.

Few-shot learning method		Accuracy (%)	Precision (%)	Recall (%)	F_1-measure (%)
Data augmentation	Adv. ResNet [172]	80.71	80.84	80.90	80.77
	Delta-encoder [173]	80.77	80.95	81.01	80.86
	AFHN [174]	81.07	81.66	81.42	81.36
Memory network	Meta Networks [159]	80.88	81.09	81.24	80.99
	TADAM [160]	80.95	81.34	81.26	81.15
	MCFS [161]	80.96	81.39	81.39	81.18
	MRN [162]	82.09	83.09	82.78	82.59
Gradient descent	MAML [163]	81.14	81.89	81.43	81.51
	Meta-LSTM [164]	81.29	82.05	81.70	81.67
	MetaGAN [165]	81.30	82.15	81.83	81.73
	LEO [166]	81.60	82.53	82.41	82.06
	LGM-Net [167]	81.75	82.57	82.62	82.16
	CTM [168]	81.92	82.80	82.68	82.36
	MetaOptNet [169]	82.31	83.16	82.79	82.74
	SIB + E3BM [170]	82.43	83.24	83.10	82.83
	LSBC [171]	82.62	83.27	83.47	82.94
Metric learning	MatchingNets [133]	81.47	82.27	82.01	81.87
	ProtoNets [134]	81.50	82.29	82.20	81.89
	RelationNets [73]	81.51	82.44	82.24	81.97
	Graph neural network [135]	84.08	84.10	84.31	84.09
	Ridge regression [136]	83.25	83.53	83.60	83.39
	TransductiveProp [137]	85.13	85.96	85.06	85.54
	Fine-tuning Baseline [138]	84.19	84.27	84.52	84.23
	URT [139]	84.07	83.74	83.78	83.90
	DSN-MR [140]	85.08	85.58	84.76	85.33
	CDFS [141]	84.00	83.69	83.74	83.84
	DeepEMD [142]	84.37	84.93	84.60	84.65
	EPNet [143]	85.27	86.06	85.45	85.66
	ACC + Amphibian [144]	85.53	86.40	86.33	85.96
	FEAT [145]	85.49	86.34	85.94	85.92
	MsSoSN + SS + SD + DD [146]	84.07	83.88	84.24	83.97
	RFS [147]	84.33	84.29	84.54	84.31
	RFS + CRAT [148]	85.72	86.76	87.15	86.24
	IDA [149]	83.98	83.60	83.70	83.79
	LR + ICI [150]	85.37	86.08	85.58	85.72
	FEAT + MLMT [151]	83.04	83.31	83.55	83.17
	BOHB [152]	83.13	83.37	83.60	83.25
	CSPN [153]	84.07	83.80	84.11	83.93
	SUR [154]	84.90	85.09	84.70	84.99
	SKD [155]	82.79	83.29	83.50	83.04
	TAFSSL [156]	83.89	83.55	83.62	83.72
	TRPN [157]	84.97	85.16	84.75	85.06
	TransMatch [158]	85.43	86.26	85.63	85.84
	Ours	**98.99**	**98.97**	**99.23**	**99.10**

★ HOSP-GNN [180]: high-order structure-preserving graph neural network, which can explore the productive structure of the samples to predict the label of the queried data on the graph.

★ GPN [181]: this model learns to propagate messages between prototypes of different classes on the graph.

★ TPN [137]: this model brings the transductive setting into graph-based few-shot learning, which performs a Laplacian matrix to propagate labels from support set to query set in the graph.

★ MetaConcept [182]: this model learns to abstract concepts via the concept graph.

From Table 12, ours is 12.58%, 12.77%, 12.63%, 12.28%, 12.16%, 12.78%, 12.11%, 12.85%, 13.17%, 13.21%, and 12.59% higher than AS-MAML, AdarGCN, APNet, LRNN, DPGN, EGNN, G-META, HOSP-GNN, GPN, TPN, and MetaConcept, in terms of accuracy, respectively. By analyzing other results on this dataset, we can get similar conclusions.

From the above, it can be seen that our algorithm is optimal. Most of the algorithms do not take into account the diversity of the distribution of the sample data, so they are not as effective as our algorithm; some algorithms (e.g., DPGN, EGNN) take into account the distribution of the sample data, but ignore the relationship between the sample and the distribution and do not introduce an attentional mechanism to enhance and solidify the relation.

Table 12
Comparison with different graph meta-learnings on the UTD-MHAD dataset.

Methods	Accuracy (%)	Precision (%)	Recall (%)	F_1-measure (%)
AS-MAML [175]	86.41	88.16	88.17	87.28
AdarGCN [176]	86.22	87.58	87.63	86.89
APNet [177]	86.36	88.10	87.83	87.22
LRNN [178]	86.71	88.46	88.20	87.58
DPGN [29]	86.83	88.54	88.31	87.68
EGNN [28]	86.21	87.36	87.37	86.78
G-META [179]	86.88	88.63	88.33	87.75
HOSP-GNN [180]	86.14	87.19	87.33	86.66
GPN [181]	85.82	87.01	87.27	86.41
TPN [182]	85.78	86.83	87.22	86.30
MetaConcept [182]	86.40	88.15	87.90	87.27
Ours	**98.99**	**98.97**	**99.23**	**99.10**

All in all, from Table 12, ours is better than others. *From the above discussion, it is clear that our approach is more effective than state-of-the-art graph meta-learning approaches.*

Table 13
Comparison with different dual graph convolutional networks on the UTD-MHAD dataset.

Methods	Accuracy (%)	Precision (%)	Recall (%)	F_1-measure (%)
DSGCN [183]	79.01	78.99	79.26	79.00
DGCN [184]	79.09	79.19	79.27	79.14
DPGCNN [185]	79.10	79.29	79.33	79.19
TwinGCN [186]	79.35	79.40	79.37	79.37
Ours	**98.99**	**98.97**	**99.23**	**99.10**

4.8. Discussion about different dual graph convolutional networks

In this subsection, we compare with some state-of-the-art dual graph convolutional networks on the UTD-MHAD dataset to verify the effectiveness of the proposed dual graph convolutional networks. Firstly, we introduce state-of-the-art dual graph convolutional networks. Then, we analyze discussion experiment results.

We review the theory and implementation of state-of-the-art dual graph convolutional networks, like the following:

★ DSGCN [183]: this model proposed new convolutions in the spectral domain with a custom frequency profile while applying them in the spatial domain.

★ DGCN [184]: a scalable and straightforward semi-supervised learning method for graph-structured data in which only a tiny portion of the training data are labeled.

★ DPGCNN [185]: a graph convolutional architecture that alternates convolution-like operations on the graph and its dual.

★ TwinGCN [186]: this method can transform the primal graph into its dual form and have implemented two pipelines based on these two forms of the graph.

From Table 13, ours is 19.98%, 19.90%, 19.89%, and 19.64% higher than DSGCN, DGCN, DPGCNN, and TwinGCN , in terms of accuracy, respectively. By analyzing other results on this dataset, we can get similar conclusions.

From the above, it can be seen that our algorithms are optimal. Although these algorithms consider the distribution of the sample data, they ignore the relationship between the sample and the distribution and do not introduce an attentional mechanism to enhance and solidify the relation. Thus they are not as good as our algorithm.

All in all, from Table 13, ours is better than others. *From the above discussion, it is clear that our approach is more effective than state-of-the-art dual graph convolutional networks.*

4.9. Discussion about different attention mechanisms for dual graphs

In this subsection, we compare with some state-of-the-art dual graph attention mechanisms on the UTD-MHAD dataset to verify the effectiveness of the proposed dual graph attention mechanisms. Firstly, we introduce state-of-the-art dual graph attention mechanisms. Then, we analyze discussion experiment results.

We review the theory and implementation of state-of-the-art dual graph attention mechanisms, like the following:

★ DBAF-Net [187]: this model uses an adaptive center-offset sampling strategy, which allows each patch to adaptively determine the neighborhood range by finding the texture structure of the pixel to be classified.

★ DAGCN [188]: this model learns the importance of neighbors at different hops using a new attention graph convolution layer.

★ DPGN [29]: this model builds a bridge connection between labeled and unlabeled samples in the form of similarity distribution.

★ DAANE [189]: this model can preserve the consistency and complementarity between structures and attributes, and reduce the conflict caused by their discrepancy.

★ CADE [190]: a context-aware dual encoding framework, which can generate representations of nodes.

Table 14
Comparison with different attention mechanisms for dual graphs on the UTD-MHAD dataset.

Method	Accuracy (%)	Precision (%)	Recall (%)	F_1-measure (%)
DBAF-Net [187]	79.38	79.57	79.67	79.48
DAGCN [188]	79.49	79.59	79.75	79.54
DPGN [29]	79.82	79.88	79.78	79.85
DAANE [189]	79.88	80.27	79.95	80.07
CADE [190]	79.91	80.35	80.19	80.13
hpGAT [191]	80.20	80.40	80.24	80.30
GraphSAGE [192]	80.30	80.49	80.66	80.39
SNEA [193]	80.54	80.60	80.74	80.57
SDB-Net [194]	80.61	80.70	80.74	80.66
SpGAT [195]	80.67	80.82	80.81	80.75
Ours	**98.99**	**98.97**	**99.23**	**99.10**

★ hpGAT [191]: this model can incorporate high-order proximity information extracted from the hierarchical topological structure of the input graph.

★ GraphSAGE [192]: this model can leverage node feature information to generate node embeddings efficiently.

★ SNEA [193]: a masked self-attentional model, which can leverage the self-attention mechanism to estimate the importance coefficient for a pair of nodes.

★ SDB-Net [194]: a novel dual branch architecture, which has well alleviated the mismatch issue.

★ SpGAT [195]: this model learns representations for different frequency components regarding weighted filters and graph wavelet bases.

From Table 14, ours is 19.61%, 19.50%, 19.17%, 19.11%, 19.08%, 18.79%, 18.69%, 18.45%, 18.38%, and 18.32% higher than DBAF-Net, DAGCN, DPGN, DAANE, CADE, hpGAT, GraphSAGE, SNEA, SDB-Net, and SpGAT , in terms of accuracy, respectively. By analyzing other results on this dataset, we can get similar conclusions.

From the above, it can be seen that our algorithms are optimal. These algorithms are not as good as ours, although they consider the sample and distribution of the embedding vector and attentional mechanisms, respectively, and do not model the relation between the sample and distribution.

All in all, from Table 14, ours is better than others. *From the above discussion, it is clear that our approach is more effective than state-of-the-art dual graph attention mechanisms.*

4.10. Verification and discussion on the WIFI gesture recognition task

To better verify the robust performance of our algorithm, we extend our model to the Wireless Fidelity (WIFI) gesture recognition task. Firstly, we introduce the WIFI gesture recognition task and used datasets. Then, we analyze discussion experiment results.

❶ Description of The Task and Dataset

The WIFI gesture recognition [47] could recognize human gestures by analyzing the gesture pattern information involved in the influenced wireless signals. This task employs widespread WIFI signals to recognize human gestures without privacy leakage.

Widar 3.0 [47] contains raw channel state information (CSI) data as well as extracted signal features (Doppler frequency shift (DFS) and body-coordinate velocity profile(BVP)), including about 260,000 motion instances collected in 75 different scenarios (including different locations, orientations and environments) with a total duration of more than 144 hours and a data size of about 325 GB. We use four evaluation metrics: accuracy (%), precision (%), recall (%), F_1-Measure (%) on this discussion.

❷ Discussion on WIFI Gesture Recognition Task

We conduct experiments with the following baselines to compare them with our proposed model, including ST-GAN [196] and WiHF [197]. Besides, it is evident that our approach is better than others in

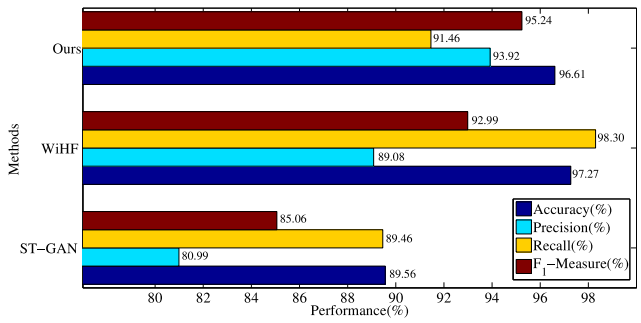

Fig. 11. The results of verification and discussion on the WIFI gesture recognition task.

Table 15
F_1-measure (%) for cross modal human action recognition results of all compared methods by using the MMAct dataset.

Method	Cross subject	Cross view	Cross scene	Cross session
Student [48]	64.44	62.21	57.91	69.2
Multi-Teachers [48]	62.67	68.13	67.31	70.53
SMD [198]	63.89	66.31	61.56	71.23
MMD [48]	64.33	68.19	62.23	72.08
MMAD [48]	66.45	70.33	64.12	74.58
Ours	**74.56**	**74.76**	**74.74**	**74.59**

Fig. 11. Specifically, our approach is 10.18% and 2.25% higher than ST-GAN and WiHF, in terms of F_1-Measure, respectively.

From above, our approach has more robust performance than the state-of-the-arts on the WIFI gesture recognition task.

4.11. Verification and discussion on the cross-modal human action understanding task

To better verify the robust performance of our algorithm, we extend our model to the cross-modal human action understanding task. Firstly, we introduce the cross-modal human action understanding task and used datasets. Then, we analyze discussion experiment results.

❶ Description of The Task and Dataset

The cross-modal human action understanding [48] could recognize human action by analyzing the multimodal action information, including the vision-based and sensor-based information. This task employs multimodal action information to understand human action.

MMAct dataset [48] contains more than 36,000 trimmed clips featuring seven modes captured from 20 subjects, including RGB video, keypoints, acceleration, gyroscope, direction, WIFI, and pressure signals. MMAct is designed according to a semi-natural data collection protocol that performs random wandering between the end of the current operation and the start of the next one. The action is performed only after the external monitor has issued a start flag. The protocol ensures that the action will occur randomly in the action area in order to provide various action videos in different camera views. We use the evaluation metrics: F_1-Measure (%) on this discussion. Following the settings of the MMAct, we evaluate the model from four perspectives: Cross Subject, Cross View, Cross Scene, and Cross Session. Specifically, Cross Subject: samples from 80% of the subjects (subject numbers from 1 to 16) have been used for the training model and the remaining 20% for testing; Cross View: samples from 3 views of all subjects have been used to train the model, and a fourth view has been used for testing; Cross Scene: samples of scenes other than the occluded scenes for all objects have been used to train the model and occluded scenes for all objects have been used for testing; Cross Session: samples drawn from the top 80% of sessions in ascending order of session ID for each topic have been used for the training model, and the remaining sessions have been used for testing.

❷ Discussion on Cross-Modal Human Action Understanding Task

We conduct experiments with the following baselines to compare them with our proposed model, including Student [48], Multi-Teachers [48], SMD [198], MMD [48], and MMAD [48]. Besides, it is obvious that our approach is better than others in Table 15. Specifically, our approach is 10.12%, 11.89%, 10.67%, 10.23%, and 8.11% higher than Student, Multi-Teachers, SMD, MMD, and MMAD, in terms of Cross Subject, respectively. By analyzing other results on the MMAct dataset, we can get similar conclusions *From above, our approach has more robust performance than the state-of-the-arts on the cross-modal human action understanding task.*

5. Conclusion and future work

In this paper, we propose a novel solution, a meta-learning-based graph prototypical model with priority attention mechanism, to address the challenges of sensor-based human activity recognition in IoT. The meta-learning-based graph prototypical model focuses on the sample features and sample distribution characteristics. Priority attention mechanism, similar to priority map in human attention, aims to gain the sample-level, distribution-level, relation embeddings from sample features, sample distribution characteristics, and their combinations simultaneously. We conduct extensive experiments and demonstrate the effectiveness of our model compared with a number of baseline methods. Moreover, we also extend our model to other human activity recognition tasks.

In the future, following this direction of this model, we will tackle more challenging and complex scenarios such as intelligent transportation [199]. On the other hand, we will collect data ourselves to form datasets to further explore the model for fine-grained human activity classification tasks.

Ethical approval

This article does not contain any studies with human participants or animals performed by any of the authors.

CRediT authorship contribution statement

Wenbo Zheng: Conceptualization, Methodology, Software, Validation, Investigation, Writing – original draft, Writing – review & editing. **Lan Yan:** Methodology, Investigation, Writing – original draft. **Chao Gou:** Conceptualization, Resources, Writing – review & editing, Project administration, Funding acquisition. **Fei-Yue Wang:** Supervision, Funding acquisition.

Declaration of competing interest

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

Acknowledgment

This work is supported in part by National Key R&D Program of China (2020YFB1600400), in part by the Key Research and Development Program of Guangzhou (202007050002), in part by the National Natural Science Foundation of China (61806198, 61533019, U1811463), and in part by National Key R&D Program of China (2018AAA0101502).

References

[1] X. Deng, Y. Jiang, L.T. Yang, M. Lin, L. Yi, M. Wang, Data fusion based coverage optimization in heterogeneous sensor networks: A survey, Inf. Fusion 52 (2019) 90–105, http://dx.doi.org/10.1016/j.inffus.2018.11.020, URL http://www.sciencedirect.com/science/article/pii/S1566253518306535.

[2] W. Ding, X. Jing, Z. Yan, L.T. Yang, A survey on data fusion in internet of things: Towards secure and privacy-preserving fusion, Inf. Fusion 51 (2019) 129–144, http://dx.doi.org/10.1016/j.inffus.2018.12.001, URL http://www.sciencedirect.com/science/article/pii/S1566253518304731.

[3] J. Lu, X. Zheng, M. Sheng, J. Jin, S. Yu, Efficient human activity recognition using a single wearable sensor, IEEE Internet Things J. (2020) 1.

[4] R. Chavarriaga, H. Sagha, A. Calatroni, S.T. Digumarti, G. Tröster, J. del R. Millán, D. Roggen, The opportunity challenge: A benchmark database for on-body sensor-based activity recognition, Pattern Recognit. Lett. 34 (15) (2013) 2033–2042, http://dx.doi.org/10.1016/j.patrec.2012.12.014, smart Approaches for Human Action Recognition. URL http://www.sciencedirect.com/science/article/pii/S0167865512004205.

[5] K. Chen, D. Zhang, L. Yao, B. Guo, Z. Yu, Y. Liu, Deep learning for sensor-based human activity recognition: overview, challenges and opportunities, 2020, arXiv preprint arXiv:2001.07416.

[6] N.Y. Hammerla, S. Halloran, T. Plötz, Deep, convolutional, and recurrent models for human activity recognition using wearables, in: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, 2016, pp. 1533–1540.

[7] E.J. Escobedo Cardenas, G.C. Chavez, Multimodal hand gesture recognition combining temporal and pose information based on cnn descriptors and histogram of cumulative magnitudes, J. Vis. Commun. Image Represent. 71 (2020) 102772, http://dx.doi.org/10.1016/j.jvcir.2020.102772, URL http://www.sciencedirect.com/science/article/pii/S1047320320300225.

[8] Naveenkumar M., Domnic S., Deep ensemble network using distance maps and body part features for skeleton based action recognition, Pattern Recognit. 100 (2020) 107125, http://dx.doi.org/10.1016/j.patcog.2019.107125, URL http://www.sciencedirect.com/science/article/pii/S0031320319304261.

[9] P. Wang, Z. Li, Y. Hou, W. Li, Action recognition based on joint trajectory maps using convolutional neural networks, in: Proceedings of the 24th ACM International Conference on Multimedia, MM '16, Association for Computing Machinery, New York, NY, USA, 2016, pp. 102–106, http://dx.doi.org/10.1145/2964284.2967191.

[10] M. Liu, F. Meng, C. Chen, S. Wu, Joint dynamic pose image and space time reversal for human action recognition from videos, in: AAAI Conference on Artificial Intelligence (AAAI), 2019.

[11] C. Si, Y. Jing, W. Wang, L. Wang, T. Tan, Skeleton-based action recognition with hierarchical spatial reasoning and temporal stack learning network, Pattern Recognit. 107 (2020) 107511, http://dx.doi.org/10.1016/j.patcog.2020.107511, URL http://www.sciencedirect.com/science/article/pii/S0031320320303149.

[12] R. Zhao, K. Wang, H. Su, Q. Ji, Bayesian graph convolution lstm for skeleton based action recognition, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019.

[13] H. Qian, S.J. Pan, C. Miao, Sensor-based activity recognition via learning from distributions, in: Thirty-Second AAAI Conference on Artificial Intelligence, 2018.

[14] H. Qian, S.J. Pan, C. Miao, Distribution-based semi-supervised learning for activity recognition, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 7699–7706.

[15] N.-n. Zheng, Z.-y. Liu, P.-j. Ren, Y.-q. Ma, S.-t. Chen, S.-y. Yu, J.-r. Xue, B.-d. Chen, F.-y. Wang, Hybrid-augmented intelligence: collaboration and cognition, Front. Inf. Technol. Electron. Eng. 18 (2) (2017) 153–179, http://dx.doi.org/10.1631/FITEE.1700053.

[16] W. Zheng, L. Yan, C. Gou, Z.-C. Zhang, J. Jason Zhang, M. Hu, F.-Y. Wang, Pay attention to doctor–patient dialogues: Multi-modal knowledge graph attention image-text embedding for covid-19 diagnosis, Inf. Fusion 75 (2021) 168–185, http://dx.doi.org/10.1016/j.inffus.2021.05.015.

[17] W. Zheng, L. Yan, C. Gou, F.-Y. Wang, Km4: Visual reasoning via knowledge embedding memory model with mutual modulation, Inf. Fusion 67 (2021) 14–28, http://dx.doi.org/10.1016/j.inffus.2020.10.007.

[18] W. Zheng, L. Yan, C. Gou, F.-Y. Wang, Two heads are better than one: Hypergraph-enhanced graph reasoning for visual event ratiocination, in: M. Meila, T. Zhang (Eds.), Proceedings of the 38th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 139, PMLR, 2021, pp. 12747–12760.

[19] J. Xu, C. Li, B. Cui, K. Yang, Y. Xu, Pfgdf: Pruning filter via gaussian distribution feature for deep neural networks acceleration, 2020, arXiv:2006.12963.

[20] S. Dey, N. Roy, W. Xu, R.R. Choudhury, S. Nelakuditi, Accelprint: Imperfections of accelerometers make smartphones trackable, in: 21st Annual Network and Distributed System Security Symposium, NDSS 2014, San Diego, California, USA, February 23-26, 2014, The Internet Society, 2014, URL https://www.ndss-symposium.org/ndss2014/accelprint-imperfections-accelerometers-make-smartphones-trackable.

[21] L. Minh Dang, K. Min, H. Wang, M. Jalil Piran, C. Hee Lee, H. Moon, Sensor-based and vision-based human activity recognition: A comprehensive survey, Pattern Recognit. 108 (2020) 107561, http://dx.doi.org/10.1016/j.patcog.2020.107561, URL http://www.sciencedirect.com/science/article/pii/S0031320320303642.

[22] N.Y. Hammerla, S. Halloran, T. Plötz, Deep, convolutional, and recurrent models for human activity recognition using wearables, in: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI'16, AAAI Press, 2016, pp. 1533–1540.

[23] N. Golestani, M. Moghaddam, Human activity recognition using magnetic induction-based motion signals and deep recurrent neural networks, Nature Commun. 11 (1) (2020) 1551, http://dx.doi.org/10.1038/s41467-020-15086-2.

[24] C. Luo, W. Li, X. Fan, H. Yang, J. Ni, X. Zhang, G. Xin, P. Shi, Positioning technology of mobile vehicle using self-repairing heterogeneous sensor networks, J. Netw. Comput. Appl. 93 (2017) 110–122, http://dx.doi.org/10.1016/j.jnca.2017.05.012, URL http://www.sciencedirect.com/science/article/pii/S1084804517302096.

[25] H.H. Zhuo, Y. Zha, S. Kambhampati, X. Tian, Discovering underlying plans based on shallow models, ACM Trans. Intell. Syst. Technol. (TIST) 11 (2) (2020) 1–30.

[26] Y. Wang, Q. Yao, J.T. Kwok, L.M. Ni, Generalizing from a few examples: A survey on few-shot learning, ACM Comput. Surv. 53 (3) (2020) 1–34.

[27] N. Bendre, H.T. Marín, P. Najafirad, Learning from few samples: A survey, 2020, arXiv preprint arXiv:2007.15484.

[28] J. Kim, T. Kim, S. Kim, C.D. Yoo, Edge-labeling graph neural network for few-shot learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 11–20.

[29] L. Yang, L. Li, Z. Zhang, X. Zhou, E. Zhou, Y. Liu, Dpgn: Distribution propagation graph network for few-shot learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 13390–13399.

[30] T.C. Sprague, J.T. Serences, Attention modulates spatial priority maps in the human occipital, parietal and frontal cortices, Nature Neurosci. 16 (12) (2013) 1879–1887.

[31] C. Mo, D. He, F. Fang, Attention priority map of face images in human early visual cortex, J. Neurosci. 38 (1) (2018) 149–157.

[32] P.C. Klink, P. Jentgens, J.A. Lorteije, Priority maps explain the roles of value, attention, and salience in goal-oriented behavior, J. Neurosci. 34 (42) (2014) 13867–13869.

[33] G.J. Zelinsky, J.W. Bisley, The what, where, and why of priority maps and their interactions with visual working memory, Ann. New York Acad. Sci. 1339 (1) (2015) 154.

[34] C. Chen, R. Jafari, N. Kehtarnavaz, Utd-mhad: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor, in: 2015 IEEE International Conference on Image Processing (ICIP), 2015, pp. 168–172.

[35] B. Barshan, M.C. Yüksek, Recognizing daily and sports activities in two open source machine learning environments using body-worn sensor units, Comput. J. 57 (11) (2014) 1649–1667.

[36] O. Banos, R. Garcia, J.A. Holgado-Terriza, M. Damas, H. Pomares, I. Rojas, A. Saez, C. Villalonga, Mhealthdroid: A novel framework for agile development of mobile health applications, in: L. Pecchia, L.L. Chen, C. Nugent, J. Bravo (Eds.), Ambient Assisted Living and Daily Activities, Springer International Publishing, Cham, 2014, pp. 91–98.

[37] A. Anjum, M.U. Ilyas, Activity recognition using smartphone sensors, in: 2013 IEEE 10th Consumer Communications and Networking Conference (CCNC), 2013, pp. 914–919.

[38] A. Reiss, D. Stricker, Introducing a new benchmarked dataset for activity monitoring, in: 2012 16th International Symposium on Wearable Computers, 2012, pp. 108–109.

[39] F. Cruciani, C. Sun, S. Zhang, C. Nugent, C. Li, S. Song, C. Cheng, I. Cleland, P. Mccullagh, A public domain dataset for human activity recognition in free-living conditions, in: 2019 IEEE SmartWorld, Ubiquitous Intelligence Computing, Advanced Trusted Computing, Scalable Computing Communications, Cloud Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), 2019, pp. 166–171.

[40] M. Bachlin, M. Plotnik, D. Roggen, I. Maidan, J.M. Hausdorff, N. Giladi, G. Troster, Wearable assistant for parkinson's disease patients with the freezing of gait symptom, IEEE Trans. Inf. Technol. Biomed. 14 (2) (2010) 436–446.

[41] A. Stisen, H. Blunck, S. Bhattacharya, T.S. Prentow, M.B. Kjærgaard, A. Dey, T. Sonne, M.M. Jensen, Smart devices are different: Assessing and mitigating-mobile sensing heterogeneities for activity recognition, in: Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems, SenSys '15, Association for Computing Machinery, New York, NY, USA, 2015, pp. 127–140, http://dx.doi.org/10.1145/2809695.2809718.

[42] D. Micucci, M. Mobilio, P. Napoletano, Unimib shar: A dataset for human activity recognition using acceleration data from smartphones, Appl. Sci. 7 (10) (2017) 1101, http://dx.doi.org/10.3390/app7101101.

[43] G. Vavoulas, C. Chatzaki, T. Malliotakis, M. Pediaditis, M. Tsiknakis, The mobiact dataset: Recognition of activities of daily living using smartphones, in: Proceedings of the International Conference on Information and Communication Technologies for Ageing Well and E-Health - Volume 1: ICT4AWE, (ICT4AGEINGWELL 2016), INSTICC, SciTePress, 2016, pp. 143–151, http://dx.doi.org/10.5220/0005792401430151.

[44] M. Malekzadeh, R.G. Clegg, A. Cavallaro, H. Haddadi, Mobile sensor data anonymization, in: Proceedings of the International Conference on Internet of Things Design and Implementation, IoTDI '19, Association for Computing Machinery, New York, NY, USA, 2019, pp. 49–58, http://dx.doi.org/10.1145/3302505.3310068.

[45] J.-L. Reyes-Ortiz, L. Oneto, A. Samà, X. Parra, D. Anguita, Transition-aware human activity recognition using smartphones, Neurocomputing 171 (2016) 754–767, http://dx.doi.org/10.1016/j.neucom.2015.07.085, URL http://www.sciencedirect.com/science/article/pii/S0925231215010930.

[46] M. Zhang, A.A. Sawchuk, Usc-had: A daily activity dataset for ubiquitous activity recognition using wearable sensors, in: Proceedings of the 2012 ACM Conference on Ubiquitous Computing, UbiComp '12, Association for Computing Machinery, New York, NY, USA, 2012, pp. 1036–1043, http://dx.doi.org/10.1145/2370216.2370438.

[47] Y. Zheng, Y. Zhang, K. Qian, G. Zhang, Y. Liu, C. Wu, Z. Yang, Zero-effort cross-domain gesture recognition with wi-fi, in: Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services, MobiSys '19, Association for Computing Machinery, New York, NY, USA, 2019, pp. 313–325, http://dx.doi.org/10.1145/3307334.3326081.

[48] Q. Kong, Z. Wu, Z. Deng, M. Klinkigt, B. Tong, T. Murakami, Mmact: A large-scale dataset for cross modal human action understanding, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 8658–8667.

[49] J. Liu, Z. Wang, H. Liu, Hds-sp: A novel descriptor for skeleton-based human action recognition, Neurocomputing 385 (2020) 22–32, http://dx.doi.org/10.1016/j.neucom.2019.11.048, URL http://www.sciencedirect.com/science/article/pii/S0925231219316509.

[50] J. Liu, N. Akhtar, A. Mian, Skepxels: Spatio-temporal image representation of human skeleton joints for action recognition, in: CVPR Workshops, 2019.

[51] B.A. Ahmed, A.M. Jamel, Human activity recognition based on parallel approximation kernel k-means algorithm, Comput. Syst. Sci. Eng. 35 (6) (2020) 441–456, http://dx.doi.org/10.32604/csse.2020.35.441, URL http://www.techscience.com/csse/v35n6/40722.

[52] M. Humayun Kabir, Keshav Thapa, Jae-Young Yang, Sung-Hyun Yang, State-space based linear modeling for human activity recognition in smart space, Intell. Autom. Soft Comput. 25 (4) (2019) 673–681, http://dx.doi.org/10.31209/2018.100000035, URL http://www.techscience.com/iasc/v25n4/39694.

[53] S.-H.K. Jeong-Sik Park, Noise cancellation based on voice activity detection using spectral variation for speech recognition in smart home devices, Intell. Autom. Soft Comput. 26 (1) (2020) 149–159, http://dx.doi.org/10.31209/2019.100000136, URL http://www.techscience.com/iasc/v26n1/39851.

[54] J.-H.K. Sun-Taag Choe, We-Duke Cho, K.-H. Kim, Reducing operational time complexity of k-nn algorithms using clustering in wrist-activity recognition, Intell. Autom. Soft Comput. 26 (4) (2020) 679–691, http://dx.doi.org/10.32604/iasc.2020.010102, URL http://www.techscience.com/iasc/v26n4/40272.

[55] Y. Li, R. Xia, X. Liu, Learning shape and motion representations for view invariant skeleton-based action recognition, Pattern Recognit. 103 (2020) 107293, http://dx.doi.org/10.1016/j.patcog.2020.107293, URL http://www.sciencedirect.com/science/article/pii/S0031320320300972.

[56] C. Si, Y. Jing, W. Wang, L. Wang, T. Tan, Skeleton-based action recognition with spatial reasoning and temporal stack learning, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018.

[57] M.E. Hussein, M. Torki, M.A. Gowayyed, M. El-Saban, Human action recognition using a temporal hierarchy of covariance descriptors on 3d joint locations, in: Twenty-Third International Joint Conference on Artificial Intelligence, 2013.

[58] Y. Hou, Z. Li, P. Wang, W. Li, Skeleton optical spectra-based action recognition using convolutional neural networks, IEEE Trans. Circuits Syst. Video Technol. 28 (3) (2018) 807–811.

[59] B. Zhang, Y. Yang, C. Chen, L. Yang, J. Han, L. Shao, Action recognition using 3d histograms of texture and a multi-class boosting classifier, IEEE Trans. Image Process. 26 (10) (2017) 4648–4660.

[60] H. Yang, D. Yan, L. Zhang, D. Li, Y. Sun, S. You, S.J. Maybank, Feedback graph convolutional network for skeleton-based action recognition, 2020, arXiv preprint arXiv:2003.07564.

[61] H. Qian, S.J. Pan, B. Da, C. Miao, A novel distribution-embedded neural network for sensor-based activity recognition, in: Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19, International Joint Conferences on Artificial Intelligence Organization, 2019, pp. 5614–5620, http://dx.doi.org/10.24963/ijcai.2019/779.

[62] W. Gao, L. Zhang, Q. Teng, H. Wu, F. Min, J. He, Danhar: Dual attention network for multimodal human activity recognition using wearable sensors, 2020, arXiv preprint arXiv:2006.14435.

[63] Y. Tang, Q. Teng, L. Zhang, F. Min, J. He, Efficient convolutional neural networks with smaller filters for human activity recognition using wearable sensors, 2020, arXiv preprint arXiv:2005.03948.

[64] J.-H. Choi, J.-S. Lee, Embracenet: A robust deep learning architecture for multimodal classification, Inf. Fusion 51 (2019) 259–270, http://dx.doi.org/10.1016/j.inffus.2019.02.010, URL http://www.sciencedirect.com/science/article/pii/S1566253517308242.

[65] Y. Guan, T. Plötz, Ensembles of deep lstm learners for activity recognition using wearables, Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 1 (2) (2017) http://dx.doi.org/10.1145/3090076.

[66] K. Xia, J. Huang, H. Wang, Lstm-cnn architecture for human activity recognition, IEEE Access 8 (2020) 56855–56866.

[67] C. Pham, S. Nguyen-Thai, H. Tran-Quang, S. Tran, H. Vu, T. Tran, T. Le, Senscapsnet: Deep neural network for non-obtrusive sensing based human activity recognition, IEEE Access 8 (2020) 86934–86946.

[68] Q. Teng, K. Wang, L. Zhang, J. He, The layer-wise training convolutional neural networks using local loss for sensor-based human activity recognition, IEEE Sens. J. 20 (13) (2020) 7265–7274.

[69] C. Hu, Y. Chen, X. Peng, H. Yu, C. Gao, L. Hu, A novel feature incremental learning method for sensor-based activity recognition, IEEE Trans. Knowl. Data Eng. 31 (6) (2019) 1038–1050.

[70] C. Hu, Y. Chen, L. Hu, X. Peng, A novel random forests based class incremental learning method for activity recognition, Pattern Recognit. 78 (2018) 277–290, http://dx.doi.org/10.1016/j.patcog.2018.01.025, URL http://www.sciencedirect.com/science/article/pii/S0031320318300360.

[71] L. Bai, L. Yao, X. Wang, S.S. Kanhere, B. Guo, Z. Yu, Adversarial multi-view networks for activity recognition, Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 4 (2) (2020) http://dx.doi.org/10.1145/3397323.

[72] Abdu Gumaei, Mabrook Al-Rakhami, Hussain AlSalman, Sk. Md. Mizanur Rahman, Atif Alamri, Dl-har: Deep learning-based human activity recognition framework for edge computing, Comput. Mater. Contin. 65 (2) (2020) 1033–1057, http://dx.doi.org/10.32604/cmc.2020.011740, URL http://www.techscience.com/cmc/v65n2/39861.

[73] F. Sung, Y. Yang, L. Zhang, T. Xiang, P.H.S. Torr, T.M. Hospedales, Learning to compare: Relation network for few-shot learning, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 1199–1208, http://dx.doi.org/10.1109/CVPR.2018.00131.

[74] T. Hospedales, A. Antoniou, P. Micaelli, A. Storkey, Meta-learning in neural networks: A survey, 2020, arXiv preprint arXiv:2004.05439.

[75] L. Ning, T.T. Georgiou, A. Tannenbaum, Matrix-valued monge-kantorovich optimal mass transport, in: 52nd IEEE Conference on Decision and Control, 2013, pp. 3906–3911, http://dx.doi.org/10.1109/CDC.2013.6760486.

[76] N. Lei, K. Su, L. Cui, S.-T. Yau, D. Xianfeng Gu, A geometric view of optimal transportation and generative model, 2017, ArXiv e-prints arXiv:1710.05488.

[77] N. Lei, Z. Luo, S.-T. Yau, D. Xianfeng Gu, Geometric understanding of deep learning, 2018, ArXiv e-prints arXiv:1805.10451.

[78] R. Flamary, N. Courty, Pot python optimal transport library, 2017, URL https://github.com/rflamary/POT.

[79] M.Z. Alaya, M. Berar, G. Gasso, A. Rakotomamonjy, Screening sinkhorn algorithm for regularized optimal transport, in: Advances in Neural Information Processing Systems, 2019, pp. 12169–12179.

[80] M. Tan, R. Pang, Q.V. Le, Efficientdet: Scalable and efficient object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 10781–10790.

[81] M. Tan, Q. Le, EfficientNet: Rethinking model scaling for convolutional neural networks, in: K. Chaudhuri, R. Salakhutdinov (Eds.), Proceedings of the 36th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 97, PMLR, Long Beach, California, USA, 2019, pp. 6105–6114, URL http://proceedings.mlr.press/v97/tan19a.html.

[82] T. Van de Cruys, Two multivariate generalizations of pointwise mutual information, in: Proceedings of the Workshop on Distributional Semantics and Compositionality, 2011, pp. 16–20.

[83] G. Bouma, Normalized (pointwise) mutual information in collocation extraction, in: Proceedings of GSCL, 2009, pp. 31–40.

[84] F. Scarselli, M. Gori, A.C. Tsoi, M. Hagenbuchner, G. Monfardini, The graph neural network model, IEEE Trans. Neural Netw. 20 (1) (2008) 61–80.

[85] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780.

[86] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, 2016, arXiv preprint arXiv:1609.02907.

[87] X. Glorot, A. Bordes, Y. Bengio, Deep sparse rectifier neural networks, in: Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, 2011, pp. 315–323.

[88] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, Y. Bengio, Graph attention networks, in: International Conference on Learning Representations, 2018, URL https://openreview.net/forum?id=rJXMpikCZ.

[89] M. Jaderberg, K. Simonyan, A. Zisserman, et al., Spatial transformer networks, in: Advances in Neural Information Processing Systems, 2015, pp. 2017–2025.

[90] P.R. Halmos, Finite-Dimensional Vector Spaces, Courier Dover Publications, 2017.

[91] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016, http://www.deeplearningbook.org.

[92] X. Ma, H. Huang, Y. Wang, S. Romano, S. Erfani, J. Bailey, Normalized loss functions for deep learning with noisy labels, 2020, arXiv preprint arXiv:2006.13554.

[93] Y. Wang, X. Ma, Z. Chen, Y. Luo, J. Yi, J. Bailey, Symmetric cross entropy for robust learning with noisy labels, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 322–330.

[94] J. Cao, W. Li, C. Ma, Z. Tao, Optimizing multi-sensor deployment via ensemble pruning for wearable activity recognition, Inf. Fusion 41 (2018) 68–79, http://dx.doi.org/10.1016/j.inffus.2017.08.002, URL http://www.sciencedirect.com/science/article/pii/S1566253517304803.

[95] S. Ha, S. Choi, Convolutional neural networks for human activity recognition using multiple accelerometer and gyroscope sensors, in: 2016 International Joint Conference on Neural Networks (IJCNN), 2016, pp. 381–388.

[96] M. Niepert, M. Ahmed, K. Kutzkov, Learning convolutional neural networks for graphs, in: M.F. Balcan, K.Q. Weinberger (Eds.), Proceedings of the 33rd International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 48, PMLR, New York, New York, USA, 2016, pp. 2014–2023, URL http://proceedings.mlr.press/v48/niepert16.html.

[97] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, arXiv preprint arXiv:1412.6980.

[98] Wikipedia contributors, Linear interpolation — Wikipedia, the free encyclopedia, 2020, [Online; accessed 6-August-2020]. URL https://en.wikipedia.org/w/index.php?title=Linear_interpolation&oldid=956137759.

[99] Wikipedia contributors, Normalization (statistics) — Wikipedia, the free encyclopedia, 2020, [Online; accessed 6-August-2020]. URL https://en.wikipedia.org/w/index.php?title=Normalization_(statistics)&oldid=946937216.

[100] A.A. Rusu, D. Rao, J. Sygnowski, O. Vinyals, R. Pascanu, S. Osindero, R. Hadsell, Meta-learning with latent embedding optimization, 2018, arXiv preprint arXiv:1807.05960.

[101] H.-J. Ye, H. Hu, D.-C. Zhan, F. Sha, Learning embedding adaptation for few-shot learning, 2018, arXiv preprint arXiv:1812.03664.

[102] C. Li, Y. Hou, P. Wang, W. Li, Joint distance maps based action recognition with convolutional neural networks, IEEE Signal Process. Lett. 24 (5) (2017) 624–628.

[103] F. Meng, H. Liu, Y. Liang, J. Tu, M. Liu, Sample fusion network: An end-to-end data augmentation network for skeleton-based human action recognition, IEEE Trans. Image Process. 28 (11) (2019) 5281–5295.

[104] R. Mojarad, F. Attal, A. Chibani, Y. Amirat, Automatic classification error detection and correction for robust human activity recognition, IEEE Robot. Autom. Lett. 5 (2) (2020) 2208–2215.

[105] Y. Dong, X. Li, J. Dezert, M.O. Khyam, M. Noor-A-Rahim, S.S. Ge, Dezert-smarandache theory-based fusion for human activity recognition in body sensor networks, IEEE Trans. Ind. Inf. 16 (11) (2020) 7138–7149.

[106] S. Feng, M.F. Duarte, Few-shot learning-based human activity recognition, Expert Syst. Appl. 138 (2019) 112782, http://dx.doi.org/10.1016/j.eswa.2019.06.070, URL http://www.sciencedirect.com/science/article/pii/S0957417419304786.

[107] X. Zhang, Y. Wong, M.S. Kankanhalli, W. Geng, Hierarchical multi-view aggregation network for sensor-based human activity recognition, PLOS ONE 14 (9) (2019) 1–20, http://dx.doi.org/10.1371/journal.pone.0221390.

[108] Y. Zhang, Z. Zhang, Y. Zhang, J. Bao, Y. Zhang, H. Deng, Human activity recognition based on motion sensor using u-net, IEEE Access 7 (2019) 75213–75226.

[109] S. Mahmud, M. Tonmoy, K.K. Bhaumik, A. Rahman, M.A. Amin, M. Shoyaib, M.A.H. Khan, A.A. Ali, Human activity recognition from wearable sensor data using self-attention, 2020, arXiv preprint arXiv:2003.09018.

[110] C.F.S. Leite, Y. Xiao, Improving cross-subject activity recognition via adversarial learning, IEEE Access 8 (2020) 90542–90554.

[111] S. Mohamad, M. Sayed-Mouchaweh, A. Bouchachia, Online active learning for human activity recognition from sensory data streams, Neurocomputing 390 (2020) 341–358, http://dx.doi.org/10.1016/j.neucom.2019.08.092, URL http://www.sciencedirect.com/science/article/pii/S0925231219314493.

[112] W. Cheng, S. Erfani, R. Zhang, R. Kotagiri, Learning datum-wise sampling frequency for energy-efficient human activity recognition, in: Thirty-Second AAAI Conference on Artificial Intelligence, 2018.

[113] M.Z. Uddin, M.M. Hassan, A. Alsanad, C. Savaglio, A body sensor data fusion and deep recurrent neural network-based behavior recognition approach for robust healthcare, Inf. Fusion 55 (2020) 105–115, http://dx.doi.org/10.1016/j.inffus.2019.08.004, URL http://www.sciencedirect.com/science/article/pii/S1566253519302581.

[114] K. Chen, L. Yao, D. Zhang, X. Wang, X. Chang, F. Nie, A semisupervised recurrent convolutional attention model for human activity recognition, IEEE Trans. Neural Netw. Learn. Syst. 31 (5) (2020) 1747–1756.

[115] Z. Qin, Y. Zhang, S. Meng, Z. Qin, K.-K.R. Choo, Imaging and fusing time series for wearable sensor-based human activity recognition, Inf. Fusion 53 (2020) 80–87, http://dx.doi.org/10.1016/j.inffus.2019.06.014, URL http://www.sciencedirect.com/science/article/pii/S1566253519302180.

[116] S. Savvaki, G. Tsagkatakis, A. Panousopoulou, P. Tsakalides, Matrix and tensor completion on a human activity recognition framework, IEEE J. Biomed. Health Inf. 21 (6) (2017) 1554–1561.

[117] W. Lu, F. Fan, J. Chu, P. Jing, S. Yuting, Wearable computing for internet of things: A discriminant approach for human activity recognition, IEEE Internet Things J. 6 (2) (2019) 2749–2759.

[118] A. Youssef, J. Aerts, B. Vanrumste, S. Luca, A localised learning approach applied to human activity recognition, IEEE Intell. Syst. (2020) 1.

[119] H. Zhang, Z. Xiao, J. Wang, F. Li, E. Szczerbicki, A novel iot-perceptive human activity recognition (har) approach using multihead convolutional attention, IEEE Internet Things J. 7 (2) (2020) 1072–1080.

[120] M.A. Alsheikh, A. Selim, D. Niyato, L. Doyle, S. Lin, H.-P. Tan, Deep activity recognition models with triaxial accelerometers, in: Workshops at the Thirtieth AAAI Conference on Artificial Intelligence, 2016.

[121] A. Saeed, T. Ozcelebi, J. Lukkien, Multi-task self-supervised learning for human activity detection, Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 3 (2) (2019) http://dx.doi.org/10.1145/3328932.

[122] M. Lv, W. Xu, T. Chen, A hybrid deep convolutional and recurrent neural network for complex activity recognition using multimodal sensors, Neurocomputing 362 (2019) 33–40, http://dx.doi.org/10.1016/j.neucom.2019.06.051, URL http://www.sciencedirect.com/science/article/pii/S0925231219309361.

[123] S. Xu, Q. Tang, L. Jin, Z. Pan, A cascade ensemble learning model for human activity recognition with smartphones, Sensors 19 (10) (2019) 2307, http://dx.doi.org/10.3390/s19102307.

[124] V. Bianchi, M. Bassoli, G. Lombardo, P. Fornacciari, M. Mordonini, I. De Munari, Iot wearable sensor and deep learning: An integrated approach for personalized human activity recognition in a smart home environment, IEEE Internet Things J. 6 (5) (2019) 8553–8562.

[125] Y. Ma, H. Ghasemzadeh, Labelforest: Non-parametric semi-supervised learning for activity recognition, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 4520–4527.

[126] Y. Chang, A. Mathur, A. Isopoussu, J. Song, F. Kawsar, A systematic study of unsupervised domain adaptation for robust human-activity recognition, Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 4 (1) (2020) http://dx.doi.org/10.1145/3380985.

[127] L. Bai, C. Yeung, C. Efstratiou, M. Chikomo, Motion2vector: Unsupervised learning in human activity recognition using wrist-sensing data, in: Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers, UbiComp/ISWC '19 Adjunct, Association for Computing Machinery, New York, NY, USA, 2019, pp. 537–542, http://dx.doi.org/10.1145/3341162.3349335.

[128] A. Akbari, R. Jafari, Transferring activity recognition models for new wearable sensors with deep generative domain adaptation, in: Proceedings of the 18th International Conference on Information Processing in Sensor Networks, IPSN '19, Association for Computing Machinery, New York, NY, USA, 2019, pp. 85–96, http://dx.doi.org/10.1145/3302506.3310391.

[129] V. Radu, M. Henne, Vision2sensor: Knowledge transfer across sensing modalities for human activity recognition, Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 3 (3) (2019) http://dx.doi.org/10.1145/3351242.

[130] T. Hur, J. Bang, T. Huynh-The, J. Lee, J.-I. Kim, S. Lee, Iss2image: A novel signal-encoding technique for cnn-based human activity recognition, Sensors 18 (11) (2018) 3910, http://dx.doi.org/10.3390/s18113910.

[131] H. Chen, S. Mahfuz, F. Zulkernine, Smart phone based human activity recognition, in: 2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2019, pp. 2525–2532.

[132] Q. Sun, Y. Liu, T.-S. Chua, B. Schiele, Meta-transfer learning for few-shot learning, in: CVPR, 2019.

[133] O. Vinyals, C. Blundell, T. Lillicrap, k. kavukcuoglu, D. Wierstra, Matching networks for one shot learning, in: Advances in Neural Information Processing Systems, Vol. 29, 2016, pp. 3630–3638.

[134] J. Snell, K. Swersky, R. Zemel, Prototypical networks for few-shot learning, in: I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems, Vol. 30, 2017, pp. 4077–4087.

[135] V.G. Satorras, J.B. Estrach, Few-shot learning with graph neural networks, in: International Conference on Learning Representations, 2018.

[136] L. Bertinetto, J.F. Henriques, P. Torr, A. Vedaldi, Meta-learning with differentiable closed-form solvers, in: International Conference on Learning Representations, 2019.

[137] Y. Liu, J. Lee, M. Park, S. Kim, E. Yang, S. Hwang, Y. Yang, Learning to propagate labels: Transductive propagation network for few-shot learning, in: International Conference on Learning Representations, 2019.

[138] G.S. Dhillon, P. Chaudhari, A. Ravichandran, S. Soatto, A baseline for few-shot image classification, in: International Conference on Learning Representations, 2020, URL https://openreview.net/forum?id=rylXBkrYDS.

[139] L. Liu, W. Hamilton, G. Long, J. Jiang, H. Larochelle, A universal representation transformer layer for few-shot image classification, 2020, arXiv preprint arXiv:2006.11702.

[140] C. Simon, P. Koniusz, R. Nock, M. Harandi, Adaptive subspaces for few-shot learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[141] A. Rahimpour, H. Qi, Class-discriminative feature embedding for meta-learning based few-shot classification, in: 2020 IEEE Winter Conference on Applications of Computer Vision (WACV), 2020, pp. 3168–3176.

[142] C. Zhang, Y. Cai, G. Lin, C. Shen, Deepemd: Few-shot image classification with differentiable earth mover's distance and structured classifiers, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[143] P. Rodríguez, I. Laradji, A. Drouin, A. Lacoste, Embedding propagation: Smoother manifold for few-shot classification, 2020, arXiv preprint arXiv:2003.04151.

[144] L. Song, J. Liu, Y. Qin, Fast and generalized adaptation for few-shot learning, 2019, arXiv preprint arXiv:1911.10807.

[145] H.-J. Ye, H. Hu, D.-C. Zhan, F. Sha, Few-shot learning via embedding adaptation with set-to-set functions, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[146] H. Zhang, P.H. Torr, P. Koniusz, Few-shot learning with multi-scale self-supervision, 2020, arXiv preprint arXiv:2001.01600.

[147] Y. Tian, Y. Wang, D. Krishnan, J.B. Tenenbaum, P. Isola, Rethinking few-sht image classification: a good embedding is all you need? 2020, arXiv preprint arXiv:2003.11539.

[148] P. Mazumder, P. Singh, V.P. Namboodiri, Improving few-shot learning using composite rotation based auxiliary task, 2020, arXiv preprint arXiv:2006.15919.

[149] Q. Liu, O. Majumder, A. Achille, A. Ravichandran, R. Bhotika, S. Soatto, Incremental meta-learning via indirect discriminant alignment, 2020, arXiv:2002.04162.

[150] Y. Wang, C. Xu, C. Liu, L. Zhang, Y. Fu, Instance credibility inference for few-shot learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[151] N. Fei, Z. Lu, Y. Gao, J. Tian, T. Xiang, J.-R. Wen, Meta-learning across meta-tasks for few-shot learning, 2020, arXiv preprint arXiv:2002.04274.

[152] T. Saikia, T. Brox, C. Schmid, Optimized generic feature learning for few-shot classification across domains, 2020, arXiv preprint arXiv:2001.07926.

[153] J. Liu, L. Song, Y. Qin, Prototype rectification for few-shot learning, 2019, arXiv preprint arXiv:1911.10713.

[154] N. Dvornik, C. Schmid, J. Mairal, Selecting relevant features from a multi-domain representation for few-shot classification, in: European Conference on Computer Vision (ECCV), 2020.

[155] J. Rajasegaran, S. Khan, M. Hayat, F.S. Khan, M. Shah, Self-supervised knowledge distillation for few-shot learning, 2020, arXiv preprint arXiv:2006.09785.

[156] M. Lichtenstein, P. Sattigeri, R. Feris, R. Giryes, L. Karlinsky, Tafssl: Task-adaptive feature sub-space learning for few-shot classification, 2020, arXiv preprint arXiv:2003.06670.

[157] Y. Ma, S. Bai, S. An, W. Liu, A. Liu, X. Zhen, X. Liu, Transductive relation-propagation network for few-shot learning, in: C. Bessiere (Ed.), Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20, International Joint Conferences on Artificial Intelligence Organization.

[158] Z. Yu, L. Chen, Z. Cheng, J. Luo, Transmatch: A transfer-learning scheme for semi-supervised few-shot learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[159] T. Munkhdalai, H. Yu, Meta networks, in: Proceedings of the 34th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 70, PMLR, International Convention Centre, Sydney, Australia, 2017, pp. 2554–2563.

[160] B. Oreshkin, P. Rodríguez López, A. Lacoste, Tadam: Task dependent adaptive metric for improved few-shot learning, in: Advances in Neural Information Processing Systems 31, 2018, pp. 721–731.

[161] L. Liu, T. Zhou, G. Long, J. Jiang, C. Zhang, Many-class few-shot learning on multi-granularity class hierarchy, IEEE Trans. Knowl. Data Eng. (2020) 1.

[162] J. He, X. Liu, R. Hong, Memory-augmented relation network for few-shot learning, 2020, arXiv preprint arXiv:2005.04414.

[163] C. Finn, P. Abbeel, S. Levine, Model-agnostic meta-learning for fast adaptation of deep networks, in: Proceedings of the 34th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 70, International Convention Centre, Sydney, Australia, 2017, pp. 1126–1135.

[164] S. Ravi, H. Larochelle, Optimization as a model for few-shot learning, in: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, in: Conference Track Proceedings, OpenReview.net, 2017.

[165] R. Zhang, T. Che, Z. Ghahramani, Y. Bengio, Y. Song, Metagan: An adversarial approach to few-shot learning, in: Advances in Neural Information Processing Systems 31, 2018, pp. 2365–2374.

[166] A.A. Rusu, D. Rao, J. Sygnowski, O. Vinyals, R. Pascanu, S. Osindero, R. Hadsell, Meta-learning with latent embedding optimization, in: International Conference on Learning Representations, 2019.

[167] H. Li, W. Dong, X. Mei, C. Ma, F. Huang, B.-G. Hu, LGM-net: Learning to generate matching networks for few-shot learning, in: Proceedings of the 36th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 97, PMLR, Long Beach, California, USA, 2019, pp. 3825–3834.

[168] H. Li, D. Eigen, S. Dodge, M. Zeiler, X. Wang, Finding task-relevant features for few-shot learning by category traversal, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

[169] C. Liu, Z. Wang, D. Sahoo, Y. Fang, K. Zhang, S.C. Hoi, Adaptive task sampling for meta-learning, 2020, arXiv preprint arXiv:2007.08735.

[170] Y. Liu, B. Schiele, Q. Sun, An ensemble of epoch-wise empirical bayes for few-shot learning, 2019, arXiv preprint arXiv:1904.08479.

[171] L. Zhou, P. Cui, X. Jia, S. Yang, Q. Tian, Learning to select base classes for few-shot classification, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[172] A. Mehrotra, A. Dukkipati, Generative adversarial residual pairwise networks for one shot learning, 2017, arXiv preprint arXiv:1703.08033.

[173] E. Schwartz, L. Karlinsky, J. Shtok, S. Harary, M. Marder, A. Kumar, R. Feris, R. Giryes, A. Bronstein, Delta-encoder: an effective sample synthesis method for few-shot object recognition, in: S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, R. Garnett (Eds.), Advances in Neural Information Processing Systems 31, Curran Associates, Inc., 2018, pp. 2845–2855.

[174] K. Li, Y. Zhang, K. Li, Y. Fu, Adversarial feature hallucination networks for few-shot learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[175] N. Ma, J. Bu, J. Yang, Z. Zhang, C. Yao, Z. Yu, Few-shot graph classification with model agnostic meta-learning, 2020, arXiv preprint arXiv:2003.08246.

[176] J. Zhang, M. Zhang, Z. Lu, T. Xiang, J. Wen, Adargcn: Adaptive aggregation gcn for few-shot learning, 2020, arXiv preprint arXiv:2002.12641.

[177] L. Liu, T. Zhou, G. Long, J. Jiang, C. Zhang, Attribute propagation network for graph zero-shot learning, in: AAAI, 2020, pp. 4868–4875.

[178] G. Sourek, F. Zelezny, O. Kuzelka, Beyond graph neural networks with lifted relational neural networks, 2020, arXiv preprint arXiv:2007.06286.

[179] K. Huang, M. Zitnik, Graph meta learning via local subgraphs, 2020, arXiv preprint arXiv:2006.07889.

[180] G. Lin, Y. Yang, Y. Fan, X. Kang, K. Liao, F. Zhao, High-order structure preserving graph neural network for few-shot learning, 2020, arXiv preprint arXiv:2005.14415.

[181] L. Liu, T. Zhou, G. Long, J. Jiang, C. Zhang, Learning to propagate for graph meta-learning, in: Advances in Neural Information Processing Systems, 2019, pp. 1039–1050.

[182] B. Zhang, K.-C. Leung, Y. Ye, X. Li, Metaconcept: Learn to abstract via concept graph for weakly-supervised few-shot learning, 2020, arXiv preprint arXiv:2007.02379.

[183] M. Balcilar, G. Renton, P. Héroux, B. Gauzere, S. Adam, P. Honeine, Bridging the gap between spectral and spatial domains in graph neural networks, 2020, arXiv preprint arXiv:2003.11702.

[184] C. Zhuang, Q. Ma, Dual graph convolutional networks for graph-based semi-supervised classification, in: Proceedings of the 2018 World Wide Web Conference, WWW '18, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 2018, pp. 499–508, http://dx.doi.org/10.1145/3178876.3186116.

[185] F. Monti, O. Shchur, A. Bojchevski, O. Litany, S. Günnemann, M.M. Bronstein, Dual-primal graph convolutional networks, 2018, arXiv preprint arXiv:1806.00770.

[186] F. Shi, Y. Zhao, Z. Xu, T. Liu, S.-C. Zhu, Twin graph convolutional networks: Gcn with dual graph support for semi-supervised learning, 2020, URL https://openreview.net/forum?id=SkxV7kHKvr.

[187] H. Zhu, W. Ma, L. Li, L. Jiao, S. Yang, B. Hou, A dual–branch attention fusion deep network for multiresolution remote–sensing image classification, Inf. Fusion 58 (2020) 116–131, http://dx.doi.org/10.1016/j.inffus.2019.12.013, URL http://www.sciencedirect.com/science/article/pii/S1566253518308182.

[188] F. Chen, S. Pan, J. Jiang, H. Huo, G. Long, Dagcn: Dual attention graph convolutional networks, in: 2019 International Joint Conference on Neural Networks (IJCNN), 2019, pp. 1–8.

[189] M. Liu, J. Liao, J. Wang, Q. Qi, H. Sun, Dual attention-based adversarial autoencoder for attributed network embedding.

[190] H. Zhu, X. Luo, H.H. Zhuo, Dual graph representation learning, 2020, arXiv preprint arXiv:2002.11501.

[191] Z. Liu, W. Liu, P. Chen, C. Zhuang, C. Song, Hpgat: High-order proximity informed graph attention network, IEEE Access 7 (2019) 123002–123012.

[192] W. Hamilton, Z. Ying, J. Leskovec, Inductive representation learning on large graphs, in: Advances in Neural Information Processing Systems, 2017, pp. 1024–1034.

[193] Y. Li, Y. Tian, J. Zhang, Y. Chang, Learning signed network embedding via graph attention, in: AAAI, 2020, pp. 4772–4779.

[194] J. Chen, M. Yang, G. Gao, Semi-supervised dual-branch network for image classification, Knowl.-Based Syst. 197 (2020) 105837, http://dx.doi.org/10.1016/j.knosys.2020.105837.

[195] H. Chang, Y. Rong, T. Xu, W. Huang, S. Sojoudi, J. Huang, W. Zhu, Spectral graph attention network, 2020, arXiv preprint arXiv:2003.07450.

[196] J. Wang, L. Zhang, C. Wang, X. Ma, Q. Gao, B. Lin, Device-free human gesture recognition with generative adversarial networks, IEEE Internet Things J. (2020) 1.

[197] C. Li, M. Liu, Z. Cao, Wihf: Enable user identified gesture recognition with wifi, in: IEEE INFOCOM 2020 - IEEE Conference on Computer Communications, 2020.
[198] G. Hinton, O. Vinyals, J. Dean, Distilling the knowledge in a neural network, 2015, arXiv preprint arXiv:1503.02531.
[199] L. Chen, Q. Wang, X. Lu, D. Cao, F. Wang, Learning driving models from parallel end-to-end driving data set, Proc. IEEE 108 (2) (2020) 262–273.

Wenbo Zheng received his bachelor degree in software engineering from Wuhan University of Technology, Wuhan, China, in 2017. He received his Ph.D. degree in computer science and technology from Faculty of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an, China, in 2021. His research interests include computer vision and machine learning.

Lan Yan received her bachelor degree from University of Electronic Science and Technology of China in 2017. She is currently a Ph.D. candidate in the School of Artificial Intelligence, University of Chinese Academy of Sciences as well as the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences. Her research interests include computer vision and pattern recognition.

Chao Gou received the B.S. degree from the University of Electronic Science and Technology of China, Chengdu, China, in 2012 and the Ph.D. degree from the University of Chinese Academy of Sciences (UCAS), Beijing, China, in 2017. From September 2015 to January 2017, he was supported by UCAS as a joint-supervision Ph.D. student in Rensselaer Polytechnic Institute, Troy, NY, USA. He is currently an Assistant Professor with School of Intelligent Systems Engineering, Sun Yat-Sen University. His research interests include computer vision and machine learning.

Fei-Yue Wang received his Ph.D. degree in computer and systems engineering from the Rensselaer Polytechnic Institute, Troy, NY, USA, in 1990. He joined The University of Arizona in 1990 and became a Professor and the Director of the Robotics and Automation Laboratory and the Program in Advanced Research for Complex Systems. In 1999, he founded the Intelligent Control and Systems Engineering Center at the Institute of Automation, Chinese Academy of Sciences (CAS), Beijing, China, under the support of the Outstanding Chinese Talents Program from the State Planning Council, and in 2002, was appointed as the Director of the Key Laboratory of Complex Systems and Intelligence Science, CAS. In 2011, he became the State Specially Appointed Expert and the Director of the State Key Laboratory for Management and Control of Complex Systems. His current research focuses on methods and applications for parallel intelligence, social computing, and knowledge automation. He is a fellow of IEEE, INCOSE, IFAC, ASME, and AAAS. In 2007, he received the National Prize in Natural Sciences of China and became an Outstanding Scientist of ACM for his work in intelligent control and social computing. He received the IEEE ITS Outstanding Application and Research Awards in 2009 and 2011, respectively. In 2014, he received the IEEE SMC Society Norbert Wiener Award. Since 1997, he has been serving as the General or Program Chair of over 30 IEEE, INFORMS, IFAC, ACM, and ASME conferences. He was the President of the IEEE ITS Society from 2005 to 2007, the Chinese Association for Science and Technology, USA, in 2005, the American Zhu Kezhen Education Foundation from 2007 to 2008, the Vice President of the ACM China Council from 2010 to 2011, the Vice President and the Secretary General of the Chinese Association of Automation from 2008-2018. He was the Founding Editor-in-Chief (EiC) of the *International Journal of Intelligent Control and Systems* from 1995 to 2000, the *IEEE ITS Magazine* from 2006 to 2007, the *IEEE/CAA JOURNAL OF AUTOMATICA SINICA* from 2014-2017, and the *China's Journal of Command and Control* from 2015–2020. He was the EiC of the *IEEE Intelligent Systems* from 2009 to 2012, the *IEEE TRANSACTIONS ON Intelligent Transportation Systems* from 2009 to 2016, and is the EiC of the *IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS* since 2017, and the Founding EiC of *China's Journal of Intelligent Science and Technology* since 2019. Currently, he is the President of CAA's Supervision Council, IEEE Council on RFID, and Vice President of IEEE Systems, Man, and Cybernetics Society.