CHAR: Composite Head-Body Activities Recognition With a Single Earable Device

Peizhao Zhu, Yuzheng Zhu, Wenyuan Li, Yanbo He, Yongpan Zou, *Member, IEEE*, Kaishun Wu, *Fellow, IEEE*, and Victor C. M. Leung, *Life Fellow, IEEE*

***Abstract*—The increasing popularity of earable devices stimulates great academic interest to design novel head gesture-based interaction technologies. But existing works simply consider it as a singular activity recognition problem. This is not in line with practice since users may have different body movements such as walking and jogging along with head gestures. It is also beneficial to recognize body movements during human-device interaction since it provides useful context information. As a result, it is significant to recognize such composite activities in which actions of different body parts happen simultaneously. In this paper, we propose a system called CHAR to recognize composite head-body activities with a single IMU sensor. The key idea of our solution is to make use of the inter-correlation of different activities and design a multi-task learning network to extract shared and specific representations. We implement a real-time prototype and conduct extensive experiments to evaluate it. The results show that CHAR can recognize 60 kinds of composite activities (12 head gestures and 5 body movements) with high accuracies of 89.7% and 85.1% in sufficient data and insufficient data cases, respectively.**

***Index Terms*—Composite activity recognition, earable device, multi-task learning.**

I. Introduction

Human activity recognition (HAR) has become a hotspot in academia due to its great significance for various intelligent services in our daily life such as healthcare, personalized recommendation, and human-computer interaction (HCI). Researchers have proposed a variety of HAR methods with different sensing modalities including radio frequency (RF) signals [1], [2], [3], acoustic sensors [4], [5], [6], and inertial measurement unit (IMU) [7], [8], [9], [10]. These works have considered a wide range of daily activities including walking [11], sleeping [12], eating [13], and etc.. In spite of good recognition performance, these works share a common shortcoming, that is, only considering singular activities and outputting a single activity type.

Singular activities usually involve a single body part such as a hand or head, or treat all the body parts as a whole. In singular activities recognition, researchers do not care about what actions are performed of different body parts. As a result, existing works usually assume that the whole body is static while a user is performing activities/gestures with her head or hand. But this is inconsistent with the characteristics of human activities, most of which consist of different actions of multiple body parts. For example, when people are walking, they may wave hands or nod heads to interact with others. In this process, walking and waving hands (or nodding heads) are independent activities of which both have meaningful and distinguishable semantic information. For another example, when people go upstairs, they may also pick up phone calls at the same time. Although such composite activities can be monitored with multiple sensors attached on different body parts as done in previous works [14], [15], [16], this is obviously not realistic and appealing in real-world scenarios as people may not be willing to wear multiple devices with them. Considering this, we pay attention to a more complex and practical HAR problem, that is, *how to recognize composite activities with a single wearable sensor*. Formally, we define composite activities as simultaneous actions of different body parts in this paper.

Composite activities have significant and practical value for human-computer interaction in real world. A typical example is shown in Fig. 1. When a user is doing exercises, she can perform head gestures to interact with smart earphones such as turning up volume, picking up calls, and switching songs. At the same time, the smart earphones can recommend exercises-related music or audiovisual programs if doing exercises is recognized. Similarly, when a user goes upstairs with hands occupied, she can shake her head to turn up/down audio's volume of earphones, and turn on noise reduction mode. In a summary, the user performs composite activities in these examples with head gestures and body movements happening simultaneously. Consequently, we

Received 12 November 2023; revised 9 June 2024; accepted 17 February 2025. Date of publication 6 March 2025; date of current version 5 June 2025. This work was supported in part by the China NSFC under Grant 62172286 and Grant U2001207, in part by the Key Project of Department of Education of Guangdong Province under Grant 2024ZDZX1011, in part by Guangdong NSF under Grant 2022A1515011509, in part by the Guangdong Provincial Key Lab of Integrated Communication, Sensing and Computation for Ubiquitous Internet of Things under Grant 2023B1212010007, in part by the Project of DEGP under Grant 2023KCXTD042, in part by Guangdong "Pearl River Talent Recruitment Program" under Grant 2019ZT08X603, in part by Guangdong "Pearl River Talent Plan" under Grant 2019JC01X235, in part by 111 Center under Grant D25008, in part by Shenzhen Science and Technology Foundation under Grant ZDSYS20190902092853047, and in part by Tencent "Rhinoceros Birds" - Scientific Research Fund for Young Researchers of Shenzhen University. Recommended for acceptance by J. Xu. *(Peizhao Zhu and Yuzheng Zhu contributed equally to this work.) (Corresponding author: Yongpan Zou.)*

Peizhao Zhu, Yuzheng Zhu, Wenyuan Li, Yanbo He, and Yongpan Zou are with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China (e-mail: zhupeizhao2017@email.szu.edu.cn; zhuyuzheng2022@email.szu.edu.cn; wenyuan@email.szu.edu.cn; fsalansky-walker@gmail.com; yongpan@szu.edu.cn).

Kaishun Wu is with the Information Hub, Hong Kong University of Science and Technology (Guangzhou), Guangzhou 511453, China (e-mail: wuks@hkust-gz.edu.cn).

Victor C. M. Leung is with the Artificial Intelligence Research Institute, Shenzhen MSU-BIT University, Shenzhen 518172, China (e-mail: vleung@ieee.org).

Digital Object Identifier 10.1109/TMC.2025.3548647

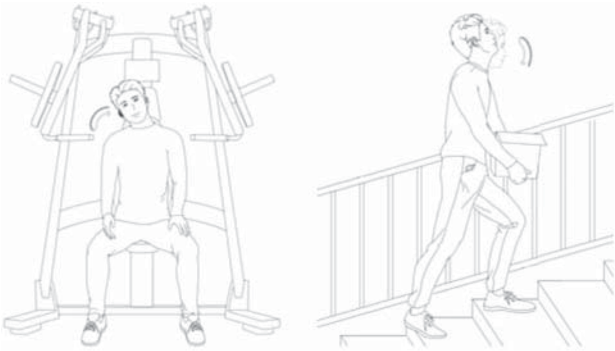


Fig. 1. The application scenarios of composite activities recognition with an earable device. Doing exercises while staying still with hands occupied (left) and going upstairs while carrying goods (right).

believe that it is meaningful to recognize both parts of activities (i.e., head gestures and body movements) due to the following reasons. On one hand, we can see from the examples that both head gestures and body movements have significant semantic information which can be used for human-device interaction. On the other hand, the commonalities and differences between different tasks are beneficial for boosting the recognition and generalization performance.

Considering the diversity of composite activities, we only focus on recognizing 60 kinds of head-body activities in this work including 12 head gestures and 5 body movements with a single earphone. There are two considerations. For one thing, a human being's head has a large degree of freedom and is very flexible to perform different gestures gently without causing any apparent discomfort. This makes head gestures appropriate for designing HCI interfaces. For another thing, compared with other body parts, head gestures happen together with body movements more naturally especially when a user is walking or jogging.

However, the solution to recognizing composite activities with a single sensor is not so straightforward due to two key obstacles. The first is the tight coupling of different tasks. Different actions will interfere with each other, which makes it difficult to recognize them. To be specific, sporadic head gestures produce notable measurements which are interference to body movements. The second obstacle is the indistinct measurements of body movements. Although an earable sensor is easy to capture head gestures, it is insensitive for body movements due to the stability of human's heads during activities. An intuitive idea is to treat each task independently and design separate learning networks to recognize different actions. But this approach ignores the correlation between head gestures and body movements which can be utilized for boosting performance. Inspired by multi-task learning, we have designed a composite activities recognition network for head gestures and body movements. Its key idea is to first extract general representation shared by coupled tasks with a shared bottom block, and then decouple them to further purify fine-grained task-specific features. As a result, it not only avoids over-fitting to a specific task, but also facilitates decision-making of each task.

Traditionally, we need to collect data from all activity categories for recognition network training in order to recognize human activity. Composite activity contains a larger number of categories, which will bring a greater data collection burden. Therefore, in addition to implementing traditional activity recognition, we specially consider a more challenging scenario, that is, *how to implement comparable composite activity recognition with insufficient data.* An innovative idea is to develop a series of data augmentation strategies to expand the dataset based on the characteristics of composite activity. Augmentation strategies are applied in the insufficient data scenario, including data synthesis, random transformation, and generative adversarial network. Through these data augmentation strategies, we not only greatly reduce the data collection burden caused by recognizing composite activity, but also realizing comparable composite activity recognition performance even with insufficient data.

Based on the above, we have built the first, as far as we know, composite head-body activities recognition system called CHAR with a single earable IMU sensor. CHAR respectively recognizes 60 kinds of activities with high accuracies of 89.7% for composite activity under training with sufficient self-collected dataset. Not only that, with a small amount of labeled data, the recognition accuracy of composite activity is 85.1%, sacrificing only 4.6%.

In a summary, the contributions of this work can be summarized as follows:

- We consider a novel HAR problem in which composite head-body activities are recognized. Instead of decomposing a composite activity into individual tasks, we take full advantage of the correlations between different tasks and design a multi-task learning network to accomplish both goals simultaneously. Compared with traditional methods, our approach outperforms with higher accuracies.
- We design and implement an earphone-based real-time prototype system with low-cost hardware and a self-developed mobile application. Extensive experiments have shown that our system can recognize 60 composite activities with an high accuracy up to 89.7% even in user-independent case.
- We further consider the insufficient data scenario and design a series of data augmentation strategies to achieve comparable composite head-body activity recognition performance. Experimental results indicate that our system still achieve 85.1% accuracy under training with insufficient data.

The remaining of this paper is organized as follows. Section II discusses the related work. Section III gives introduction to the system design. In Sections IV and V, we mainly describe the details of system implementation, experimental settings, and performance evaluation. At last, Section VII concludes the paper.

II. Related Work

A. Human Activity Recognition

Vision systems [17], [18] use cameras for recognition, but they are considered invasive. Ambient sensors [2], [3], [19] require to be installed at fixed locations with extra expense. In addition to these, wearable sensors, which are usually built in mobile devices, have attracted the attention of researchers. Some researchers take advantage of microphone [5], [6], biological signal sensor [20], [21] for HAR. The works which utilize IMU to recognize sporadic [7], [8], [9], [10] or periodic [11], [15], [22], [23] human acitivities are most related to ours. When it comes to sporadic activities, HulaMove [7] uses IMU of mobile phone for waist interaction. HeadGesture [8] proposes a hands-free input approach leveraging head gestures. GlassGesture [9] provides efficient gesture recognition and robust authentication with Google Glass. As for periodic movements, previous works [11], [24], [25], [26] use smartphone to recognize periodic movements and contribute their datasets. Other works propose some powerful networks such as DCNN [22], DeepSense [23], DeepConvLSTM [15] based on public datasets to recognize daily activities. Obviously, these works only concentrate on the recognition of singular activities. Nevertheless, human activity is usually a composite of many kinds of actions. As a contrast, our system makes use of multi-task learning to recognize sporadic head gestures and periodic body movements at the same time. In addition, X-CHAR [27] defines a complex activity that must be composed of multiple simple activities that may evolve over long periods in different orders and frequencies. Among them, a simple activity is the basic unit that can be captured within a short time window and can not be decomposed further. As a result, a complex activity is essentially the composition of a sequence of simple activities along time. Similarly, RFWash [28] recognizes gestures that are performed back-to-back in a continuous sequence. Different from them, composite activities in our work refer to multiple different types of activities occurring simultaneously.

B. Multi-Task Learning

Multi-task learning is a training paradigm which can deal with multiple tasks jointly. It has been successfully applied in many areas especially in natural language processing. However, multi-task learning has not been widely used in HAR. The works [29], [30], [31] make use of multi-task learning approach to realize gestures/activities recognition and user authentication jointly. CogAx [32] uses a contrastive and multi-task learning framework to correlate with underlying functional and cognitive health parameters of older adults. As the most relevant work, AROMA [14] defines complex activities as continuous and high-level semantic meaning activities which are combination of simple activities along temporal dimension. AROMA proposes a deep multi-task learning based method to facilitate complex activities recognition using features extracted from simple activities. In contrast, we concentrate on composite activities composed of simultaneous actions of different body parts. Consequently, we designed a multi-task recognition network which makes full use of shared information and further exploits task-specific representations of each task to facilitate decision-making.

C. Earable Applications

Recently, researchers have shown great interest in developing earable applications. EarBuddy [33] is an on-face interactive system by recognizing sounds of on-face gestures. EarEcho [34] utilizes in-ear speaker and microphone to capture cannal echo for authentication. eBP [35] proposes a wearable system for blood pressure monitoring. While prior works have explored the use of microphone or physiological sensors in ear-worn devices, not many have explored the use of IMU on ear-worn devices for complex HAR. [36] investigates head motion tracking using the eSense [37] platform. MandiPass [38] leverages IMU to achieve continuous authentication function. Similarly, our system is built with low-cost and ubiquitous IMU which can be easily deployed in commercial earphones. In contrast, instead of concentrating on head or body activities, our earable device senses head-body composite activities simultaneously.

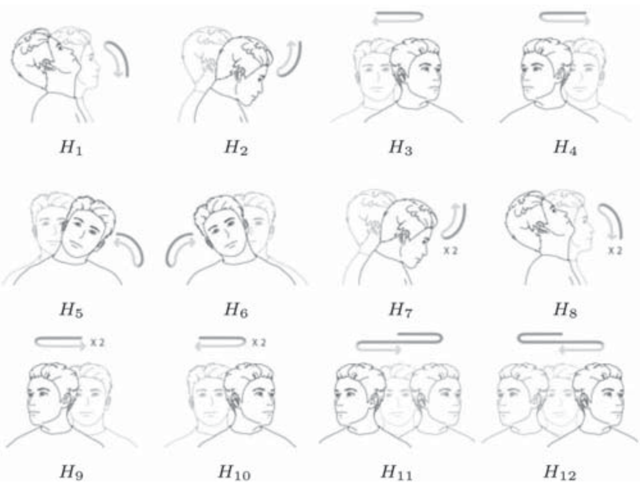


Fig. 2. The design of head gestures.

III. System Design

A. Composite Activities Taxonomy

In this paper, a composite head-body activity means the time aligned compound of a head gesture and a body movement. We refer to some HAR benchmark datasets of body movements recognition such as HHAR [24], UCI [25], MotionSense [26] and Shoaib [11], which have been commonly used in many works. Then we select 5 most common movements which are practical in various real-world scenarios and suitable for our single earable device interaction. To be specific, we consider going downstairs and upstairs, staying still, life-walking, jogging which are denoted by $B_1 \sim B_5$. As for the head part, we design a total number of 12 gestures as shown in Fig. 2 including raising head up and back for once and twice (H_1, H_8), nodding once and twice (H_2, H_7), shaking head to the left for once and twice (H_3, H_{10}), shaking head to the left first and then to the right (H_{11}), shaking head to the right for once and twice (H_4, H_9), shaking head to the right first and then to the left (H_{12}), leaning head to the left and right shoulders (H_5, H_6). The criteria of designing these head gestures is referred to previous works such as HeadGesture [8] and GlassGesture [9] since they are natural for users to perform. As a result, there are a total number of 60 composite head-body activities.

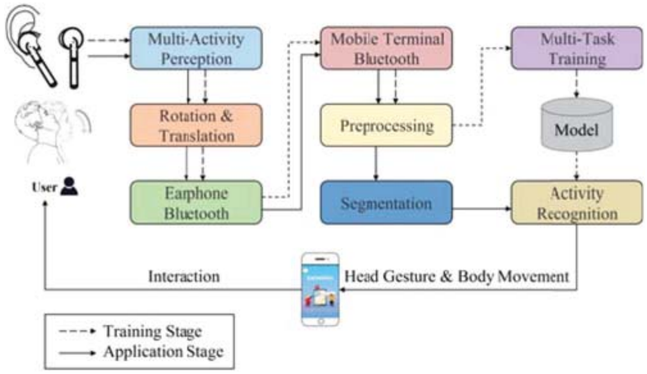


Fig. 3. The system overview of CHAR.

B. Overview

Fig. 3 shows the overview of training and testing pipelines of our system. When a user performs head gestures accompany with body movements, the earable device such as a smart earphone with an IMU sensor senses the rotational and translational movements, and transmits the data to a mobile terminal through Bluetooth. After that, signal instances corresponding to each composite activity are extracted precisely with an adaptive segmentation method. In the training stage, the extracted signal segments are used for building a self-designed multi-task learning network which is trained on a remote server. In the application stage, the trained network is deployed on a mobile device such as a smartphone, takes in signals of a composite activity, and outputs labels of a head gesture and a body movement simultaneously. According to different head-body composite activities, the system will accomplish HCI by controlling multimedia applications such as picking up phone calls, adjusting volume, and etc.. In the following, we shall give details of each key part of our system design.

C. Activity Segmentation

Activity segmentation refers to detecting the starting point (SP) and ending point (EP) of an activity. In our head-body activity recognition task, since body movement is periodic and persistent, we only need to acquire head gesture segment, which contains the composite head-body activity. Other previous works set a fixed energy threshold for activity segmentation which can not apply to different practical scenarios. In our work, we propose an adaptive segmentation method which dynamically adjusts the threshold according to different scenarios. As a result, user's head gestures with different body movements will be accurately segmented. The activity segmentation algorithm is shown in Algorithm 1. The following section gives details of this method.

Algorithm 1: Activity Segmentation Algorithm.

```
Input: 3-axis signal $Gyro = \{G_x, G_y, G_z\}$; Window
       length for EP and SP detection $L_{end}$, $L_{start}$;
       Framing stride $S$; Threshold for EP detection
       $\theta$; Scaling coefficient for SP detection $k$
Output: $SP$; $EP$
$W_{signal}$ = Frame($Gyro$, ($L_{end} + L_{start}$), $S$);
$W_{end}$ = $W_{signal}$($L_{start}$ :, :, :);
$W_{cont}$ = $W_{signal}$(: $L_{start}$, :, :);
for $i$ = 1 to FrameNum($W_{end}$) do
    for $j$ = 1 to ChannelNum($W_{end}$) do
        $E_{ch}(i, j)$ = Energy($W_{end}(i, j, :)$)
    end
    $ch(i)$ = FindMaxIndex($E_{ch}(i, :)$);
    $E_{end}(i)$ = Energy($W_{end}(i, ch(i), :)$);
    $E_d$ = Diff($E_{end}$);
    $EP$ = FindFirstIndexLess($E_d(i)$, $\theta$);
    if $EP$ is not None then
        $E_{cont}$ = Energy($W_{cont}(i, ch(i), :)$);
        $W_{start}$ = Frame($W_{end}(i, ch(i), :)$, $L_{start}$, $S$);
        for $j$ = 1 to FrameNum($W_{start}$) do
            $E_{start}(j)$ = Energy($W_{start}(j, :)$);
        end
        $SP$ = FindFirstIndexGreater($E_{start}$,
          $k \times E_{cont}$);
    end
end
return $SP$, $EP$
```

1) Signal Processing: In order to segment activity from data stream, we use 3 sliding windows W_{end}, W_{cont} and W_{start} to frame the signal with 1.0 seconds stride. Since W_{end} needs to cover entire activity for EP detection, the size of W_{end} is set to 4.0 seconds, which is slightly larger than the duration of all head gestures. W_{cont} captures the context information of the activity which follows W_{end} in time dimension and the size of W_{cont} is empirically set to 0.5 seconds. When the EP of an activity is detected, W_{start} further frames the current W_{end} with the same size of W_{cont}. After framing data stream, we calculate the average energy E by the formula as follows:

$$E = \frac{1}{T}\sum_{t}^{T}(x_t)^2 \tag{1}$$

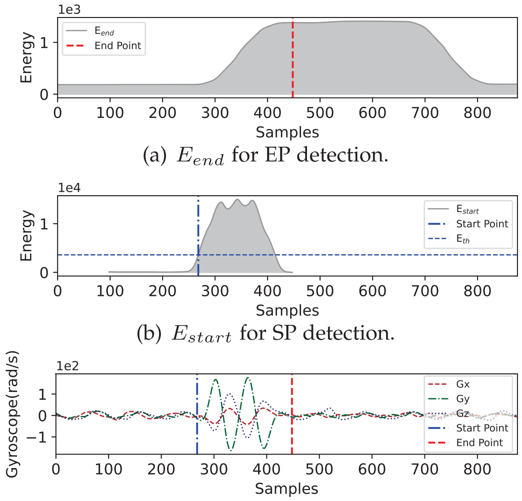
where x is the signal and t is the timestamp of x.

2) Channel Selection: In our system, we get 6 channels signal of the IMU. Since gyroscope varies significantly and its noise is relatively low, we select the signal of it as the original signal for activity segmentation. Different head gestures have different variations in the 3-axis of gyroscope. We need to get the best segmentation channel in real time. Therefore, We select the axis with the largest average energy as the best segmentation channel ch as follows:

$$ch = \arg\max_{i} E_i \tag{2}$$

where i is channel index and E_i is the energy of each channel.

3) EP and SP Detection: Fig. 4 shows an example of the activity segmentation process. We utilize E_{end} to denote the energy curve calculated by (2) for EP detection as shown in Fig. 4(a). When a user performs a head gesture, it first steps into W_{end} and then steps out. Since the size of W_{end} is set to be slightly larger than the duration of all head gestures, when a entire gesture enters W_{end}, it will not leave immediately but



(a) E_{end} for EP detection.

(b) E_{start} for SP detection.

(c) Origin signal and segmentation result.

Fig. 4. An example of the activity segmentation process.

remain in the window for a period of time. When a entire head gesture slides in W_{end}, no matter what kind of body movements is, the sensor has the same background noise and the variation of E_{end} is very small. Hence, E_{end} will first increase, then remain stable, and finally decrease as shown in Fig. 4(a). Once E_{end} remains stable, the right boundary of W_{end} is the EP of a head gesture as shown in the figure. The EP is detected when the variation of E_{end} is less than θ which is empirically set to 4.

Similarly, we utilize W_{start} for SP detection as shown in Fig. 4(b). Since W_{end} contains an entire activity, the SP will be included in it. When E_{start} is greater than a threshold E_{th}, the SP of an activity is detected. E_{th} is defined by kE_{cont} and act as a dynamic threshold, where k is a scaling factor and set to 10. By this means, E_{th} changes dynamically according to the background noise since W_{cont} captures the real-time context information. When a user has different body movements, the dynamic threshold E_{th} varies accordingly. After EP and SP detection, we calculate the midpoint of SP and EP, and extract a signal segment lasting for 3.0 seconds as shown in Fig. 4(c) in order to acquire fixed-length inputs for the network.

D. Data Augmentation

Conventionally, we need to collect a mass of data of 60 head-body activities for composite activity recognition, which brings challenges of data collection. First of all, compositing 12 head gestures and 5 body movements will generate up to $12 \times 5 = 60$ kinds of head-body activities. Moreover, it required a large amount of data to train the network because body movements generate less notable sensor measurements on earable device. To overcome these challenges, we propose some strategies to enable the system to achieve satisfactory composite activity recognition with insufficient data. In the following, we shall give details about these strategies.

1) Data Synthesis: As shown in Fig. 5, we synthesize data in two methods: splicing two head gestures and compositing head-body activity. With these methods, we can synthesize data for 60 composite head-body activities using only data from 6 singular head gestures and 5 singular body movements.

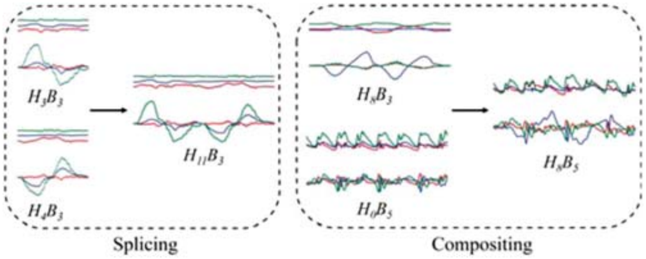


Fig. 5. Data synthesis examples. Each instance contains accelerometer (upper) and gyroscope (lower) measurements.

Head Gesture Splicing: Among the 12 designed head gestures, we respectively define the irreducible head gestures $H_1 \sim H_6$ as atomic head gestures and the pairwise spliced head gestures $H_7 \sim H_{12}$ as combined head gestures. Obviously, combined head gestures can be spliced by atomic head gestures. Specifically, atomic head gestures are spliced head-to-tail along the time dimension to synthesize combined head gestures. Fig. 5 on the left shows an example of splicing. While the body is B_3 (staying still), H_{11} (shaking the head to the left and then to the right) can be synthesized by splicing H_3 (shaking the head to the left for once) and H_4 (shaking the head to the right for once). In this way, we can synthesize all combined head gestures from the collected atomic head gestures, which relieves the burden of collecting data for some types of head gestures ($H_7 \sim H_{12}$).

Head-body Activity Compositing: As usual, we should collect data for all kinds of composite activities in order to recognize them, which creates a data collection burden. However, we propose composite singular head gestures and singular body movements to synthesize head-body activities. To be specific, the data of singular body movements is high-pass filtered and then be added to data of singular head gestures in time alignment. Fig. 5 on the right shows an example of compositing. B_3 and H_0 indicate that the body and head are staying still, respectively. Head-body activity H_8B_5 (raising head up and back for twice with jogging) can be synthesized by compositing singular head gesture H_8B_3 (raising head up and back for twice) and singular body movement H_0B_5 (jogging). In this way, we can synthesize all composite head-body activities from the collected data of singular head gestures and singular body movements. It relieves the burden compared to directly collecting data of composite activities.

2) Random Transformation: When IMU is used to characterize user activity, the differences in data are reflected in noise of the environment, intensity of activity and execution time of activity. First of all, the same activity shows diversity under different environmental noises. Taking the body movement of walking as an example, there is a difference between the sensor measurement when the user walks on flat ground and on rough ground. In order to increase the diversity of training data under environmental noise, we adds random noise to the training data to simulate different noise levels of activity. In addition, the same activity shows diversity among different users or different

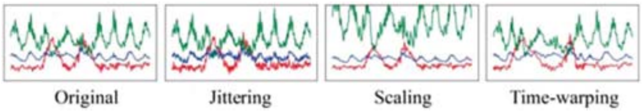


Fig. 6. Augmentation example using jitting, scaling and time-warping. Each instance only show accelerometer measurements for brevity.

physical states. Whether it is head gesture or body movement, when different users or the same user in different physical states complete the same activity, the execution intensity and execution time of the activity will be different. In order to increase the diversity of training data in the execution process, we randomly stretches and compresses the execution intensity and execution time of the training data to simulate different activity intensity and time variance.

To be specific, we use several ramdom transformation techniques to augment composite head-body activity instances. Fig. 6 show that three time series data augmentation techniques [39] are used to enlarge the data size: 1) jittering, to simulate different noise level, with the noise intensity $j \sim \mathcal{N}(0, 0.05^2)$; 2) scaling, to simulate different activity strength, with the scaling factor $s \sim \mathcal{N}(1, 0.3^2)$; 3) time-warping, to simulate activity temporal variance, with warping randomness $w \sim \mathcal{N}(1, 0.3^2)$. Fig. 6 shows a example using these three augmentation techniques. Compared with the original instance, the augmented instances show diversity based on the underlying pattern. The augmented results well simulate noise level, activity strength and temporal variance of the composite activity.

3) Generative Adversarial Network: Since body movements cause less notable sensor measurements, it requires more training data to recognize them. When users wear earable device, they will always continue to perform body movements such as walking, going up and down stairs, which generates a large amount of unlabeled body movement data. It is obviously more difficult to obtain labeled body movement data than unlabeled data. The training of recognition networks relies more on labeled data, which brings the burden of data collection. Fortunately, previous work provides a large number of data sets for IMU to recognize body movements. In order to reduce the burden of data collection, we set our sights on public datasets, hoping to obtain the large amount of data needed to supplement body movement recognition. However, the pattern of the same body movement varies greatly depending on the angle or position of the IMU sensor. In other words, recognition networks trained using public datasets do not perform well on our earable device. Therefore, public datasets cannot be used directly to train recognition networks.

In order to solve the above problems, we use public dataset and CHAR's easily accessible unlabeled data to train a generative adversarial network to achieve style transfer. Style represents the sensor placement angle and position, and style transfer eliminates the measurement value differences caused by different styles. Specifically, our goal is to learn mapping functions $G : X \rightarrow Y$ and $F : Y \rightarrow X$ between public dataset domain X and our dataset domain Y. To achieve this goal, we train a style transfer generative adversarial network CycleGAN [40] with unlabeled data from these two domains. After that, mapping function G transfer X to X', which has similar pattern to Y and serves as a supplement to our body movement dataset. Since the original CycleGAN is used for image style transfer, we modify its backbone for time-series. To be specific, we adopt a 16-unit 1-layer Bi-LSTM before a fully-connected layer as the generator, and 3 fully connected layers as the discriminator. Fig. 7 shows some transfer examples. As we can see, the mapping function G well transfer public dataset domain X to X', which closely matches the pattern of our dataset domain Y.

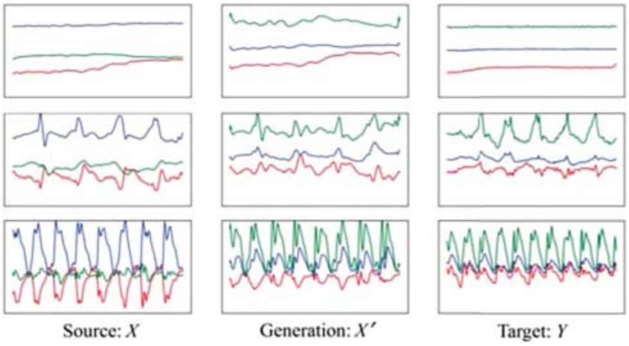


Fig. 7. CycleGAN transfer examples. Each instance only show accelerometer measurements for brevity. The top is staying still (B_3), the middle is going upstairs (B_2) and the bottom is jogging (B_5).

E. Network Design

1) Multi-Task Learning: To deal with a multi-task problem, the most intuitive idea is to split it into several independent single-task subproblems that are solved separately, and then combine the results together. However, this method has deficiency for two reasons. For one thing, there is shared information between head gestures and body movements. Making good use of the shared information is beneficial for extracting features of each task. For another thing, head gestures and body movements recognition are not entirely independent of each other. Taking full advantage of the correlations between them by joint training facilitates the decision making of each task. As a result, we turn our attention to multi-task learning.

Multi-task learning is a training paradigm where learning model is trained with data from multiple tasks simultaneously. Inspired by the insight of multi-task learning, we design a composite activities recognition network (CARN) for sporadic head gestures and periodic body movements recognition. In evaluation section, we compare the performance of some classical single-task networks for HAR and our multi-task network. The experimental results show the superiority of our network, which indicates that multi-task learning plays an important role in composite activities recognition. Fig. 8 shows the architecture of CARN, which consists of shared bottom block and specific top blocks.

2) Design of Shared Bottom Block: The bottom layers are shared across tasks to extract task-shared representation which facilitates both head gestures and body movements recognition. This kind of shared structure substantially reduces the risk of

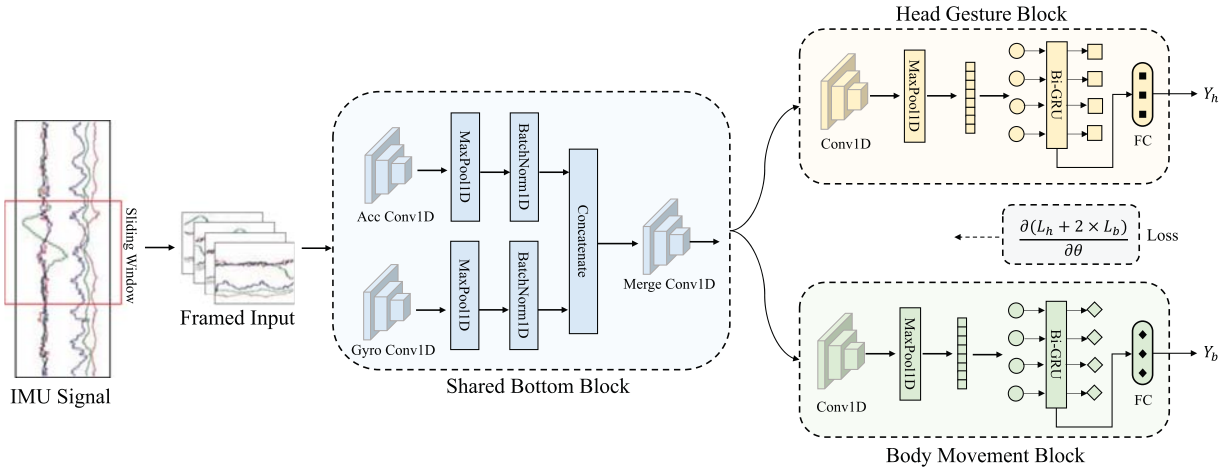


Fig. 8. The architecture of our composite-activity recognition network.

overfitting. For example, the contribution of head gesture task to the aggregated gradient in the back-propagation process can be regarded as noise for the body movement task. In other words, one task can be used as a noise source to avoid overfitting of the other one. In a nutshell, the shared bottom block improves generalization performance by learning head gesture and body movement recognition tasks in parallel using shared representation.

Inspired by LRCN [41], we treat the 3.0 seconds length segment as frame-by-frame data. To be specific, we use a 0.5 seconds length sliding window to frame the segments with 0.1 seconds stride and then feed all frames to the network. To begin with, the signals of each sensors have different signal-noise patterns and representation capability. Therefore, we need to extract intra-modality representation across time and channels of each sensor separately. Meanwhile, we pay attention to the correlation between different modalities. Thus, we further extract cross-modality representation across different sensors. These two operations avoid mutual interference between the two modalities. For implementation details, we utilize two different 1D-CNN to extract the spatial features of accelerometer and gyroscope seperately. Max pooling and batch normalization are applied on each output of 1D-CNN. After that, we concatenate the intra-modality features along channel dimension and feed them to a 1D-CNN for multi-modality fusion and cross-modality features extraction. In this way, we extract intra-modality and cross-modality spatial features as shared representation for specific tops.

3) Design of Specific Top Blocks: The specific top layers are explicitly utilized to excavate specific and fine-grained representation of each task. Since specific tasks aim to recognize diverse kind of activities, we separately adopt two different feature extractors to acquire task-specific representations. The reason is that different tasks favor different spatial and temporal information. Specifically, head gestures and body movements recognition tasks prefer to rotation and translation spatial information separately. Besides, it is obvious that head gestures and body movements are sporadic and periodic signals, respectively. Consequently, these two recognition tasks also require different temporal information.

The specific tops consist of head gesture block and body movement block. To extract specific and deeper representations, we employ two separate 1D-CNN to extract the spatial features of rotation and translation for different tasks. Max pooling is applied to reduce the dimensionality along time. Afterwards, we flatten the outputs and empirically adopt a 256-unit 1-layer Bi-GRU and a 128-unit 1-layer Bi-GRU to profile the temporal relationships for head gestures and body movements, respectively. It is worth mentioning that the time steps of Bi-GRU corresponds to the input frames. Finally, we feed the last time step of Bi-GRU into the fully-connected layers of specific task blocks to get Y_h and Y_b. We separately define cross-entropy loss L_h and L_b for head gesture and body movement task. At multi-task learning, we use $L = L_h + \lambda L_b$ for joint training, where λ is empirically set to 2.

IV. Implementation and Experiments

In this section, we describe the implementation of CHAR and give experiment details about how data are collected and generated for corresponding evaluation.

A. Hardware and Software

As commercial earphones have no access to sensor's data, we build a prototype with low-cost hardware including a JY901 IMU, an ESP32 as the microcontroller, a 3.7 V lithium battery as the power supply, and a 3D printed plastic enclosure shaped like an earphone as shown in Fig. 9. The JY901 module consists of an accelerometer, a gyroscope, and a magnetometer with sampling rate set to be 100 Hz during experiments. The ESP32 controls IMU data collection and transmission to a mobile device (e.g., a smartphone) via Bluetooth. We have also developed an Android application on a mobile device which is responsible for receiving data, segmenting activities, and running the recognition model, following the pipeline as introduced Section III. Specifically, after training the network on a server with an Intel(R) Xeon(R)

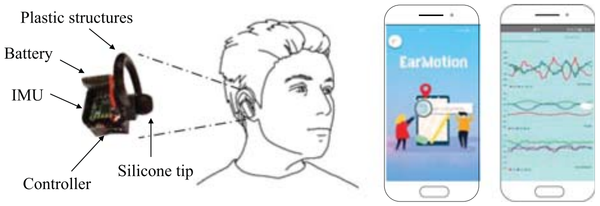


Fig. 9. The hardware and mobile application of CHAR.

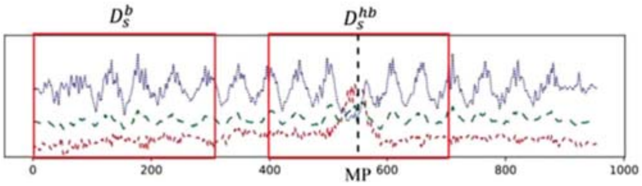


Fig. 10. Instance partition example. The instance only show accelerometer measurements for brevity.

Platinum 8260 CPU and NVIDIA GeForce RTX 2080Ti GPU, we deploy it along with the activity segmentation algorithm (see Section III-C) on a HUAWEI Mate 40 Pro through Chaquopy project. The smartphone has a HI-Silicon Kirin 9000 CPU, 8 GB RAM, and 256 GB ROM. Based on the output of the recognition model, developers can design different HCI applications.

B. Data Collection

To evaluate CHAR, we recruit 15 participants denoted by $P_1 \sim P_{15}$ aged from 18 to 25 years old. Before experiments, we instruct them to use our system and tell them necessary details. We then request each of them to perform every composite head-body activity for 5 times, and finally get a total number of 4500 (i.e., $15 \times 12 \times 5 \times 5$) raw instances, since there are 12 head gestures and 5 body movements as aforementioned. For each raw instance, we annotate the SP and EP for a composite activity.

After that, we further partition the raw instances. To be specific, we calculate the midpoint (MP) of SP and EP for each raw instance and use a 3.0 seconds non-overlapping window to obtain corresponding datasets. Fig. 10 shows a partition example. It is obvious that the window contains a whole composite head-body activity when the midpoint divides the window evenly, so we obtain self-collected head-body activity dataset D_s^{hb}. Since sporadic head gesture performs during periodic body movement, the non-overlapping windows at two ends contain singular body movement. The data from these windows make up self-collected singular body movement dataset D_s^b. In addition, when the body movement is B_3 (staying still), composite activity essentially degenerates into singular head gesture. Therefore, we extract singular head gesture dataset D_s^h from D_s^{hb}. It is worth mentioning that we only retain 6 atomic head gestures, which can synthesize 6 combined head gestures.

In addition to the self-collected dataset, we also used a public dataset RealWorld [42]. It covers acceleration and gyroscope data of the activities climbing stairs down and up, jumping, lying, standing, sitting, running/jogging, and walking of fifteen subjects. For each activity, it recorded simultaneously the body positions chest, forearm, head, shin, thigh, upper arm, and waist. We crop RealWorld to preserve 5 body movements consistent with CHAR. Similarly, we get public singular body movement dataset D_p^b using a 3.0 seconds non-overlapping sliding window.

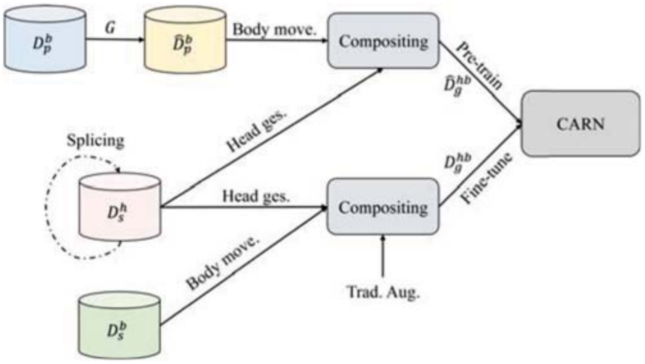


Fig. 11. The pipeline of data generation.

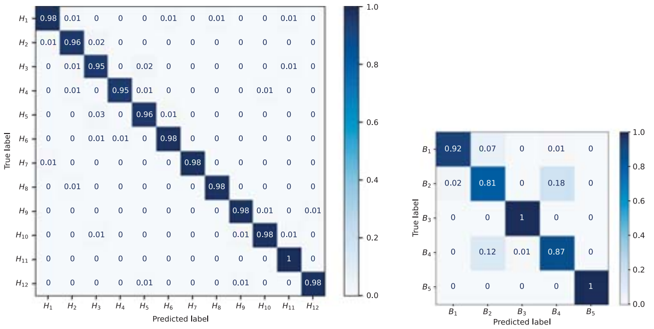
In a word, we acquire datasets as following: 1) D_s^{hb}: self-collected dataset with 60 composite head-body activities; 2) D_s^h: self-collected dataset with 6 singular atomic head gestures; 3) D_s^b: self-collected dataset with 5 singular body movements; 4) D_p^b: public dataset with 5 singular body movements.

C. Data Generation

In this part, we generate composite head-body activity data using strategies and datasets mentioned in Sections III-D and IV-B, respectively. Fig. 11 shows the pipeline of data generation. First of all, we train CycleGAN to learn the mapping function G from unlabeled D_p^b to unlabeled D_s^b and then transfer D_p^b to $\hat{D}_p^b$ as a supplement. Specifically, we use the sensor data of $P_1 \sim P_3$ on head position of RealWorld dataset as D_p^b. The reason for selecting $P_1 \sim P_3$ is that they wear the sensors at similar angles on their heads. Since D_p^b has label, $\hat{D}_p^b$ retains the original label with the pattern being transfered, which can complement D_p^b. After that, we randomly select $14 \times 6 \times M_s^h$ instances of D_s^h and $14 \times 5 \times M_s^b$ instances of D_s^b, since there are 14 training users, 6 atomic head gestures and 5 body movements. The values of M_s^h and M_s^b are set relatively small in the corresponding evaluation to simulate the insufficient data scenario. Next, we apply splicing synthesis to extend D_s^h to all 12 head gestures, and apply compositing synthesis to obtain generation datasets $\hat{D}_g^{hb}$ and D_g^{hb}, i.e., $D_s^h + \hat{D}_p^b \rightarrow \hat{D}_g^{hb}$ and $D_s^h + D_s^b \rightarrow D_g^{hb}$. Finally, we utilize random transformation techniques on D_g^{hb}. The generation dataset $\hat{D}_g^{hb}$ and D_g^{hb} are used to pre-train and fine-tune CARN, respectively.

V. Evaluation

In this section, we comprehensively evaluate CHAR under two setup, including training with self-collected data and training with generation data. In the following, we shall demonstrate the performance of each evaluation.



(a) Head gestures. (b) Body movements.

Fig. 12. Confusion matrices of head gestures and body movements recognition in user-independent scenarios.

A. *Training With Sufficient Data*

1) Evaluation Setup: Similar to other works on HAR, we build a self-collected composite activity dataset D_s^{hb} containing all 60 head-body activities. Then we train CHAR with these sufficient data and evaluate its performance. Each instance in the dataset contains both a head gesture and a body movement. In the following evaluation, we train and test CHAR using D_s^{hb}. Besides, we consider user- dependent and independent cases. In the former case, we train and test the system with 15-fold cross-validation method. While in the latter one, we train and test it with the 'leave-one-user-out' strategy. In both cases, the results are averaged over five testing rounds.

2) Overall Performance: Activity Segmentation: We manually segment signal sequences by marking SPs and EPs of activities as ground truths, and compare them with the results obtained by our proposed segmentation method (i.e., Algorithm 1). We utilize missed detection rate (MDR) and false detection rate (FDR) as the metrics for evaluation. Formally, MDR is defined as the percentage of missing activities that are not detected by the algorithm. FDR is defined as the percentage of mistakenly detected activities outside the ground truth. Experiments show that the MDR and FDR with Algorithm 1 are 1.8% and 1.2%, respectively. This means that more than 97.0% of head gestures can be detected correctly. In addition, the differences between segmentation results and the ground truth will also affect the final recognition performance of CHAR. We will evaluate this point in *Cascade Performance*.

Activity Recognition: In the user- dependent and independent cases, CHAR recognizes composite activities with high accuracies of 97.0% and 89.7%, respectively. Since users prefer to use a system immediately without retraining, we set the user-independent condition as our baseline case in the following evaluation. Fig. 12 shows the confusion matrices that CHAR recognizes head gestures and body movements in the baseline case. As we can see, the recognition accuracies of head gestures and body movements reach 97.7% and 92.0%, respectively. This indicates that CHAR can recognize composite head-body activities with high accuracy. This is mainly due to our designed network which decouples their relationship. In addition, we can also see that walking (B_1) and going upstairs (B_2) are easier to be confused, which is mainly originated from user diversity. There is user diversity of some activities in HAR, which leads to poor performance in user-independent recognition. We carefully check these data of walking and going upstairs. It was found that these two movements are relatively similar across different users. First, walking and going upstairs produce periodic signals with similar periods, which are longer than going downstairs and jogging. Then, walking on rough ground tends to produce z-axis acceleration signal analogous to that of going upstairs, which leads to confusion between these two movements. Nevertheless, accuracies of CHAR on body movements recognition is still up to 98.0% and 92.0% in user- dependent and independent situations, which is sufficient for practical needs.

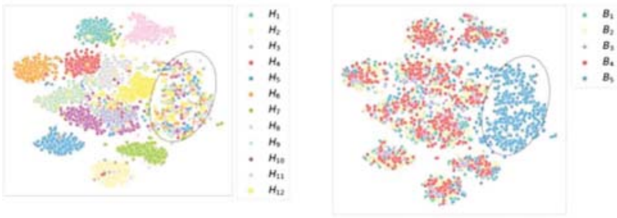
TABLE I
THE RECOGNITION ACCURACIES WITH DIFFERENT DATASETS

Training / Testing	Head gesture	Body movement	Composite activity
D_h / D_h	97.7%	92.0%	89.7%
D_h / D_a	93.7%	91.9%	86.3%
$\boldsymbol{D_a}$ / $\boldsymbol{D_a}$	**94.9%**	**89.8%**	**85.4%**

Cascade Performance: According to Fig. 3, segmentation is followed by the recognition network. Hence, its result has impact on the network's classification performance. To evaluate this, we consider two segmentation methods, namely, by Algorithm 1 and by human (i.e., ground truth). Correspondingly, we denote data obtained from D_s^{hb} by these two method as D_a and D_h, respectively. Then we train and test the network with different combinations of D_a and D_h as shown in Table I and get corresponding results. We can see that using D_a as the testing set causes accuracy drops of 4.0% and 0.1% for head gestures and body movements recognition, respectively. This indicates that our segmentation algorithm has some impact on head gestures, but has little on body movements. This is because some instances of the combined head gestures (such as H_7, H_9, H_{11}) are disjointed, so that the algorithm segments part of these gestures. Partial segmentation leads to recognition error because head gesture is sporadic. Differently, body movement is periodic, and slight segmentation differences have little affect on recognition. Besides, we explore the possibility of directly annotating instances using Algorithm 1 without human effort, by training and testing the model with D_a. It can be seen that CHAR can recognize head gestures and body movements with accuracies of 94.9% and 89.8%, respectively. Compared with the D_h/D_h case, the accuracies decrease by 2.8% and 2.2%, respectively. It validates that with the proposed segmentation method, CHAR can still reliably detect the SP and EP of an activity instance, and achieve very close performance to that of ground-truth segmentation. Hence, our proposed system can perform well in practical application scenarios and reduce the burden of annotating data.

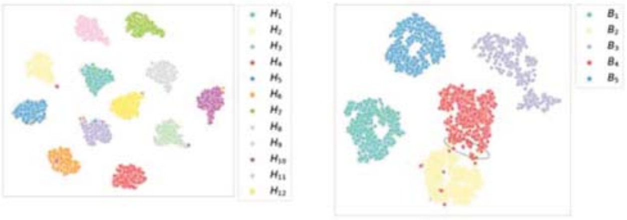
3) Embedding Visualization: To better dissect the internal mechanism of multi-task learning, we show the embeddings of CARN in two-dimension space with t-SNE algorithm.

Shared Bottom Embedding: Fig. 13 shows the shared feature representations extracted by the shared bottom block. Note



(a) Colored by head gestures. (b) Colored by body movements.

Fig. 13. Embedding visualization with t-SNE algorithm for features extracted by Shared Bottom Block of CARN. The figure corresponds to the same features and has been colored in terms of different types of activities.



(a) Head gesture embedding. (b) Body movement embedding.

Fig. 14. Embedding visualization with t-SNE algorithm for features further extracted by Head Gesture Block (a) and Body Movement Block (b) of CARN, respectively.

that feature representations shown in Fig. 13(a) and Fig. 13b are totally the same but are colored in terms of head gestures and body movements, respectively. Fig. 13(a) shows that head gestures are grouped at a coarse-grained level except for the circled data points. This means that the shared bottom block can extract distinct features for most instances. Samples mixed together in the circled zone represent jogging movements as shown in Fig. 13(b). This is because gentle body movements such as being still, walking, going upstairs and downstairs have relatively slight effect on the measurements of head gestures. But when users are jogging, the IMU measurements of head gestures tend to be overwhelmed by it. Accordingly, from Fig. 13(b), we can find that only the jogging instances are well grouped and other body movements are easily mixed together after shared bottom block. The reason is that head gestures seriously deform periodic signals generated by body movements. What is more, periodic signals generated by body movements require a model to extract temporal features, while the shared bottom block only uses CNNs to extract shared spatial features. In order to better group body movements, we use GRUs to extract specific time-domain information following the shared body bottom.

Specific Task Embedding: Fig. 14 shows the feature representations of head gestures and body movements extracted by corresponding blocks. We can observe from Fig. 14(a) that all head gestures (including in jogging movement) are well grouped. Moreover, compared to the results shown in Fig. 13(a), the distances between different categories is obviously larger. It means that the head gesture block extracts unique features for better recognition on the basis of shared feature block. Similarly, it can be seen from Fig. 14(b) that, except for a few data points in the circled zone, the body movements can be well grouped. It indicates that even though the shared bottom block cannot directly group some of the body movements, the relevant part of the network has the ability to extract unique periodic features corresponding to body movements.

TABLE II
ACCURACY OF BENCHMARKS AND OUR NETWORK

Network	Head gesture	Body movement	Composite activity
DCNN	91.2%	81.0%	73.4%
DeepSense	92.4%	76.5%	69.8%
DeepConvLSTM	88.1%	84.9%	74.4%
CARN	**97.7%**	**92.0%**	**89.7%**

4) Comparison With Benchmarks: In this section, we compare our proposed network CARN with several classical networks for HAR as benchmarks including DCNN [22], DeepSense [23], and DeepConvLSTM [15]. DCNN [22] designs a convolutional neural network to learn features from raw IMU input signals which outperforms traditional machine learning methods such as support vector machine and deep belief network. DeepSense [23] integrates CNNs and RNNs to combine different sensing modalities of mobile sensors and extract temporal features for HAR. DeepConvLSTM [15] proposes a deep activity recognition framework composed of convolutional and LSTM recurrent layers, which is suitable for the recognition of static/periodic activities and sporadic activities. As these networks deal with single-task HAR, we train them for head gestures and body movements recognition tasks separately. In contrast, CARN is trained for both tasks simultaneously. Also, in order to make our dataset suitable for those networks, we slightly modify their input and output layers.

Table II gives comparative results of CARN and other benchmarks in both recognition tasks. From our previous analysis, it can be known that sporadic head gestures can be well recognized by only using CNNs to extract spatial features. Since head gestures are accompanied by periodic body movements, using RNNs to extract temporal features will interfere with head gestures recognition instead. Therefore, the head gestures recognition accuracy of DCNN reach 91.2% with CNNs only. On the contrary, DeepConvLSTM using RNNs has the worst performance on the head gestures recognition task, but its accuracy of body movements recognition reaches 84.9%. In addition, it can be seen that CARN outperforms benchmarks in both head gestures and body movements recognition tasks. Accuracies of head gestures and body movements recognition tasks are 5.3% higher than DeepSense and 7.1% higher than DeepConvLSTM, respectively. This is because irrelevant task will interfere the target task, which results in poor recognition performance. However, our designed multi-task learning framework fully learns the commonalities of two tasks and then decouples them to learn the differences. It reduces the risk of overfitting and improves generalization ability of recognition model.

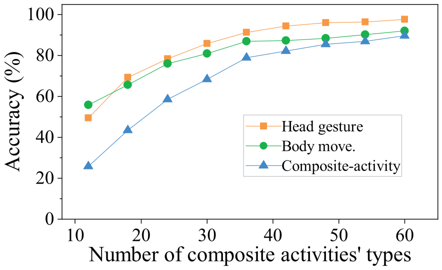


Fig. 15. Recognition accuracies with different number of training activity types.

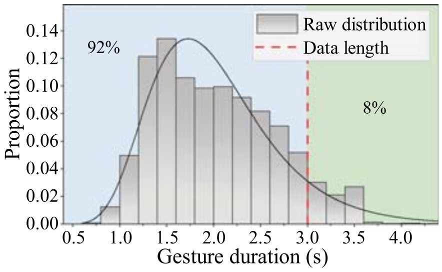


Fig. 16. Statistical histogram of head gestures performing time.

5) Impact of Composite Activity Types: Without specification, we train the network with all types of head-body composite activities (i.e., 60 types of composite activities). A natural question is whether the network can be trained with a part of composite activities, but can still recognize the whole set accurately. The intuitive rationale is that unseen composite activities are also composed of the same basic head gestures and body movements which appear in the training activities. To quantify the impact of the number of composite activities, we randomly select a certain number of composite activities types in the training set, and test CHAR's performance on the whole set. We vary the number of composite activities types in the training set from 12 to 60, and obtain the results as shown in Fig. 15. It can be seen that as the number of composite activities increases, the recognition accuracy first increases and then remains stable. The recognition performance of the system tends to be stable when more than 36 composite activities are used for model training. Experiments demonstrate that the network can be trained using data from some composite activities and applied to all, thereby reducing the burden of data collection.

6) Impact of Data Length: As mentioned in Section III-C, after detecting the EP and SP of an activity, we extract a fixed-length data segment and feed it into the network. The data length is a critical parameter for recognizing activities accurately. Intuitively, it should be large enough to cover an activity as much as possible. To determine its proper value, we first obtain a statistical histogram of head gestures' durations as shown in Fig. 16. It can be seen that more than 90% of the head gestures last less than 3.0 seconds, which indicates a proper range of data length. Although a longer segment contains more information of activities, it also brings about heavier computational overhead and more energy consumption. To achieve good trade-off, we test CHAR's performance with different data lengths from 1.0 s to 4.0 seconds. As shown in Fig. 17, the recognition accuracies rise with the data length. But the improvement is very limited after the data length exceeds 3.0 seconds. In addition, only about 50% of the head gestures duration are less than 2.0 seconds from Fig. 16. However, setting the data length to 2.0 seconds can also achieve 96.1% and 88.7% recognition accuracies for head gestures and body movements, respectively. As a result, for mobile devices with limited resources, we suggest appropriately reducing the data length to ease computing overhead and energy consumption.

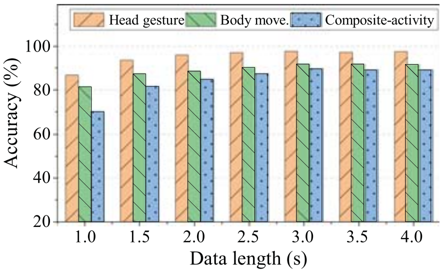


Fig. 17. Recognition accuracies with different data lengths.

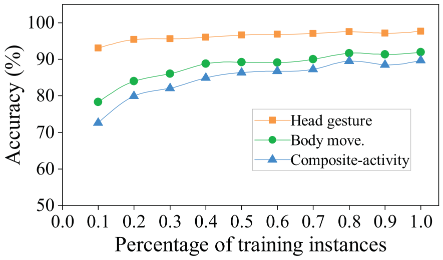


Fig. 18. Recognition accuracies with different training dataset sizes.

7) Impact of Data Size: We also evaluate the impact of training dataset size by randomly sampling a whole dataset at different percentages. Since the evaluation is conducted with 'leave-one-user-out' strategy, a whole dataset contains 4200 instances in total. Fig. 18 shows the results. With the percentage increasing, the accuracies of recognizing head gestures, body movements, and composite activities initially increase fast and then remain relatively stable, after the percentage exceeds 0.8. It means that with 80% of the dataset, CHAR's performance is close to the optimal. Moreover, compared with body movements, head gestures can be recognized more accurately with less training data. This is because IMU measurements caused by head gestures are more obvious than those body movements. As a result, even a small amount of data enables the model to learn feature patterns of head gestures. In contrast, body movements cause less notable sensor measurements which are easier to be interfered. Hence, it requires more training data to recognize body movements (and thus composite activities) with high accuracy.

8) Impact of Sensory Signal: We use two kinds of sensory signals in our system, i.e., accelerometer and gyroscope signals. It is obvious that these two signals are significant in capturing

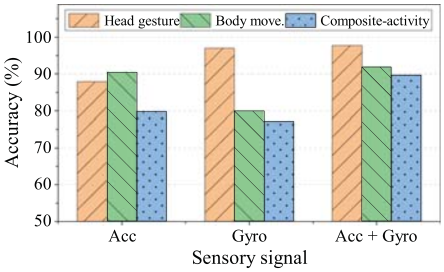


Fig. 19. Recognition accuracies with different sensory signals.

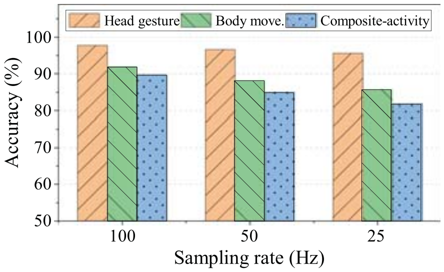
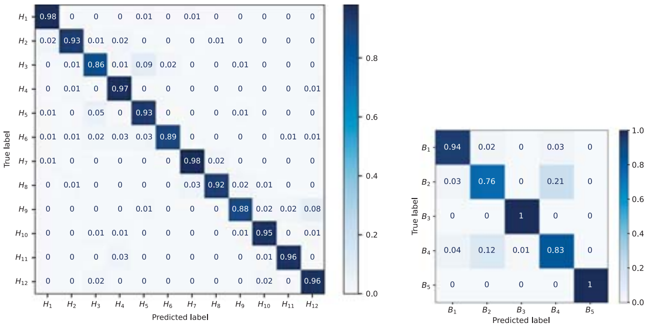


Fig. 20. Recognition accuracies with different sampling rates.

composite activities. However, accelerometer and gyroscope make different contribution to head gestures and body movements recognition. Therefore, we evaluate the recognition accuracies with different sensory signals as shown in Fig. 19. Accelerometer and gyroscope separately concentrate on capturing movement and rotation. Therefore, a single accelerometer and a single gyroscope recognize body movements and head gestures with satisfactory performance, respectively. In other words, these two sensory signals complement each other for recognizing composite head-body activities. Since our recognition network is designed with a multi-modal fusion module, using these two sensory signals simultaneously achieve the best recognition performance both in head gestures and body movements.

9) Impact of Sampling Rate: Intuitively, a higher sampling rate results in more precise measurement of activities. But it also brings about heavier computational overhead and larger energy consumption. To evaluate the impact of sampling rate, we downsample the collected data from 100 Hz to 50 Hz and 25 Hz, and test the recognition accuracies of CHAR in each case. Fig. 20 shows the results. We can see that the accuracies of recognizing head gestures remain nearly the same with different sampling rates. The underlying reason is that gyroscope measurements are sensitive to head gestures which are in low frequency range. As for body movements, with the sampling rate decreasing, the recognition accuracies drops more obviously. This is because the frequency of body movements is relatively large and information loss will happen due to downsampling. Specially, when the sampling rate is 50 Hz and 100 Hz, the corresponding recognition accuracies of body movements are 88.2% and 92.0% respectively. Therefore, for mobile devices with limited resources, we suggest appropriately reducing the sampling rate of hardware to ease computing overhead and energy consumption.



(a) Head gestures. (b) Body movements.

Fig. 21. Confusion matrices of head gestures and body movements recognition.

B. Training With Insufficient Data

1) Evaluation Setup: In the following evaluation, we study whether CHAR shows comparable composite activity recognition performance even training with insufficient data. Since recognizing composite activity requires a large amount of data, we consider data augmentation strategies. To be specific, we generate dataset $\hat{D}_g^{hb}$ and D_g^{hb} through the pipeline mentioned in Section IV-C using a small amount of data. After that, we pre-train and fine-tune CARN with $\hat{D}_g^{hb}$ and D_g^{hb}, respectively. Finally, we test CHAR with genuine self-collected dataset D_s^{hb} and only consider the more challenging user-independent cases for simplicity. Without special instruction, the following evaluation is based on this pipeline.

2) Overall Performance: To study the insufficient data scenario, we set $M_s^h = 1$ and $M_s^b = 2$, which means using a small amount of labeled data to generate data and train the network. Fig. 21 shows the confusion matrices that CHAR recognizes head gestures and body movements in user-independent case. As we can see, the recognition accuracies of head gesture and body movement reach 94.0% and 90.8%, respectively. It demonstrates that CHAR recognize composite activities with high accuracy training with generation data. In this case, we only need to provide $14 \times 6 \times 1$ labeled instances of D_s^h, $14 \times 5 \times 2$ labeled instances of D_s^b and the rest unlabeled instances of D_s^b. With a small amount of labeled singular activity data and easily accessible unlabeled body movement data, CHAR can recognize 60 kinds of composite head-body activities with accuracy of 85.1%. In addition, we compare the recognition accuracies of different types of activities under training with self-collected data and generation data as shown in Fig. 23. Training with generation data, i.e., with a small amount of labeled data, the recognition accuracy of composite activity is only sacrificed by 4.6%. Among them, the accuracies of head gesture and body movement decrease by 3.7% and 1.2%, respectively. Therefore, CHAR can achieve satisfactory composite head-body activity recognition even with insufficient data.

3) Effectiveness of Data Augmentation: To achieve comparable composite activity recognition with insufficient data, we propose some strategies to generate sufficient training data. In this part, we evaluate the effectiveness of data augmentation strategies, including data synthesis random transformation and

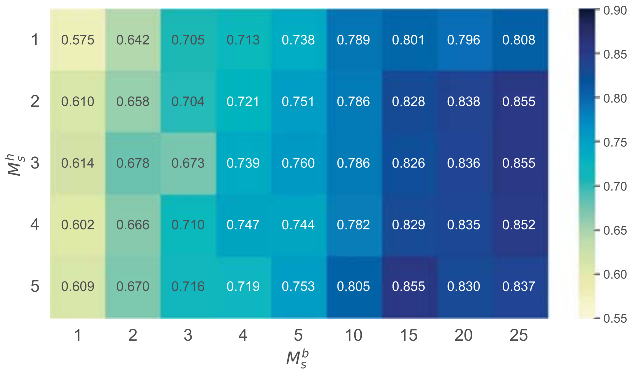


Fig. 22. Composite head-body activity recognition accuracy using only data synthesis strategy with different M_s^h and M_s^b.

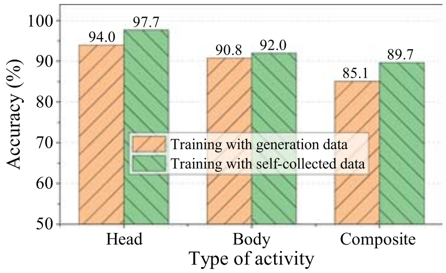


Fig. 23. Recognition accuracies under training with generation data and self-collected data.

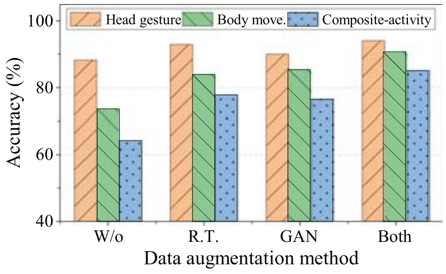


Fig. 24. Recognition accuracies under different augmentation methods.

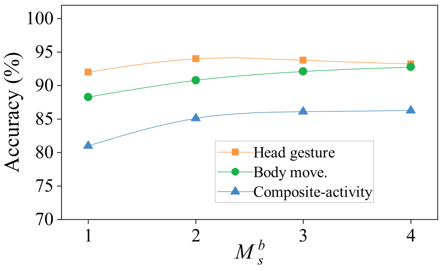


Fig. 25. Recognition accuracies using both data augmentation strategies with different M_s^b.

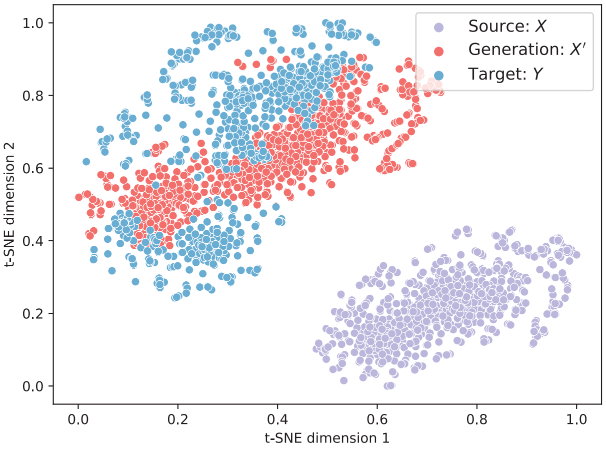


Fig. 26. The visualization of embeddings of different datasets with t-SNE.

generative adversarial network. In the first place, we train CARN using only data synthesis strategy with different M_s^h and M_s^b. From Fig. 22, it can be seen that composite head-body activity recognition accuracy comes to 85.5% when the singular activity data is sufficient. In other words, the performance of CHAR training with synthesis data is only sacrificed by 4.2% compared to that with self-collected composite activity dataset D_s^{hb}. Above results demonstrate that the data synthesis strategy can effectively generate data resembles genuine composite activity. Since composite activity data is synthesized from singular activity data, it reduces the demand for the data size to a certain extent.

To further reduce the burden of data acquisition, based on data synthesis strategy with $M_s^h = 1$ and $M_s^b = 2$, we apply data augmentation strategy and evaluate its effectiveness. Specifically, we train CARN under different augmentation methods: without augmentation, only random transformation(R.T.), only GAN-based augmentation, and both augmentations. We can see from Fig. 24 that the accuracies of recognizing composite activity in the above four cases are 64.2%, 77.9%, 76.5%, and 85.1%, respectively. In other word, random transformation and GAN-base augmentation separately bring accuracy gain of 13.7% and 12.3% in composite activity recognition. Moreover, the accuracy gain is further increased up to 20.9% when apply two augmentation strategies simultaneously. These results verify that the data augmentation strategy obtains high-quality data for the training of CARN, thereby improving the recognition accuracy. In a word, the proposed data augmentation strategies enable CHAR to achieve better performance with insufficient data.

In order to demonstrate the effectiveness of CycleGAN, we utilize t-SNE technique [43] to reduce the dimensionality of the public dataset (Source: X), the generated dataset (Generation: X'), and the self-collected dataset (Target: Y). The result is shown in Fig. 26. Intuitively, we can see that compared with X, the embeddings of X' are much closer to those of Y, which proves the effectiveness of our data generation method. To further quantify the effectiveness, we also calculate the Jensen-Shannon divergence [44], a metric used to measure the similarity between two probability distributions, between these two pairs of data distributions, namely, X and Y, X' and Y. The similarity between two distributions is greater when the JS divergence is closer to zero. The JS divergence of X and Y, X' and Y is 0.79 and 0.21, respectively. Since X' and Y are similar but not exactly the same, CycleGAN has good generalization capability.

4) Impact of Data Size: As aforementioned, we randomly select $14 \times 6 \times M_s^h$ instances of D_s^h and $14 \times 5 \times M_s^b$ instances of D_s^b to generate data for network training. To study

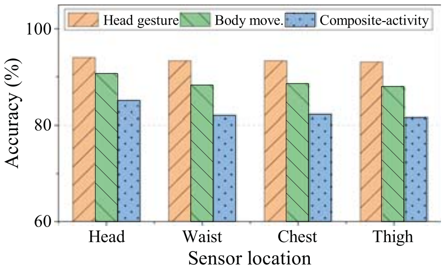


Fig. 27. Recognition accuracies with different sensor location on public dataset.

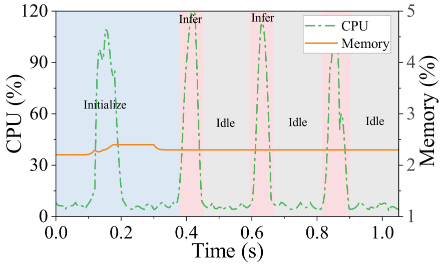


Fig. 28. CPU and memory occupation.

the insufficient data scenario, we evaluate the impact of M_s^h and M_s^b on CHAR's performance. In fact, since head gestures can be recognized more accurately with less training data compared with body movements, we can set M_s^h to remain constant at 1 and set M_s^b from 1 to 4. Fig. 25 shows the recognition accuracies with different M_s^b. As we can see, the recognition accuracies rise with M_s^b in accordance with expectation. But the improvement is very limited when M_s^b exceeds 2. In other word, setting M_s^b to 2 is a trade-off proposal, which balances the requirements between data size and performance. In addition, when $M_s^b = 1$, the recognition accuracies of head gesture and body movement can also reach 92.0% and 88.3%, respectively. It means that CHAR can also meet the needs of HAR performance even with insufficient data.

5) Impact of Sensor Location on Public Dataset: In Section IV-C, by training CycleGAN, the mapping function G can transfer the patterns from D_p^b to $\hat{D}_p^b$, which is a supplement to D_s^b. Since we recognize activities with a single earable device, it is appropriate to choose D_p^b which has closest pattern with D_s^b. Therefore, we select the sensor data at head position in the RealWord dataset as D_p^b. In practice, it will achieve similar supplementary effect in spite of the large patterns difference between our earable device and sensors in other locations. To evaluate the impact of sensor location on public dataset D_p^b, we seperately select the sensory data of head, waist, chest and thigh as D_p^b. Fig. 27 shows the recognition acuracies of CHAR using data from different sensor locations for data augmentation. It can be seen that augmentation works best with data where the sensor is located on the head. This is because the sensor of head in RealWord has the closest pattern to our earable device. At the same time, the accuracy gains obtained by other three different locations are slightly lower than that of the head, but they can also achieve a good augmentation effect. The above results prove that our data augmentation method is applicable to different sensor locations, which is a good property for taking advantage of those datasets where the sensor locations do not match ours.

C. System-Running Performance

In this section, we evaluate CHAR's real-time running performance including response time, CPU and memory occupation, and energy consumption. As the response time is closely related with a smartphone's hardware specifications, we conduct experiments on four kinds of smartphones, namely, HUAWEI Mate 40 Pro, Redmi K30S, SAMSUNG Galaxy S8, and Vivo X20, which are with various specifications and released in different years. To be specific, we measure CHAR's real-time performance by continuously running the testing pipeline with data fed into the system for one hour. During this process, we turn off all the other applications and set the screen brightness to the medium level. After that, we calculate the average result for each smartphone as shown in Table III.

1) Response Time: We insert a piece of codes in the Android application to measure the response time of CHAR which is defined as the duration of outputting the result after performing an activity. From the perspective of data processing, the response time mainly consists of activity segmentation time and network inference time. We can see from Table III that CHAR on HUAWEI Mate 40 Pro finishes segmenting and recognizing an activity within 28.0 ms and 47.7 ms respectively which induces a response time of about 75.7 ms. As for other smartphones, we can observe that although the response time is longer due to lower CPU capacity, it is still less than 200 ms even on the most out-of-date device. These results verify that the response time of CHAR can meet the demand of most human-computer interaction applications on present commercial smartphones.

2) CPU and Memory Occupation: At the same time, we utilize Android Debug Brige to acquire CPU and memory occupation during the above testing. Table III shows the average CPU and memory occupation of each testing repetition. We can see that the average CPU cost on different smartphones is very close, with an average percentage of about 66.2%. However, memory cost varies among different smartphones, that is, a smartphone with larger memory capacity will allocate more memory resource to CHAR. The CPU and memory cost on HUAWEI Mate 40 Pro are about 66.6% and 230.4 MB, respectively. In order to show more details of CHAR running at different stages, we further conduct experiment on this smartphone. Fig. 28 shows the CPU and memory occupation at different stages including initialization, inference, and being idle. We can observe that the memory occupation of is only about 2.4% in average and slightly increases during initialization stage where the recognition model is loaded. The CPU occupation increases rapidly during the inference stage, and decreases to a low level of about 10% when CHAR becomes idle. This because the recognition model is implemented with CPU without using GPU. In the future, we can make full use of the GPU capacity of smartphones and reduce the burden of CPU.

3) Energy Consumption: We measure the energy consumption through Android BatteryManager. Fig. 29 shows how the battery level of HUAWEI Mate 40 Pro varies with running time

TABLE III
THE REAL-TIME RUNNING PERFORMANCE OF CHAR ON DIFFERENT SMARTPHONES

Metric / Smartphone	Release	CPU model	Battery (mAh)	Segment time (ms)	Infer time (ms)	CPU cost (%)	Memory (MB)	Energy (mAh/trial)
HUAWEI Mate 40 Pro	2020	Kirin 9000	4400	28.0	47.7	66.6	230.4	0.124
Redmi K30S	2020	Snapdragon 865	5000	33.3	78.8	66.7	132.8	0.082
SAMSUNG Galaxy S8	2017	Snapdragon 835	3000	58.2	140.9	66.4	136.4	0.112
Vivo X20	2017	Snapdragon 660	3245	57.0	131.9	64.9	62.8	0.325

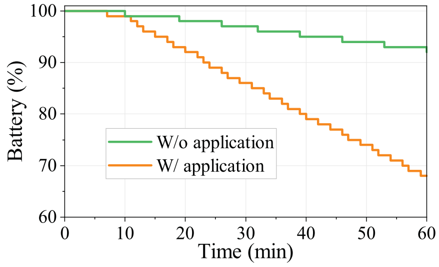
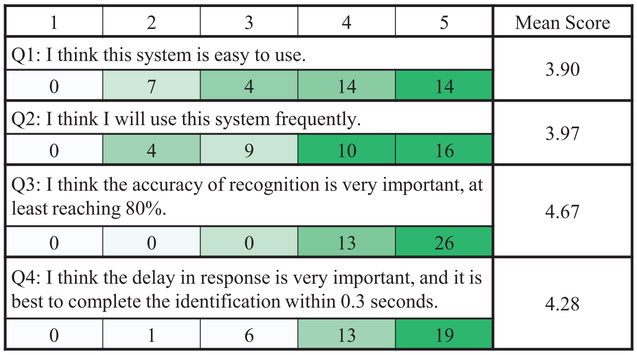


Fig. 29. Energy consumption.

when CHAR is turned on (yellow line) and off (green line). As we can see, when CHAR is running continuously, the average energy consumption per minute can be estimated by $\frac{4400\times(100\%-68\%)}{60}$ equalling 23.47 mAh/min. Note that this energy consumption includes the part of lighting the screen which can be calculated by $\frac{4400\times(100\%-93\%)}{60}$ equalling 5.13 mAh/min. Consequently, CHAR consumes 18.34 mAh per minute which is relatively low. Actually, in real-world usage cases, activities are performed once in a while instead of continuously. Therefore, we further calculate the average energy cost of each testing repetition (i.e., energy per trial) which equals 0.124 mAh per trial. Similarly, we measure energy consumption of CHAR on other three smartphones as shown in Table III. It can be seen that Vivo X20 has the largest energy consumption since it is used most frequently and has the most severe battery degradation. Other three smartphones consume an average of 0.106 mAh per trail, which is relatively low for present commercial smartphones.

D. User Study

In order to fully understand CHAR's practical usefulness, we conduct a user-study experiment in the wild to evaluate users' opinions towards CHAR. We recruit a total number of 39 volunteers (20 males and 19 females) with ages ranging from 19 to 50 to participate in the experiment. Before delivering the hardware and mobile application to them, we make a detailed introduction of how to use CHAR in daily life to them. We request each volunteer to try this system for one week and then answer the questions shown in Fig. 30 with rating scores from 1 to 5, indicating 'strongly disagree', 'disagree', 'average', 'agree', and 'strongly agree', respectively. The overall results are shown in Fig. 30. Note that the value in each cell represents the number of volunteers who choose the corresponding option. We can see that 71.2% of volunteers think CHAR is easy to use, and 66.7% of them are willing to use it frequently. The reason is two-fold. On the one hand, CHAR can achieve composite behaviour recognition through a single ear-worn device, and users can use it in daily life without wearing additional devices. On the other hand, CHAR's interaction process is very natural. Users only need to make simple head gestures to interact in different motion states. In addition, all the volunteers believe that the recognition accuracy of CHAR should exceed 80%, and 82.1% of volunteers believe that the response time of CHAR should be less than 0.3 s. According to the aforementioned evaluation results, CHAR's recognition accuracy is up to 85.1%, and its inference time on an outdated mobile phone is 0.19 s, both of which can meet the need of most users.



1	2	3	4	5	Mean Score
Q1: I think this system is easy to use.					3.90
0	7	4	14	14	
Q2: I think I will use this system frequently.					3.97
0	4	9	10	16	
Q3: I think the accuracy of recognition is very important, at least reaching 80%.					4.67
0	0	0	13	26	
Q4: I think the delay in response is very important, and it is best to complete the identification within 0.3 seconds.					4.28
0	1	6	13	19	

Fig. 30. Overall experience of using the system.

We also investigate users' willingness to perform different composite activities for HCI purpose with rating scores from 1 to 5 indicating 'very unwilling', 'unwilling', 'average', 'willing', and 'very willing'. The average rating scores of different composite activities are shown in Fig. 31. We can see that volunteers' favourite head gestures include H_2, H_5, H_6, H_3, H_4, and H_7. This is because these gestures are simple and fast, making them more convenient for daily use. Overall, users' willingness to perform head gestures during running decreases, and they prefer to use H_2, H_5, and H_6. The underlying reason is that these gestures are simple and will not change the line of sight during running. In a nutshell, users' willingness to use head gestures highly depends on the difficulty of executing them and their current body movement.

VI. DISCUSSION AND FUTURE WORK

A. Comparison With Alternative Approaches

There are some other alternatives to solve the problem of reducing data-collection burden such as self-supervised learning, transfer learning, and semi-supervised learning.

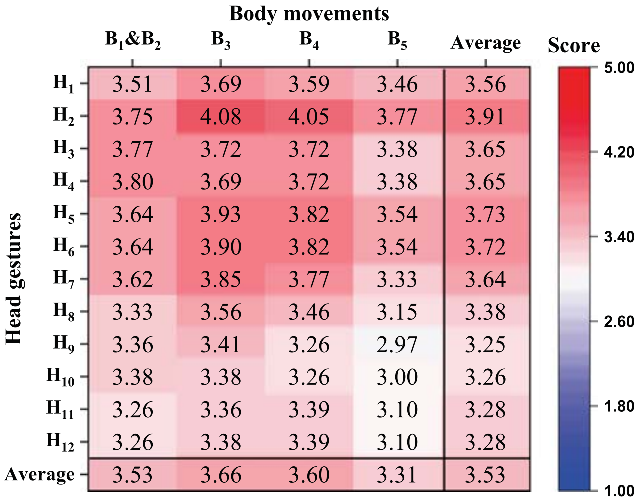


Fig. 31. Willingness to use various composite activities.

Self-supervised learning makes use of a large amount of unlabelled data to pre-train a network on auxiliary tasks, and then fine-tune it on the target task with few labelled data. LIMU-BERT [45] is a typical example of adopting self-supervised learning for human activity recognition. It takes masked IMU data as input to train a feature extractor in the self-supervision stage, and fine-tune a simple classifier with downstream tasks in the supervision phase. The difficulty of this approach lies in designing appropriate self-supervised tasks. As for the network shown in Fig. 8, if we keep the fully connected layers of specific top blocks, it is difficult to design appropriate self-supervised tasks for them individually. However, if we remove these layers, a small amount of labelled data in the supervision stage are not adequate for fine-tuning the two specific top layers. Consequently, self-supervised learning is not very appropriate for our task.

Transfer learning is a machine learning technique where a model developed for a particular task is reused as the starting point for a model on a second task. It leverages pre-existing knowledge from one domain (i.e., source domain) to improve learning efficiency and performance in another domain (i.e., target domain). As the body movement dataset is easily accessible, we can synthesize a composite activity dataset with little but sufficient head gesture data to train a network. However, this process does not take full advantage of the large amount of body movement unlabelled data that exists in the target domain, which can promote the training of the target task. Therefore, transfer learning is not an optimal choice for our problem.

Semi-supervised learning usually first uses a small amount of labelled data to train a network, and generates pseudo labels for the unlabelled data to retrain the network iteratively. In this work, with a small amount of labelled data ($M_s^h = 1$ and $M_s^b = 2$) used for training, the recognition accuracy of composite activities is only 64.2% as shown Fig. 22. As a result, the confidence of pseudo labels generated by this network is relatively low. The predication errors of unlabelled samples will be propagated to subsequent training, resulting in limited performance improvement. Therefore, semi-supervised learning is not appropriate for the task in this work.

Based on the analysis, we choose to augment the body movement dataset with style transferring method, which not only adapts to the characteristics of the data, but also introduces information from public dataset to achieve better training result. Nevertheless, it is possible to incorporate other data augmentation techniques and the above learning methods in our network to make the most of available data and further improve the recognition accuracy. We leave this as part of our future work.

B. Limitation and Future Work

User behaviour in the wild is very complex with more diverse categories involving different body parts. Due to the use of a single IMU sensor in an earable device, CHAR is limited to recognizing whole-body movements and head gestures, but unable to handle upper-limb activities such as hand and finger gestures. This is because the measurement of an IMU sensor is highly dependent on its location. Upper-limb activities induce minute variations of the measurements of an IMU sensor worn in an ear. In order to further enhance the capability of CHAR, a possible solution is to integrating multiple types of sensors or modalities on different body parts. For example, by fusing sensors in a wrist-worn device, it is possible to expand CHAR's activity recognition scope to include hand and even finger gestures. What is more, it is beneficial for improving the accuracy and robustness with multiple sensors.

Another limitation of CHAR is the relatively heavy CPU burden when running the network on commercial smartphones as shown in Table III. This is because when we implement CHAR on the above devices, we only make use of the smartphones' CPU as computing units. In future optimization, the deep learning network can be implemented with GPU for acceleration. In addition, in the present version, the signal processing, activity segmentation, and recognition network are implemented with Python codes and run on Android platforms with Java through Chaquopy project. The running efficiency of both two programming codes are not high. In the future, we can rewrite the whole project with more efficient programming languages such as C and C++. With these engineering optimization tricks, we believe that CHAR shall have better real-time running performance.

VII. Conclusion

Composite activities are rather common and significant for human beings, but has not been carefully investigated yet. In this paper, we put forward a composite activity recognition system called CHAR that can recognize a variety of head-body activities based on a single IMU measurement output by an earphone device. The high-level idea of our solution is to take full advantage of the inter correlation between head gestures and body movements, and design a multi-task learning network to extract shared and task-specific feature representations. We have implemented a real-time prototype and conduct extensive experiments to evaluate its performance. The results show that

CHAR can recognize 60 head-body composite activities with a high accuracy even in sufficient data and insufficient data cases at the same time. We envision that our approach can be also utilized for other composite activities recognition.

REFERENCES

[1] Y. Yang, J. Cao, and Y. Wang, “Robust RFID-based respiration monitoring in dynamic environments,” *IEEE Trans. Mobile Comput.*, vol. 22, no. 3, pp. 1717–1730, Mar. 2023.

[2] X. Li, Y. Zhang, I. Marsic, A. Sarcevic, and R. S. Burd, “Deep learning for RFID-based activity recognition,” in *Proc. ACM Conf. Embedded Netw. Sensor Syst.*, 2016, pp. 164–175.

[3] Y. Wang and Y. Zheng, “Modeling RFID signal reflection for contact-free activity recognition,” *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 2, no. 4, pp. 1–22, 2018.

[4] Y. Jin et al., “SonicASL: An acoustic-based sign language gesture recognizer using earphones,” *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 5, no. 2, pp. 1–30, 2021.

[5] K. Ling, H. Dai, Y. Liu, A. X. Liu, W. Wang, and Q. Gu, “UltraGesture: Fine-grained gesture sensing and recognition,” *IEEE Trans. Mobile Comput.*, vol. 21, no. 7, pp. 2620–2636, Jul. 2022.

[6] Y. Ren, C. Wang, J. Yang, and Y. Chen, “Fine-grained sleep monitoring: Hearing your breathing with smartphones,” in *Proc. IEEE Conf. Comput. Commun.*, 2015, pp. 1194–1202.

[7] X. Xu et al., “HulaMove: Using commodity IMU for waist interaction,” in *Proc. ACM CHI Conf. Hum. Factors Comput. Syst.*, 2021, pp. 1–16.

[8] Y. Yan, C. Yu, X. Yi, and Y. Shi, “HeadGesture: Hands-free input approach leveraging head movements for HMD devices,” *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 2, no. 4, pp. 1–23, 2018.

[9] S. Yi, Z. Qin, E. Novak, Y. Yin, and Q. Li, “GlassGesture: Exploring head gesture interface of smart glasses,” in *Proc. IEEE Conf. Comput. Commun.*, 2016, pp. 1–9.

[10] T. Hachaj and M. Piekarczyk, “Evaluation of pattern recognition methods for head gesture-based interface of a virtual reality helmet equipped with a single IMU sensor,” *Sensors*, vol. 19, no. 24, 2019, Art. no. 5408.

[11] M. Shoaib, S. Bosch, O. D. Incel, H. Scholten, and P. J. Havinga, “Fusion of smartphone motion sensors for physical activity recognition,” *Sensors*, vol. 14, no. 6, pp. 10146–10176, 2014.

[12] C. Liu, J. Xiong, L. Cai, L. Feng, X. Chen, and D. Fang, “Beyond respiration: Contactless sleep sound-activity recognition using RF signals,” *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 3, no. 3, pp. 1–22, 2019.

[13] S. Zhang et al., “NeckSense: A multi-sensor necklace for detecting eating activities in free-living conditions,” *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 4, no. 2, pp. 1–26, 2020.

[14] L. Peng, L. Chen, Z. Ye, and Y. Zhang, “Aroma: A deep multi-task learning based simple and complex human activity recognition method using wearable sensors,” *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 2, no. 2, pp. 1–16, 2018.

[15] F. J. Ordóñez and D. Roggen, “Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition,” *Sensors*, vol. 16, no. 1, 2016, Art. no. 115.

[16] C. Wang et al., “Leveraging activity recognition to enable protective behavior detection in continuous data,” *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 5, no. 2, pp. 1–27, 2021.

[17] P. Voigt, M. Budde, E. Pescara, M. Fujimoto, K. Yasumoto, and M. Beigl, “Feasibility of human activity recognition using wearable depth cameras,” in *Proc. ACM Int. Symp. Wearable Comput.*, 2018, pp. 92–95.

[18] R. Khurana, K. Ahuja, Z. Yu, J. Mankoff, C. Harrison, and M. Goel, “GymCam: Detecting, recognizing and tracking simultaneous exercises in unconstrained scenes,” *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 2, no. 4, pp. 1–17, 2018.

[19] Y. Wang, K. Wu, and L. M. Ni, “WiFall: Device-free fall detection by wireless networks,” *IEEE Trans. Mobile Comput.*, vol. 16, no. 2, pp. 581–594, Feb. 2017.

[20] T. Zhao, J. Liu, Y. Wang, H. Liu, and Y. Chen, “PPG-based finger-level gesture recognition leveraging wearables,” in *Proc. IEEE Conf. Comput. Commun.*, 2018, pp. 1457–1465.

[21] Q. Zhang, J. Jing, D. Wang, and R. Zhao, “WearSign: Pushing the limit of sign language translation using inertial and EMG wearables,” *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 6, no. 1, pp. 1–27, 2022.

[22] J. 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. Joint Conf. Artif. Intell.*, 2015, pp. 3995–4001.

[23] S. Yao, S. Hu, Y. Zhao, A. Zhang, and T. Abdelzaher, “DeepSense: A unified deep learning framework for time-series mobile sensing data processing,” in *Proc. 26th Int. Conf. World Wide Web*, 2017, pp. 351–360.

[24] A. Stisen et al., “Smart devices are different: Assessing and mitigatingmobile sensing heterogeneities for activity recognition,” in *Proc. ACM Conf. Embedded Netw. Sensor Syst.*, 2015, pp. 127–140.

[25] 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, 2016.

[26] M. Malekzadeh, R. G. Clegg, A. Cavallaro, and H. Haddadi, “Mobile sensor data anonymization,” in *Proc. Int. Conf. Internet Things Des. Implementation*, 2019, pp. 49–58.

[27] J. V. Jeyakumar, A. Sarker, L. A. Garcia, and M. Srivastava, “X-char: A concept-based explainable complex human activity recognition model,” *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 7, no. 1, pp. 1–28, 2023.

[28] A. Khamis, B. Kusy, C. T. Chou, M.-L. McLaws, and W. Hu, “RFWash: A weakly supervised tracking of hand hygiene technique,” in *Proc. ACM Conf. Embedded Netw. Sensor Syst.*, 2020, pp. 572–584.

[29] C. Li, M. Liu, and Z. Cao, “WiHF: Enable user identified gesture recognition with WiFi,” in *Proc. IEEE Conf. Comput. Commun.*, 2020, pp. 586–595.

[30] H. Kong et al., “MultiAuth: Enable multi-user authentication with single commodity WiFi device,” in *Proc. 22nd Int. Symp. Theory, Algorithmic Found., Protocol Des. Mobile Netw. Mobile Comput.*, 2021, pp. 31–40.

[31] L. Chen, Y. Zhang, and L. Peng, “Metier: A deep multi-task learning based activity and user recognition model using wearable sensors,” *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 4, no. 1, pp. 1–18, 2020.

[32] S. R. Ramamurthy et al., “CogAx: Early assessment of cognitive and functional impairment from accelerometry,” in *Proc. IEEE Int. Conf. Pervasive Comput. Commun.*, 2022, pp. 66–76.

[33] X. Xu et al., “Earbuddy: Enabling on-face interaction via wireless earbuds,” in *Proc. ACM CHI Conf. Hum. Factors Comput. Syst.*, 2020, pp. 1–14.

[34] Y. Gao, W. Wang, V. V. Phoha, W. Sun, and Z. Jin, “Earecho: Using ear canal echo for wearable authentication,” *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 3, no. 3, pp. 1–24, 2019.

[35] N. Bui et al., “EBP: A wearable system for frequent and comfortable blood pressure monitoring from user’s ear,” in *Proc. ACM Annu. Int. Conf. Mobile Comput. Netw.*, 2019, pp. 1–17.

[36] A. Ferlini, A. Montanari, C. Mascolo, and R. Harle, “Head motion tracking through in-ear wearables,” in *Proc. 1st Int. Workshop Earable Comput.*, 2019, pp. 8–13.

[37] F. Kawsar, C. Min, A. Mathur, and A. Montanari, “Earables for personal-scale behavior analytics,” *IEEE Pervasive Comput.*, vol. 17, no. 3, pp. 83–89, Third Quarter 2018.

[38] J. Liu, W. Song, L. Shen, J. Han, and K. Ren, “Secure user verification and continuous authentication via earphone IMU,” *IEEE Trans. Mobile Comput.*, vol. 22, no. 11, pp. 6755–6769, Nov. 2023.

[39] B. K. Iwana and S. Uchida, “An empirical survey of data augmentation for time series classification with neural networks,” *PLoS One*, vol. 16, no. 7, 2021, Art. no. e0254841.

[40] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2242–2251.

[41] J. Donahue et al., “Long-term recurrent convolutional networks for visual recognition and description,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 2625–2634.

[42] T. Sztyler and H. Stuckenschmidt, “On-body localization of wearable devices: An investigation of position-aware activity recognition,” in *Proc. IEEE Int. Conf. Pervasive Comput. Commun.*, 2016, pp. 1–9.

[43] L. Van der Maaten and G. Hinton, “Visualizing data using t-SNE,” *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 2579–2605, 2008.

[44] J. Lin, “Divergence measures based on the shannon entropy,” *IEEE Trans. Inf. Theory*, vol. 37, no. 1, pp. 145–151, Jan. 1991.

[45] H. Xu, P. Zhou, R. Tan, M. Li, and G. Shen, “Limu-BERT: Unleashing the potential of unlabeled data for IMU sensing applications,” in *Proc. ACM Conf. Embedded Netw. Sensor Syst.*, 2021, pp. 220–233.

Peizhao Zhu received the master's degree from the School of Computer Science and Software Engineering, Shenzhen University, in 2024. His research interests include gesture recognition, mobile computing, and Internet of Things.

Yuzheng Zhu is currently working toward the master's degree with the Colledge of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China. His research interests covers mobile computing and intelligent sensing.

Wenyuan Li received the bachelor's degree from the School of Computer Science and Software Engineering, Shenzhen University, in 2024. His research interests include gesture recognition, mobile computing, and Internet of Things.

Yanbo He is currently working toward the bachelor's degree with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China. His research interests include mobile computing, machine learning, and Internet of Things (IoT).

Yongpan Zou (Member, IEEE) received the PhD degree from the Department of Computer Science and Engineering (CSE), Hong Kong University of Science and Technology, in 2017. He is currently an associate professor with the College of Computer Science and Software Engineering, Shenzhen University. His research interests include ubiquitous sensing, mobile computing, and human-computer interaction.

Kaishun Wu (Fellow, IEEE) received the PhD degree from the Hong Kong University of Science and Technology, Hong Kong, in 2011. He is currently a professor in information hub with the Hong Kong University of Science and Technology (Guangzhou). His research interests include wireless communications and mobile computing. He won several best paper awards of international conferences, such as IEEE Globecom 2012 and IEEE MASS 2014.

Victor C. M. Leung (Life Fellow, IEEE) received the PhD degree in electrical engineering from the University of British Columbia, in 1981. He is currently the dean with Artificial Intelligence Research Institute, Shenzhen MSU-BIT University. His research focuses on wireless networks and mobile systems with more then 60,000 citations. He has received numerous accolades, such as the APEBC Gold Medal, NSERC-PostgraduateScholarships, and IEEE Vancouver Section Centennial Award.