KiCi: A Knowledge Importance Based Class Incremental Learning Method for Wearable Activity Recognition

Shuai Guo
The Beijing Key Laboratory of Mobile Computing and Pervasive Device
Institute of Computing Technology, Chinese Academy of Sciences
University of Chinese Academy of Sciences
Beijing, China
guoshuai20g@ict.ac.cn

Yang Gu*
The Beijing Key Laboratory of Mobile Computing and Pervasive Device
Institute of Computing Technology, Chinese Academy of Sciences
University of Chinese Academy of Sciences
Peng Cheng Laboratory
Beijing, China
guyang@ict.ac.cn

Shijie Wen
The Beijing Key Laboratory of Mobile Computing and Pervasive Device
Institute of Computing Technology, Chinese Academy of Sciences
University of Chinese Academy of Sciences
Beijing, China
wenshijie20s@ict.ac.cn

Yuan Ma
The Beijing Key Laboratory of Mobile Computing and Pervasive Device
Institute of Computing Technology, Chinese Academy of Sciences
University of Chinese Academy of Sciences
Beijing, China
mayuan20z@ict.ac.cn

Yiqiang Chen
The Beijing Key Laboratory of Mobile Computing and Pervasive Device
Institute of Computing Technology, Chinese Academy of Sciences
University of Chinese Academy of Sciences
Peng Cheng Laboratory
Beijing, China
yqchen@ict.ac.cn

Jiwei Wang
Shandong Artificial Intelligence Institute, Qilu University of Technology (Shandong Academy of Sciences)
Institute of Computing Technology, Chinese Academy of Sciences
Jinan, China
wangjw@sdas.org

Chunyu Hu
School of Computer Science and Technology, Qilu University of Technology (Shandong Academy of Sciences)
Jinan, China
hcy@qlu.edu.cn

ABSTRACT

Wearable-based human activity recognition (HAR) is commonly employed in real-world scenarios such as health monitoring, auxiliary diagnosis, etc. As implementing activity recognition is a daunting challenge in an open dynamic environment, incremental learning has become a common method to adapt to variable behavior patterns of users and create dynamic modeling in activity recognition. However, catastrophic forgetting is a significant challenge with incremental learning. This is contrary to our expectations of identifying new activity classes while remembering existing ones. To address this problem, we propose a knowledge importance-based class incremental learning method called KiCi and construct an incremental learning model based on the framework of self-iterative knowledge distillation for dynamic activity recognition. To eliminate the prediction bias of the teacher model on the old knowledge, we utilize the trained weights of previous incremental steps generated by the teacher model as the prior knowledge to obtain knowledge importance. Then use it to make the student model have a reasonable trade-off between old and new knowledge and mitigate catastrophic forgetting by avoiding negative transfer. We conduct extensive experiments on four public HAR datasets and our method consistently outperforms the existing state-of-the-art methods by a large margin.

*Contact Author

CIKM '22, October 17–21, 2022, Atlanta, GA, USA

ACM ISBN 978-1-4503-9236-5/22/10.
https://doi.org/10.1145/3511808.3557371

CCS CONCEPTS

• **Human-centered computing** → **Ubiquitous and mobile computing**; • **Applied computing** → *Health informatics*; • **Computing methodologies** → *Knowledge representation and reasoning*.

KEYWORDS

activity recognition, class incremental learning, knowledge distillation

ACM Reference Format:
Shuai Guo, Yang Gu, Shijie Wen, Yuan Ma, Yiqiang Chen, Jiwei Wang, and Chunyu Hu. 2022. KiCi: A Knowledge Importance Based Class Incremental Learning Method for Wearable Activity Recognition. In *Proceedings of the 31st ACM Int'l Conference on Information and Knowledge Management (CIKM '22), Oct. 17–21, 2022, Atlanta, GA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3511808.3557371

1 INTRODUCTION

HAR techniques facilitate the development of enormous pervasive applications such as health monitoring, auxiliary diagnosis, etc. With the development of embedded hardware in recent years, various motion sensors (accelerometers, gyroscopes, magnetometers, infrared sensors, etc.) are embedded into unobtrusive wearable devices (e.g., smartphones, smartwatches, wristbands, armbands, and shoes). These devices supply a flow of streaming sensor readings correlated with the activity categories. Consequently, recognizing activities using wearable devices has become increasingly popular and widely utilized in recent years [11, 13].

Conventional activity recognition utilizes fixed activity recognition models, which are trained with offline collected data, to classify user activities online. However, user activity pattern changes always lead to the failure of fixed activity recognition models. With a fixed activity recognition model, new activity patterns exhibited by users will be misclassified as previously known activities. It is challenging to accurately classify both newly emerging classes and known classes. To adapt to changes in user activity patterns, a novel model needs to be trained with both the new class data and the known class data. This process requires a large memory space to store the data that are used for initial training. And the new class data is only available when the user activity pattern increases. Moreover, retraining a novel model from scratch requires more time. Incremental learning is a promising method for handling changes in activity patterns in less time and with fewer storage requirements.

Incremental learning allows a model to be continually updated with new data, instead of being trained once on a whole dataset. In order to achieve this goal, a common method is to fine-tune the old model on new data by setting the number of output nodes to be that of current classes (including old and new classes) in user activities. However, this naive method suffers from a serious problem known as catastrophic forgetting [23]. The knowledge distillation method has been widely used in this field to try to preserve the knowledge in the original model [20].

Knowledge distillation (KD) [15] uses the softened logits generated by a teacher model as the targets to train a student model. The existing methods enforce the student model to fully mimic the behavior of the teacher model to classify user activities. Specifically, the distilled student model aims to imitate the teacher model's logits on the training exemplars. This is a proxy to the student model's true goal, which is mimicking these logits on test exemplars, to attain similar generalization performance as the teacher model. However, fully imitating the outputs of the teacher model may not be optimal, as the teacher model may confidently mispredict some classes [12]. Thus the bias contained in the teacher model is also transferred to the student model and the bias may be amplified by the student. This is a potential limitation of knowledge distillation, and transferring this kind of bias contained in the pretrained teacher model to the student model will hurt the student model which causes catastrophic forgetting.

In this paper, we propose a novel incremental learning method, namely KiCi, a knowledge importance-based class incremental learning method, to adapt to variable behavior patterns of users and create dynamic modeling in activity recognition. First of all, we utilize the softened logits generated by the teacher model as the prior knowledge to achieve knowledge extraction and divide it into weights subsets. Then, knowledge importance is evaluated through weights subsets to measure the correctness of the teacher model and reduce the impact of the inherently inaccurate teacher model on the student model. Finally, the student model was transformed into a knowledge distillation loss-weighted model using knowledge importance to avoid negative transfer and mitigate catastrophic forgetting.

The contributions of our paper are summarized as follows:

- Based on knowledge importance class incremental learning method (KiCi) is proposed to adapt to variable behavior patterns of users and create dynamic modeling in activity recognition.
- To eliminate the bias the teacher model transfer to the student model, we make full use of the information contained in the output logits generated by the teacher model to evaluate knowledge importance.
- Extensive experiments are conducted based on four public HAR datasets, which verify the effectiveness and robustness.

2 RELATED WORK

2.1 Incremental Learning

In incremental learning, a learner addresses a sequence of these tasks without storing the complete datasets of these tasks. The main challenge in incremental learning is to avoid catastrophic forgetting interference, that is, the learner forgetting previously acquired knowledge when learning a new incremental step. The state-of-the-art methods in incremental learning can be categorized into three classes.

First, parameter-based incremental learning methods [2] add regularization term in the loss function, which prevents the weights from changing drastically when moving from one task to the next task. The approaches of this strategy such as EWC [19], SI [34], and MAS [2] manage to constrain the important weights of the old model when facing new data. These methods expect small changes in the important parameters. While efficient in terms of memory and computation, these methods suffer from brittleness due to feature drift for a large number of tasks [29].

Second, Rehearsal based incremental learning methods [7, 10, 31, 32, 35, 36] store a part of training data and re-train on this stored data while training on new steps. More recently, several works [9] have shown that directly optimizing the loss of rehearsal-based incremental learning methods is cheaper than parameter-based approaches and improves the prediction performance. Other methods construct a generative model, e.g., GANs [14], to generate

exemplars for rehearsal instead of storing old data directly [22]. However, in these methods, an additional generative model needs to be trained simultaneously. Therefore, they rely heavily on the quality of the generated model.

Third, knowledge distillation methods [1, 30, 32, 35, 36] attempt to preserve the performance of the old incremental steps was presented in LwF [20] using distillation loss in combination with the standard cross-entropy loss. iCaRL [25] and EE2L [8] compute the distillation loss on the network outputs. UCIR [16] uses normalized feature vectors to apply the distillation loss instead of the prediction of the network. BiC [31] adds a bias correction layer to correct the model's outputs, where the layer is trained on a separate validation set. WA [35] corrects the biased weights by aligning the norms of the weight vectors for new classes to those for old classes. DER2 [32] keep the previously learned representation fixed and augment it with novel features parameterized by a new feature extractor, and re-use the structure via the final classifier to mitigate forgetting. Coil [36] learns to relate across incremental tasks with the class-wise semantic relationship by using optimal transport. Both prospective transport and Retrospective transport are included.

2.2 Incremental Learning Method for Wearable-based HAR

Wearable-based HAR has been a hot research topic for decades, and activity recognition methods based on incremental learning methods have been widely studied. [17]proposed a class incremental random forest (CIRF) to enable existing models to identify new activities. A separating-axis-theorem-based splitting strategy was adopted to insert new nodes without reconstructing a subtree. The Gini index and information gain were employed to split the leaves of the decision tree in the random forest. [21] proposed a method OALELM to decrease the extent of model adjustment but did not dig deep enough to find the correlation of unstable performance with the adaptation process. [6] proposes a hybrid semi-supervised and knowledge-based system for real-time activity recognition incremental learning method CAVIAR, to apply semantic reasoning on context-data to refine the predictions of an incremental classifier. [4] put-together deep learning techniques with incremental learning to obtain personalized models that perform better concerning the user-independent model and personalized model obtained using traditional machine learning techniques. In addition, [28] introduces an ensemble-based personalized human activity recognition method that can not only learn from streaming data but also adapt to different contexts and changes in context. [18] proposes a semi-supervised class incremental learning method Di-CIRF, which can detect newly emerging classes and update a previously established activity recognition model through streaming data.

KiCi differs from previous works in a key aspect: more than simply combining different objective terms to balance old and new classes in user activities, we eliminate the teacher model's prediction bias of old classes in knowledge distillation simply and effectively. Without any additional model parameters, hyperparameters, or a reserved validation set, our method achieves better performance than previous methods.

3 METHOD

3.1 Definition of Incremental Learning

In incremental learning, we are given a set of data X with labels Y belonging to classes in C. This defines the dataset $D = \{(\mathbf{x}, y)|\mathbf{x} \in X, y \in Y\}$. In class incremental learning, we want to expand an existing model to classify new classes. Given T incremental steps and split C into T subsets $C^1, C^2, \cdots, C^T$, where $C{=}C^1 \cup C^2 \cup \cdots \cup C^T$ and $C^i \cap C^j = \emptyset$ for $i \neq j$. By defining incremental step t as introducing new classes C^t using dataset $D^t = \{(\mathbf{x}, y)|y \in C^t\}$. It can be denoted that $X^t = \{\mathbf{x}|(\mathbf{x}, y) \in D^t\}$ and $Y^t = \{y|(\mathbf{x}, y) \in D^t\}$ as the training data and labels used at incremental step t. The goal is to train a classifier that accurately classifies examples belonging to the new set of classes C^t, while still being able to correctly classify examples belonging to classes C^i, where $i < t$.

3.2 Self-iterative Knowledge Distillation Framework

In this subsection, we introduce an activity recognition framework in class incremental learning, which utilizes both the rehearsal strategy and the knowledge distillation strategy. As shown in Figure 1, for an incremental step with n old classes and m new classes, we learn a new model to perform classification on $n + m$ classes, by using the knowledge distillation from an old model that classifies the old n classes. The new activity recognition model is learned by using a distilling loss and a classification loss.

Assume there are B incremental steps of train data $\{D^1, \cdots, D^B\}$, with $D^b = \{(\mathbf{x}_1^b, y_1^b), \cdots, (\mathbf{x}_{n_b}^b, y_{n_b}^b)\}$ for the b^{th} incremental step, where $\mathbf{x}_i^b$ and y_i^b represent the input data and the label respectively, n_b is the number of samples in the set D^b. In the b^{th} step of class incremental learning, the goal is to learn knowledge from new data D^b, while retaining the previous experiences learned from old data $\{D^1, \cdots, D^{b-1}\}$. For each step, the trained model is evaluated in all seen classes.

For the b^{th} incremental step, initialize the model with the parameters learned in the previous step and add new output nodes (weights in the Fully Connected layer are initialized randomly). Then, it attempts to learn new classes and meanwhile preserve the original capabilities with the new data D^b and a few rehearsal data D_{old}^b. It is assumed that the new data D^b comes from C^b new classes, and the rehearsal data D_{old}^b comes from C_{old}^b old classes, where $C_{old}^b = \sum_{k=1}^{b-1} C^k$. Let the output logits of the old and current model as $\hat{\mathbf{o}}(\mathbf{x}) = \left(\hat{o}_1(\mathbf{x}), \cdots, \hat{o}_{C_{old}^b}(\mathbf{x})\right)^T$ and $\mathbf{o}(\mathbf{x}) = \left(o_1(\mathbf{x}), \cdots, o_{C_{old}^b}(\mathbf{x}), o_{C_{old}^b+1}(\mathbf{x}), \cdots, o_{C_{old}^b+C^b}(\mathbf{x})\right)^T$ respectively. The distilling loss is given by:

$$\mathcal{L}_{KD}(\mathbf{x}) = \sum_{c=1}^{C_{old}^b} -\hat{q}_c(\mathbf{x}) \log\left(q_c(\mathbf{x})\right), \tag{1}$$

where $\hat{q}_c(\mathbf{x}) = \frac{e^{\hat{o}_c(\mathbf{x})/T}}{\sum_{j=1}^{C_{old}^b} e^{\hat{o}_j(\mathbf{x})/T}}$, $q_c(\mathbf{x}) = \frac{e^{o_c(\mathbf{x})/T}}{\sum_{j=1}^{C_{old}^b} e^{o_j(\mathbf{x})/T}}$; T is the temperature scalar. The distilling loss is computed for all samples from the new classes and the rehearsal data from the old classes.

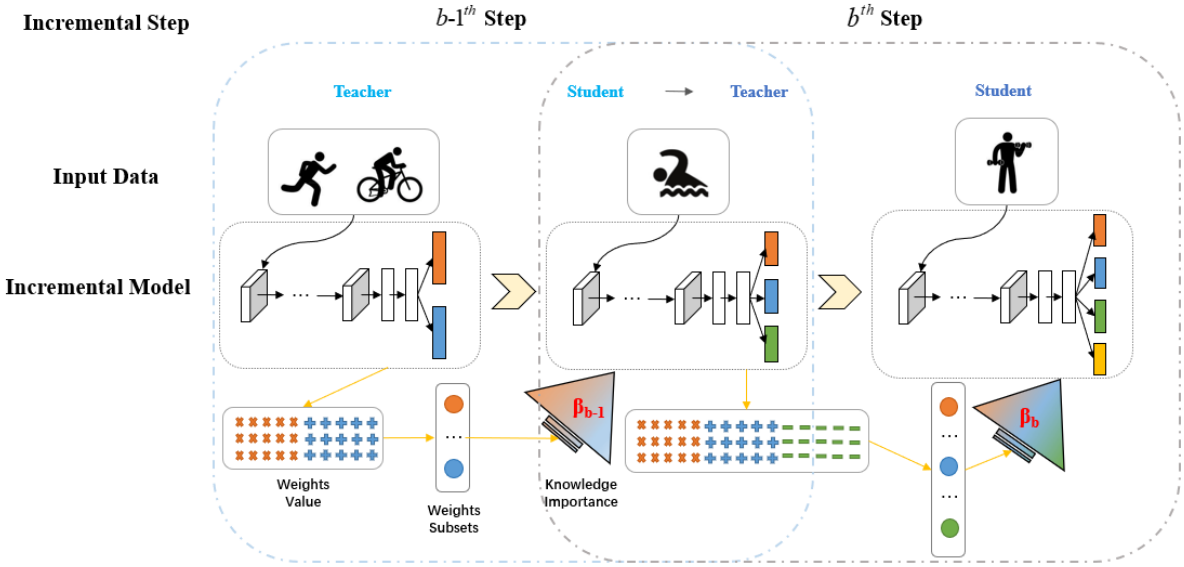


Figure 1: Overview of KiCi method based on the framework of self-iterative knowledge distillation class incremental learning model. In the $b-1^{th}$ incremental step, we utilize the output logits generated by the teacher model as the prior knowledge weights value. By using weights subsets to evaluate knowledge importance, make the student model has a reasonable trade-off between old (running and riding) and new (swimming) knowledge. Finally, the student model in the $b-1^{th}$ incremental step will be used as the teacher model in the b^{th} incremental step. (Best viewed in color)

We use the softmax cross-entropy as the classification loss, which is given by:

$$\mathcal{L}_{CE}(\mathbf{x}, y) = \sum_{c=1}^{C^b+C^b_{old}} -\delta_{c=y} \log\left(p_c(\mathbf{x})\right), \tag{2}$$

where $\delta_{c=y}$ is the indicator function and $p_c(\mathbf{x})$ is the output probability for the c^{th} class in $n+m$ old and new classes.

The overall loss combines the cross-entropy loss $\mathcal{L}_{CE}$ and the knowledge distillation loss $\mathcal{L}_{KD}$ as follows:

$$\mathcal{L}(\mathbf{x}, y) = (1-\lambda)\mathcal{L}_{CE}(\mathbf{x}, y) + \lambda\mathcal{L}_{KD}(\mathbf{x}), \tag{3}$$

where λ is a hyper-parameter governing the balance between the two losses. The scalar λ is used with the value of $\frac{C^b_{old}}{C^b+C^b_{old}}$. λ is 0 for the first step since all classes are new. For the extreme case where $n \gg m$, λ is nearly 1, indicating the importance to maintain the old classes.

3.3 Knowledge Importance based Class Incremental Learning

Weights Subsets. Knowledge distillation (KD) uses the softened logits generated by a teacher model as the targets to train a student model. And the existing methods which use KD enforce the student model to fully mimic the behavior of the teacher model like in Eq.(3). However, fully imitating the outputs of the teacher model may not be optimal, as the teacher model may be incorrectly predicted for some samples and generate prediction bias. Meanwhile, the bias may be amplified by the student. To solve this problem, we treat the predicted value of the teacher model for each training sample as a weight value. In the b^{th} step of class incremental learning, we has the train data $D^b = \{\left(\mathbf{x}^b_1, y^b_1\right), \cdots (\mathbf{x}^b_{n_b}, y^b_{n_b})\}$, and C^b is all classes number. For any $\mathbf{x}_i$ and the corresponding y_i belongs to $C^b_{y_i}$, $i \in [1, n_b]$,the weights value is formulated as follows:

$$\varphi^b_{i,C^b_{y_i}} = \left[p_{C^b_{y_i}}(\mathbf{x}_i) - \frac{1}{C^b - 1} \sum\nolimits_{C^b_{y_i}, \neq C^b_{y_i}} p_{C^b_{y_i},}(\mathbf{x}_i) \right]. \tag{4}$$

The weights values of all train data $\{D^1, \cdots D^B\}$ as the weights set and by using the teacher model's predicted values for samples of the same class in the training data as a weights subset.

Knowledge Importance. To measure the correctness of the teacher model and reduce the impact of inherently inaccurate teacher models on the student models, we use weights subsets to evaluate knowledge importance. In the b^{th} step of class incremental learning, we take the weights subsets of all old data $\{D^1, \cdots D^{b-1}\}$ as prior knowledge. So the value of the weights in old data can be formulated as follow:

$$\varphi^{b-1}_{i,C^{b,y_i}_{old}} = \left[p_{C^{b,y_i}_{old}}(\mathbf{x}_i) - \frac{1}{C^b_{old}-1} \sum\nolimits_{C^{b,y_{i'}}_{old} \neq C^{b,y_i}_{old}} p_{C^{b,y_{i'}}_{old}}(\mathbf{x}_i) \right], \tag{5}$$

where the y_i corresponding to $\mathbf{x}_i$ belongs to C^{b,y_i}_{old}. The knowledge importance to the teacher model in the b^{th} step can be calculated as :

$$\beta = \frac{\sum_{C^{b,y_i}_{old}=1}^{C^b_{old}} \max[0, \mathbb{E}(\varphi^{b-1}_{i,C^{b,y_i}_{old}})]}{C^b_{old}}. \tag{6}$$

Table 1: Description of four public activity datasets

Dataset	Samples	Features	Classes	Subjects
OPPORTUNITY	5,022	459	16	4
PAMAP2	7,352	243	12	9
HARUSDS	10,299	561	6	30
DSADS	9,120	405	19	8

In words, Eq.(5) places greater faith in the teacher model for those classes which it predicts correctly with confidence. When $\mathbb{E}(\varphi^{b-1}_{i,C^{b,y_i}_{old}}) < 0$, which can occur on classes that are rare in the training data, we set $\mathbb{E}(\varphi^{b-1}_{i,C^{b,y_i}_{old}}) = 0$, and completely ignore the teacher predictions.

By using the knowledge importance to indicate the credibility of the teacher model. The student model will make a reasonable trade-off between old and new data to eliminate the teacher model's prediction bias of old knowledge, the final loss we aim to optimize becomes:

$$L(\mathbf{x}, y) = (1-\beta)L_{CE}(\mathbf{x}, y) + \beta L_{KD}(\mathbf{x}). \tag{7}$$

In the standard distillation configuration Eq.(3), only have a single weight λ that is common for all classes. But Eq.(7) has high confidence that the teacher model can accurately predict the classes, thus eliminating the teacher model's bias in predicting old knowledge and mitigating catastrophic forgetting.

4 EXPERIMENTS

We compared our KiCi method with classical (LwF [20] and CIRF [17]) and current state-of-the-art (BiC [31], HAL [10], DER [7], WA [35], DER2 [32], and Coil [36]) incremental learning methods. We evaluate the methods on OPPORTUNITY [27], PAMAP2 [26], HARUSDS [5], and DSADS [3], which are widely used in the study of class incremental learning for wearable-based human activity recognition.

4.1 Datasets

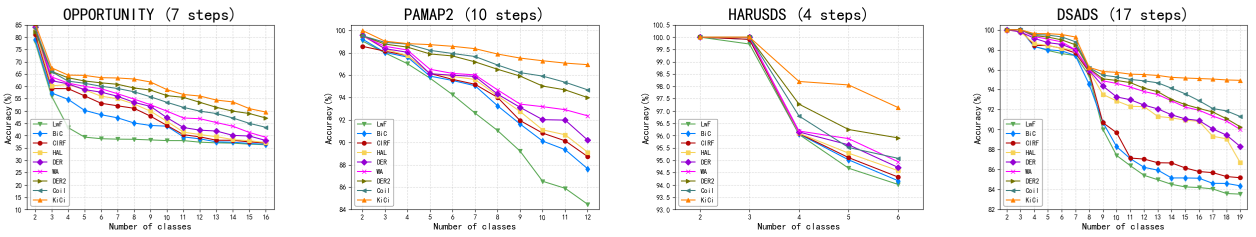
To evaluate the performance of the proposed method, comparative experiments are designed using three real-world human activity datasets which are detailed in Table 1:

- **OPPORTUNITY [27]:** The dataset is comprised of typical daily activities performed by 4 subjects. Six different runs (including five runs of ADLs and a drill run) are recorded for each subject. Three types of sensors are employed to acquire data in a smart house. In our experiments, we pay more attention to the activity recognition problem. Thus, we use data acquired from on-body sensors. As there are plenty of missing data from Bluetooth sensors, we only use five XSense inertial measurement units mounted on a motion jacket and two commercial InertiaCube3 inertial sensors placed on feet. Each unit includes accelerometer, gyroscope, and magnetometer sensors. In the OPPORTUNITY dataset, activities are divided into 4 levels: modes of locomotion, low-level actions, middle-level gestures, and high-level activities. In this paper, we conduct experiments on middle-level gestures to evaluate the effectiveness of our method.
- **PAMAP2 [26]:** The dataset was collected from 9 participants, which includes 12 activities ("walking", "lying down", "standing", etc) and 6 optional activities ("watching TV", "folding laundry", etc). Since those optional activities are rarely performed by the participants, we excluded them from the analysis. This dataset recorded signals from three inertial measurement units (IMUs) located on the hand, chest, and ankle and a heart rate monitor located on the chest. As there are many missing values in the data collected by the heart rate monitor, we only use the data from IMU in the experiments. Each IMU consists of two accelerometers, one magnetometer, and one gyroscope. Thus, readings collected from 12 sensors are used in the experiments.
- **HARUSDS [5]:** The Human Activity Recognition Using Smart-phones Data Set (HARUSDS) was collected with 30 subjects. A smartphone was fixed on the subject's waist to acquire readings of the 3-axial accelerometer and 3-axial gyroscope for six activities (walking, walking upstairs, walking downstairs, sitting, standing, lying). This dataset has been preprocessed and divided into training and testing sets by the authors. In our experiments, we use the provided feature directly. However, the partition of the dataset is not consistent with our experimental goals. Thus, we merge the training and testing sets together and redistribute the dataset on demand.
- **DSADS [3]:** The Daily and Sports Activities Data Set (DSADS) from UCI Machine Learning Repository consists of 19 daily and sports activities performed by 8 subjects (4 males and 4 females). Three types of sensors (3-axial accelerometer, 3-axial gyroscope, and 3-axial magnetometer) were placed on five different body parts including the torso, right arm, left arm, right leg, and left leg. Thus, there are 45 sensor readings in each frame. All subjects performed the activities in their styles with no restriction.

As features are provided in HARUSDS, we extract features only for the other datasets. In our experiments, we first synthesize the reading of the 3 axes from one sensor into a synthetic value using the formula $a = \sqrt{(x^2 + y^2 + z^2)}$, which eliminates the impact caused by the difference in orientation among the sensors. Then, we adopt the sliding window technique to segment the sensor readings. In our experiments, we set the window size to 5 seconds, 2 seconds, and 5.12 seconds for DSADS, OPPORTUNITY, and PAMAP2 respectively with 50% overlap between consecutive windows. For each window, we extract 27 features, from both the time domain and frequency domain for a single sensor. After preprocessing the four HAR datasets, we conducted full experiments on these HAR datasets. We also perform experiments to analyze the effect of different components of our approach.

4.2 Implementation Details

Our method is implemented with Pytorch. For a fair comparison, all compared methods adopt an 18-layer ResNet and train 256 epochs. The learning rate starts from 0.1 and reduces to 1/10 of the previous learning rate after 100, 150, and 200 epochs. The models are optimized by SGD with batch size 256, and the temperature scalar is set to 2. We select rehearsal exemplars based on herding selection which is also the same as the previous work. [25]. The total number of exemplars for the old classes is fixed. The magnitudes of both



(a) 1 class / step (b) 1 class / step (c) 1 class / step (d) 1 class / step

Figure 2: The performance of single-class incremental methods on four HAR datasets. The average and standard deviations are obtained over five runs. (Best viewed in color)

embeddings and biases for the new classes are much greater than those for the old classes due to the class imbalance. When adopting cosine normalization in the last layer, the ReLU in the penultimate layer is removed to allow the features to take both positive and negative values. Furthermore, we run experiments on three different class orders and followed metrics for accuracy, standard deviations, and forgetting from recent works [33] in the results. Although this paper is based on sensor data from wearable devices as input data for the model, this method can also produce good results when images, etc. are used as input data. We also provide the experimental results on CIFAR-100 based on ResNet-18 following RPSNet [24] in the appendix, which proves the superiority of our method again. We note that most previous works use a modified 32-layer ResNet, which has fewer channels and residual blocks compared to standard ResNet-32. We argue that such a small network is not suitable because it cannot achieve competitive results on CIFAR100 compared with standard 18-layer ResNet and may underestimate the performance of methods.

4.3 Experiments on Single-class Increments

To verify the effectiveness of KiCi on single-class incremental methods, we compare our KiCi method with the state-of-the-art methods on four HAR datasets (OPPORTUNITY, PAMAP2, HARUSDS, and DSADS). We store 200 exemplars from old classes as the same as the previous work [31]. We select rehearsal exemplars based on herding selection [25] which is also the same as the previous work. More classes have been seen, fewer data can be retained per class. As a result, the problem of catastrophic forgetting becomes more serious. The incremental learning results on four HAR datasets are shown in Figure 2. We can infer from the results that our proposed KiCi outperforms the current SOTA methods in terms of the final incremental accuracy and the average incremental accuracy. Although KiCi has a small gain for the first few couples of incremental steps compared with other methods. However, the gain of KiCi increases as more incremental steps arrive. This suggests that as the number of old classes and old knowledge increases, the bias generated by the teacher model also gradually grows. Regarding the final incremental classifier on all classes, our KiCi method outperforms the SOTA method on four HAR datasets by 2.25%, 2.40%, 1.23%, and 3.62% respectively. On average over all the incremental steps except the first steps (because the first step is not relevant for class incremental learning), KiCi outperforms the SOTA method on four HAR datasets by 0.92%, 2.47%, 0.79%, and 1.01% respectively. These results demonstrate our Kici method is more effectively reducing the bias from the teacher model's prediction and mitigating the problem of catastrophic forgetting during the incremental process to improve the classification accuracy.

Table 2: Incremental learning results (accuracy %) on OPPORTUNITY dataset with an increment of 2 classes. The average and standard deviations are obtained over five runs.

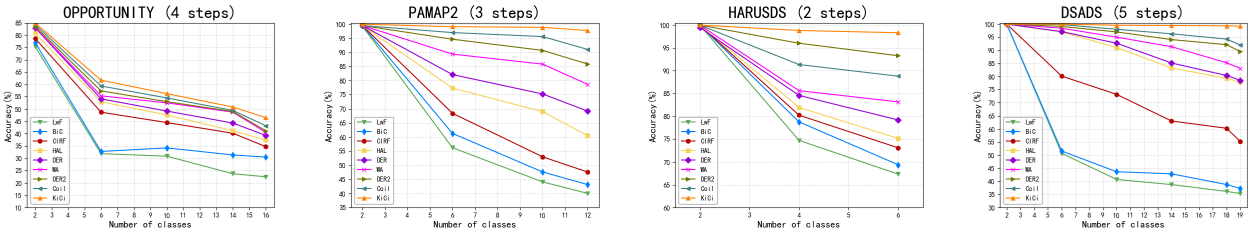
Classes	LwF	BiC	CIRF	HAL	DER	WA	DER2	Coil	Ours
2	77.48	78.53	80.46	81.46	82.15	82.61	82.17	83.69	**84.84**
4	55.77	56.27	57.36	58.52	60.78	62.14	64.08	63.98	**65.17**
6	42.14	48.32	50.46	51.62	54.38	59.03	60.50	58.14	**61.27**
8	39.50	44.35	47.60	47.72	51.90	54.69	56.20	53.27	**60.02**
10	38.63	39.72	41.13	43.28	48.36	53.12	52.45	50.06	**56.16**
12	37.57	38.23	39.82	40.11	44.27	48.37	48.86	45.17	**50.55**
14	35.26	36.13	36.21	38.22	41.64	43.15	44.37	42.82	**50.30**
16	32.46	33.15	34.23	35.84	38.79	40.18	42.50	40.79	**46.30**
AVG	44.85	46.84	48.41	49.60	52.78	55.41	56.30	54.74	**59.33**
SD	13.97	13.82	14.09	13.92	12.95	12.42	11.99	13.17	**11.33**

Table 3: Incremental learning results (accuracy %) on PAMAP2 dataset with an increment of 2 classes. The average and standard deviations are obtained over five runs.

Classes	LwF	BiC	CIRF	HAL	DER	WA	DER2	Coil	Ours
2	99.06	99.53	99.06	99.53	99.53	99.53	99.53	99.53	**99.53**
4	81.84	85.78	89.32	92.36	94.58	95.51	96.68	97.03	**98.88**
6	67.01	74.23	80.72	85.48	92.17	94.29	93.01	95.18	**97.64**
8	62.92	63.17	73.29	80.62	89.83	92.34	91.30	92.62	**94.96**
10	61.18	62.66	68.59	74.10	83.64	89.87	89.72	91.04	**94.16**
12	59.34	61.44	63.24	69.32	76.23	84.48	85.89	88.67	**93.57**
AVG	71.89	75.97	79.04	83.57	89.33	92.64	92.69	94.01	**96.46**
SD	14.22	13.03	12.26	10.30	7.57	4.70	4.47	3.66	**2.33**

4.4 Experiments on Multi-class Increments

Previous incremental learning methods for wearable activity recognition excel in achieving single-class increments, but when performing multi-class increments performance is not good [17]. So in this subsection, we verify the effectiveness of KiCi on multi-class incremental methods. By conducting the experiments on the four



(a) 4 classes / step (b) 4 classes / step (c) 2 classes / step (d) 4 classes / step

Figure 3: The performance of multi-class incremental methods on four HAR datasets. The average and standard deviations are obtained over five runs. (Best viewed in color))

Table 4: Class incremental learning performance (accuracy %) on OPPORTUNITY. We initialize the model with 6 classes, followed by 5 incremental steps using 2 classes per step. The gains based on Variation1 are also reported in parentheses. 'Full' is obtained with all training data for all classes. The average results over all the incremental steps except the first step are obtained over five runs and also reported here.

#classes	6	8		10		12		14		16		Average	
Variation1 (CE)	63.73	54.21		47.98		44.53		41.40		35.42		44.71	
Variation2(CE+KD)	63.75	56.09	(+1.88)	50.60	(+2.62)	46.74	(+2.21)	43.21	(+1.81)	38.35	(+2.93)	47.01	(+2.30)
Ours(CE+KD+KiCi)	63.66	**61.18**	(+6.97)	**58.61**	(+10.63)	**57.22**	(+12.69)	**50.99**	(+9.59)	**46.84**	(+11.42)	**54.97**	(+10.26)
Full	-									60.42		-	

Table 5: Incremental learning results (accuracy %) on HARUSDS dataset with an increment of 2 classes. The average and standard deviations are obtained over five runs.

Classes	LwF	BiC	CIRF	HAL	DER	WA	DER2	Coil	Ours
2	100.00	100.00	100.00	10.000	99.48	100.00	100	100	**100.00**
4	74.73	78.79	80.25	81.96	84.59	85.63	96.02	91.35	**98.82**
6	67.33	69.38	73.10	75.16	79.20	83.17	93.28	88.80	**98.32**
AVG	80.69	82.72	84.45	85.71	87.76	89.60	96.43	93.38	**99.05**
SD	13.99	12.81	11.38	10.48	8.58	7.42	2.75	4.79	**0.70**

Table 6: Incremental learning results (accuracy %) on DSADS dataset with an increment of 2 classes. The average and standard deviations are obtained over five runs.

Classes	LwF	BiC	CIRF	HAL	DER	WA	DER2	Coil	Ours
2	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	**100.00**
4	75.26	76.58	85.37	89.24	93.89	98.17	99.17	99.86	**100.00**
6	65.09	71.40	83.37	86.32	90.28	95.20	97.89	98.73	**99.93**
8	58.42	65.00	81.48	84.17	88.17	92.25	95.59	97.28	**99.65**
10	57.67	64.11	74.05	82.08	85.96	89.09	93.82	97.00	**99.60**
12	55.44	61.44	70.37	80.37	82.03	84.93	92.46	96.92	**99.55**
14	56.97	59.65	65.18	77.62	78.33	80.39	91.02	96.59	**99.54**
16	54.74	58.27	62.13	70.08	73.06	74.64	90.73	94.83	**99.47**
18	54.54	57.17	60.87	64.64	65.17	68.91	88.14	94.28	**99.30**
19	51.88	57.12	59.33	61.18	63.85	65.18	87.05	93.76	**99.16**
AVG	63.00	67.07	74.22	79.57	82.07	84.88	93.59	96.93	**99.62**
SD	13.89	12.52	12.52	11.13	11.37	11.63	4.28	2.07	**0.27**

HAR datasets. We start from a model trained on 2 randomly selected classes. Then, with each incremental step, 2 and 4 random classes are added respectively. We store 200 exemplars from old classes and select rehearsal exemplars based on herding selection the same as the previous work. The class incremental learning results are shown in Tables 2 through 6 and Figure 3. We report the performance of every incremental step, as well as the average and standard deviation results for all incremental steps except the first step. We can see that our KiCi method consistently outperforms other methods by a sizable margin at different incremental splits, either in the trend of classification accuracy curve, average incremental accuracy, or standard deviation results. It is observed that our method consistently surpasses other methods at every step for different splits. Moreover, the gap between our method and other methods increases with the continuous adding of novel classes. Compare to the SOTA method with an incremental of 2 classes, the overall performance on the total classes at the end of incremental learning is improved by 4.90%, 3.80%, 5.04%, and 5.40% respectively. And on average over all the incremental steps except the first steps, KiCi outperforms the SOTA method on four HAR datasets by 2.45%, 2.84%, 2.62%, and 2.69% respectively. As the number of classes added in each incremental step becomes larger in the multi-class incremental case, the bias in the teacher model's prediction of the old knowledge increases, and therefore the teacher model becomes less reliable. In this scenario having the student model exactly replicate the output of the teacher model for the old knowledge would greatly reduce the classification accuracy. This shows the importance of our KiCi approach in solving the multi-class incremental situation.

4.5 Comparison on Forgetting Indicators

We followed metrics for the forgetting from recent works [33] that indicate the decrease in performance for each of the incremental

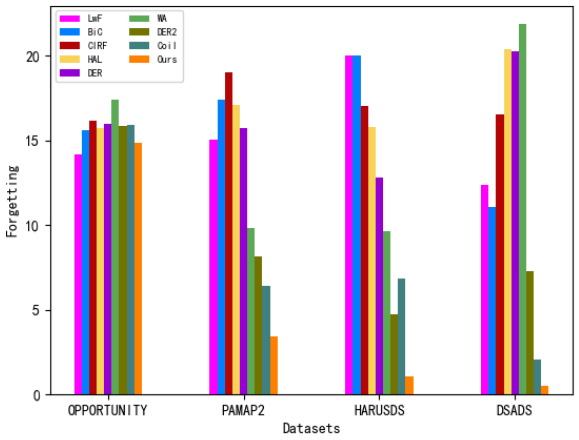
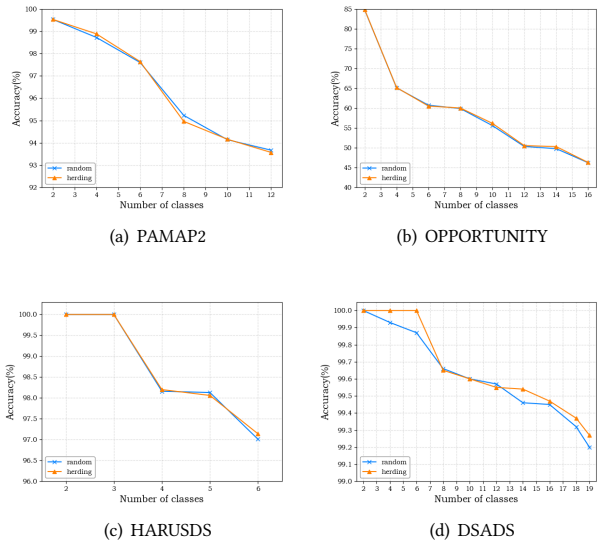


Figure 4: The forgetting value on four datasets with an increment of 2 classes. As shown in the figure, our method is lower than the state-of-the-art methods. (Best viewed in color)



(a) PAMAP2 (b) OPPORTUNITY

(c) HARUSDS (d) DSADS

Figure 5: Incremental learning results on four datasets for different exemplar selection strategies. (Best viewed in color)

steps between their peak accuracy and their accuracy after the continual learning experience has finished. We start from a model trained on 2 randomly selected classes. Then, with each incremental step, 2 classes are added. As shown in Figure 4, our proposed KiCi is the method for achieving minimal Forgetting. This indicates that our method forgets the old knowledge more slowly in the process of increments and is effective to handle catastrophic forgetting in class incremental learning.

4.6 Effect of KiCi

We now analyze the components of our KiCi method and demonstrate their impact. To analyze the effect of KiCi, we perform experiments on the OPPORTUNITY dataset. We start from a model trained on 6 randomly selected classes. Then, with each incremental step, 2 random classes are added. We compare our method with two variations in the following: Variation1, training with the cross-entropy loss alone; variation2, training with both the distilling loss and cross-entropy loss; Ours, training with the combined loss and our method KiCi.

The incremental learning results are shown in Table 4. With the help of the knowledge distillation, variation2 is slightly better than variation1 since it retains the classification capability of the old classes. However, both variation1 and variation2 have low accuracy on the final step to classify all 16 classes. This is mainly because of the prediction bias of the teacher model on the old knowledge. When using the knowledge importance-based class incremental learning method, KiCi improves the accuracy of all incremental steps. The classification accuracy on the final step (16 classes) is boosted from 38.35% to 46.84%. This demonstrates that the prediction bias of the teacher model is a big issue and our method is effective to address it.

4.7 The Sensitivity of Exemplar Selection

We also study the impact of different exemplar selection strategies. We compare two strategies: (a) random selection and (b) herding selection the exemplar selection strategy proposed by iCaRL [25]. iCaRL maintains the exemplars that are closed to the class center in the feature space. Both strategies store 200 exemplars from old classes. The incremental learning results are shown in Figure 5. The herding exemplar selection strategy performs slightly better than the random selection. This demonstrates that our method is not sensitive to exemplar selection.

4.8 Experiment on CIFAR-100

Although the KiCi method in this paper is mainly used in the field of sensor-based activity recognition applications, from a methodological point of view, KiCi is also applicable to solve class incremental learning in other domains, such as the CV. The experimental results of the KiCi method on the CIFAR-100 dataset are given in Table 7. We start from a model trained on 10 randomly selected classes. Then, with each incremental step, 10 random classes are added. We store 2000 exemplars from old classes and select rehearsal exemplars based on herding selection the same as the previous work. We adopt an 18-layer ResNet and train 256 epochs. The learning rate starts from 0.1 and reduces to 1/10 of the previous learning rate after 100, 150, and 200 epochs. The models are optimized by SGD with batch size 128, and the temperature is set to 2. The average and standard deviations are obtained over five runs. We can infer from the results that our proposed KiCi outperforms the existing SOTA methods by a large margin in terms of the final incremental accuracy and the average incremental accuracy. Regarding the final incremental classifier on all classes, our KiCi method outperforms the SOTA method on CIFAR-100 by 2.06%. On average over all the incremental steps except the first steps (because the first step is

Table 7: Class incremental learning performance (accuracy %) on CIFAR-100. We initialize the model with 10 classes, followed by 9 incremental steps using 10 classes per step. The average results over all the incremental steps except the first step are obtained over five runs and also reported here.

#classes	10	20	30	40	50	60	70	80	90	100	AVG±SD
WA	90.61	79.28	73.55	68.03	66.91	64.17	63.54	63.27	62.09	60.68	69.21±8.95
DER2	90.82	79.89	74.01	72.30	71.17	68.76	65.18	64.82	64.67	63.02	71.45±8.13
Coil	90.80	80.30	76.45	73.85	72.32	70.04	68.97	67.31	65.28	64.67	73.08±7.52
KiCi	90.76	80.67	77.92	75.47	73.83	71.34	70.00	69.76	68.11	**66.73**	**74.46±6.85**

not relevant for class incremental learning), KiCi outperforms the SOTA method on CIFAR-100 by 1.38%. These results demonstrate our Kici method in image data is also more effectively reducing the bias from the teacher model's prediction and mitigating the problem of catastrophic forgetting during the incremental process to improve the classification accuracy.

5 CONCLUSIONS AND FUTURE WORK

Wearable-based human activity recognition (HAR) is commonly adopted in the real world. In this paper, we propose a new incremental learning approach, KiCi, for dynamic activity recognition. KiCi enables student models to have a reasonable trade-off between old and new knowledge by employing a subset of weights to evaluate the importance of knowledge. As a result, the teacher model's prediction bias of old knowledge is effectively eliminated, and negative transfer is avoided, preventing catastrophic forgetting. We plan to investigate the use of unsupervised or semi-supervised learning in future research to enable the model to actively discriminate the number of new classes throughout the incremental process and achieve accurate classification of old and new classes. In addition, we consider constructing hybrid increments of data, feature, and class increments to eliminate catastrophic forgetting and improve the classification accuracy of old and new classes during the incremental process.

ACKNOWLEDGMENTS

This work is supported by the National Key Research and Development Plan of China (No. 2020YFC2007104), Natural Science Foundation of China (No. 61902377, No. 62101530, No.62002187), Science and Technology Service Network Initiative, Chinese Academy of Sciences (No. KFJ-STS-QYZD-2021-11-001), Youth Innovation Promotion Association CAS.

REFERENCES

[1] Kian Ahrabian, Yishi Xu, Yingxue Zhang, Jiapeng Wu, Yuening Wang, and Mark Coates. Structure aware experience replay for incremental learning in graph-based recommender systems. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 2832–2836, 2021.

[2] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 139–154, 2018.

[3] Kerem Altun, Billur Barshan, and Orkun Tunçel. Comparative study on classifying human activities with miniature inertial and magnetic sensors. *Pattern Recognition*, 43(10):3605–3620, 2010.

[4] Hamza Amrani, Daniela Micucci, and Paolo Napoletano. Personalized models in human activity recognition using deep learning. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 9682–9688. IEEE, 2021.

[5] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, and Jorge L Reyes-Ortiz. Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine. In *International workshop on ambient assisted living*, pages 216–223. Springer, 2012.

[6] Claudio Bettini, Gabriele Civitarese, and Riccardo Presotto. Caviar: Context-driven active and incremental activity recognition. *Knowledge-Based Systems*, 196:105816, 2020.

[7] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. *Advances in neural information processing systems*, 33:15920–15930, 2020.

[8] Francisco M Castro, Manuel J Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In *Proceedings of the European conference on computer vision (ECCV)*, pages 233–248, 2018.

[9] Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K Dokania, Philip HS Torr, and M Ranzato. Continual learning with tiny episodic memories. 2019.

[10] Arslan Chaudhry, Albert Gordo, Puneet K Dokania, Philip Torr, and David Lopez-Paz. Using hindsight to anchor past knowledge in continual learning. *arXiv preprint arXiv:2002.08165*, 3, 2020.

[11] Tai-Te Chu, An-Zi Yen, Wei-Hong Ang, Hen-Hsen Huang, and Hsin-Hsi Chen. Vidlife: A dataset for life event extraction from videos. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 4436–4444, 2021.

[12] Xiang Deng and Zhongfei Zhang. Comprehensive knowledge distillation with causal intervention. *Advances in Neural Information Processing Systems*, 34, 2021.

[13] Yuntao Du, Jindong Wang, Wenjie Feng, Sinno Pan, Tao Qin, Renjun Xu, and Chongjun Wang. Adarnn: Adaptive learning and forecasting of time series. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 402–411, 2021.

[14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[15] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[16] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 831–839, 2019.

[17] Chunyu Hu, Yiqiang Chen, Lisha Hu, and Xiaohui Peng. A novel random forests based class incremental learning method for activity recognition. *Pattern Recognition*, 78:277–290, 2018.

[18] Chunyu Hu, Yiqiang Chen, Lisha Hu, Han Yu, and Dianjie Lu. Disagreement-based class incremental random forest for sensor-based activity recognition. *Knowledge-Based Systems*, 239:108044, 2022.

[19] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

[20] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017.

[21] HU Lisha, WANG Suzhen, CHEN Yiqiang, HU Chunyu, JIANG Xinlong, CHEN Zhenyu, and GAO Xingyu. Objective equilibrium measurement based kernelized incremental learning method for fall detection. *Journal of Computer Applications*, 38(4):928, 2018.

[22] He Lyu, Ningyu Sha, Shuyang Qin, Ming Yan, Yuying Xie, and Rongrong Wang. Advances in neural information processing systems. *Advances in neural information processing systems*, 32, 2019.

[23] German I Parisi, Ronald Kemker, Jose L Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113:54–71, 2019.

[24] Jathushan Rajasegaran, Munawar Hayat, Salman H Khan, Fahad Shahbaz Khan, and Ling Shao. Random path selection for continual learning. *Advances in Neural Information Processing Systems*, 32, 2019.

[25] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017.

[26] Attila Reiss and Didier Stricker. Introducing a new benchmarked dataset for activity monitoring. In *2012 16th international symposium on wearable computers*, pages 108–109. IEEE, 2012.

[27] Daniel Roggen, Alberto Calatroni, Mirco Rossi, Thomas Holleczek, Kilian Förster, Gerhard Tröster, Paul Lukowicz, David Bannach, Gerald Pirkl, Alois Ferscha, et al. Collecting complex activity datasets in highly rich networked sensor environments. In *2010 Seventh international conference on networked sensing systems (INSS)*, pages 233–240. IEEE, 2010.

[28] Pekka Siirtola and Juha Röning. Context-aware incremental learning-based method for personalized human activity recognition. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–15, 2021.

[29] Michalis K Titsias, Jonathan Schwarz, Alexander G de G Matthews, Razvan Pascanu, and Yee Whye Teh. Functional regularisation for continual learning using gaussian processes. 2019.

[30] Yuening Wang, Yingxue Zhang, and Mark Coates. Graph structure aware contrastive knowledge distillation for incremental learning in recommender systems. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 3518–3522, 2021.

[31] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 374–382, 2019.

[32] Shipeng Yan, Jiangwei Xie, and Xuming He. Der: Dynamically expandable representation for class incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3014–3023, 2021.

[33] Jaehong Yoon, Wonyong Jeong, Giwoong Lee, Eunho Yang, and Sung Ju Hwang. Federated continual learning with weighted inter-client transfer. In *International Conference on Machine Learning*, pages 12073–12086. PMLR, 2021.

[34] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International Conference on Machine Learning*, pages 3987–3995. PMLR, 2017.

[35] Bowen Zhao, Xi Xiao, Guojun Gan, Bin Zhang, and Shu-Tao Xia. Maintaining discrimination and fairness in class incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13208–13217, 2020.

[36] Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. Co-transport for class-incremental learning. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 1645–1654, 2021.