Heterogeneous Mutual Knowledge Distillation for Wearable Human Activity Recognition

Zhiwen Xiao, *Member, IEEE*, Huanlai Xing, *Member, IEEE*, Rong Qu, *Senior Member, IEEE*, Hui Li, Xinzhou Cheng, Lexi Xu, *Senior Member, IEEE*, Li Feng, and Qian Wan

Abstract—Recently, numerous deep learning algorithms have addressed wearable human activity recognition (HAR), but they often struggle with efficient knowledge transfer to lightweight models for mobile devices. Knowledge distillation (KD) is a popular technique for model compression, transferring knowledge from a complex teacher to a compact student. Most existing KD algorithms considered homogeneous architectures, hindering performance in heterogeneous setups. This is an under-explored area in wearable HAR. To bridge this gap, we propose a heterogeneous mutual KD (HMKD) framework for wearable HAR. HMKD establishes mutual learning within the intermediate and output layers of both teacher and student models. To accommodate substantial structural differences between teacher and student, we employ a weighted ensemble feature approach to merge the features from their intermediate layers, enhancing knowledge exchange within them. Experimental results on the HAPT, WISDM, and UCI_HAR datasets show HMKD outperforms 10 state-of-the-art KD algorithms in terms of classification accuracy. Notably, with ResNetLSTMaN as the teacher and MLP as the student, HMKD increases by 9.19% in MLP's F_1 score on the HAPT dataset.

Index Terms—Data Mining, Human Activity Recognition, Knowledge Distillation, Model Compression, Wearable Sensors

1 Introduction

Human activity recognition (HAR) involves identifying individuals' actions by analyzing their interactions with the environment [1]. This technology has found extensive applications across diverse real-world domains, including electroencephalography (EEG) detection [2], spectrum map prediction [3], and healthcare applications [4]. With the widespread use of mobile devices, such as smartphones and watches, the collection of wearable HAR data has become accessible and convenient. Consequently, wearable sensor-based HAR has become one of the primary research focuses in HAR [5]. Wearable HAR data consists of a sequence of time-ordered data points gathered by wearable sensor(s), for example, a triaxial accelerometer with three sensors generating X-, Y-, and Z-axis data simultaneously. The series is associated with one or more time-dependent variables, encompassing both univariate and multivariate aspects. A HAR algorithm captures both local and global patterns from a given time series, including those associated with a single variable and those across multiple variables [6], [7], [8].

Over the years, a plethora of algorithms has been developed to tackle wearable-HAR-oriented challenges, primarily employing traditional and deep learning techniques [5], [8]. Traditional algorithms typically rely on statistical or machine learning methods, with an emphasis on extracting remarkable representations from HAR data. For instance, in [9], a hierarchical hidden Markov algorithm was designed to extract semantic relationships between contexts. In [10], a system based on coordinate transformation, principal component analysis (PCA) and online support vector machine (SVM) was proposed to address various HAR problems. In contrast, deep learning algorithms have the capability to uncover inherent relationships among representations through the construction of an internal hierarchy of data [11]. For example, Shu *et al.* [12] put forward a graph long short-term memory (LSTM)-in-LSTM algorithms to build person-level actions and group-level activity. Al-qaness *et al.* [13] presented a multi-level residual attention model to extract intrinsic connections among activities. Xia *et al.* [14] designed a multiple-level domain model with a single inertial measurement sensor for HAR. Xu *et al.* [15] devised a deformable convolutional network model for extracting salient features from the data. In [16], a fully-convolutional network(FCN)-LSTM-attention-based network (FCNLSTMaN) was introduced to extract local and global patterns of HAR data.

Existing deep learning models for wearable HAR face the following challenges. *Many of them were designed for particular wearable HAR tasks, showcasing strong feature extraction capabilities for problem-specific tasks. However, they lack efficient knowledge transfer from heavy and complex models to lightweight and simple ones, which is quite crucial for resource-constrained mobile devices, like smartwatches and tablets.* Hinton *et al.* [17] introduced knowledge distillation (KD) to transfer knowledge from a large-scale neural network to

Manuscript received XX,XX. This work was partially supported by the Natural Science Foundation of Hebei Province (No. F2022105027), the Natural Science Foundation of Sichuan Province (No. 2022NSFSC0568, No. 2022NSFSC0944, and No. 2023NSFSC0459), and the Fundamental Research Funds for the Central Universities, P. R. China (Corresponding Author: Huanlai Xing).
Z. Xiao, H. Xing, L. Feng, and Q. Wan are with the School of Computing and Artificial Intelligence, Southwest Jiaotong University, Chengdu 610031, China, with the Tangshan Institute of Southwest Jiaotong University, Tangshan 063000, China, and with the Engineering Research Center of Sustainable Urban Intelligent Transportation, Ministry of Education, Chengdu 611756, China (Emails: xiao1994zw@163.com; hxx@home.swjtu.edu.cn; fengli@swjtu.edu.cn; qianwan@my.swjtu.edu.cn).
R. Qu is with the School of Computer Science, University of Nottingham, Nottingham NG7 2RD 455356, UK (Email: rong.qu@nottingham.ac.uk).
H. Li is with the School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an 710049, China (Email: lihui10@mail.xjtu.edu.cn).
X. Cheng and L. Xu are with the Research Institute, China United Network Communications Corporation, Beijing 100048, China (Email: chengxz11@chinaunicom.cn and xulx29@chinaunicom.cn).

a smaller one, often referred to as the teacher-and-student model. Unlike traditional compression and acceleration techniques, e.g., parameter pruning/sharing and low-rank factorization methods, KD makes a student model adaptly replicate the knowledge embedded in a teacher model. KD acts as a regularization technique for both the teacher and student, fostering effective knowledge transfer between them. Response-based, feature-based, and relation-based algorithms are three main research streams [18]. Response-based algorithms encourage knowledge transfer from the output (i.e., logits) of a teacher to that of a student, e.g., resolution-aware KD [19], HAR contrastive distillation [20], and lightweight HAR [21]. Feature-based algorithms facilitate knowledge transfer between intermediate layers of a teacher and its student, as opposed to a direct transfer from output to output, such as FitNet [22] and distillation methods using layer-calibration and task-disentangle distillation [23]. Unlike response- and feature-based algorithms, relation-based algorithms pay attention to extracting the connections among layers in teacher and student models, e.g., relation-based metric learning [24]. However, most KD algorithms face the following challenges:

- *While these algorithms have demonstrated remarkable performance, their success is often contingent upon the assumption of homogeneity in the architectures of both teacher and student models. Challenges emerge when confronted with heterogeneous architectures, as existing approaches may falter due to the distinct features inherent in both teacher and student.*
- *Currently, there are limited number of heterogeneous distillation algorithms that account for the diversity in the teacher and student architectures. An exemplary algorithm in this category is the effective one-for-All KD [25], which transfers knowledge from the teacher's intermediate layers to the output of the student model. While this approach underscores the importance of imparting the teacher's knowledge to the student, it overlooks the reciprocal nature of knowledge exchange, where the student's knowledge also contributes to the teacher's understanding. KD serves as a mechanism for fostering mutual learning between teacher and student models [26], [27]. Effectively promoting mutual learning between teacher and student models seems a promising solution to heterogeneous distillation.*
- *To the best of our knowledge, there has been insufficient research attention dedicated to heterogeneous distillation in the wearable HAR field. This underscores the potential for further exploration and development in this specific domain.*

To address the challenges above, we propose a heterogeneous mutual KD (HMKD) framework for wearable HAR. Unlike the effective one-for-All KD, HMKD not only establishes mutual learning within the intermediate layers of both teacher and student models but also extends it to the output layers of these models. This inclusive approach promotes efficient knowledge flow between the teacher and student, facilitating a comprehensive exchange of information. In this work, we choose two well-known dual-network-based teacher models and two straightforward, foundational student models. The two teacher models are FCNLSTMaN [16] and ResNetLSTMaN [28]. FCNLSTMaN comprises a fully-convolutional network (FCN) composed of three ConvBlocks and an LSTMaN structure consisting of two LSTM-based attention layers, as depicted in Fig. 1 (a). ResNetLSTMaN comprises three residual blocks and two LSTM-based attention layers, illustrated in Fig. 1 (c). On the other hand, the two student models are convolutional neural network (CNN) with three convolutional blocks in Fig. 1 (b) and multi-layer perceptron (MLP) with three dense (i.e., fully-connected) layers in Fig. 1 (d), respectively.

Our major contributions are listed as follows.

- We propose HMKD for wearable HAR, fostering mutual learning not only within the intermediate layers of both teacher and student models but also at the output layers. This approach enhances the knowledge transfer efficiency between the teacher and its student.
- Given the substantial structural disparities between the teacher and student models, we introduce a weighted ensemble feature approach designed to merge the features extracted from the intermediate layers of these models. This approach aims to circumvent potential information loss when employing intricate distillation links, particularly in scenarios involving heterogeneous teacher and student models. Consequently, this method promotes knowledge exchange within the intermediate layers of both models.

  Additionally, unlike most KD algorithms that use Kullback-Leibler (KL) divergence as the distillation function for teacher and student models [17], [18], [19], we adopts the Jensen-Shannon (JS) divergence function. This function quantifies the knowledge variability between teacher and student. It also offers a distinct perspective on evaluating and leveraging the differences and similarities between the models for enhanced KD.
- The experimental findings reveal that employing FCNLSTMaN and ResNetLSTMaN as teacher models and CNN and MLP as student models, HMKD demonstrates superior performance compared to 10 state-of-the-art (SOTA) KD algorithms on three renowned wearable HAR datasets regarding the F_1 value. Three HAR datasets include the smartphone-based recognition of human activities and postural transitions dataset (HAPT), wireless sensor data mining (WISDM), and University of California Irvine activity recognition using smartphones (UCI_HAR). In particular, with ResNetLSTMaN and MLP as teacher and student, the F_1 value of MLP increases by approximately 9.19% on the HAPT dataset.

The remainder of the paper is structured as follows. Section 2 reviews the most relevant studies. The overall structure of HMKD and its components are presented in Section 3. Section 4 analyzes the experimental results. Section 5 summarizes the findings and draws conclusions.

2 Related Work

This section provides a review of pertinent studies on the wearable HAR and KD.

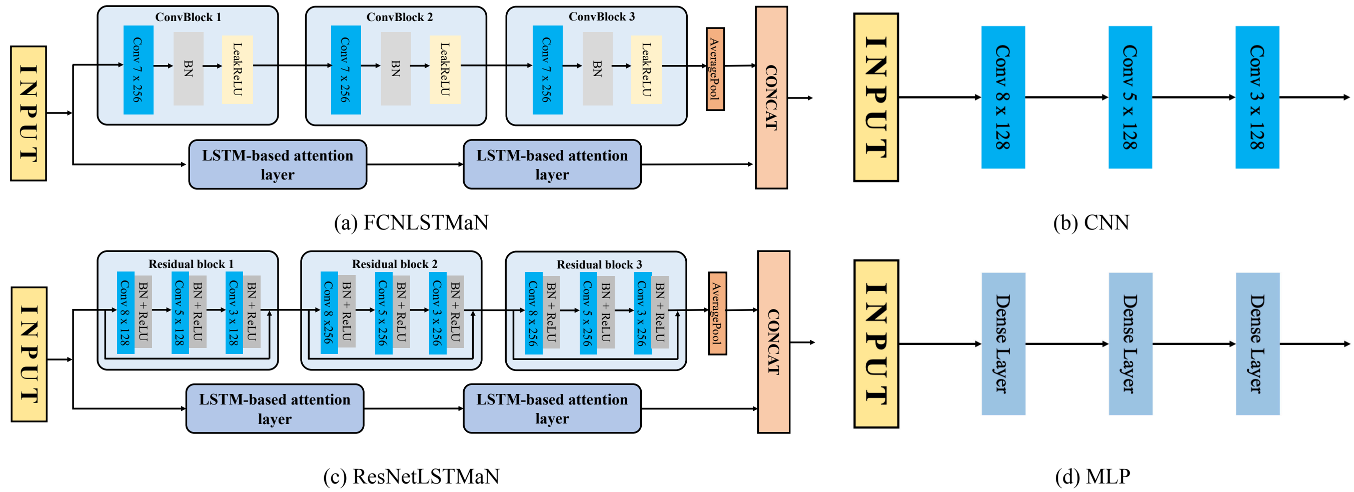


Fig. 1. Architectures of the teacher and student models. (a) FCNLSTMaN [16]. It comprises a FCN consisting of three ConvBlocks and an LSTMaN consisting of two LSTM-based attention layers. Each ConvBlock is composed of a 1-dimensional convolutional layer, a batch normalization (BN) layer, and the leaky rectified linear unit (ReLU) function. (b) CNN with three convolutional blocks. (c) ResNetLSTMaN [28]. It comprises a ResNet with three residual blocks and an LSTMaN consisting two LSTM-based attention layers. (d) MLP model with three dense layers. Note: 'Conv 7 x 256' represents a 1-dimensional convolutional layer with a kernel size of 7 and a channel size of 256.

2.1 Wearable HAR Algorithms

Numerous algorithms have been developed to tackle wearable HAR problems, falling into two main categories: traditional and deep learning-based approaches [5], [8]. Traditional algorithms typically employ statistical or machine learning methods to extract shallow features from HAR data, e.g., fuzzy temporal window approach, PCA, Bagging, Bayes method, J48, decision tree, random forest, SVM, logistic regression, gradient boosting machine, KNN, AdaBoost, collaboration method, k-means, Markov regression, and logic-based reasoning [9], [10], [29], [30], [31], [32], [33].

On the other hand, deep learning algorithms aim to extract inherent relationships among representations through building an internal hierarchy of data [11]. For example, Luo *et al.* proposed a binarized neural network, called BinaryDilatedDenseNet, for low-latency and low-memory HAR. In [34], a temporal convolutional method was designed to address low-power activity recognition. In [35], a multi-head convolutional attention method was used to model multi-dimensional representations from the data. Typical deep learning algorithms include LSTM-in-LSTM [12], bi-directional LSTM [36], kernel density estimation-based model [37], CNN-LSTM-based model [38], SelfHAR [39], multiple-level domain model [15], multi-level residual attention model [14], CSSHAR [40], stacked denoising autoencoder [41], selective kernel convolution [42], deformable convolutional model [15], Lego convolutional model [43], CapMatch [44], and ColloSSL [45].

2.2 Knowledge Distillation

KD, regarded as one of the widely used regularization techniques, is designed to facilitate knowledge transfer from a more complex network (teacher) to a simpler one (student). Researchers categorize the existing KD algorithms into three main groups based on the form of knowledge transfer: response-based, feature-based, and relation-based [18]. Response-based algorithms enable the knowledge transfer from the output (i.e., logits) of a teacher to that of a student [17]. For instance, Xu *et al.* proposed a contrastive distillation framework with regularized knowledge (ConDRK) for HAR. Zhao *et al.* [46] introduced a decoupled KD method to transfer the target and non-target knowledge from the output of a teacher to its student. The resolution-aware KD [19], lightweight HAR [21], expert embedding KD [47], correlation-based KD with a stronger teacher [48], and collaborative KD [49] are representative response-based approaches.

Feature-based algorithms facilitate knowledge transfer between intermediate layers of a teacher and its student [22]. For example, Peng *et al.* [50] presented a correlation congruence KD framework to transfer the instance-level information and the correlation between instances. Hao *et al.* [51] devised a collaborative feature sharing approach for multi-level knowledge sharing. Tian *et al.* [52] introduced a contrastive representation distillation method to transfer the structural knowledge of a teacher to its student.

Relation-based algorithms pay attention to understanding and leveraging the relationships between layers to enhance the knowledge transfer process. For instance, in [53], a relation-based metric learning model was designed to improve the representation of image embedding. In [24], a multi-level KD method based attention was devised to extract intrinsic relationships between teacher and student models. In [27], a cross-layer mutual distillation approach was presented to facilitate the dense mutual learning between teacher and student.

3 The Proposed HMKD

This section overviews the structure of the proposed HMKD and details its key components, including teacher and student models and heterogeneous distillation architecture.

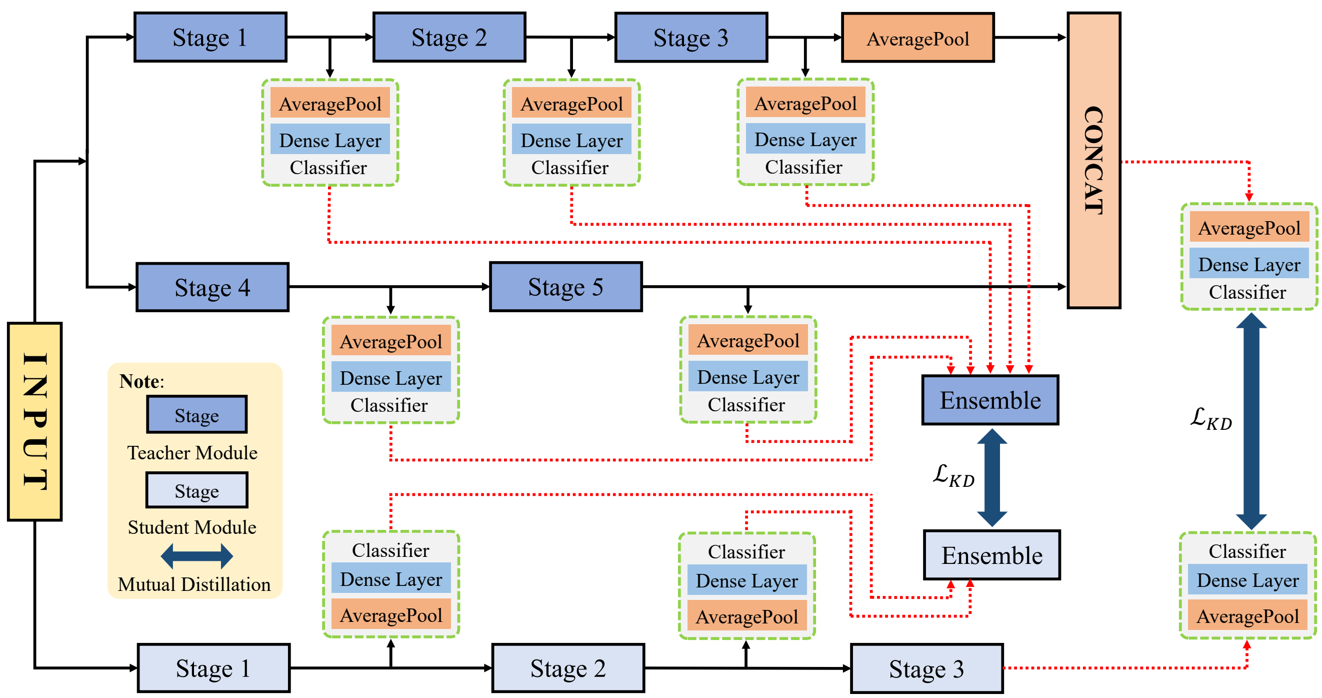


Fig. 2. Overview of the proposed HMKD. The teacher and student modules are critical components representing the neural network structures in teacher and student models, respectively. For example, in the FCNLSTMaN teacher model, 'stage 1', 'stage 2', and 'stage 3' correspond to 'ConvBlock 1', 'ConvBlock 2', and 'ConvBlock 3', respectively, while 'stage 4' and 'stage 5' denote the first and second LSTM-based attention layers, respectively. In the CNN student model, 'stage 1', 'stage 2', and 'stage 3' represent the first, second, and third 1-dimensional convolutional layers, respectively.

3.1 Overview

HMKD establishes mutual learning not only within the intermediate layers of both teacher and student but also extends to the output layers of the two models. This approach promotes efficient knowledge flow between the teacher and student, facilitating a comprehensive exchange of information. The structure of the proposed HMKD is shown in 2. To handle the significant structural differences between the teacher and student models, we design a weighted ensemble feature approach to fuse the features extracted from the intermediate layers of these models. This approach enhances the knowledge exchange within the intermediate layers of both models. To measure the knowledge variability between teacher and student models, we employ the JS divergence function, which offers a distinct perspective on evaluating and leveraging the differences and similarities between the models for enhanced KD.

3.2 Teacher Model

In this work, we choose two well-known dual-network-based teacher models: FCNLSTMaN [16] and ResNetLSTMaN [28].

3.2.1 FCNLSTMaN

FCNLSTMaN comprises a FCN and an LSTMaN in parallel, as depicted in Fig. 1 (a).

Fully Convolutional Network FCN consists of three Convblocks for local feature extraction, namely 'ConvBlock 1', 'ConvBlock 2', and 'ConvBlock 3'. Each ConvBlock is comprised of a 1-dimensional convolutional layer, a batch normalization (BN) layer, and the leaky rectified linear unit (ReLU) function. An arbitrary Convblock is defined as:

$$f_{ConvB}(x) = LeakyReLU(BN(CNN(x))) \tag{1}$$

where, $CNN()$, $BN()$, and $LeakyReLU()$ represent the 1-dimensional convolutional, batch normalization, and leaky ReLU functions, respectively. x is the input data.

LSTM-based Attention Network LSTMaN is composed of two LSTM-based attention layers for global relation extraction. Each layer embeds LSTM networks into attention structure [54], as shown in Fig. 3. This structure involves mapping a query, $Query$ and a set of key–value pairs, Key-$Value$, to an output, O_{Latt}. $Query$, Key, and $Value$ correspond to the feature vectors extracted by the three LSTM networks. O_{Latt} is defined in Eq. (2).

$$O_{Latt} = Softmax(Query \cdot Key^T) \cdot Value \tag{2}$$

where, $Softmax()$ computes the possibilities of a given vector. Key^T is the transpose of Key.

3.2.2 ResNetLSTMaN

ResNetLSTMaN integrates a ResNet and an LSTMaN in parallel, illustrated in Fig. 1 (c).

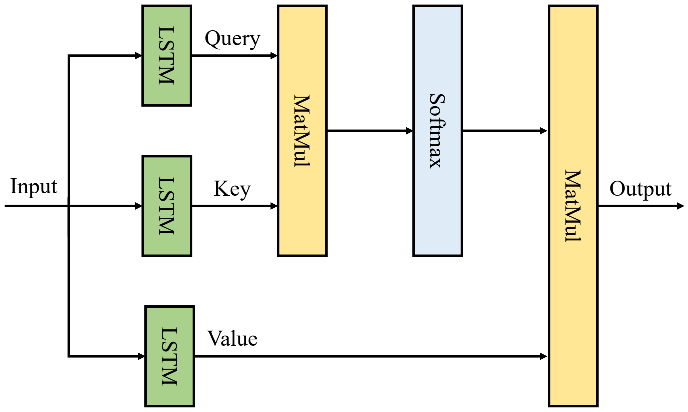


Fig. 3. Architecture of an LSTM-based attention layer [54]. Note: 'MatMul' represent the matrix multiplication operation, and 'Softmax' outputs the probability of a give matrix.

Residual Network ResNet contains three residual blocks, i.e., 'residual block 1', 'residual block 2', and 'residual block 3', to extract local features from the data. Each residual block consists of three 1-dimensional convolutional layers, the BN layer, and the ReLU function. In particular, a residual structure is incorporated into each block to mitigate the risk of information loss and gradient degradation during training.

LSTM-based Attention Network Similar to FCNLSTMaN, LSTMaN in ResNetLSTM also incorporates two LSTM-based attention layers to capture global patterns in the data. The first layer is responsible for extracting fundamental relationships from the data. Subsequently, the second layer focuses on capturing intricate connections among the relationships obtained in the previous step. This hierarchical approach contributes to a more nuanced understanding and representation of the relationships within the data.

3.3 Student Model

In this study, two basic yet fundamental student models are considered: the CNN model depicted in Fig. 1 (b) and the MLP model illustrated in Fig. 1 (d). More specifically, the CNN model comprises three 1-dimensional layers, labeled as 'Conv 8 x 128', 'Conv 5 x 128', and 'Conv 3 x 128'. On the other hand, the MLP model consists of three dense (fully-connected) layers.

3.4 Mutual Knowledge Distillation

HMKD establishes mutual learning not only within the intermediate layers of both teacher and student models but also extends to the output layers of these models. This inclusive approach promotes efficient knowledge flow between the teacher and student, facilitating a comprehensive exchange of information across different levels of the models. The architecture of HMKD details in Fig. 2. Let $V_{i,j}^{\mathcal{T}}$, $i = 1, 2, ..., N$, $j = 1, 2, 3, 4, 5$, denote the output of the i-th output feature vector of j-th teacher module after passing the corresponding classifier, where N is the size of input samples. The classifier typically consists of an average pooling layer and a dense layer. $V_i^{\mathcal{T}}$ is the i-th output vector of the teacher model. Let $V_{i,j}^{\mathcal{S}}$, $i = 1, 2, ..., N$, $j = 1, 2$, be the output of the i-th output feature vector of j-th student module after passing the corresponding classifier. $V_i^{\mathcal{S}}$ stands for the i-th output vector of the student model.

3.4.1 Weighed Ensemble Feature

Considering the substantial structural disparities between teacher and student models, we design a weighted ensemble feature approach to fuse the features extracted from the intermediate layers of these models. This approach promotes knowledge exchange within the intermediate layers of both models.

The teacher's weighted ensemble feature, $V_{WEF,i}^{\mathcal{T}}$, is calculated as:

$$V_{WEF,i}^{\mathcal{T}} = \sum_{j=1}^{5} \alpha_j^{\mathcal{T}} V_{i,j}^{\mathcal{T}} \tag{3}$$

where, $\alpha_j^{\mathcal{T}}$ is the weighted coefficient of $V_{i,j}^{\mathcal{T}}$, as defined in Eq. (4).

$$\alpha_j^{\mathcal{T}} = \frac{V_{i,j}^{\mathcal{T}}}{\sum_{j=1}^{5} V_{i,j}^{\mathcal{T}}} \tag{4}$$

The student's weighted ensemble feature, $V_{WEF,i}^{\mathcal{S}}$, is defined in Eq. (5).

$$V_{WEF,i}^{\mathcal{S}} = \sum_{j=1}^{2} \alpha_j^{\mathcal{S}} V_{i,j}^{\mathcal{S}} \tag{5}$$

where, $\alpha_j^{\mathcal{S}}$ is the weighted coefficient of $V_{i,j}^{\mathcal{S}}$. It is defined as:

$$\alpha_j^{\mathcal{S}} = \frac{V_{i,j}^{\mathcal{S}}}{\sum_{j=1}^{2} V_{i,j}^{\mathcal{S}}} \tag{6}$$

3.4.2 Teacher's Loss Function

The teacher's loss function, $\mathcal{L}^{\mathcal{T}}$, is combination of a supervised loss, $\mathcal{L}_{sup}^{\mathcal{T}}$, and a teacher distillation loss, $\mathcal{L}_{TDL}^{\mathcal{T}}$, as defined in Eq. (7).

$$\mathcal{L}^{\mathcal{T}} = \mathcal{L}_{sup}^{\mathcal{T}} + \beta^{\mathcal{T}} \mathcal{L}_{TDL}^{\mathcal{T}} \tag{7}$$

where, $\beta^{\mathcal{T}}$ is the constant coefficient of $\mathcal{L}^{\mathcal{T}}$. Following [26], [27], we set $\beta^{\mathcal{T}}$ = 1.0 in this paper.

$\mathcal{L}_{sup}^{\mathcal{T}}$ is defined in Eq. (8).

$$\mathcal{L}_{sup}^{\mathcal{T}} = -\frac{1}{N} \sum_{i=1}^{N} y_i log(V_i^{\mathcal{T}}) \tag{8}$$

where, y_i is the i-th ground truth label.

$\mathcal{L}_{TDL}^{\mathcal{T}}$ is defined as:

$$\mathcal{L}_{TDL}^{\mathcal{T}} = \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}_{i,TDL}^{\mathcal{T}} \tag{9}$$

where,

$$\begin{aligned} \mathcal{L}_{i,TDL}^{\mathcal{T}} = &\mathcal{L}_{KD}(V_{WEF,i}^{\mathcal{T}}/t_{KD}, V_{WEF,i}^{\mathcal{S}}/t_{KD}) \\ &+ \mathcal{L}_{KD}(V_i^{\mathcal{T}}/t_{KD}, V_i^{\mathcal{S}}/t_{KD}) \end{aligned} \tag{10}$$

where, t_{KD} serves as a scaling coefficient for the features of the teacher and student, playing a crucial role in facilitating the knowledge flow between teacher and student models. $\mathcal{L}_{KD}(p, q)$ is based on the JS divergence function to measure the average difference between the outputs of teacher and student models, as defined in Eq. (11).

$$\mathcal{L}_{KD}(p,q) = \frac{KL(\frac{p+q}{2},p) + KL(\frac{p+q}{2},q)}{2} \tag{11}$$

where, $KL(p,q)$ is the KL function.

3.4.3 Student's Loss Function

The student's loss function, $\mathcal{L}^{\mathcal{S}}$, includes two components: a supervised loss, $\mathcal{L}^{\mathcal{S}}_{sup}$ and a student distillation loss, $\mathcal{L}^{\mathcal{S}}_{SDL}$. It is calculated as:

$$\mathcal{L}^{\mathcal{S}} = \mathcal{L}^{\mathcal{S}}_{sup} + \beta^{\mathcal{S}}\mathcal{L}^{\mathcal{S}}_{SDL} \tag{12}$$

where, $\beta^{\mathcal{S}}$ represents the constant coefficient of $\mathcal{L}^{\mathcal{S}}$. As suggested in [26], [27], we set $\beta^{\mathcal{S}}$ = 1.0 in this paper.

$\mathcal{L}^{\mathcal{S}}_{sup}$ is shown in Eq. (13).

$$\mathcal{L}^{\mathcal{S}}_{sup} = -\frac{1}{N}\sum_{i=1}^{N} y_i log(V_i^{\mathcal{S}}) \tag{13}$$

$\mathcal{L}^{\mathcal{S}}_{SDL}$ is calculated in Eq. (14).

$$\mathcal{L}^{\mathcal{S}}_{SDL} = \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}^{\mathcal{S}}_{i,SDL} \tag{14}$$

where,

$$\begin{aligned} \mathcal{L}^{\mathcal{S}}_{i,SDL} = \mathcal{L}_{KD}(V^{\mathcal{S}}_{WEF,i}/t_{KD}, V^{\mathcal{T}}_{WEF,i}/t_{KD}) \\ + \mathcal{L}_{KD}(V_i^{\mathcal{S}}/t_{KD}, V_i^{\mathcal{T}}/t_{KD}) \end{aligned} \tag{15}$$

Following mutual learning algorithms in [26], [27], we utilize gradient descent to jointly optimize the teacher's and student's parameters. Let $\theta^{\mathcal{T}}_k$ and $\theta^{\mathcal{S}}_k$ represent the teacher's and student's parameters during the k-th training epoch, respectively. $\theta^{\mathcal{T}}_k$ and $\theta^{\mathcal{S}}_k$ are defined in Eq. (16).

$$\begin{aligned} \theta^{\mathcal{T}}_k &= \theta^{\mathcal{T}}_{k-1} - \eta^{\mathcal{T}}\nabla_{\theta^{\mathcal{T}}_{k-1}}\mathcal{L}(\theta^{\mathcal{T}}_{k-1}), \\ \theta^{\mathcal{S}}_k &= \theta^{\mathcal{S}}_{k-1} - \eta^{\mathcal{S}}\nabla_{\theta^{\mathcal{S}}_{k-1}}\mathcal{L}(\theta^{\mathcal{S}}_{k-1}) \end{aligned} \tag{16}$$

where, $\nabla_{\theta^{\mathcal{T}}_{k-1}}$ and $\nabla_{\theta^{\mathcal{S}}_{k-1}}$ denote the teacher's and student's gradients during the (k-1)-th training epoch, respectively. $\eta^{\mathcal{T}}$ and $\eta^{\mathcal{S}}$ represent the learning rates of the teacher and student, respectively. The pseudo-code of HMKD is shown in Algorithm 1.

Algorithm 1 HMKD

Input: $\mathcal{D} = \{\mathcal{D}_{train}, \mathcal{D}_{val}, \mathcal{D}_{test}\}$; ▷ $\mathcal{D}_{train}$, $\mathcal{D}_{val}$, and $\mathcal{D}_{test}$ are the training, validation, and testing datasets, respectively.
Output: $Y^{\mathcal{T}}$ and $Y^{\mathcal{S}}$; ▷ $Y^{\mathcal{T}}$ and $Y^{\mathcal{S}}$ are the teacher's and student's predictions, respectively.

1: Initialize the teacher's and student's parameters, $\theta^{\mathcal{T}}_0$ and $\theta^{\mathcal{S}}_0$;
2: //Training and validation
3: **for** $k = 1$ to $Epochs$ **do** ▷ $Epochs$ is the size of training epochs.
4:    Feedforward $\mathcal{D}_{train}$ into the teacher and student;
5:    Obtain $\mathcal{L}^{\mathcal{T}}$ and $\mathcal{L}^{\mathcal{S}}$ using Eq. (7)(12);
6:    Update $\theta^{\mathcal{T}}_k$ and $\theta^{\mathcal{S}}_k$ using Eq. (16);
7:    **if** $k > 1$ **then**
8:       Validate the teacher and student using $\mathcal{D}_{val}$;
9:    **end if**
10: **end for**
11: //Testing
12: Obtain $Y^{\mathcal{T}}$ and $Y^{\mathcal{S}}$ using $\mathcal{D}_{test}$.

4 Experiments

This section introduces the experimental setup and performance metrics. Following that, it delves into verification of hyper-parameter sensitivity and the corresponding ablation study. Finally, it assesses the overall performance of HMKD and visualizes the representations learned.

4.1 Experimental Setting

4.1.1 Data Description

To assess the performance of HMKD, we select three well-known wearable HAR datasets, as outlined below:

- **HAPT**: collected from 30 volunteers aged 19-48 years, is a smartphone-based recognition dataset for human activities and postural transitions (HAPT) [55]. The sensor signals, which include accelerometer and gyroscope with noise filters, were sampled in fixed-width sliding windows of 2.56 seconds with a 50% overlap (128 readings per window). Each sample is represented as a 561-feature vector, encompassing time and frequency domain variables. The dataset includes six basic activities: Walking (Wk), Walking_Upstairs (Wu), Walking_Downstairs (Wd), Sitting (St), Standing (Sd), and Laying (Ly). Additionally, it features six static postures, namely Stand-to-Sit (DtS), Sit-to-Stand (StD), Sit-to-Lie (StL), Lie-to-Sit (LtS), Stand-to-Lie(DtL), and Lie-to-Stand (LtD).
- **WISDM**: the Wireless Sensor Data Mining (WISDM) [56] lab collected accelerometer data at a rate of every 50ms, with a signal sample rate set to 20Hz. The dataset comprises 1,098,207 examples of multiple physical activities, each characterized by six attributes: user, activity, timestamp, x-acceleration, y-acceleration, and z-acceleration. The dataset includes six activities: Walking (Wk), Jogging (Jg), Upstairs (Us), Downstairs (Ds), Sitting (St), and Standing (Sd).
- **UCI_HAR**: the Human Activity Recognition using smartphones dataset from the University of California Irvine Machine Learning Repository (UCI_HAR) [57] was gathered from 30 volunteers aged 19-48 years. Each volunteer wore a smartphone (Samsung Galaxy S II) on their waist and performed six activities: Walking (Wk), Walking_Upstairs (Wu), Walking_Downstairs (Wd), Sitting (St), Standing (Sd), and Laying (Ly). The dataset includes 3-axial linear acceleration and 3-axial angular velocity measurements recorded at a constant rate of 50Hz, with the signal sample rate set to 20Hz.

We collect the detailed information of the three datasets in Table 1.

TABLE 1
Details of three HAR datasets.

Dataset	Sample Rate	Activities	Classes	Samples
HAPT	50Hz	Walking (Wk), Walking_Upstairs (Wu), Walking_Downstairs (Wd), Sitting (St), Standing (Sd), Laying (Ly), Stand-to-Sit (DtS), Sit-to-Stand (StD), Sit-to-Lie (StL), Lie-to-Sit (LtS), Stand-to-Lie(DtL), and Lie-to-Stand (LtD)	12	10,929
WISDM	20Hz	Walking (Wk), Jogging (Jg), Upstairs (Us), Downstairs (Ds), Sitting (St), and Standing (Sd)	6	1,098,207
UCI_HAR	50Hz	Walking (Wk), Walking_Upstairs (Wu), Walking_Downstairs (Wd), Sitting (St), Standing (Sd), and Laying (Ly)	6	10,299

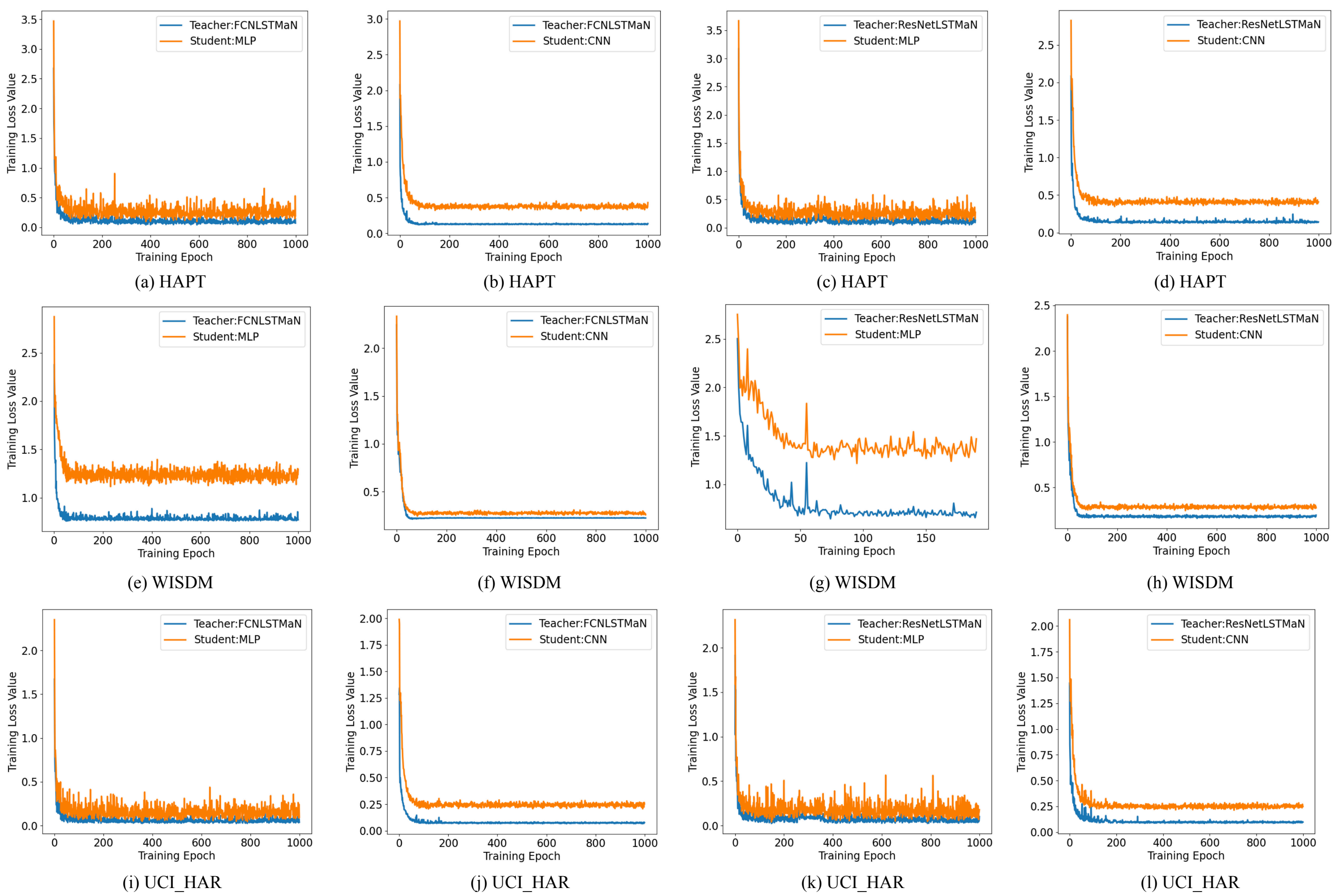


Fig. 4. Training loss values obtained by various teacher and student models during training on three HAR datasets.

4.1.2 Data Preprocessing

Following [5], [8], [9], [10], [16], we utilize the fixed time window method to integrate activity data gathered from diverse sensors. Each sensor involves employing filtering technologies like Kalman, low-pass, and wavelet filters for noise elimination and achieving stable sampling and data frequency. Simultaneously, the sequence data within a fixed time window is fed to HAR models as input, which aims for effective sensor data fusion within a specific time window to enhance activity recognition robustness. The representation of sensor data at the m-th timestamp is denoted by D_m, while WZ represents the size of the fixed time window. The time series data gathered within this fixed time window is represented as x_i, defined in Eq. (17).

$$x_i = [D_1, D_2, ..., D_{WZ}], \quad i = 1, 2, ..., N \tag{17}$$

As recommended by [5], [8], [10], [16], [44], [56], [57], we set WZ on the HAPT, WISDM, and UCI_HAR datasets to 561, 48, and 561, respectively.

4.1.3 Data Partition

As suggested in [5], [8], [15], [16], [35], [37], [41], [42], [43], [44], [58], each dataset is partitioned into two groups using a 7:3 ratio. To determine the optimal hyper-parameters for HMKD, the first group is further divided into training and validation sets, with an 8:2 ratio. Meanwhile, the second one is designated as the testing set. This partition scheme facilitates a systematic approach to hyper-parameter tuning and model evaluation, ensuring a comprehensive assessment of the proposed approach.

4.1.4 Implementation details

In FCNLSTMaN and ResNetLSTMaN, we set the unit number of each LSTM-based attention layer to 128. In this paper, RMSPropOptimizer is employed as the optimizer, with the momentum term, initial learning rate, and decay value set to 0.9, 0.001, and 0.9, respectively. The experiments are conducted using a computer with Ubuntu 18.04 OS, equipped with an Nvidia RTX 2080Ti GPU featuring 22GB, and an AMD R5 1400 CPU with 32GB RAM. To depict the specific training of HMKD, we outline the training loss values acquired from distinct teacher and student models throughout the entire training process on the three HAR datasets in Fig. 4.

4.2 Performance Metrics

Following [5], [8], [13], [15], [16], [35], [41], [43], [44], we consider two widely adopted metrics, namely, $Accuracy$ and F-measure (F_1), in performance comparison. These metrics are defined in Eqs. (18) and (19).

$$Accuracy = \frac{NTP + NTN}{NTP + NTN + NFP + NFN} \times 100\% \tag{18}$$

$$F_1 = \frac{Precision \times Recall}{Precision + Recall} \tag{19}$$

where,

$$\begin{aligned} Precision &= \frac{NTP}{NTP + NFN} \times 100\% \\ Recall &= \frac{NTP}{NTP + NTN} \times 100\% \end{aligned} \tag{20}$$

where, NTP and NTN denote the numbers of true positive and true negative instances, respectively. NFP and NFN are the numbers of false positive and false negative instances, respectively.

4.3 Hyper-parameter Sensitivity

We study the influence of hyper-parameter settings on the performance of HMKD on the HAPT, WISDM, and UCI_HAR datasets.

4.3.1 HMKD with different t_{KD} values

t_{KD} operates as a scaling coefficient for the features of both the teacher and student, playing a pivotal role in promoting the knowledge flow between these models. As illustrated in Table 2, the optimal setting for t_{KD} is found to be 1.0. This setting proves to be the most effective for HMKD, resulting in the highest F_1 value across each HAR dataset.

TABLE 2
F_1 results obtained by HMKD with different t_{KD} on three HAR datasets.

Teacher	Student	t_{KD}	HAPT (%)	WISDM (%)	UCI_HAR (%)
FCNLSTMaN	MLP	0.10	88.89	80.86	86.46
		0.50	89.49	80.32	87.84
		1.00	**92.43**	**83.90**	**90.21**
		2.00	90.97	80.13	89.57
		5.00	88.24	78.91	87.66
	CNN	0.10	88.53	88.38	89.57
		0.50	91.25	90.04	90.18
		1.00	**93.04**	**91.00**	**92.58**
		2.00	90.97	89.99	90.89
		5.00	88.89	89.14	88.57
ResNetLSTMaN	MLP	0.10	88.54	80.92	87.34
		0.50	90.02	83.93	89.62
		1.00	**92.77**	**85.69**	**91.72**
		2.00	90.04	83.93	88.94
		5.00	89.49	82.89	88.23
	CNN	0.10	89.49	88.74	89.12
		0.50	90.73	89.99	90.89
		1.00	**93.15**	**92.13**	**93.90**
		2.00	89.49	89.31	90.18
		5.00	88.89	88.92	89.02

TABLE 3
F_1 results obtained by HMKD with different KD losses on three HAR datasets.

Teacher	Student	Loss	HAPT (%)	WISDM (%)	UCI_HAR (%)
FCNLSTMaN	MLP	L_1	90.04	81.06	88.89
		L_2	90.97	82.15	89.02
		KL	92.04	83.05	89.12
		CE	88.19	80.92	85.96
		JS	**92.43**	**83.90**	**90.21**
	CNN	L_1	90.97	89.99	90.18
		L_2	91.25	90.04	90.18
		KL	92.16	90.02	92.19
		CE	88.89	89.14	88.57
		JS	**93.04**	**91.00**	**92.58**
ResNetLSTMaN	MLP	L_1	90.02	84.44	89.62
		L_2	91.63	84.68	89.57
		KL	92.16	84.85	90.34
		CE	87.13	81.79	86.33
		JS	**92.77**	**85.69**	**91.72**
	CNN	L_1	90.73	90.02	90.89
		L_2	91.46	90.63	91.17
		KL	92.85	91.38	92.58
		CE	88.53	89.19	89.19
		JS	**93.15**	**92.13**	**93.90**

4.3.2 HMKD with different KD losses

Selecting an appropriate KD loss function is crucial for quantifying the average difference between the outputs of teacher and student models. Table 3 presents the F_1 results achieved by HMKD using 5 different KD losses on three HAR datasets, namely, KL, JS, L_1 (Mean Absolute Error), CE (Cross Entropy), and L_2 (Mean Squared Error) losses. Among them, JS outperforms the other four. Consequently, JS loss is chosen as the preferred option to enhance the knowledge transfer between teacher and student models.

4.4 Ablation Study

We investigate the effects of different components on HMKD on three HAR datasets.

4.4.1 Effectiveness of Mutual Learning

To verify the effectiveness of mutual learning on HMKD, we compare it with three variants in terms of F_1, listed below.

TABLE 4
F_1 results obtained by vairous HMKD variants on three HAR datasets.

Teacher	Student	Method	HAPT (%)	WISDM (%)	UCI_HAR (%)
FCNLSTMaN	MLP	MLP	83.58	78.91	85.07
		HUKD	88.24	79.49	88.49
		HMKD w/o WEF	88.89	81.86	87.46
		HMKD (Avg.)	91.63	82.72	89.49
		HMKD	**92.43**	83.90	**90.21**
		HMKD (Oline)	92.39	**84.28**	90.18
	CNN	CNN	85.88	88.00	87.05
		HUKD	88.24	89.19	88.49
		HMKD w/o WEF	90.49	89.38	89.94
		HMKD (Avg.)	92.77	89.49	90.89
		HMKD	93.04	91.00	**92.58**
		HMKD (Oline)	**93.28**	**91.79**	91.82
ResNetLSTMaN	MLP	MLP	83.58	78.91	85.07
		HUKD	88.19	82.72	89.14
		HMKD w/o WEF	89.00	83.89	89.23
		HMKD (Avg.)	90.73	84.39	89.91
		HMKD	**92.77**	85.69	**91.72**
		HMKD (Oline)	91.46	**86.39**	90.89
	CNN	CNN	85.88	88.00	87.05
		HUKD	88.53	89.19	89.19
		HMKD w/o WEF	90.89	89.89	89.57
		HMKD (Avg.)	92.85	90.63	91.17
		HMKD	93.15	92.13	**93.90**
		HMKD (Oline)	**93.69**	**93.01**	92.89

- *MLP*: the pure MLP with three dense (fully-connected) layers, without the incorporation of any distillation method.
- *CNN*: the pure CNN with three convolutional layers, without the incorporation of any distillation method.
- *HUKD*: a heterogeneous unidirectional KD method, transfering knowledge from teacher to student.

Table 4 collects the results of the algorithms above on three HAR datasets. Notably, the performance enhancement of KD for the student is evident. KD facilitates knowledge transfer from the teacher to the student, thereby regularizing the student model and improving its feature extraction capabilities. For instance, when FCNLSTMaN serves as the teacher and MLP as the student, both HUKD and HMKD contribute to an improvement in the F_1 value of MLP on the HAPT dataset, by 4.66% and 8.85%, respectively.

Meanwhile, HMKD outperforms HUKD on each HAR dataset. This is because that HUKD neglects the reciprocal nature of the distillation process, prioritizing the teacher's importance to the student's model while overlooking the student's significance to the teacher. Conversely, HMKD acknowledges and underscores the mutual learning aspect, recognizing the importance of both the teacher and the student in the distillation process. This perspective results in the HMKD's superiority over HUKD.

4.4.2 Effectiveness of Weighted Ensemble Feature

To study the impact of weight ensemble feature on HMKD, we compare it with two variants as follows.

- *HMKD w/o WEF*: HMKD without weight ensemble feature.
- *HMKD (Avg.)*: HMKD with the average feature method instead of weight ensemble feature.

Leveraging the weighted ensemble feature to enhance knowledge transfer in the intermediate layers of the teacher and student, HMKD demonstrates superiority over HMKD w/o WEF on all datasets in Table 4. In contrast to the average feature method, the weighted ensemble feature takes into account the actual proportion of each feature in the intermediate layers of the model, well representing the corresponding features. This is why HMKD is better than HMKD (Avg.) on all datasets.

4.4.3 Online vs. Offline

During the distillation process, if a teacher model is pre-trained, it conducts online distillation for its student; otherwise, it performs offline distillation for its student. HMKD (Online) refers to the online distillation within the HMKD framework.

Leveraging prior knowledge, HMKD (Online) demands less training time and converges faster than HMKD, e.g., when ResNetLSTMaN and CNN serve as the teacher and student, the training time of HMKD (Online) on the HAPT dataset is approximately 40% shorter than that of HMKD. As shown in Table 4, there is minimal difference in overall performance between HMKD (Online) and HMKD. While HMKD (Online) shows slight improvement on a few datasets compared to HMKD, the pre-trained teacher model in HMKD (Online) tends to be time-consuming and consumes significant computational resources. This is why the offline HMKD approach is chosen.

In summary, the mutual learning, weight ensemble feature, and offline strategy are all crucial components for HMKD.

4.5 Experimental Comparisons and Analysis

To verify the performance of HMKD, we compare it with a number of KD algorithms against F_1 value, listed below.

- *MLP*: the pure MLP with three dense (fully-connected) layers, without the incorporation of any distillation method.
- *CNN*: the pure CNN with three convolutional layers, without the incorporation of any distillation method.
- *FCNLSTMaN*: the pure FCNLSTMaN, without the incorporation of any distillation method.
- *ResNetLSTMaN*: the pure ResNetLSTMaN, without the incorporation of any distillation method.
- *ResponseKD*: a vanilla response-based KD method for HAR [17].
- *FitNet*: a hint-based KD method for HAR [22].
- *CC*: a correlation congruence-based KD method for HAR [50].
- *CRD*: a contrastive representation distillation method for HAR [17].
- *RelaKD*: a relational KD method for HAR [59].
- *DKD*: a decoupled KD method for HAR [46].
- *MD*: a mutual distillation method for HAR [26].
- *DMD*: a dense cross-layer mutual-distillation method for HAR [27].
- *DIST*: a correlation-based KD method with a stronger teacher for HAR [48].
- *OFA*: a one-for-all KD framework for HAR [25].

Table 5 shows the F_1 results obtained by various KD algorithms on three HAR datasets. Evidently, HMKD outperforms all compared KD algorithms on each dataset. For

TABLE 5
F_1 results obtained by various state-of-the-art KD algorithms on three HAR datasets.

Teacher	Student	Method	HAPT (%)	WISDM (%)	UCI_HAR (%)
FCNLSTMaN	MLP	FCNLSTMaN	96.14	98.96	96.92
		MLP	83.58	78.91	85.07
		RsponseKD	84.86	80.13	85.86
		FitNet	85.93	80.04	86.46
		CC	86.87	80.32	86.89
		CRD	88.53	80.58	86.46
		RelaKD	88.67	80.86	86.99
		DKD	88.19	79.49	85.96
		MD	88.24	74.17	88.49
		DMD	88.89	80.86	86.46
		DIST	89.49	80.86	87.66
		OFA	90.04	81.06	88.89
		HMKD	**92.43**	**83.90**	**90.21**
	CNN	FCNLSTMaN	96.14	98.96	96.92
		CNN	85.88	88.00	87.05
		RsponseKD	86.87	88.26	87.66
		FitNet	88.19	88.92	88.49
		CC	88.67	88.74	89.02
		CRD	87.85	89.04	89.12
		RelaKD	88.24	89.19	88.49
		DKD	88.89	89.14	88.57
		MD	85.21	89.31	87.84
		DMD	89.49	88.38	88.94
		DIST	88.53	89.19	89.57
		OFA	90.97	89.99	90.18
		HMKD	**93.04**	**91.00**	**92.58**
ResNetLSTMaN	MLP	ResNetLSTMaN	96.89	99.12	97.49
		MLP	83.58	78.91	85.07
		RsponseKD	85.05	80.92	85.59
		FitNet	85.83	80.93	86.35
		CC	87.13	77.12	87.00
		CRD	86.32	82.93	88.23
		RelaKD	88.19	82.72	89.14
		DKD	87.13	81.79	86.33
		MD	88.54	74.25	87.34
		DMD	88.00	82.89	88.23
		DIST	88.89	83.93	88.94
		OFA	90.02	84.44	89.62
		HMKD	**92.77**	**85.69**	**91.72**
	CNN	ResNetLSTMaN	96.89	99.12	97.49
		CNN	85.88	88.00	87.05
		RsponseKD	86.39	88.26	87.84
		FitNet	86.89	88.74	89.02
		CC	88.19	89.31	89.57
		CRD	87.34	89.19	90.21
		RelaKD	89.49	87.84	89.12
		DKD	88.53	89.19	89.19
		MD	88.89	88.38	88.49
		DMD	87.34	88.89	88.57
		DIST	90.04	89.99	89.49
		OFA	90.73	90.02	90.89
		HMKD	**93.15**	**92.13**	**93.90**

instance, when FCNLSTMaN serves as the teacher and MLP as the student, HMKD achieves the highest F_1 value on the HAPT dataset, namely 92.43%. OFA ranks the second, while ResponseKD yields the least favorable performance.

The observations above can be attributed to the following factors. HMKD establishes mutual learning not only within the intermediate layers of both teacher and student models but also extends to the output layers of them. This approach promotes efficient knowledge flow between the teacher and student, facilitating a comprehensive exchange of information. OFA effectively enhances the knowledge flow from the teacher to the student under heterogeneous architectures, thanks to its ability to project intermediate features into an aligned latent space and its adaptive target enhancement scheme. On the other hand, ResponseKD establishes a simple link between the outputs of teacher and student models through KL. However, this link may pose challenges in transferring sufficient knowledge from the teacher to the student.

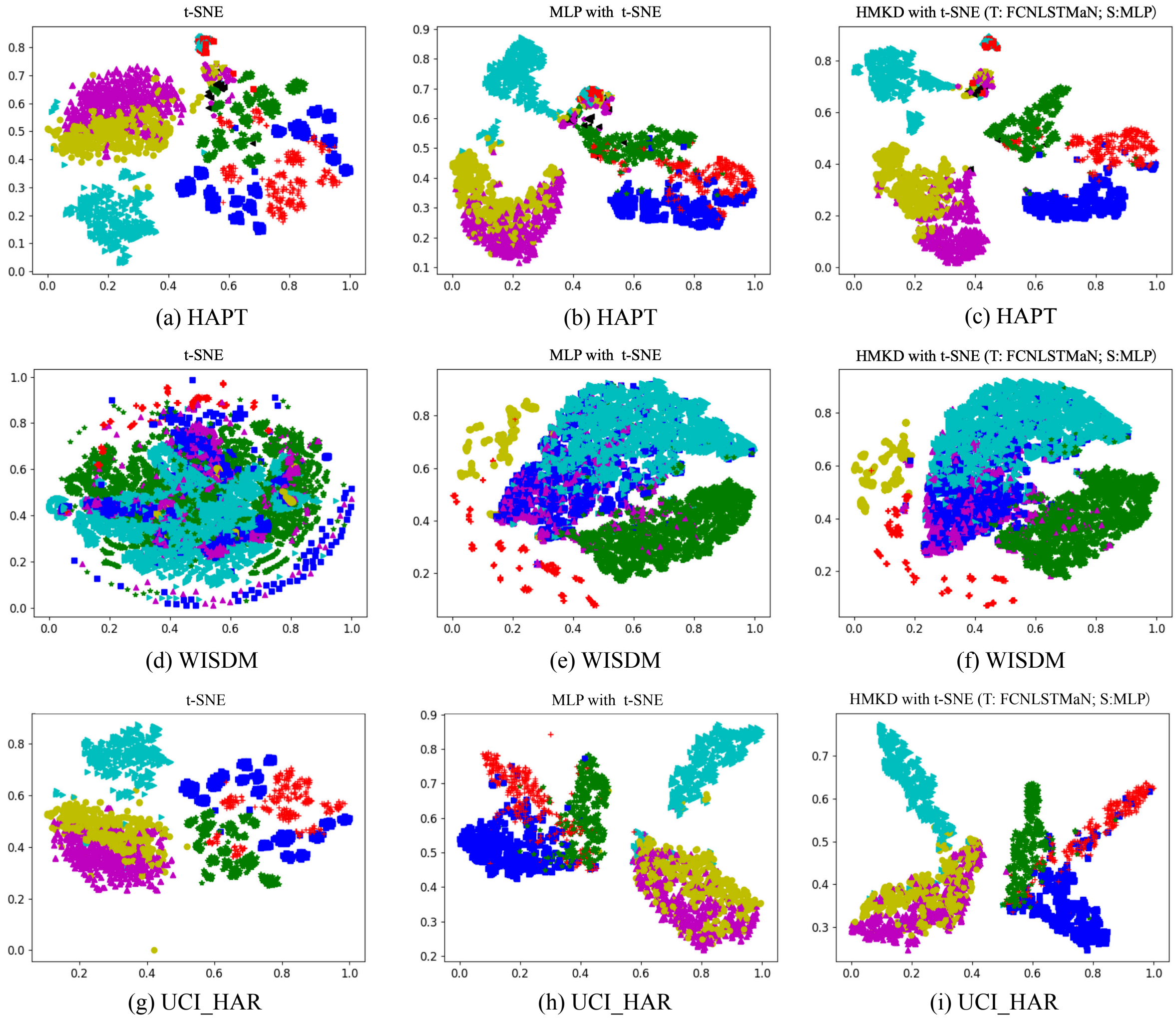


Fig. 5. Visualization of representations learned by t-SNE, MLP with t-SNE, and HMKD with t-SNE on three HAR datasets, where HMKD is based on FCNLSTMaN as the teacher and MLP as the student, namely '(T: FCNLSTMaN; S: MLP)'.

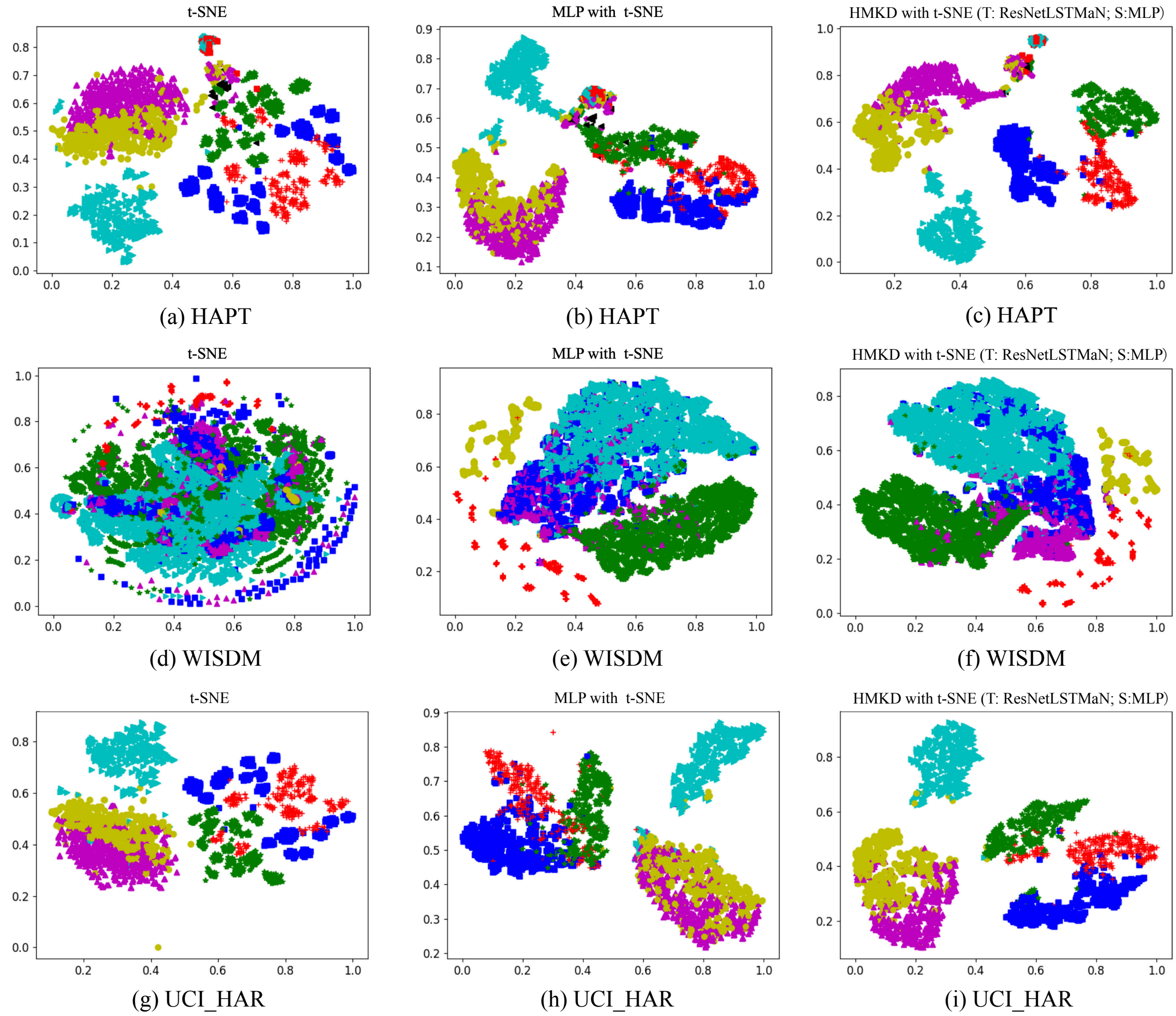
4.6 Representation Visualization

To examine the effectiveness of representation learning in HMKD, we apply t-distributed stochastic neighbor embedding (t-SNE) [60], an unsupervised nonlinear method, to visually represent the learned features. The visualization of representations learned by t-SNE, MLP/CNN with t-SNE, and HMKD with t-SNE on three HAR datasets is presented in Figs. 5, 6, 7, and 8. Figs. 5 and 6 correspond to HMKD with FCNLSTMaN as the teacher, and MLP and CNN as students, respectively, while Figs. 7 and 8 represent HMKD with ResNetLSTMaN as the teacher, and MLP and CNN as students, respectively.

In a comparative analysis with t-SNE alone, HMKD showcases a superior clustering effect by efficiently grouping samples with similar characteristics, enhancing clustering coherence. For instance, as emphasized in Figs. 5 (a), 5 (b), and 5 (c), HMKD with t-SNE successfully clusters HAPT instances that exhibit similar features, highlighting the model's ability to mine distinct patterns. Similar scenarios are seen in Figs. 6, 7, and 8.

5 Conclusion

HMKD encourages mutual interaction between the teacher and student models at intermediate and output layers. This approach helps promote efficient knowledge sharing between teacher and student, enabling a thorough exchange of information across various levels within the models. Given the significant structural differences between teacher and student models, the weighted ensemble feature approach can amalgamate the features extracted from the intermediate layers of these models, which facilitates the knowledge exchange within the intermediate layers of both models. Experimental results show that compared with 10 SOTA KD algorithms, HMKD showcases superior performance on three HAR datasets, in terms of F_1 score. Notably, when employing ResNetLSTMaN and MLP as teacher and student, the F_1 score of MLP sees a growth of approximately 9.19% with the application of HMKD on the HAPT dataset. These results indicate the potential of HMKD for addressing various real-world HAR problems.



(a) HAPT (b) HAPT (c) HAPT

(d) WISDM (e) WISDM (f) WISDM

(g) UCI_HAR (h) UCI_HAR (i) UCI_HAR

Fig. 6. Visualization of representations learned by t-SNE, MLP with t-SNE, and HMKD with t-SNE on three HAR datasets, where HMKD is based on FCNLSTMaN as the teacher and CNN as the student, namely '(T: FCNLSTMaN; S: CNN)'.

REFERENCES

[1] D. Anguita, L. O. A. Ghio, X. Parra, and J. Reyes-Ortiz, "A public domain dataset for human activity recognition using smartphones," in *Proc. 21st Eur. Symposium Artif. Neural Netw., Comput. Intell. Mach. Learn.*, 2013, pp. 437–442.

[2] X. Li, H. Deng, J. Ouyang, H. Wan, W. Yu, and D. Wu, "Act as what you think: Towards personalized eeg interaction through attentional and embedded lstm learning," *IEEE Trans. Mobile Comput.*, pp. 1–13, 2023.

[3] X. Li, X. Wang, T. Song, and J. Hu, "Robust online prediction of spectrum map with incomplete and corrupted observations," *IEEE Trans. Mobile Comput.*, vol. 21, no. 12, pp. 4583–4594, 2022.

[4] B. D. Deebak and S. O. Hwang, "Healthcare applications using blockchain with a cloud-assisted decentralized privacy-preserving framework," *IEEE Trans. Mobile Comput.*, pp. 1–18, 2023.

[5] F. Serpush, M. B. Menhaj, B. Masoumi, and B. Karasfi, "Wearable sensor-based human activity recognition in the smart healthcare system," *Comput. Intel. Neurosc.*, pp. 1–31, 2022.

[6] H. Xing, Z. Xiao, R. Qu, Z. Zhu, and B. Zhao, "An efficient federated distillation learning system for multitask time series classification," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–12, 2022.

[7] Z. Xiao, H. Xing, B. Zhao, R. Qu, S. Luo, P. Dai, K. Li, and Z. Zhu, "Deep contrastive representation learning with self-distillation," *IEEE Trans. Emerg. Top. Comput. Intell.*, pp. 1–13, 2023.

[8] Y. Wang, S. Cang, and H. Yu, "A survey on wearable sensor modality centred human activity recognition in health care," *Expert Syst. Appl.*, vol. 127, pp. 167–190, 2018.

[9] C. Zhu and W. Sheng, "Wearable sensor-based hand gesture and daily activity recognition for robot-assisted living," *IEEE Trans. Syst. Man Cybern. Part A Syst. Humans*, vol. 41, no. 3, pp. 569–573, 2011.

[10] Z. Chen, Q. Zhu, Y. C. Soh, and L. Zhang, "Robust human activity recognition using smartphone sensors via ct-pca and online svm," *IEEE Trans. Ind. Inform.*, vol. 13, no. 6, pp. 3070–3080, 2017.

[11] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, pp. 436–444, 2015.

[12] X. Shu, L. Zhang, Y. Sun, and J. Tang, "Host–parasite: Graph lstm-in-lstm for group activity recognition," *IEEE Trans. Neur. Net. Lear.*, vol. 32, no. 2, pp. 663–674, 2021.

[13] M. A. A. Al-qaness, A. Dahou, M. A. Elaziz, and A. M. Helmi, "Multi-resatt: Multilevel residual network with attention for human activity recognition using wearable sensors," *IEEE Trans. Ind. Inform.*, vol. 19, no. 1, pp. 144–152, 2023.

[14] S. Xia, L. Chu, L. Pei, Z. Zhang, W. Yu, and R. C. Qiu, "Learning disentangled representation for mixed-reality human activity recognition with a single imu sensor," *IEEE Trans. Neur. Net. Lear.*, vol. 70, pp. 1–14, 2021.

[15] S. Xu, L. Zhang, W. Huang, H. Wu, and A. Song, "Deformable convolutional networks for multimodal human activity recognition using wearable sensors," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–14, 2022.

[16] Z. Xiao, X. Xu, H. Xing, F. Song, X. Wang, and B. Zhao, "A federated learning system with enhanced feature extraction for human activity recognition," *Knowl.-Based Syst.*, vol. 229, pp. 1–14, 2021.

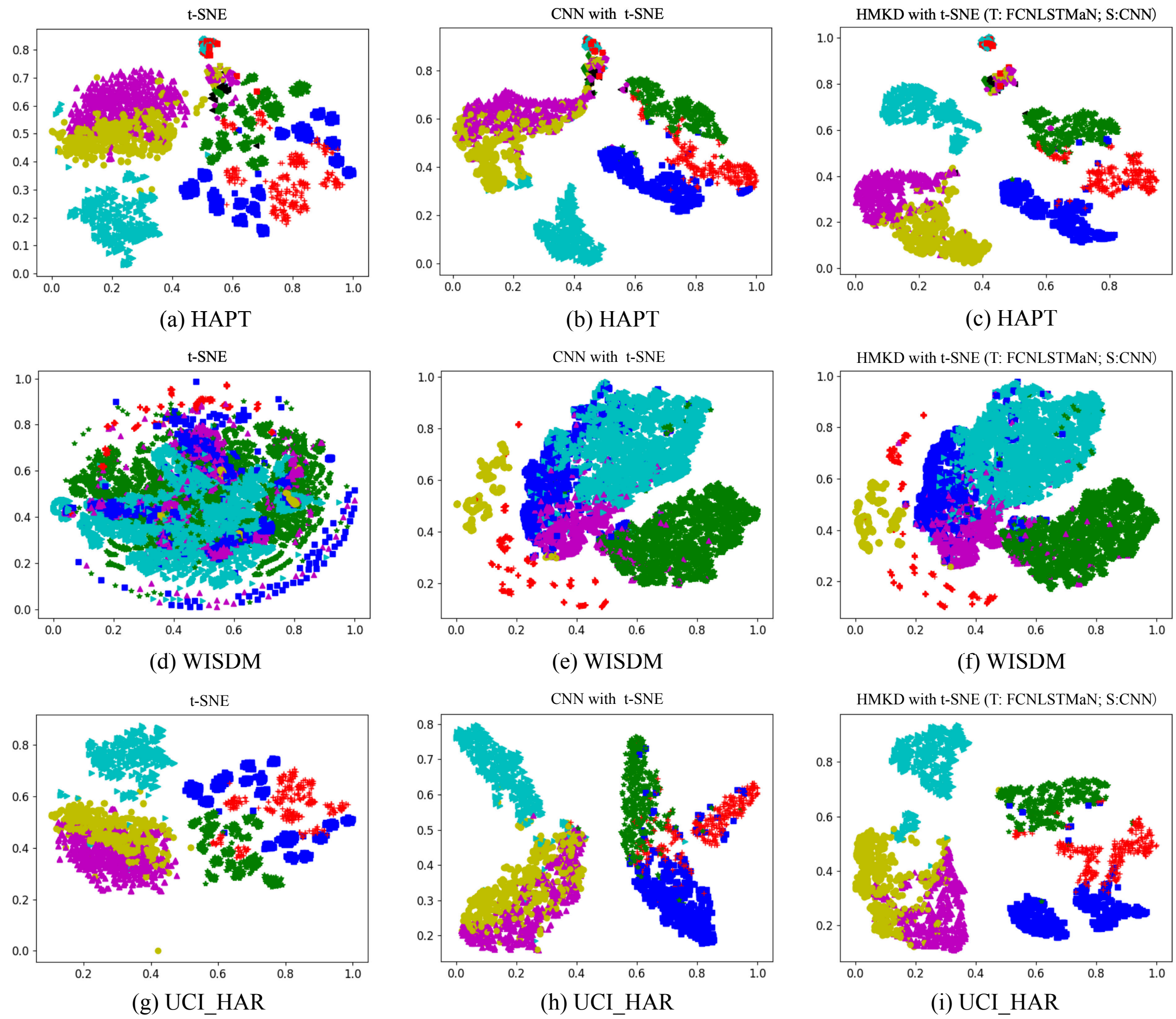


Fig. 7. Visualization of representations learned by t-SNE, CNN with t-SNE, and HMKD with t-SNE on three HAR datasets, where HMKD is based on ResNetLSTMaN as the teacher and MLP as the student, namely '(T: ResNetLSTMaN; S: MLP)'.

[17] G. Hinton, O. Vinyals, and J. Dean, "Distillation the knowledge in a neural network," *arXiv preprint arXiv: 1503.02531*, 2015.

[18] J. Gou, B. Yu *et al.*, "Knowledge distillation: A survey," *Int. J. Comput. Vis.*, vol. 129, pp. 1789–1819, 2021.

[19] Z. Feng, J. Lai, and X. Xie, "Resolution-aware knowledge distillation for efficient inference," *IEEE Trans. Image Process.*, vol. 30, pp. 6985–6996, 2021.

[20] Q. Xu, M. Wu, X. Li, K. Mao, and Z. Chen, "Contrastive distillation with regularized knowledge for deep model compression on sensor-based human activity recognition," *IEEE Trans. Ind. Cyber-Phys. Syst.*, vol. 1, pp. 217–226, 2023.

[21] S. Deng, J. Chen, D. Teng, C. Yang, D. Chen, T. Jia, and H. Wang, "Lhar: Lightweight human activity recognition on knowledge distillation," *IEEE J. Biomed. Health. Inf.*, pp. 1–10, 2023.

[22] A. Romero, N. Ballas, S. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "Fitnet: hints for thin deep nets," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–13.

[23] Y. Zhang, Z. Yan, X. Sun, X. Lu, J. Li, Y. Mao, and L. Wang, "Bridging the gap between cumbersome and light detectors via layer-calibration and task-disentangle distillation in remote sensing imagery," *IEEE Trans. Geosci. Remote*, vol. 61, pp. 1–18, 2023.

[24] J. Gou, L. Sun, B. Yu, S. Wan, W. Ou, and Z. Yi, "Multilevel attention-based sample correlations for knowledge distillation," *IEEE Trans. Ind. Inform.*, vol. 19, no. 5, pp. 7099–7109, 2023.

[25] Z. Hao, J. Guo, K. Han, Y. Tang, H. Hu, Y. Wang, and C. Xu, "One-for-all: Bridge the gap between heterogeneous architectures in knowledge distillation," in *Proc. Adv. Neural Inf. Proces. Syst.*, 2023, pp. 1–13.

[26] Y. Zhang, T. Xiang, T. M. Hospedales, and H. Lu, "Deep mutual learning," in *Proc IEEE Comput. Soc. Conf. Comput. Vision Pattern Recognit. (CVPR)*, 2018, pp. 4320–4328.

[27] A. Yao and D. Sun, "Knowledge transfer via dense cross-layer mutual-distillation," in *Proc. Lect. Notes Comput. Sci., ECCV*, 2020, pp. 294–311.

[28] H. Xing, Z. Xiao, D. Zhan, S. Luo, P. Dai, and K. Li, "Selfmatch: Robust semisupervised time-series classification with self-distillation," *Int. J. Intell. Syst.*, pp. 1–28, 2022.

[29] A. Gupta, A. Kembhavi, and L. S. Davis, "Observing human-object interactions: Using spatial and functional compatibility for recognition," *IEEE Trans. Pattern Anal.*, vol. 31, no. 10, pp. 1775–1789, 2009.

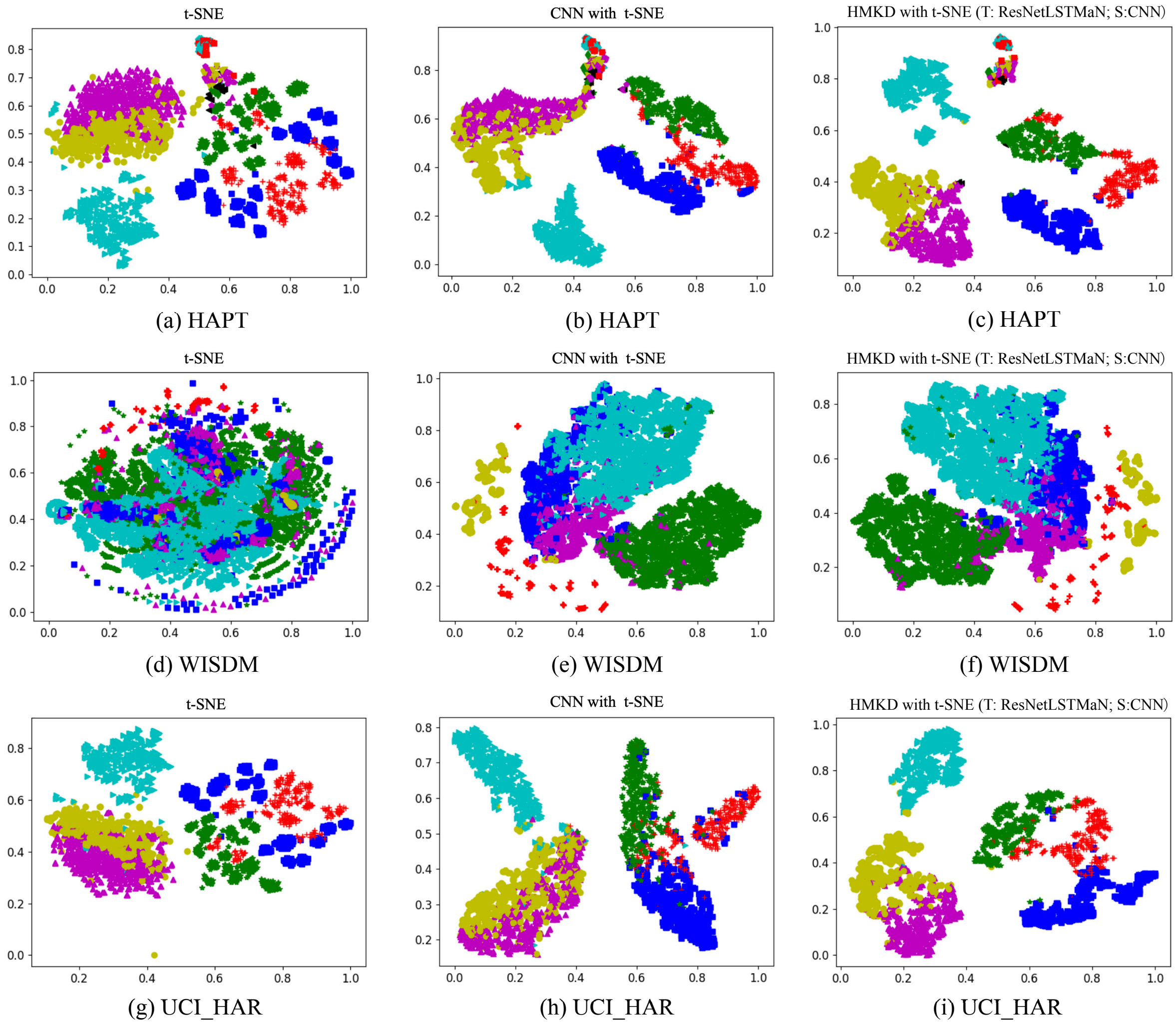
[30] S. Xia, L. Chu, L. Pei, Z. Zhang, W. Yu, and R. C. Qiu, "Learning disentangled representation for mixed- reality human activity recognition with a single imu sensor," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–14, 2021.

[31] D. Tao, L. Jin, Y. Wang, and X. L, "Rank preserving discriminant analysis for human behavior recognition on wireless sensor networks," *IEEE Trans. Ind. Inform.*, vol. 10, no. 1, pp. 813–823, 2014.

[32] R. A. Hamad, A. S. Hidalgo, M. R. Bouguelia, M. E. Estevez, and J. M. Quero, "Efficient activity recognition in smart homes using delayed fuzzy temporal windows on binary sensors," *IEEE J. Biomed. Health*, vol. 24, no. 2, pp. 387–395, 2020.

[33] S. Khalifa, G. Lan, M. Hassan, A. Seneviratne, and S. K. Das, "Harke: Human activity recognition from kinetic energy harvesting data in wearable devices," *IEEE Trans. Mobile Comput.*, vol. 17, no. 6, pp. 1353–1368, 2018.

[34] D. Ravi, C. Wong, B. Lo, and G. Yang, "Deep learning for human



(a) HAPT (b) HAPT (c) HAPT

(d) WISDM (e) WISDM (f) WISDM

(g) UCI_HAR (h) UCI_HAR (i) UCI_HAR

Fig. 8. Visualization of representations learned by t-SNE, CNN with t-SNE, and HMKD with t-SNE on three HAR datasets, where HMKD is based on ResNetLSTMaN as the teacher and CNN as the student, namely '(T: ResNetLSTMaN; S: CNN)'.

activity recognition: a resource efficient implementation on low-power devices," in *In Proc. Annual Body Sens. Netw. Conf.*, 2016, pp. 71–76.

[35] H. Zhang, Z. Xiao, J. Wang, F. Li, and E. Szczerbicki, "A novel iot-perceptive human activity recognition (har) approach using multihead convolutional attention," *IEEE Internet Things J.*, vol. 7, no. 2, pp. 1072–1080, 2020.

[36] Z. Chen, L. Zhang, C. Jiang, Z. Cao, and W. Cui, "Wifi csi based passive human activity recognition using attention based blstm," *IEEE Trans. Mobile Comput.*, vol. 18, no. 11, pp. 2714–2724, 2019.

[37] Y. Dong, X. Li, J. Dezert, M. O. Khyam, M. Noor-A-Rahim, and S. S. Ge, "Dezert-smarandache theory-based fusion for human activity recognition in body sensor networks," *IEEE Trans. Ind. Inform.*, vol. 16, no. 11, pp. 7138–7149, 2020.

[38] S. Abbaspour, F. Fotouhi, A. Sedaghatbaf, H. Fotouhi, M. Vahabi, and M. Linden, "A comparative analysis of hybrid deep learning models for human activity recognition," *Sensors*, vol. 20, no. 19, pp. 1–14, 2020.

[39] C. Tang, I. Perez-Pozuelo *et al.*, "Selfhar: Improving human activity recognition through self-training with unlabeled data," in *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 2021, pp. 1–30.

[40] B. Khaertdinov, E. Ghaleb, and S. Asteriadis, "Contrastive self-supervised learning for sensor-based human activity recognition," in *Proc. IEEE Int. Jt. Conf. Biom.*, 2021, pp. 1–8.

[41] F. Gu, K. Khoshelham, S. Valaee, J. Shang, and R. Zhang, "Locomotion activity recognition using stacked denoising autoencoders," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 2085–2093, 2018.

[42] W. Gao, L. Zhang, W. Huang, F. Min, J. He, and A. Song, "Deep neural networks for sensor-based human activity recognition using selective kernel convolution," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–13, 2021.

[43] Y. Tang, Q. Teng, L. Zhang, F. Min, and J. He, "Layer-wise training convolutional neural networks with smaller filters for human activity recognition using wearable sensors," *ArXiv Preprint arXiv:2005.03948*, 2020.

[44] Z. Xiao, H. Tong, R. Qu, H. Xing, S. Luo, Z. Zhu, F. Song, and L. Feng, "Capmatch: Semi-supervised contrastive transformer capsule with feature-based knowledge distillation for human activity recognition," *IEEE Trans. Neur. Net. Lear.*, pp. 1–16, 2024.

[45] Y. Jain *et al.*, "Collossl: Collaborative self-supervised learning for human activity recognition," in *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 2022, pp. 1–28.

[46] B. Zhao, Q. Cui, R. Song, Y. Qiu, and J. Liang, "Decoupled knowledge distillation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recognit. (CVPR)*, 2022, pp. 11 943–11 952.

[47] W.-C. Kao, H.-X. Xie, C.-Y. Lin, and W.-H. Cheng, "Specific expert learning: Enriching ensemble diversity via knowledge distillation," *IEEE Trans. Cybernetics*, vol. 53, no. 4, pp. 2494–2505, 2023.

[48] T. Huang, S. You, F. Wang, C. Qian, and C. Xu, "Knowledge distillation from a stronger teacher," in *Proc. Adv. Neural Inf. Proces. Syst.*, 2022, pp. 1–12.

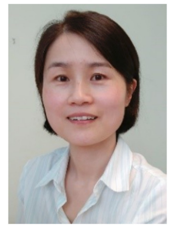
[49] H. Zhao, X. Sun, J. Dong, C. Chen, and Z. Dong, "Highlight every step: Knowledge distillation via collaborative teaching," *IEEE Trans. Cybernetics*, vol. 52, no. 4, pp. 2070–2081, 2022.

[50] B. Peng, X. Jin, D. Li, S. Zhou, Y. Wu, J. Liu, Z. Zhang, and Y. Liu, "Correlation congruence for knowledge distillation," in *Proc. IEEE*

Int. Conf. Comput. Vision (ICCV), 2019, pp. 5006–5015.
[51] Z. Hao, Y. Luo, Z. Wang, H. Hu, and J. An, "Cdfkd-mfs: Collaborative data-free knowledge distillation via multi-level feature sharing," *IEEE Trans. Multimedia*, vol. 24, pp. 4262–4274, 2022.
[52] Y. Tian, D. Krishnan, and P. Isola, "Contrastive representation distillation," in *Proc. Int. Conf. Learn. Repre.*, 2020, pp. 1–19.
[53] L. Yu, V. Yazici, X. Liu, J. Weijer, Y. Chen, and A. Ramisa, "Learning metrics from teachers: compact networks for image embedding," in *Proc IEEE Comput. Soc. Conf. Comput. Vision Pattern Recognit. (CVPR)*, 2019, pp. 2902–2911.
[54] Z. Xiao, X. Xu, H. Xing, S. Luo, P. Dai, and D. Zhan, "Rtfn: A robust temporal feature network for time series classification," *Inf. Sci.*, vol. 571, pp. 65–86, 2021.
[55] J.-L. Reyes-Ortiz, L. Oneto, A. Sam, X. Parra, and D. Anguita, "Transition-aware human activity recognition using smartphones," *Neurocomputing*, vol. 171, pp. 754–767, 2016.
[56] J. R. Kwapisz, G. M. Weiss, and S. Moore, "Activity recognition using cell phone accelerometers," in *Proc. 14th International Workshop on Knowledge Discovery from Sensor Data (at KDD-10)*, 2010, pp. 72–84.
[57] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine," in *Proc. Lect. Notes Comput. Sci., IWAAL*, 2012, pp. 216–223.
[58] S. Xia, L. Chu, L. Pei, Z. Zhang, W. Yu, and R. C. Qiu, "Learning disentangled representation for mixed- reality human activity recognition with a single imu sensor," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–14, 2021.
[59] W. Park, D. Kim, Y. Lu, and M. Cho, "Relational knowledge distillation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recognit. (CVPR)*, 2019, pp. 3962–3971.
[60] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *J. Mach. Learn. Res.*, vol. 9, no. 11, 2008.

Zhiwen Xiao (M) received the B.Eng. degree in network engineering from the Chengdu University of Information Technology, Chengdu, China, and the M.Eng. degree in computer science from the Northwest A & F University, Yangling, China. He is pursuing the Ph.D. degree in computer science at Southwest Jiaotong University, Chengdu, China. His research interests include semantic communication, federated learning (FL), representation learning, data mining, and computer vision.

Huanlai Xing (M) received Ph.D. degree in computer science from University of Nottingham (Supervisor: Dr Rong Qu), Nottingham, U.K., in 2013. He was a Visiting Scholar in Computer Science, The University of Rhode Island (Supervisor: Dr. Haibo He), USA, in 2020-2021. Huanlai Xing is with the School of Computing and Artificial Intelligence, Southwest Jiaotong University (SWJTU), and Tangshan Institute of SWJTU. He was on Editorial Board of SCIENCE CHINA INFORMATION SCIENCES. He was a member of several international conference program and senior program committees, such as ECML-PKDD, MobiMedia, ISCIT, ICCC, TrustCom, IJCNN, and ICSINC. His research interests include semantic communication, representation learning, data mining, reinforcement learning, machine learning, network function virtualization, and software defined networking.



Rong Qu (SM'12) is a full Professor at the School of Computer Science, University of Nottingham. She received her B.Sc. in Computer Science and Its Applications from Xidian University, China in 1996 and Ph.D. in Computer Science from The University of Nottingham, U.K. in 2003. Her research interests include the modelling and optimisation for logistics transport scheduling, personnel scheduling, network routing, portfolio optimization and timetabling problems by using evolutionary algorithms, mathematical programming, constraint programming in operational research and artificial intelligence. These computational techniques are integrated with knowledge discovery, machine learning and data mining to provide intelligent decision support on logistic fleet operations at SMEs, workforce scheduling at hospitals, policy making in education, and cyber security for connected and autonomous vehicles.

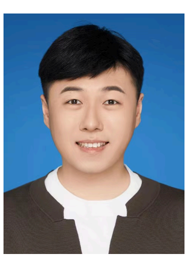
Dr. Qu is an associated editor at Engineering Applications of Artificial Intelligence, IEEE Computational Intelligence Magazine, IEEE Transactions on Evolutionary Computation, Journal of Operational Research Society and PeerJ Computer Science. She is a Senior IEEE Member since 2012 and the Vice-Chair of Evolutionary Computation Task Committee since 2019 and Technical Committee on Intelligent Systems Applications (2015-2018) at IEEE Computational Intelligence Society. She has guest edited special issues on the automated design of search algorithms and machine learning at the IEEE Transactions on Pattern Analysis and Machine Intelligence and IEEE Computational Intelligence Magazine.

Hui Li received the B.Sc. and M.Sc. degrees in applied mathematics from the School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an, China, in 1999 and 2002, respectively, and the Ph.D. degree in computer science from the University of Essex, Colchester, U.K., in 2008. He is currently a Professor with the School of Mathematics and Statistics, Xi'an Jiaotong University. His current research interests include evolutionary computation, multiobjective optimization, and machine learning. Dr. Li was a recipient of the 2010 IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION Outstanding Paper Award as one of the inventors for MOEA/D.

Xinzhou Cheng is the professor level senior engineer, the head of network intelligent operation R&D center at China Unicom Research Institute. From 2004 to 2013, he worked at Beijing Telecom Planning & Designing Institute, where he is the chief engineer in wireless department and the head of the network optimization center. From 2013 to 2020, he worked at the China Unicom Network Technology Research Institute, where he is the senior specialist, professor level senior engineer, the head of big data R&D Center, the head of network operation and big data R&D center. He received M.S. degree from Beijing University of Posts and Telecommunications in 2004. He has published more than 150 technical papers. His research interests include telecom big data, network planning & optimization, network intelligent operation.

Lexi Xu (SM) received PhD degree from Queen Mary University of London, London, United Kingdom in 2013. He is now a senior engineer at Research Institute, China United Network Communications Corporation (China Unicom). He is also a China Unicom delegate in ITU, ETSI, 3GPP, CCSA. His research interests include big data, self-organizing networks, satellite system, radio resource management in wireless system, etc.

Li Feng received his PhD degree from Xi'an Jiaotong University under the supervision of Prof. Xiaohong Guan (Academian of CAS, IEEE Fellow). He is a Research Professor and PhD supervisor with the School of Computing and Artificial Intelligence, Southwest Jiaotong University, Chengdu. His research interests include artificial intelligence, cyber security and its applications.



Qian Wan received the B.Eng. degree in computer science from the Wuhan University, Wuhan, China, and the M.Des. degree in interaction design from the China University of Geosciences, Wuhan, China. He is pursuing the Ph.D. degree in computer science at Southwest Jiaotong University, Chengdu, China. His research interests include data model and mining, virtual reality, and augmented reality.