Received September 14, 2021, accepted October 12, 2021, date of publication October 22, 2021, date of current version November 1, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3122298

HiHAR: A Hierarchical Hybrid Deep Learning Architecture for Wearable Sensor-Based Human Activity Recognition

NGUYEN THI HOAI THU AND DONG SEOG HAN, (Senior Member, IEEE)
School of Electronic and Electrical Engineering, Kyungpook National University, Daegu 41566, Republic of Korea

Corresponding author: Dong Seog Han (dshan@knu.ac.kr)

This work was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2021-2020-0-01808) supervised by the IITP (Institute of Information & Communications Technology Planning & Evaluation).

This work involved human subjects or animals in its research. The authors confirm that all human/animal subject research procedures and protocols are exempt from review board approval.

ABSTRACT Wearable sensor-based human activity recognition (HAR) is the study that deals with sensor data to understand human movement and behavior. In a HAR model, feature extraction is widely considered to be the most essential and challenging part as the sensor signals contain important information in both spatial and temporal contexts. In addition, because people often carry out an activity for a while before changing to another activity, the sensor data also contain long-term context dependencies. In this paper, in order to enhance the long, short-term and spatial features from the sensor data, we propose a hierarchical deep learning-based HAR model (HiHAR) which is constructed from two powerful deep neural network architectures: convolutional neural network (CNN) and bidirectional long short-term memory network (BiLSTM). With the hierarchical structure, HiHAR contains two stages: local and global. In the local stage, a CNN and a BiLSTM are applied on the window-data level to extract local spatiotemporal features. The global stage with another BiSLTM is used to extract long-term context information from adjacent windows in both forward and backward time directions, then performs activity classification task. Our experiment results on two public datasets (UCI HAPT and MobiAct scenario) indicate that the proposed hybrid model achieves competitive performance compared to other state-of-the-art HAR models with an average accuracy of 97.98% and 96.16%, respectively.

INDEX TERMS Human activity recognition, wearable sensor, deep learning, CNNs, bidirectional LSTMs, context dependence.

I. INTRODUCTION

Human activity recognition (HAR) has been attracting considerable interest due to its wide-range applications in surveillance, smart environments and healthcare domains. Research in HAR can be organized into three main approaches: vision sensor-based, radio-based and wearable sensor-based.

In the first approach, videos collected from cameras such as RGB and depth videos are used to classify the activity by segmenting the human subject from the background and calculating the human skeleton [1]. Although this approach is currently one of the most active research areas in computer vision, challenging problems that arise in this approach are user privacy and computational complexity.

In the radio-based approach, attenuation of the radio strength and change of communication patterns caused by the existence and motions of users in a radio field are analysed to distinguish human activities [2]. This method provides a device-free solution for HAR [3] and utilizes the communication infrastructure such as wireless transceivers, thus helps to improve the user experience and to reduce the deployment cost. A key limitation of this approach is that the system is highly sensitive to the environmental interferences.

The third approach uses sensors such as inertial measurement units (IMUs), barometers and heart rate sensors embedded in wearable devices to perceive human movement. Recently, with the technological advancement in the area of microelectronics and the ubiquitousness of smart devices, discussions regarding this approach have dominated research in recent years, especially in the healthcare domain.

The associate editor coordinating the review of this manuscript and approving it for publication was Lorenzo Mucchi.

A typical application of HAR is health informatics mobile apps which are widely deployed in wearable smart devices [4]. By recognizing user activities, these apps can track user exercise intensity, health conditions, whereby help promote a healthier lifestyle. Another remarkable application of HAR is abnormal activity detection and alarm systems. Freezing of gait detection in Parkinson's disease can monitor the patients' movement and help stimulates the patients to resume walking by providing a rhythmic auditory signal [5], [6]. Automatic recognition of falls plays a vital role in providing earlier responses, thus, help to reduce the serious consequences especially for elderly people [7].

In general, a wearable sensor-based HAR system often contains three main stages: preprocessing, feature extraction and classification. In the first stage, the data (e.g., acceleration, angular velocity), after being collected from sensors, is preprocessed to remove noise, then split into small segments. From this, feature extraction methods are applied to the processed data, in order to extract essential features before being fed into classifiers.

Traditionally, conventional pattern recognition (PR) methods have been widely applied for HAR. In this approach, statistical and structural feature extraction methods are applied to extract features in the time domain (e.g., mean, standard deviation), frequency domain (e.g., Fourier transform) and time-frequency domain (e.g., short-time Fourier transform, Wavelet transform) [8]. Machine learning algorithms such as *k*-nearest neighbours (KNNs), Naïve Bayes (NB), decision tree techniques and support vector machine (SVM) are, then, adopted for the classification task. In some experimental environments, where the data is well-preprocessed and there are only a few labelled data, these HAR models are fully able to gain good performance. However, in real-life scenarios, this approach may suffer from a number of pitfalls as it heavily depends on heuristic hand-engineered feature extraction methods [9]. This domain knowledge requirement also makes it difficult to apply the HAR system to new domains.

With the rising of computing power and the number of available datasets, the past decade has witnessed a huge growth in deep learning algorithms which achieve remarkable performance in various fields such as computer vision (CV) and natural language processing (NLP) [10]. Deep learning models such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs) are believed to have the ability to learn data representation and automatically extract abstract features. In addition, their deep architectures also help promote the re-use of features [11]. Thus, deep learning has been used as an ideal approach to overcome the challenges of conventional pattern recognition methods in the field of HAR.

Different from the conventional approach, deep learning-based HAR frameworks often contain only two main parts: preprocessing and classification. Firstly, the raw sensor data is split into fix-length windows with an overlap. In the second part, feature extraction and classification are unified into one model and are simultaneously updated during the training process.

Similar to other classification tasks such as spam filtering and image classification, in the testing phase, most of the studies in HAR have only focused on the raw sensor data of single window for predicting the corresponding activity label. Although this approach has been proved to perform well on the well-cleaned experimental data, it fails to take into account the real-life scenario where the data contain not only basic activities but also transitions. Moreover, sensor data in HAR contain long-term dependency as people commonly carry out an activity for a while before changing to another activity. Thus, considering only context-independent data can make the system face the problem of inadequate information for activity prediction. In this paper, we explore the possibility of utilizing long-term dependency in human activity by introducing a hierarchical hybrid deep learning-based human activity recognition model (HiHAR).

The main contributions of this paper are the followings:

- We propose a novel hybrid end-to-end deep learning model which leverages two of the most popular deep network architectures: CNN and BiLSTM to efficiently extract both spatial and temporal features from multimodal raw sensor data.
- A hierarchical architecture with two stages: local and global is proposed to automatically extract local features from single windows and global temporal features from adjacent windows to enhancing the long-term dependency of human activity's time-sequential data, thus, improve the classification confidence.
- The proposed model is tested on two public datasets which provide real-life scenarios with a sequence of activities and transitions among them instead of using clean preprocessed data.

The rest of the paper is organized as follows. In Section II, we review some previous works which use deep learning methods and provide a background of hybrid DL models and some opening gaps in HAR. In Section III, we propose a novel HAR framework in which a hierarchical hybrid deep learning model is deployed in order to leverage the long-term dependency of human activity and improve the accuracy of the HAR system. Section IV shows the experimental results and discussions. Our conclusions and some notes on future works are drawn in the final section.

II. RELATED WORK

A. CONVOLUTIONAL NEURAL NETWORKS

In the last few years, several studies on HAR have delved into deep learning and achieved great successes. With the advantage of automatically extracting the correlation of local groups in array data, CNN has become a favourable deep learning architecture in HAR. One of the first examples of using CNN in HAR is presented in [12] by Yang *et al.* The authors indicate that through the deep architecture of CNN, higher-level representation of raw sensor data could be extracted. Moreover, mutually enhancing feature learning and classification in one model makes the learned features are more discriminative. Ronao and Cho [13] have pointed out that a deep CNN working with 1D convolutional operations can outperform traditional pattern recognition methods on activity classification using smartphone sensors. Whereas, Jiang *et al.* [14] instead of using 1D convolution, resized the sensor data to a virtual 2D image before feeding it into a 2D CNN in order to extract both temporal and spatial features from the activity images for the classification task. [15] proposes a two-stage CNN model to improve the recognition accuracy of activities that have complex patterns and less training data.

Recently, several state-of-the-art CNN architectures proposed in the computer vision field have been successfully applied to HAR. The work in [16] proposed a HAR model based on U-Net [17] to perform prediction at sampling point level, hence overcome the problem of multi-class. Mahmud *et al.* [18] transform 1D time-series sensor data to 2D data, then apply residual block-based CNN to extract features and classify activities. Tang *et al.* [19] use the Lego filter [20] to construct a lightweight deep convolutional neural network for HAR.

B. RECURRENT NEURAL NETWORKS

With the ability to contain information about the history of all the past elements in the sequence, recurrent neural networks (RNNs) are widely used for processing the time series sensor data in HAR. Zeng *et al.* [21] proposed a continuous attention-based long short-term memory (LSTM) model which pays more attention to important sensor modalities and salient parts of the sensor signal in HAR. Barut *et al.* [22] design a multitask framework using stacked LSTM layers to perform activity classification and intensity estimation from raw sensor data. Authors in [23] and [24], instead of using raw data, use feature data extracted from the principal component analysis (PCA) and discrete wavelet transform (DWT), respectively, as input to the bidirectional LSTM recurrent neural network. [25] propose a spectrogram-based feature extraction and data augmentation method to deal with the scarcity of label data, then carry out the classification task by using a deep LSTM. In [26], it was shown that fusing automatically learned features extracted from a stacked LSTM RNN and hand-crafted features can boost the system performance. Chen *et al.* [27] proposed a local feature-based LSTM network which is capable of encoding temporal dependency and learning features from high sampling frequency acceleration data.

C. HYBRID MODELS

In recent years, several studies have suggested that using hybrid models which are combined from different types of deep learning architectures can achieve high performance in HAR. A combination of inception module-based CNN and gated recurrent units (GRUs) [28] has been proposed in [29] for extracting sequential temporal dependencies in complex human activity recognition. Chen *et al.* [30] deployed a 1D-CNN-LSTM model for extracting deep features from long acceleration sequences, then used an attention mechanism to combine with the handcrafted features of heart rate variability data in a sleep-wake detection framework. A semi-supervised framework was proposed in [31] using a recurrent convolutional attention model to deal with the imbalance of the labelled data. [32] applies CNN on small slices of window data, the extracted features are then fed into an LSTM layer for activity recognition. Different from other hybrid HAR models that use LSTM on the features extracted from CNN, Xia *et al.* [33] proposed an LSTM-CNN in which a two-layer LSTM is applied directly to the raw sensor data before deploying 2D convolutional layers.

Although the literature shows the success of conventional pattern recognition and state-of-the-art deep learning methods, a closer look at the literature reveals a number of gaps and shortcomings. Most of the studies have only focused on the data of individual windows to predict the activities separately without considering the correlation of adjacent windows. Although this approach can achieve high accuracy in multi-class classification tasks such as face recognition and medical image classification, it can lose the property of long-term dependency in HAR sensor data. To address this shortcoming, Chen *et al.* [26] have proposed an algorithm called maximum full a posterior (MFAP) to consider both the past and the current a posterior information. The activity sequence is assumed as a first-order Markov chain so that the current observation is conditionally independent of the previous observation given the current activity. However, this method requires an extra manual task on the output of softmax layer from the deep neural network.

In addition, a little work has been done to utilize and validate the ideas under real-life scenario while a vast majority of prior works uses clean dataset in which the data of each activity is collected, processed and stored separately without considering the transitions between activities. In fact, activities are carried out in sequence and there are some activities that cannot be carried out adjacently without transition such as lying and running. To address these shortcomings, we propose a hierarchical hybrid model which we referred to as HiHAR. With the hierarchical architecture, the model can extract local temporal, spatial features, and global temporal dependency in window sequences.

III. METHODOLOGY

The HAR framework contains two main components: windowing-sequencing and the HiHAR model. In the windowing-sequencing step, collected sensor data are split into fix-length windows with overlaps. The window data is, then, segmented into sequences of windows. The HiHAR model consists of 2 stages: the first stage is a 2D CNN-BiLSTM subnet which is applied to single windows to extract local temporal and spatial features, the second stage is constructed from a BiLSTM and softmax layer to learn the global dependency and perform the classification task.

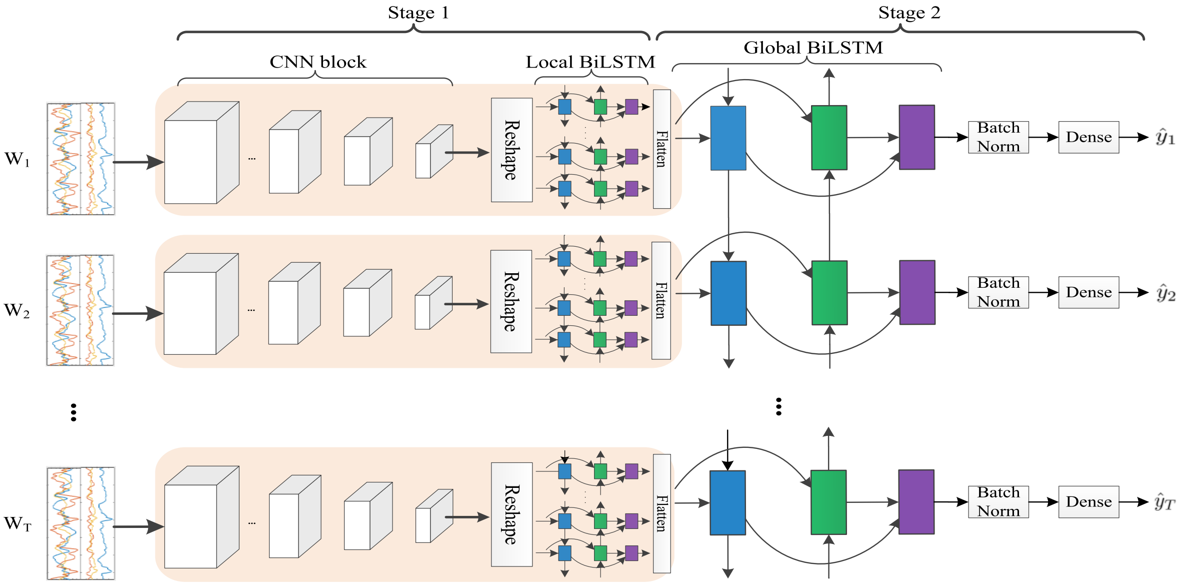


FIGURE 1. Detailed and unrolled architecture of the proposed hierarchical hybrid deep learning model. In stage 1, each window data is processed individually by the subnet contains a CNN block and a local BiLSTM to extract local features. Stage 2 contains a global BiLSTM which analyses the whole window sequence to extract global features and perform classification task.

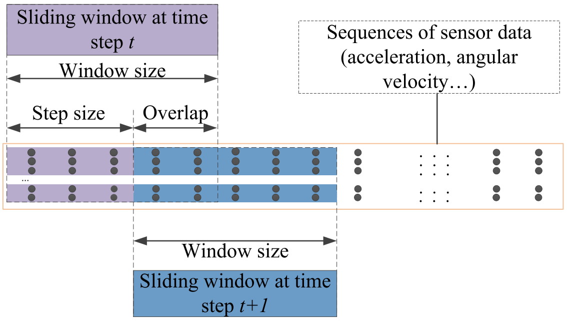

Sliding window at time step t
Window size
Step size
Overlap
Sequences of sensor data (acceleration, angular velocity…)
Window size
Sliding window at time step t+1

FIGURE 2. Illustration of sliding window method in HAR.

The detailed architecture of the proposed hybrid deep learning model is shown in Fig. 1.

A. WINDOWING-SEQUENCING

Windowing is considered as one of the most popular segmentation techniques used in HAR for the recognition of periodic activities (e.g., running, walking) and static activities (e.g., standing, sitting, lying) [34]. The raw sensor signals are split into fixed-length windows. There is an overlap between adjacent windows to increase the number of training data samples and avoid missing the transition of one activity to another. The windowing process is described in Fig. 2.

The window data at time t is a 2D matrix with a size of $(N \times K)$ and is presented as

$$\mathbf{W}_t = \left[\, \mathbf{a}_t^1 \ \mathbf{a}_t^2 \ \ldots \ \mathbf{a}_t^K \,\right] \in \mathbb{R}^{N \times K} \tag{1}$$

where column vector $\mathbf{a}_t^k = \left(\mathbf{a}_{t1}^k, \mathbf{a}_{t2}^k, \ldots, \mathbf{a}_{tN}^k\right)^\top$ is the sequence data of sensor k at window time t, $\top$ is the transpose operator, K is the number of sensor sequences and N is the length of the window. In order to leverage the correlations among windows and apply the training process, the window data is split into sequences of windows

$$\mathbf{S} = \left\{ \left(\mathbf{W}_1, y_1\right), \left(\mathbf{W}_2, y_2\right), \ldots, \left(\mathbf{W}_T, y_T\right) \right\} \tag{2}$$

where T is the length of the window sequence and y_t is the corresponding activity label of window $\mathbf{W}_t$.

B. STAGE 1: CNN-BiLSTM SUBNET

1) CONVOLUTIONAL BLOCK

With a structure in which each unit in feature maps is connected to local patches in the feature maps of the previous layer [10], CNNs using 2D kernels have the ability to extract local spatial and temporal features from raw sensor data. In addition, CNNs are also capable of identifying multimodal correlations among sensors [35]. Furthermore, if a pattern can appear in one part of a window sensor data, it could appear anywhere in other windows in term of time, hence the idea of sharing the same weights between units at different locations helps CNNs not only reduce the number of parameters but also can detect the same motif in varied temporal positions of the sensor array. Therefore, in the hybrid model, we implement a convolutional block that contains a set of 1D, 2D convolutional and pooling layers. The detailed architecture of this convolutional block is shown in Table 1.

Firstly, in order to input the sensor data into the 2-dimension convolutional block, the window data split from the previous step is considered as a virtual image $\mathbf{W}_t \in \mathbb{R}^{N \times K \times 1}$ with height is the length of the window: N, width is the number of sensor signals: K (e.g., accelerometer: x, y, z; and gyroscope: x, y, z), and depth is equal to 1.

The block contains 1D, 2D convolutional operations, pooling operations and activation functions. In the first two convolutional layers, a filter size of (3×3) is applied to extract both local temporal, spatial features and multimodal correlation from the sensor data. As the number of sensor signals is small compares to the window length (*i.e.*, $K \ll N$), a same-padding is used to preserve the input size spatially as well as the information near the edge of the input data. The last three layers MaxPool(2,1), Conv(3,1), MaxPool(2,2) are deployed without padding to distill the deep features, reduce representation and computational cost. In addition, by operating over each activation map independently without overlapping, and reduce the size of feature maps, pooling layers help the representations be more manageable and also reduce the risk of overfitting.

Output tensor of the last MaxPool layer has a size of $N' \times K' \times 32$ where the time information is contained in the first dimension with a size of N'. However, the local BiLSTM requires 2D input data (*timesteps* × *features*). Thus, the output is reshaped into a 2D tensor with a size of $\left(N' \times \left(K' \times 32\right)\right)$ before being fed into the local BiLSTM layer.

2) LOCAL BiLSTM

Bidirectional recurrent neural network (BRNN) was first proposed by Schuster and Paliwal [36] and is widely used in sequential data processing applications such as automatic speech recognition and machine translation. The network is constructed by splitting the state neurons of an RNN into two states with identical structures but opposite directions: forward and backward. The forward state deals with the positive time direction while the backward state deals with the negative time direction. Outputs from two states are combined using merging mode (e.g., concatenation, summation). In this study, in an effort to avoid vanishing and exploding gradient problems of standard RNN, an LSTM recurrent network proposed in [37] is deployed in the bidirectional recurrent neural network for learning temporal features in a single window.

Output hidden states of the local bidirectional long short-term memory layer at window W_t after merging is defined as

$$\mathbf{H}_t^{\langle L\rangle} = \left[\mathbf{h}_{t1}\ \mathbf{h}_{t2}\ \ldots\ \mathbf{h}_{tN'}\right]^\top \in \mathbb{R}^{N' \times R} \tag{3}$$

where the superscript $\langle L\rangle$ indicates the local stage, N' is the temporal dimension of the output tensor from the convolutional block, and R is the number of hidden units in BiLSTM. This output tensor $\mathbf{H}_t^{\langle L\rangle}$ of window $\mathbf{W}_t$ is then reshaped into a vector $\mathbf{o}_t^{\langle L\rangle}$ before being fed into the global BiLSTM layer. Finally, outputs of a window sequence can be defined as

$$\mathbf{O}^{\langle L\rangle} = \left[\mathbf{o}_1^{\langle L\rangle}\ \mathbf{o}_2^{\langle L\rangle}\ \ldots\ \mathbf{o}_T^{\langle L\rangle}\right] \tag{4}$$

C. STAGE 2: GLOBAL BiLSTM

Most of the previous studies consider the context-independence method where each window is analysed independently. However, the context information can play an important role in the classification of current activity. For example, if the predicted activity of the past window $\mathbf{W}_{t-1}$ and the future window $\mathbf{W}_{t+1}$ are lying, there will be a high possibility that the activity of the current window $\mathbf{W}_t$ is lying. Therefore, to be able to learn long-term dependency, a second BiLSTM is applied to a sequence of windows, instead of a single window. The global BiLSTM is implemented as a multiple-input and multiple-output layer where the hidden state of each input is used for predicting the corresponding output activity. This layer takes the output vectors of the window sequence from stage 1 as input for long-term dependency interpretation

$$\left(\mathbf{h}_1^{\langle G\rangle}, \ldots, \mathbf{h}_T^{\langle G\rangle}\right) = BiLSTM^{\langle G\rangle}\left(\mathbf{o}_1^{\langle L\rangle}, \ldots, \mathbf{o}_T^{\langle L\rangle}\right) \tag{5}$$

where $\mathbf{h}_t^{\langle G\rangle}$ is the hidden state of the global BiLSTM at window time t. A batch normalization method proposed by Ioffe and Szegedy [38] is applied to the output hidden states as a regularization mechanism in order to avoid overfitting and increase the stability of the model. Finally, a fully connected layer with a softmax activation function is deployed for activity classification.

Structure of the two BiLSTM layers used in this paper is different from the structure of a normal stacked RNN-based HAR model in two aspects: input data and inter-layer connection. In the normal stacked BiLSTM layers which are described in Fig. 3, the first BiLSTM layer is applied to the sensor data of every time step inside a window. An output hidden state of each time step is then sequentially input to the next BiLSTM layer. However, in the proposed HiHAR model, the local BiLSTM layer is applied to every time step of the distilled features extracted from the CNN block. All the output hidden states of a window are then fed into the global BiLSTM as a feature vector of one time step (*i.e.*, one window). Thus, the HiHAR model can process on a sequence of windows while the normal stacked BiLSTM model can only be applied for individual windows.

TABLE 1. Structure of the convolutional block.

Type	Filter Shape	Stride/Padding	Activation
Conv2D	3x3x32	(1, 1)/ same	ReLU
Conv2D	3x3x32	(1, 1)/ same	ReLU
MaxPool2D	Pool 2x1	(2, 1)/ no	
Conv2D	3x1x32	(1, 1)/ no	ReLU
MaxPool2D	Pool 2x2	(2, 2)/ no	
Reshape			

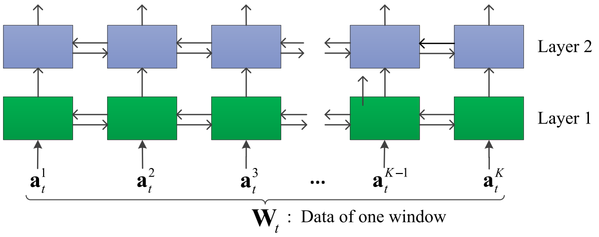


FIGURE 3. Normal structure of two stacked BiLSTM layers.

TABLE 2. Dataset description.

Dataset	UCI HAPT	MobiAct (Scenario)
Sensors	3-axis Accelerometer, Gyroscope	3-axis Accelerometer, Gyroscope, Orientation sensors
Device	Samsung Galaxy S2	Samsung Galaxy S3
Position	Waist	Trouser pocket
Sampling rate	50Hz	$\sim$200Hz$\xrightarrow{\text{downsample}}$50Hz
Window size	2.56s (128 readings)	2.56s (128 readings)
Overlap	1.28s (50%)	1.28s (50%)
No. subjects (Train/Test)	30 (1-24/ 25-30)	19 (1-14/ 15-19)
Activities	12 Activities (6 BAs & 6 PTs)	11 Activities (7 BAs & 4 PTs)
PT group 1	stand-to-sit, sit-to-lie, stand-to-lie	stand-to-sit, car-step in
PT group 2	lie-to-sit, sit-to-stand, lie-to-stand	sit-to-stand, car-step out

IV. EXPERIMENT RESULTS

A. DATASETS

The paper concentrates on building a model that can handle and utilize real-world circumstances where activities are carried out in a continuous way, thus, we selected two widely-used public datasets that provide raw data with sequences of actions and transitions: UCI HAPT [39] and MobiAct [40]. The details of the two datasets are listed in Table 2.

1) UCI HAPT

The public UCI HAPT dataset was collected from a group of 30 volunteers with an age range of 19-48 years old. The dataset provides raw inertial signals collected from 3-axial linear acceleration and 3-axial angular velocity sensors embedded in a smartphone mounted on the user's waist. It contains a set of 6 basics activities (BAs): standing, sitting, lying, walking, walking upstairs, walking downstairs; and 6 postural transitions (PTs) that occurred between three static postures: stand-to-sit, sit-to-stand, sit-to-lie, lie-to-sit, stand-to-lie and lie-to-stand.

2) MobiAct DATASET

The MobiAct dataset was collected and published by the Biomedical Informatics and eHealth Laboratory (BMI lab) in [40]. The dataset contains raw sensor data from a smartphone when participants carry out different types of activities of daily living and a range of falls. The phone was put in a trouser pocket freely chosen by the participant in random orientation. In this paper, we only use the scenario data in order to simulate real-world situations. The scenario data contains 5 sub-scenarios of daily living: leaving the home, being at work, leaving work, doing exercise and returning at home. The data was collected from 19 participants with 11 different activities: 7 basic activities (standing, sitting, walking, jogging, jumping, walking downstairs and walking upstairs) and 4 transitions (stand-to-sit, sit-to-stand, car-step in and car-step out). The original data was collected with the highest sampling rate of the smartphone which approximates to 200 samples per second (Hz).

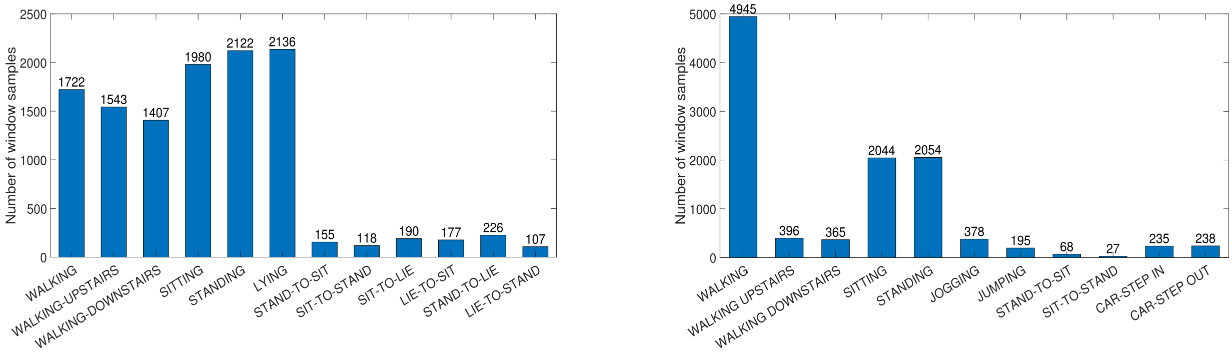
3) PREPROCESSING

In order to reduce the computational complexity, the scenario data in the MobiAct dataset is downsampled to 50Hz. With the sampling frequency of 50 Hz, we split the sequential data into windows with a fixed width of 2.56 seconds (128 readings/window) and 50% overlap. For the windows that contain more than one activity, the most frequent sample activity will be selected as the label of the window. Finally, the total number of windows obtained in UCI HAPT and MobiAct datasets are 11,883 and 10,945, respectively. The window data distribution on activities of two datasets is shown in Fig. 4. As we are only interested in recognizing basic activities (BAs) and the window number of postural transitions (PTs) are small compared to the BAs, in order to avoid the class imbalanced problem of PTs, a grouping method proposed in [24] is applied to cluster PTs that have similar pattern into groups.

B. EXPERIMENTAL SETTINGS

Shuffling dataset before splitting into training and testing sets is a widely-used method in HAR and other classification tasks. However, this method does not work in the case of time series data in HAR because it ignores the long-term correlation inherent in human physical activities. Moreover, human activity has a high intra-class variance since each person has a unique way to perform an activity. Thus, shuffling dataset gives the model a chance to look at the data from all subjects in the dataset and helps it predict well on the test set. However, in practical situations, the model is often applied to new users whose activity sensor data might have a big difference compared to the dataset. In addition, this method can cause difficulties in model investigation and implementation as researchers might struggle for reproducing the result. Therefore, in this study, in order to evaluate the model flexibility in working with different users, the models are trained on a set of subjects and tested on another set of subjects in the dataset. In the UCI HAPT dataset, data of the first 24 users are used for training and data of the remaining 6 users are used for testing while in the MobiAct (scenario) dataset, this training/testing split is 14 users/5 users.

As the BiLSTM network is applied over a sequence of windows instead of a single window, the window sequence data is shuffled randomly every epoch in order to obtain faster convergence [41] and provide the most general testing scenario. Due to randomness in the training process (e.g., weight initialization, data shuffling), the accuracy can slightly change after each running time, thus, for each model that we consider in this research, we run 10 experiments, the average accuracy and standard deviation obtained from the experiments are considered to assess model performance.



(a) UCI HAPT Dataset

(b) MobiAct Scenario Dataset

FIGURE 4. Window data distribution on different activities of the two datasets.

TABLE 3. Hyper-parameters and training configuration.

Parameter	Value
Initializer	Glorot uniform
Loss function	Categorical cross-entropy
Optimizer	Adam
Mini-batch size	64
No. epochs	250
LSTM hidden state size	128

All the deep learning models considered in this work are implemented using the TensorFlow framework [42] and are trained from the scratch. The weights and bias are initialized using Glorot uniform initializer [43]. The adaptive moment estimation (Adam) [44] is used as the gradient descent optimizer. A learning rate scheduling method is applied to adjust the learning rate during the training process. At the beginning of the training process, the learning rate is initialized to 0.001 and is decreased to 0.1 of the prior learning rate (*i.e.*, divided by 10) when the learning stagnates (*i.e.*, the loss has stopped decreasing for 10 epochs). The details of hyper-parameters and model training configuration are represented in Table 3.

C. RESULTS AND DISCUSSION

1) MERGING MODES IN BiLSTM

As outputs from the forward and backward direction in the BiLSTM are not connected but later are merged into one, four merging modes are examined in this study: averaging, concatenation, multiplication and summation with each mode represents a corresponding operator applied to the outputs of two unidirectional LSTM layers. In Table 4, the average accuracies of a HiHAR-8 model which uses sequences with a size of 8 windows in different merge modes are provided. The parameters column indicates the number of parameters in the last fully connected layer which takes the outputs of the global BiLSTM layer as input in the case of $\mathbf{W}_t \in \mathbb{R}^{128\times 6}$. It can be seen that although the concatenation method has double the number of parameters compared to the other merging modes, it does not achieve the best accuracy. Therefore, the summation mode is selected for further experiments as it has the least computational cost compared to other modes.

TABLE 4. Comparison of different merge modes of BiLSTM in HiHAR-8.

Merge mode	Accuracy (%) UCI HAPT	Accuracy (%) MobiAct-scenario	Parameters
Summation	97.98 ±0.24	96.16 ±0.22	1548
Averaging	97.88 ±0.24	95.81 ±0.39	1548
Multiplication	97.47 ±0.22	96.07 ±0.25	1548
Concatenation	97.82 ±0.28	95.70 ±0.15	3084

2) EFFECT OF SEQUENCE LENGTH

Because BiLSTM requires the whole sequence data to perform the forward and backward operation, considerable attention must be paid when selecting the size of the window sequences. A comparison of how different sequence lengths affect the model accuracy is conducted and the results are provided in Fig. 5. The box plots indicate the classification results from 10 attempts of each sequence size, while the dash lines present the average accuracies. It can be seen that the accuracy is improved significantly when the sequence length is increased. In the UCI HAPT dataset, the average accuracy increases by 2% by increasing window sequence length from $T = 1$ to $T = 3$. This is because longer sequences provide more past and future information for the model to analyse. However, long sequences can contain redundant information and make the training process be difficult as the model tends to memorize the order of activities carried out in the training data. Therefore, after taking the highest accuracy at sequence length $T = 8$, the accuracy becomes stagnant and starts decreasing. Since the model has to wait for the whole window sequence to come in order to analyse past and future information, long sequences lead to a high delay of the system. Therefore, the proposed model with window sequence length $T = 8$ (HiHAR-8) is selected as our final model for the best trade-off between accuracy and latency.

3) COMPARISON MODELS

We compare our proposed model with conventional pattern recognition methods as well as state-of-the-art deep learning

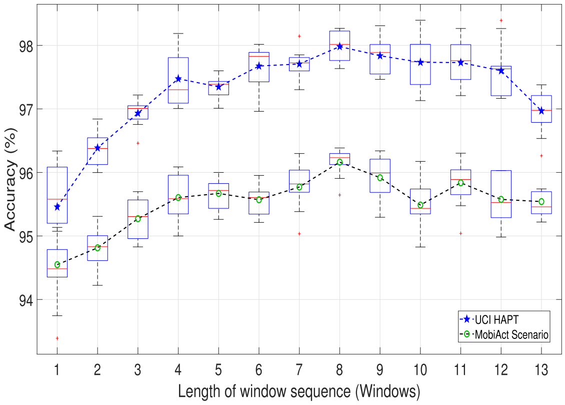


FIGURE 5. Accuracy of HiHAR model with different lengths of the window sequence.

TABLE 5. Average accuracy comparison between different machine learning methods and models of ablation study.

Model	Average Accuracy (%)	
	UCI HAPT	MobiAct-scenario
Conventional ML models		
KNNs ($k = 7$)	75.62	65.86
SVM	89.26	63.60
Single DL models		
LSTM	91.02 ±1.50	83.38 ±2.95
BiLSTM	92.63 ±1.07	90.43 ±1.48
CNN [12]	94.40 ±0.36	93.69 ±0.34
Hybrid DL models		
InnoHAR [29]	95.09 ±0.49	93.70 ±0.49
LSTM-CNN [33]	90.49 ±1.01	91.15 ±0.79
Ablation models		
CNN-BiLSTM Subnet	95.97 ±0.28	93.78 ±0.67
Unidirectional HiHAR	97.09 ±0.22	95.11 ±0.46
HiHAR-8	**97.98** ±0.24	**96.16** ±0.22

models in the field of HAR. In Table 5, different machine learning methods include conventional pattern recognition: KNNs, SVM; deep learning: LSTM, BiLSTM, stacked BiLSTM, CNN [12]; and hybrid deep learning methods: CNN-GRU [29], LSTM-CNN [33] are employed in order to make a comparison with the proposed HiHAR-8 network. Three referenced models used in this part are implemented based on the description in the corresponding papers. The results indicate that deep learning methods outperform conventional ML methods with a big gap in the average accuracy. The proposed HiHAR-8 achieves the best results on both datasets with 97.98% and 96.16% which are 2% higher compared to the hybrid inception module CNN-GRU-based InnoHAR model. These results have strengthened our hypothesis that the local spatio-temporal and long-term context features extracted by our hybrid deep learning model provide a better understanding of sensor data, hence improve the classification accuracy. In addition, the results suggest that CNN models are able to work well on raw signal while LSTM models perform well on the abstract sensor signal which is distilled by CNN.

TABLE 6. Complexity of the deep learning models.

Model	Complexity	
	No. Parameters	FLOPs
Single DL models		
LSTM	88,329	371,980
BiLSTM	158,985	610,577
CNN	56,509	112,784
Hybrid DL models		
InnorHAR	1,354,009	2,898,623
LSTM-CNN	90,633	196,881
CNN-BiLSTM Subnet	323,881	907,598
HiHAR-8	4,484,905	1,186,235

Table 6 illustrates the computational complexity including the number of trainable parameters and the number of floating-point operations (FLOPs) that are required for inferring a single window of the MobiAct (scenario) dataset with 9 activity labels. The number of FLOPs is calculated using the ProfileOptionBuilder API provided by the TensorFlow framework. As the HiHAR-8 model infers 8 windows at one time, the number of FLOPs of this model is divided by 8 to obtain the average FLOPs for inferring a single window. From the results, it can be seen that most of the hybrid models have higher complexity than the single DL models. The HiHAR-8 model has the highest number of parameters while the InnoHAR model requires the most number of FLOPs. In the HiHAR-8 model, although the same subnet at the local stage is used for every single window to reduce the size of the model, the connection between the two stages obtains the most parameters because all the output hidden states from the local BiLSTM are input to the global BiLSTM.

4) ABLATION STUDY

For a thorough examination of the proposed HiHAR model, an ablation study with two models is considered: CNN-BiLSTM subnet and unidirectional HiHAR. The former is implemented by removing the global BiLSTM from the HiHAR model with the aim of evaluating the efficiency in learning local spatial and temporal features. By using only sensor data of single windows for recognizing the corresponding activity, the model also helps to illustrate how the adjacent windows contribute to the activity prediction of the current window. The unidirectional HiHAR is implemented by replacing the global BiLSTM layer with an LSTM layer in order to delve into how the future information affects the current classification. The results of the ablation study are depicted in Table 5.

The subnet achieves competitive performance with the innoHAR model and outperforms the LSTM-CNN model on both datasets. This result reveals that using 2D convolutional kernels and a bidirectional recurrent network helps to improve the data representation learning of the model. However, using only the current window for predicting activity limits the view of the model, hence the average accuracy of the subnet is 2% lower than the HiHAR. When only the past information is provided, the unidirectional HiHAR achieves an average accuracy of 97.09% which is 1% lower compared to the HiHAR. Since the model requires no future

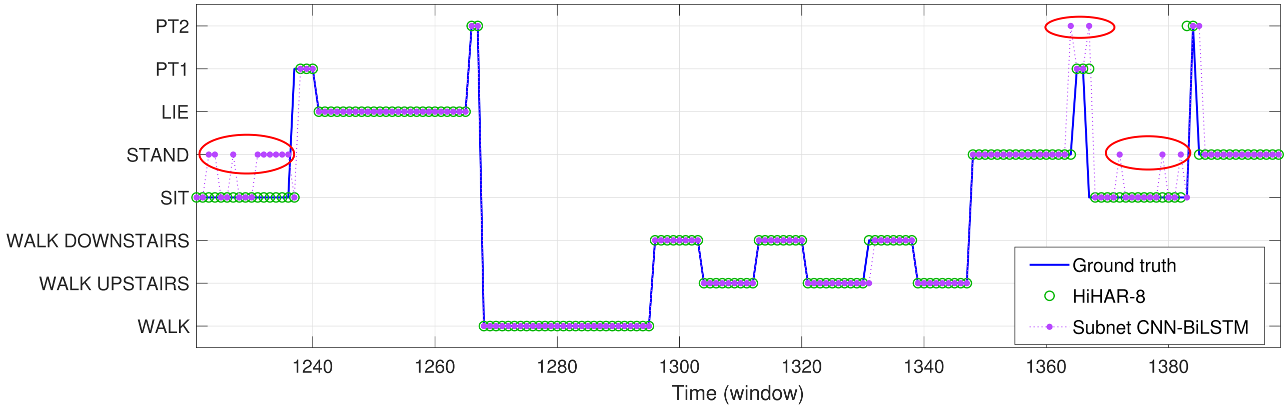
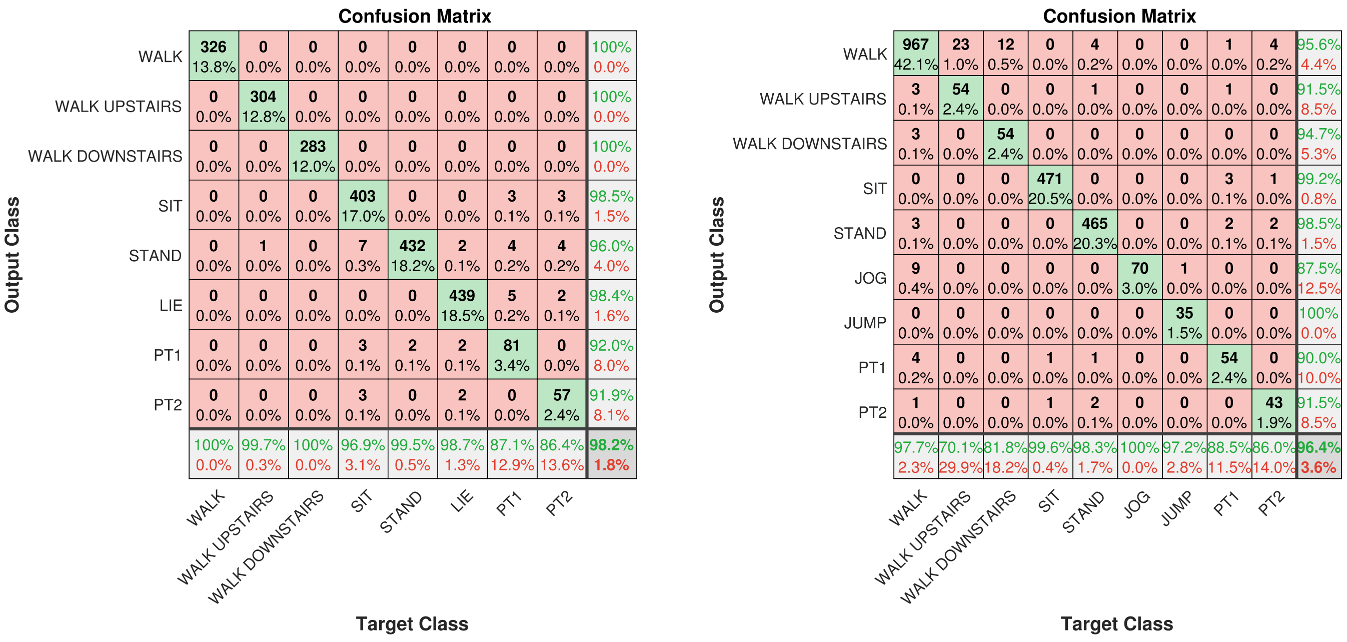


FIGURE 6. Activity classification results of the proposed HiHAR-8 model and the proposed subnet CNN-BiLSTM on the UCI HAPT dataset.



	WALK	WALK UPSTAIRS	WALK DOWNSTAIRS	SIT	STAND	LIE	PT1	PT2	
WALK	326 13.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
WALK UPSTAIRS	0 0.0%	304 12.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
WALK DOWNSTAIRS	0 0.0%	0 0.0%	283 12.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
SIT	0 0.0%	0 0.0%	0 0.0%	403 17.0%	0 0.0%	0 0.0%	3 0.1%	3 0.1%	98.5% 1.5%
STAND	0 0.0%	1 0.0%	0 0.0%	7 0.3%	432 18.2%	2 0.1%	4 0.2%	4 0.2%	96.0% 4.0%
LIE	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	439 18.5%	5 0.2%	2 0.1%	98.4% 1.6%
PT1	0 0.0%	0 0.0%	0 0.0%	3 0.1%	2 0.1%	2 0.1%	81 3.4%	0 0.0%	92.0% 8.0%
PT2	0 0.0%	0 0.0%	0 0.0%	3 0.1%	0 0.0%	2 0.1%	0 0.0%	57 2.4%	91.9% 8.1%
	100% 0.0%	99.7% 0.3%	100% 0.0%	96.9% 3.1%	99.5% 0.5%	98.7% 1.3%	87.1% 12.9%	86.4% 13.6%	98.2% 1.8%

(a) UCI HAPT Dataset

	WALK	WALK UPSTAIRS	WALK DOWNSTAIRS	SIT	STAND	JOG	JUMP	PT1	PT2	
WALK	967 42.1%	23 1.0%	12 0.5%	0 0.0%	4 0.2%	0 0.0%	0 0.0%	1 0.0%	4 0.2%	95.6% 4.4%
WALK UPSTAIRS	3 0.1%	54 2.4%	0 0.0%	0 0.0%	1 0.0%	0 0.0%	0 0.0%	1 0.0%	0 0.0%	91.5% 8.5%
WALK DOWNSTAIRS	3 0.1%	0 0.0%	54 2.4%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	94.7% 5.3%
SIT	0 0.0%	0 0.0%	0 0.0%	471 20.5%	0 0.0%	0 0.0%	0 0.0%	3 0.1%	1 0.0%	99.2% 0.8%
STAND	3 0.1%	0 0.0%	0 0.0%	0 0.0%	465 20.3%	0 0.0%	0 0.0%	2 0.1%	2 0.1%	98.5% 1.5%
JOG	9 0.4%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	70 3.0%	1 0.0%	0 0.0%	0 0.0%	87.5% 12.5%
JUMP	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	35 1.5%	0 0.0%	0 0.0%	100% 0.0%
PT1	4 0.2%	0 0.0%	0 0.0%	1 0.0%	1 0.0%	0 0.0%	0 0.0%	54 2.4%	0 0.0%	90.0% 10.0%
PT2	1 0.0%	0 0.0%	0 0.0%	1 0.0%	2 0.1%	0 0.0%	0 0.0%	0 0.0%	43 1.9%	91.5% 8.5%
	97.7% 2.3%	70.1% 29.9%	81.8% 18.2%	99.6% 0.4%	98.3% 1.7%	100% 0.0%	97.2% 2.8%	88.5% 11.5%	86.0% 14.0%	96.4% 3.6%

(b) MobiAct Scenario Dataset

FIGURE 7. Confusion matrices of HiHAR-8 on UCI HAPT and MobiAct scenario datasets. (The rightmost columns present the class-wise precision values, and the bottom rows present the class-wise recall values. The cells on the bottom right of the plots show the overall accuracies.)

information, the latency is reduced to just half of the window size (1.28s). Thus, in case that a fast response is required, unidirectional HiHAR can be deployed with high performance compared to other state-of-the-art models.

Fig. 6 illustrates the activity recognition results of the HiHAR-8 and the subnet CNN-BiLSTM on the UCI HAPT dataset. Although the subnet performs well in recognizing most of the activities and tracking the activity change, there are some misclassifications between standing and sitting. The reason is that in the UCI HAPT dataset, the phone was mounted at the user's waist, thus, the acceleration and angular velocity data between sitting and standing are sometimes similar to each other. This similarity causes confusion in the subnet as the output predicted activity keeps changing between standing and sitting. In contrast, by exploiting the past and future information, the HiHAR model has higher confidence in differentiating the two activities. This result has further strengthened our hypothesis that there exists a correlation among human activities and by leveraging this context information from adjacent windows, the model has higher confidence in predicting activity.

5) CONFUSION MATRICES

The confusion matrices of the proposed HiHAR model in one experiment on two datasets are shown in Fig. 7. The model performs well on classifying most of the basic activities with high precision and recall values. For example, an approximation of 100% of precision and recall are obtained for walking, walking downstairs, walking upstairs, standing in the UCI HAPT dataset, and for jogging, jumping in the MobiAct scenario dataset. However, one limitation of the proposed model is found in the case of the MobiAct scenario dataset. Because the phone is kept at a random side of the trousers and in a random direction, there are some windows that the model confuses in differentiating walking, walking upstairs and walking downstairs. In both datasets, the transition groups are sometimes misclassified with their corresponding static postures. This is because there is no exact boundary among the preceding posture, transition, and the following posture, which leads to a problem that there may exist multiple activities in one window data.

V. CONCLUSION

In this paper, we presented a novel hierarchical hybrid deep learning-based model to enhance temporal, spatial features and utilize long-term context information in wearable sensor-based human activity recognition systems. The proposed system incorporates the use of a convolutional neural network in time-space information extraction and two bidirectional long short-term memory networks in learning local and global context in both forward and backward time directions. The experimental results show that by focusing on the multi-modality characteristic of the sensor signal and leveraging the strengths of the CNN, BiLSTM network, our proposed method can significantly improve the classification accuracy of the HAR model in the two public datasets. Importantly, our results provide evidence for the potential of using context information in activity recognition. In future work, we intend to concentrate on how to reduce the system latency, complexity and improve the accuracy of recognition between static postures. In addition, further research will be needed to overcome the problem of inter-class similarity and reduce the impact of device position on system performance.

REFERENCES

[1] L. Wang, D. Q. Huynh, and P. Koniusz, "A comparative review of recent kinect-based action recognition algorithms," *IEEE Trans. Image Process.*, vol. 29, pp. 15–28, 2020.

[2] S. Wang and G. Zhou, "A review on radio based activity recognition," *Digit. Commun. Netw.*, vol. 1, no. 1, pp. 20–29, 2015.

[3] D. Wu, D. Zhang, C. Xu, H. Wang, and X. Li, "Device-free WiFi human sensing: From pattern-based to model-based approaches," *IEEE Commun. Mag.*, vol. 55, no. 10, pp. 91–97, Oct. 2017.

[4] M. Milosevic, M. T. Shrove, and E. Jovanov, "Applications of smartphones for ubiquitous health monitoring and wellbeing management," *JITA J. Inf. Technol. Appl. (Banja Luka) (APEIRON)*, vol. 1, no. 1, pp. 7–15, Jun. 2011.

[5] L. Sigcha, N. Costa, I. Pavón, S. Costa, P. Arezes, J. M. López, and G. De Arcas, "Deep learning approaches for detecting freezing of gait in Parkinson's disease patients through on-body acceleration sensors," *Sensors*, vol. 20, no. 7, p. 1895, Mar. 2020.

[6] M. Bachlin, M. Plotnik, D. Roggen, I. Maidan, J. M. Hausdorff, N. Giladi, and G. Troster, "Wearable assistant for parkinson's disease patients with the freezing of gait symptom," *IEEE Trans. Inf. Technol. Biomed.*, vol. 14, no. 2, pp. 436–446, Mar. 2010.

[7] N. Noury, A. Fleury, P. Rumeau, A. K. Bourke, G. O. Laighin, V. Rialle, and J. E. Lundy, "Fall detection–Principles and methods," in *Proc. 29th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Aug. 2007, pp. 1663–1666.

[8] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 3, pp. 1192–1209, 3rd Quart., 2013.

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

[10] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.

[11] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[12] J. B. Yang, M. N. Nguyen, P. P. San, X. L. Li, and S. Krishnaswamy, "Deep convolutional neural networks on multichannel time series for human activity recognition," in *Proc. 24th Int. Conf. Artif. Intell. (IJCAI)*, 2015, pp. 3995–4001.

[13] C. A. Ronao and S.-B. Cho, "Human activity recognition with smartphone sensors using deep learning neural networks," *Expert Syst. Appl.*, vol. 59, pp. 235–244, Oct. 2016.

[14] W. Jiang and Z. Yin, "Human activity recognition using wearable sensors by deep convolutional neural networks," in *Proc. 23rd Int. Conf. Multimedia ACM*. Oct. 2015, pp. 1307–1310.

[15] J. Huang, S. Lin, N. Wang, G. Dai, Y. Xie, and J. Zhou, "TSE-CNN: A two-stage end-to-end CNN for human activity recognition," *IEEE J. Biomed. Health Informat.*, vol. 24, no. 1, pp. 292–299, Jan. 2020.

[16] Y. Zhang, Z. Zhang, Y. Zhang, J. Bao, Y. Zhang, and H. Deng, "Human activity recognition based on motion sensor using U-Net," *IEEE Access*, vol. 7, pp. 75213–75226, 2019.

[17] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. 18th Int. Conf. Med. Image Comput. Comput.-Assist. Intervent (MICCAI)*. Cham, Switzerland: Springer, 2015, pp. 234–241.

[18] T. Mahmud, A. Q. M. Sazzad Sayyed, S. A. Fattah, and S.-Y. Kung, "A novel multi-stage training approach for human activity recognition from multimodal wearable sensor data using deep neural network," *IEEE Sensors J.*, vol. 21, no. 2, pp. 1715–1726, Jan. 2021.

[19] 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," *IEEE Sensors J.*, vol. 21, no. 1, pp. 581–592, Jan. 2021.

[20] Z. Yang, Y. Wang, C. Liu, H. Chen, C. Xu, B. Shi, C. Xu, and C. Xu, "LegoNet: Efficient convolutional neural networks with Lego filters," in *Proc. 36th Int. Conf. Mach. Learn.*, vol. 97, 2019, pp. 7005–7014.

[21] M. Zeng, H. Gao, T. Yu, O. J. Mengshoel, H. Langseth, I. Lane, and X. Liu, "Understanding and improving recurrent networks for human activity recognition by continuous attention," in *Proc. ACM Int. Symp. Wearable Comput.*, Oct. 2018, pp. 56–63.

[22] O. Barut, L. Zhou, and Y. Luo, "Multitask LSTM model for human activity recognition and intensity estimation using wearable sensor data," *IEEE Internet Things J.*, vol. 7, no. 9, pp. 8760–8768, Sep. 2020.

[23] A. A. Aljarrah and A. H. Ali, "Human activity recognition using PCA and BiLSTM recurrent neural networks," in *Proc. 2nd Int. Conf. Eng. Technol. Appl. (IICETA)*, Aug. 2019, pp. 156–160.

[24] N. T. H. Thu and D. S. Han, "Utilization of postural transitions in sensor-based human activity recognition," in *Proc. Int. Conf. Artif. Intell. Inf. Commun. (ICAIIC)*, Feb. 2020, pp. 177–181.

[25] O. S. Eyobu and D. Han, "Feature representation and data augmentation for human activity classification based on wearable IMU sensor data using a deep LSTM neural network," *Sensors*, vol. 18, no. 9, p. 2892, Aug. 2018.

[26] Z. Chen, S. Xiang, J. Ding, and X. Li, "Smartphone sensor-based human activity recognition using feature fusion and maximum full a posteriori," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 7, pp. 3992–4001, Jul. 2020.

[27] Z. Chen, M. Wu, K. Gao, J. Wu, J. Ding, Z. Zeng, and X. Li, "A novel ensemble deep learning approach for sleep-wake detection using heart rate variability and acceleration," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 5, no. 5, pp. 803–812, Oct. 2021.

[28] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Doha, Qatar, Oct. 2014, pp. 1724–1734.

[29] C. Xu, D. Chai, J. He, X. Zhang, and S. Duan, "InnoHAR: A deep neural network for complex human activity recognition," *IEEE Access*, vol. 7, pp. 9893–9902, 2019.

[30] Z. Chen, M. Wu, W. Cui, C. Liu, and X. Li, "An attention based CNN-LSTM approach for sleep-wake detection with heterogeneous sensors," *IEEE J. Biomed. Health Informat.*, vol. 25, no. 9, pp. 3270–3277, Sep. 2021.

[31] K. Chen, L. Yao, D. Zhang, X. Wang, X. Chang, and F. Nie, "A semisupervised recurrent convolutional attention model for human activity recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 5, pp. 1747–1756, May 2020.

[32] R. Mutegeki and D. S. Han, "A CNN-LSTM approach to human activity recognition," in *Proc. Int. Conf. Artif. Intell. Inf. Commun. (ICAIIC)*, Feb. 2020, pp. 362–366.

[33] K. Xia, J. Huang, and H. Wang, "LSTM-CNN architecture for human activity recognition," *IEEE Access*, vol. 8, pp. 56855–56866, 2020.

[34] O. Banos, J.-M. Galvez, M. Damas, H. Pomares, and I. Rojas, "Window size impact in human activity recognition," *Sensors*, vol. 14, no. 4, pp. 6474–6499, Apr. 2014.

[35] E. Sansano, R. Montoliu, and Ó. B. Fernández, "A study of deep neural networks for human activity recognition," *Comput. Intell.*, vol. 36, no. 3, pp. 1113–1139, Aug. 2020.

[36] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.

[37] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[38] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, vol. 37, Jul. 2015, pp. 448–456.

[39] J.-L. Reyes-Ortiz, L. Oneto, A. Samà, X. Parra, and D. Anguita, "Transition-aware human activity recognition using smartphones," *Neurocomputing*, vol. 171, pp. 754–767, Jan. 2016.

[40] C. Chatzaki, M. Pediaditis, G. Vavoulas, and M. Tsiknakis, "Human daily activity and fall recognition using a smartphone's acceleration sensor," in *Information and Communication Technologies for Ageing Well and e-Health*. Cham, Switzerland: Springer, 2017, pp. 100–118.

[41] Y. Bengio, "Practical recommendations for gradient-based training of deep architectures," in *Neural Networks: Tricks Trade*. Berlin, Germany: Springer, 2012, pp. 437–478.

[42] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, and S. Ghemawat. (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. [Online]. Available: tensorflow.org

[43] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Stat.*, 2010, pp. 249–256.

[44] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, 2015, pp. 1–15.

NGUYEN THI HOAI THU received the B.S. degree in electronics and communications engineering from Vietnam National University (VNU), Hanoi, Vietnam, in 2017, and the M.S. degree in electronic and electrical engineering from Kyungpook National University (KNU), Daegu, Republic of Korea, in 2021, where she is currently pursuing the Ph.D. degree. Her research interests include signal processing, machine learning, and communications for e-health, biomedical engineering applications, and autonomous vehicle.

DONG SEOG HAN (Senior Member, IEEE) received the B.S. degree in electronic engineering from Kyungpook National University (KNU), Daegu, South Korea, in 1987, and the M.S. and Ph.D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology, Daejon, South Korea, in 1989 and 1993, respectively. From 1987 to 1996, he was with Samsung Electronics Company Ltd., where he developed the transmission systems for QAM HDTV and Grand Alliance HDTV receivers. He was a Courtesy Associate Professor with the Department of Electrical and Computer Engineering, University of Florida, in 2004. He was the Director of the Center of Digital TV and Broadcasting, Institute for Information Technology Advancement, from 2006 to 2008. Since 1996, he has been with the School of Electronics Engineering, KNU, as a Professor. His research interests include intelligent signal processing and autonomous vehicles.