Bidirectional Transfer Learning Between Activity and User Identity Recognition Tasks via 2D CNN-LSTM Model for Wearables

Billur Barshan and Enes Koşar

***Abstract*—We implement transfer learning (TL) to attain efficiency in training and feature extraction by freezing the connection weights of shallow layers of deep learning (DL) models, already trained for a given task, to be used in another. This eliminates the need to retrain the whole model from scratch for the new task, resulting in significantly reduced training times, while maintaining the classification accuracy at acceptable levels. Specifically, we investigate bidirectional feature transfer between the human activity recognition (HAR) and user identity recognition (UIR) tasks through the use of wearable motion sensors. Both tasks have recently drawn considerable attention and are significant because of their relevance in a broad range of applications. Knowledge transfer between tasks via TL brings many advantages. We employ the features extracted by our recently proposed hybrid DL model which has proven superior to commonly used state-of-the-art (SoTA) DL models. The model can extract features more effectively because of its parallel-running 2D convolutional neural network (CNN) and long short-term memory (LSTM) branches that receive complementary input types, unlike many hybrid models that implement a series connection that receives a single type of input. To our knowledge, TL between the above-mentioned tasks has not been implemented to date. We demonstrate high-accuracy bidirectional feature transfer between the HAR and UIR domains that shortens the training time by a factor that varies between 14.3–34.4, without much degradation in the classification accuracy. To provide a typical figure, the accuracy degrades by 2.45% for a group of three dynamic activities. The key technical challenge of this work is maintaining the classification accuracy at acceptable levels while benefiting from the advantages that TL offers. We expect the results of this work to have impact by way of reducing the demanding computational requirements in training resource-limited wearable edge devices.**

***Index Terms*—Convolutional neural network (CNN), deep learning (DL), feature extraction, human activity recognition (HAR), hybrid DL models, long short-term memory (LSTM), machine learning (ML), transfer learning (TL), user identity recognition (UIR), wearable motion sensors.**

I. Introduction

ADVANCEMENTS in ubiquitous (pervasive) computing and computational intelligence have accelerated the integration of sensors in devices, accessories, and textiles used on a daily basis. Wearable sensors are one of the key elements of the Internet of Things (IoT). With the development of the micro-electro-mechanical systems (MEMS) technology, wearables have become more compact, lighter, and less costly, making them more convenient to use and, hence, more conducive to the IoT. Data acquired from wearable sensors are highly informative about the user state, well being, activities, and identity. Real-time processing and analysis of the recorded data enables informed and improved decision making. However, involvement of wearable sensors in IoT entails multiple challenges, such as dealing with their limited resources, management of data acquired from a network of communicating sensors, interoperability of different sensor modalities, and ensuring the security and privacy of personal/sensitive data.

A research area that has benefitted considerably from the rapid developments in wearable technology and sensor analytics is human activity recognition (HAR) [1]. HAR pertains to automated detection and monitoring of activities through the processing of sensory input [2]. Wearable motion sensors are affixed to different body parts (e.g., torso and limbs), where each body part is considered to be rigid. Movements of the body parts during activities are directly captured in 3D and transformed into signal patterns which are recorded. To be able to discern the activities by way of machine learning (ML) or deep learning (DL), ideally, each different motion type or activity of interest must be associated with a distinct signal pattern. Since the recordings are 1D time sequences, this approach eliminates the correspondence problem, occlusion effects, and any privacy issues that may arise when using camera systems.

Accelerometers, gyroscopes, and magnetometers are among the widely employed wearable motion sensor types. Besides wearables-based solutions, activities can be recognized by specially designed smart environments as well [3]. However, such environments confine the user to a limited space, typically indoors, while wearables can be used both indoors and outdoors without any restriction. HAR finds a broad range of applications in ambient intelligence, context-aware systems, digital well-being and healthcare, assistive technologies, biomechanics, sports science, entertainment, and ergonomics [4].

User identity recognition (UIR) is another recognition task of interest that aims to detect user identities automatically based on their biometric features or activities. Recognizing

Received 21 February 2025; revised 19 April 2025; accepted 9 May 2025. Date of publication 9 June 2025; date of current version 7 November 2025. *(Corresponding author: Billur Barshan.)*

The authors are with the Department of Electrical and Electronics Engineering, Bilkent University, 06800 Ankara, Türkiye (e-mail: billur@ee.bilkent.edu.tr; enes.kosar@bilkent.edu.tr).

Digital Object Identifier 10.1109/JIOT.2025.3577925

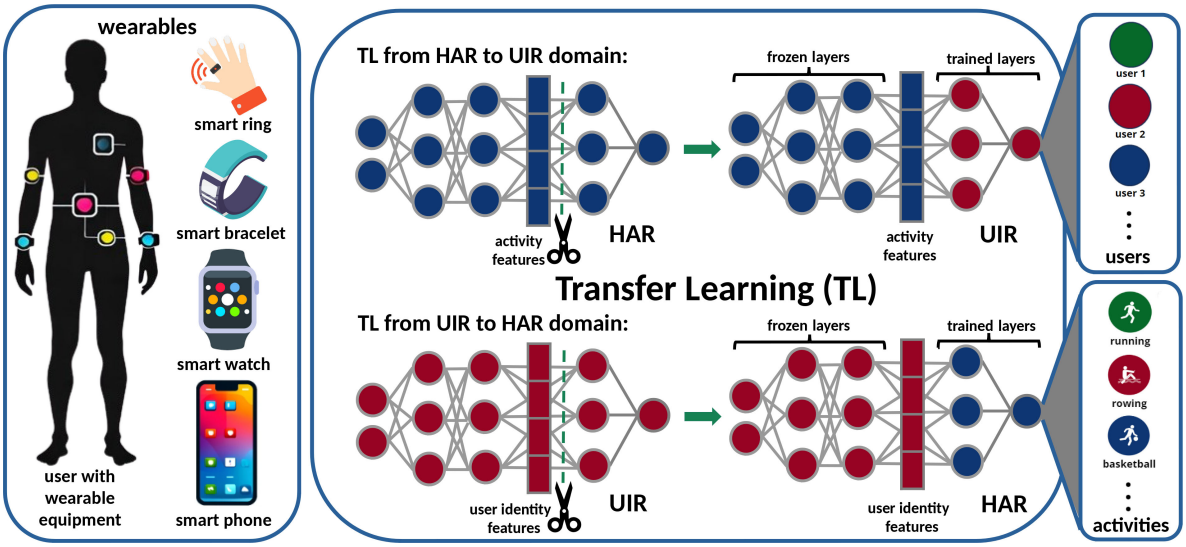

Fig. 1. Overview of bidirectional TL between the HAR and UIR domains. The upper pipeline illustrates the use of TL in a scenario where a pretrained model is available for the HAR task but not for UIR; in this case, only the final layers of the HAR model need to be fine-tuned for the UIR task. Conversely, the lower pipeline depicts how TL is exploited for the HAR task in a scenario where a pretrained model exists for the UIR task but not for HAR.

user identities may be necessary to prevent third party access to certain services, devices, vehicles, buildings, inventories, and other facilities. Recordings of daily and sports activities, gait and sleep patterns, heart and respiration rates contain identifying information about individuals. Wearable devices, such as smart watches and bracelets offer convenient access to services, such as electronic payment systems and social media. However, the personal and sensitive nature of the data they handle raises significant concerns regarding authentication, security, and privacy. Therefore, developing robust, efficient, and user-friendly UIR systems is essential. Some of the application areas of UIR are digital security and surveillance systems [5], detection of crime and violence [6], personalized services and technologies [7], [8], [9], digital healthcare [10], [11], and telemedicine [12].

Both the HAR and UIR problems involve classification tasks (of activities and users, respectively) for which extraction of discriminative features is essential [13]. In our recent work [14], [15], we have investigated the best DL model architecture to extract highly representative features for accurate HAR, based on data acquired from wearable motion sensors. In the present study, we first explore whether the proposed model's learning capacity extends beyond HAR and extracted features are usable in another related problem domain—the UIR domain—or not. We demonstrate that besides HAR, our recently introduced hybrid [2D Convolutional Neural Network (CNN)-long short-term memory (LSTM)] DL model attains enhanced accuracy in the UIR task as well. Then, we employ transfer learning (TL) and elaborate on bidirectional transfer of features, extracted by our hybrid DL model, between the HAR and UIR domains. This allows identification of salient features that are common to both tasks without relinquishing high classification accuracies. Our approach reduces the required training effort considerably and the number of parameters to some extent, while maintaining an acceptable level of accuracy.

An overview of the method we follow is depicted in Fig. 1. We assume the existence of a pretrained model for one of two tasks and adapt it for the other task by merely adjusting the connection weights of its classification layers. For instance, if a pretrained model for HAR is already available, we demonstrate that by tuning the weights of only its final layers, the model can be effectively adapted for the UIR task. Conversely, if we start with a pretrained model for UIR, we demonstrate that training only the last couple of layers enables successful adaptation for the HAR task. For both scenarios—availability of a pretrained model for HAR but not for UIR and vice versa—instead of developing a completely new model from scratch, we can efficiently repurpose the existing pretrained model for the other task that the model is not trained for.

The key contributions of this study are multifold.

1) Wearable sensor-based HAR and UIR tasks are often associated with limited computational resources and small datasets. Using pretrained models for new tasks can offer significant benefits. However, the lack of studies that investigate feature transfer between the HAR and UIR tasks presents a research gap in the wearable sensing area. We demonstrate the feasibility of feature transfer between these two tasks with our hybrid DL model, setting a baseline for future research. This will lead to more efficient training and inference, which is crucial in fields with limited data and computing power.
2) We implement bidirectional TL between the HAR and UIR domains for the first time which results in considerably reduced training times without significant degradation in the classification accuracy. Consequently, the demanding requirements in training wearables with low computational resources and limited memory space can be considerably reduced.
3) We compare the results of bidirectional feature transfer between the UIR and HAR domains using our DL

model and four basic state-of-the-art (SoTA) models: a) 1D CNN; b) 2D CNN; c) LSTM, which are single models, and d) a standard 1D CNN-LSTM hybrid model. Our results indicate that while most of the comparison models do not generalize to the new domain adequately, our model adapts well and maintains acceptable accuracy levels, degraded by small amounts. This extends our previous findings and demonstrates that our model's features are not only useful across datasets and subjects but also for new tasks.

4) Effectively, we design a unified encoder capable of handling both the HAR and UIR tasks within a single framework. This streamlined approach significantly reduces the memory burden on wearable edge devices that have limited processing and storage capabilities. By consolidating the feature extraction process for both tasks into a single encoder, we simplify the system architecture while improving the efficiency and practicality of deploying advanced functionalities on resource-constrained wearables.

In the next section, we provide background and overview the related work on UIR, DL, and TL, based on processing wearable motion sensor signals. In Section III, we investigate the feasibility of feature transfer from the HAR domain to the UIR domain. In Section IV, we describe the implementation details of bidirectional TL between the HAR and UIR domains and present our results. We summarize, draw conclusions, and specify some future research directions in the final section.

II. Related Work

A. Related Work on User Identity Recognition

UIR systems rely on user characteristics that fall under three categories: 1) Physiological biometrics (such as fingerprint scans or heartbeat measurements); 2) behavioral biometrics (e.g., walking styles or other activity-based movement patterns); and 3) physical fingerprints (such as device hardware that are commonly used in the IoT) [16]. UIR studies based on behavioral biometrics tend to employ wearable motion sensors for signal acquisition. Acquired signals are usually preprocessed via signal processing techniques, followed by the application of ML/DL methods for feature extraction and signal classification, to identify users [17].

Analysis of gait patterns of individuals has proven to be a reliable source of information for UIR in a number of studies. The work reported in [18] uses an LSTM network with an attention mechanism to extract gait-related features from the recordings of an inertial measurement unit (IMU). Reference [19] formulates the problem as an image classification task where smart phone-embedded accelerometer signals are converted into spectro-temporal image representations by using the short-time Fourier transform (STFT). The study reported in [20] presents a similar approach. Although IMUs embedded in smart phones, smart watches, and other wearables simplify system implementation, the authors of [21] argue that these devices are unable to acquire the detailed biomechanical information necessary for reliable UIR. Therefore, besides an IMU, they install 10 force sensors into specific locations in the user's shoes to capture comprehensive kinetic and kinematic information.

Activity types other than gait can also produce distinct signal patterns that allow the identification of individuals. The work reported in [5] demonstrates that common daily activities and fall states can be used for accurate UIR, based on processing two open-access benchmark activity datasets [University of California Irvine HAR (UCI HAR) and University of Milano Bicocca Smartphone-based HAR (UniMiB SHAR) Datasets]. The authors of [22] and [23] have implemented a variety of ML/DL techniques as feature extractors and classifiers for UIR, processing recorded activity signals. Most studies train the models only once, with minimal end-user involvement. This results in static models that cannot be updated in time. To address this issue, Reference [24] proposes an interactive UIR method based on reinforcement learning. Because employing particular movement signals for UIR imposes a limitation, Reference [25] proposes a method for activity-free UIR that combines HAR with UIR based on computer vision techniques and wearables. This approach allows greater flexibility in UIR since it does not restrict the users to particular movement types.

B. Related Work on Deep Learning

DL models have recently attracted much attention from the research community and achieved favorable outcomes in many different fields. Compared to ML techniques, they possess the ability to automatically extract discriminative and diverse sets of features that are data-informed, based on recorded sensor data. However, DL models need high computational resources, large amounts of quality data, and long training times. Despite these demanding requirements, DL models display higher robustness and superior classification capability. Single and hybrid DL models have been developed for various purposes. While single LSTM architectures exhibit outstanding performance on time-series data, single CNN models are superior in processing images. The input to the latter can be real images or time sequences represented in the form of images [26]. Hybrid structures combine the strengths of single CNN and LSTM models and typically involve a cascaded (series) connection of CNN and LSTM units, such as CNN followed by LSTM (CNN-LSTM) or vice versa (LSTM-CNN), in which the output of the initial module in the sequential arrangement is provided as input to the second one.

Authors have recently introduced a novel 2D CNN-LSTM hybrid DL model that demonstrates superior performance in HAR by processing data acquired from wearable motion sensors [14], [15]. The model, illustrated in Fig. 2, can extract features more effectively because of its parallel-running 2D CNN and LSTM branches, unlike many hybrid models that implement a series connection. While providing the raw time sequences to the LSTM branch, the 2D CNN branch receives the spectrogram images of the raw time sequences as input. We have compared the HAR performance of our 2D CNN-LSTM DL model to those of six well-known DL architectures. These are three single (1D CNN, 2D CNN, and

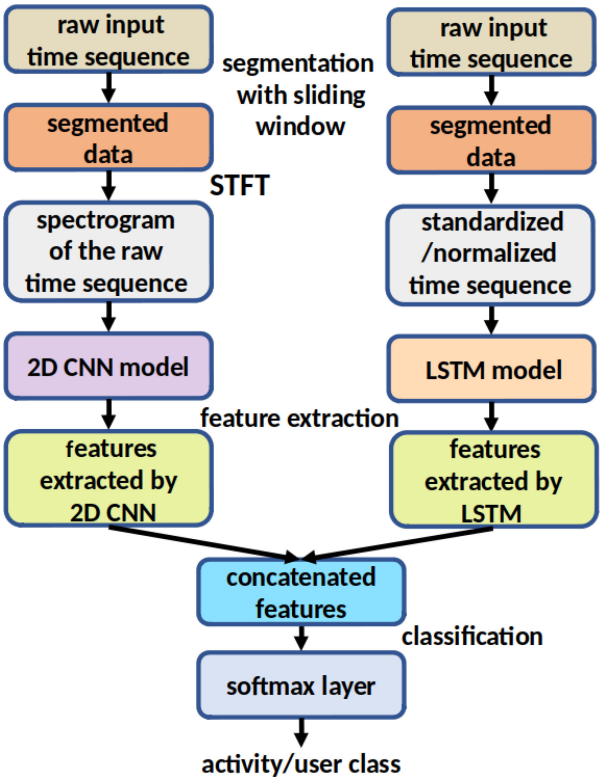

Fig. 2. Structure of the 2D CNN-LSTM hybrid DL model [14].

LSTM) and three hybrid models (standard 1D CNN-LSTM, Ordóñez & Roggen's 1D CNN-LSTM [27], and the alternative 1D CNN-LSTM [28]). In the first two of the hybrid models, the 1D CNN and LSTM modules are connected in series whereas in the third, they are connected in parallel, forming the two branches of the model architecture. After implementing all seven models, we have tuned their hyperparameters and tested their generalization capability and robustness for the HAR task on two open-access datasets. The datasets contain data from multiple (19 and 30) subjects. Our 2D CNN-LSTM hybrid DL model has proven superior with respect to four performance metrics, mainly because of its two parallel-running branches receiving inputs of inherently different nature. This enables extracting a broader range of representative and generalizable features compared to existing SoTA architectures [14], [15]. In comparison to the alternatives, the model offers an acceptable level of complexity measured by the number of model parameters, training time, and occupied memory space. For these reasons, the DL model that we use in this work is the same as the model that we have proposed in [14].

C. Transfer Learning for Human Activity Recognition

The research and development phase of DL models involves a large number of experiments to evaluate different design alternatives and parameter settings. Training a DL model is a computationally intensive and time-consuming task that requires large quantities of labeled data. The more complex and deeper a DL model is, the larger the amount of training data needed to adequately learn the discerning features for a particular task. However, such vast quantities of data may not always be available in every domain. In the realm of wearables, data are scarce, computational power is low, and memory space is limited, slowing down the training process. Under such conditions, it would be advantageous to transfer knowledge from a domain where vast quantity of data is available to another domain where it is not. TL helps reduce the burden by enabling faster experimentation with smaller, more efficient models, allowing researchers to iterate quickly and improve model design more effectively. Besides, given the pervasiveness of DL applications, it would not be practical to train a separate model from scratch for each and every new DL task.

To overcome the above-mentioned challenges, employing pretrained models in conjunction with TL offers considerable benefits. Many well-established pretrained models exist for vision and language tasks, providing a solid foundation for unexplored research topics. These models, which have been extensively trained to serve as feature extractors, can be fine-tuned for specific tasks and datasets. The use of TL may alleviate the heavy computational burden of training DL models by employing previously learned knowledge in a given domain (the source domain) to train a model in another domain (the target domain) [29], [30], [31]. In other words, TL can facilitate a learning task in the target domain by exploiting the knowledge acquired from the source domain. Feature transfer from the source domain to the target domain enables bypassing the feature extraction stage of model training which significantly reduces the computational requirements.

Feature extraction in deep neural networks (DNNs) is conducted in a hierarchical way, meaning that while the initial layers extract elementary or primitive features, the last few layers bring out the more complex and domain-specific features [32], [33]. Thus, the earlier layers of DNNs operate as feature extractors while the last few handle the classification process. When the tasks and problem types have sufficient similarity, different DL models can benefit from using the same feature extraction layers and the features extracted by them. Based on this concept, parameter control strategy [34] proposes that for similar tasks, many of the layers of a DNN, trained in the source domain, can be frozen and only the last few layers of the network need to be tuned for the task in the target domain.

HAR has benefitted from TL techniques to overcome some of the challenges in this area. Within the HAR context, TL can be implemented in the form of cross-body part, cross-sensor, cross-dataset, cross-user (cross-subject), and cross-activity knowledge transfer [15], [35], [36].

Users carry smart bracelets, smart watches, and smart phones on their body at different configurations. However, the way these devices are worn on the body affects the resulting signal patterns considerably. In our earlier work, we have investigated the effect of position and orientation changes of motion sensor units on the HAR accuracy [37], [38], [39], [40]. In those works, we have developed algorithms that employ features that are invariant to position and orientation shifts and also investigated the interchangeability of sensor units carried on different body parts. Within the context of TL, it is highly advantageous if a model trained for HAR with

a particular sensor configuration on a certain body part is capable of transferring the acquired knowledge to alternative sensor configurations and other body parts. These would be examples of cross-body part and cross-sensor TL.

Cross-user (cross-subject) TL is the most commonly exploited type of TL for HAR [41]. Since the age, gender, height, weight, and physical state of individuals affect the way they perform activities, ML/DL models trained on a group of subjects' data may not perform sufficiently well on new (unseen) users [42]. A number of studies have demonstrated favorable results in cross-user TL to improve model performance on new users [43], [44], [45].

III. Feasibility of Feature Transfer from the HAR Domain to the UIR Domain

Since individuals perform physical activities in their own characteristic style, signals recorded from wearable motion sensors contain cues for recognizing the person. To investigate the feasibility of feature transfer from the HAR domain to the UIR domain, we first consider each activity type in the two datasets separately to recognize user identities, employing our recently proposed hybrid DL model which we have developed for the HAR task [14], [15]. This also helps to identify the activity types that are informative and useful for the UIR task, which we will address later in Section IV-A4. We kept the layer structure of our hybrid DL model the same as in [14] and [15] but trained it from scratch with only the signals recorded from a particular activity type and evaluated the UIR accuracy for that activity. While training the model for UIR, we label the activity recordings with the corresponding user labels and provide them as input to the model. Table I presents the resulting accuracies. (The reader is referred to [14], [15], [46], [47], [48], and [49] for detailed descriptions of the UCI HAR and DSA Datasets that we have used in this study. DSA: Daily and Sports Activities.) The results in Table I(a), obtained by processing the UCI HAR Dataset, indicate that while we attain UIR accuracies above 93.4% with the three dynamic (walking related) activities, static activities (sitting, standing, and laying down) result in accuracies below 54.6%. On the other hand, Table I(b) shows that all the UIR accuracies that we have obtained by processing the DSA Dataset are above 84.7%. The reason for this difference between the results of the two datasets could be that while the UCI HAR Dataset was collected using only a single waist-worn sensor unit embedded in a smart phone, the DSA Dataset was acquired with a total of five sensor units, each placed on a different body part (the torso and each of the four limbs). Besides, the five sensor units contain an additional sensor modality which is the magnetometer. This allows extraction of additional information from the activity signals that supports the recognition of users as they perform both static and dynamic activities.

The generally high UIR accuracy values that we have obtained employing a DL model developed for the HAR task indicate that HAR and UIR tasks have some features in common and it is worthwhile to transfer features from the HAR domain to the UIR domain. Note that the experiments reported in this section were conducted solely to explore the feasibility of TL between the two domains. We have investigated which activity signals provide the most informative representations for UIR. These experiments are not intended to be part of the bidirectional TL scheme between the two domains in a real-world application. In the following section, we elaborate on bidirectional feature transfer between these two domains.

TABLE I
UIR Accuracy Plus/Minus One Standard Deviation for Each Type of Activity, Based on Processing the (a) UCI HAR Dataset (b) DSA Dataset

(a)

Activity Name		Accuracy ± std (%)
sitting		35.39 ± 3.51
standing	**(static activities)**	46.05 ± 5.95
laying down		54.52 ± 3.59
walking		98.10 ± 0.97
walking upstairs	**(dynamic activities)**	96.51 ± 0.86
walking downstairs		93.41 ± 0.92
dynamic (walking-related) activities combined		**96.83 ± 0.31**

(b)

Activity Name		Accuracy ± std (%)
sitting		99.72 ± 0.55
standing	**(static activities)**	100.00 ± 0.00
lying on back side		100.00 ± 0.00
lying on right side		100.00 ± 0.00
ascending stairs		99.44 ± 0.52
descending stairs		99.86 ± 0.28
standing in an elevator still		99.17 ± 0.81
moving around in an elevator		89.58 ± 1.58
walking in a parking lot		99.44 ± 0.81
walking on a flat treadmill at a speed of 4 km/h		100.00 ± 0.00
walking on a 15° inclined treadmill at 4 km/h		100.00 ± 0.00
running on a flat treadmill at a speed of 8 km/h		100.00 ± 0.00
exercising on a stepper		99.58 ± 0.56
exercising on a cross trainer		100.00 ± 0.00
cycling on an exercise bike in horizontal position		99.72 ± 0.34
cycling on an exercise bike in vertical position		99.31 ± 0.62
rowing		84.72 ± 1.46
jumping		85.83 ± 2.18
playing basketball		86.25 ± 2.76
all activities combined		**94.59 ± 0.16**

IV. Bidirectional Transfer Learning Between the HAR and UIR Domains

In this section, we demonstrate through TL that the features extracted by our hybrid DL model, originally developed for the HAR task, can provide high quality representations for another recognition task (which is UIR based on user activities), and vice versa. To do this, we implement a single encoder for joint feature extraction for both HAR and UIR instead of two separate encoders for HAR and UIR, enabling TL between the two tasks (Fig. 3). Assuming the availability of a model already trained using our encoder for one task (HAR), we directly use the features it extracts to train only a minimal classifier for the other task (UIR), and vice versa. This way, we attain efficiency in training and feature extraction and demonstrate the practical utility of reusing a pretrained model across these two domains. Our approach eliminates the need to retrain the whole model from scratch as well as the use of

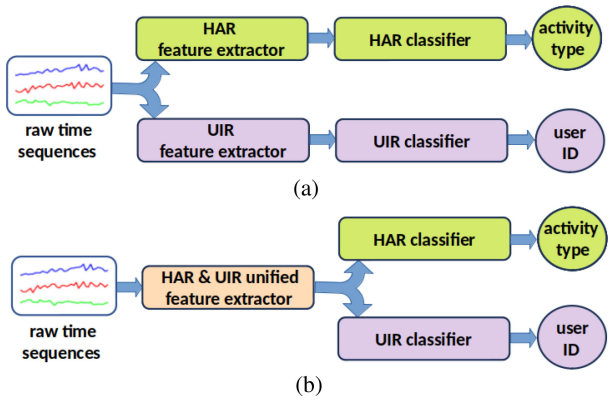

Fig. 3. Feature extraction via encoders: (a) two separate encoders for HAR and UIR, and (b) a single unified encoder for joint feature extraction for both HAR and UIR.

two encoders, as in Fig. 3(a), for feature extraction, which is critical for resource-limited wearables. Consequently, training times and memory space are considerably reduced, while an acceptable level of accuracy is maintained.

In Section IV-A, we investigate feature transfer from the HAR domain to the UIR domain, whereas in Section IV-B, we explore feature transfer in the opposite direction.

A. *Feature Transfer from HAR Domain to UIR Domain*

To transfer features from the HAR domain to the UIR domain, we use our recently proposed hybrid DL model, developed for the HAR task [14], [15]. Note that in the cited work, a separate DL model is developed for each of the two datasets, trained with all of the activities of the corresponding dataset.

1) Selecting the Features to be Transferred: We need to determine at which layer of each branch of the model to transfer the extracted features from the HAR domain to the UIR domain. Transferring features only from a couple of shallow layers may require retraining a larger part of the DL model for the new task, which is costly. On the other hand, since the last few layers of the network extract more domain-specific features, it is usually not feasible to use those features for another task. Thus, there is an inevitable tradeoff between the depth of the layers whose features will be transferred to the UIR domain (by freezing their weights) and the required training time.

To identify the best layers for feature transfer, we have conducted experiments by using features of each possible layer combination of the two parallel-running branches of our DL model. We form the layer combinations by taking one layer from each branch and present the resulting accuracies in Table II. Rows and columns of Table II(a), respectively, represent the 2D CNN and LSTM branch layers of our hybrid DL model developed by processing the UCI HAR Dataset. The top left corner of the table corresponds to the shallowest layers of the model that extract more elementary or primitive features, while the bottom right corner corresponds to the deepest layers and more advanced, domain-specific features. For the 2D CNN-LSTM model illustrated on the left side of

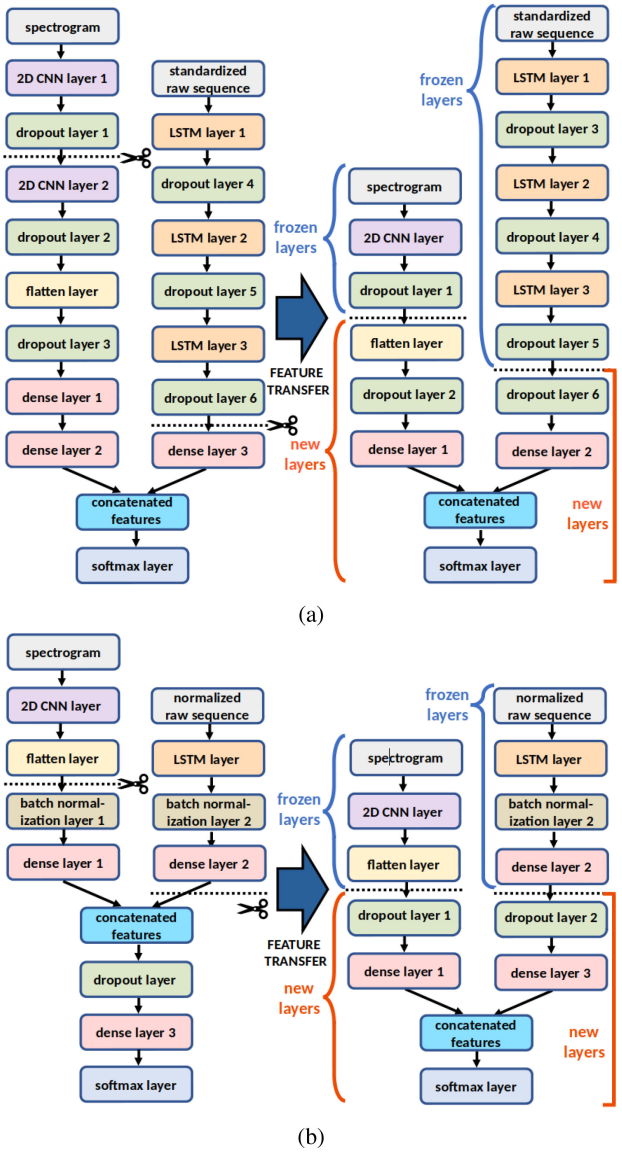

Fig. 4. TL from the HAR domain to the UIR domain for the (a) UCI HAR Dataset (b) DSA Dataset.

Fig. 4(a), we obtain the maximum classification accuracy by using the features extracted after dropout layer 1 which is the second layer of the 2D CNN branch and dropout layer 6 which is the second-to-last layer of the LSTM branch. This indicates that only one 2D CNN layer is sufficient to extract useful features for HAR and the features extracted by the 2D CNN branch of the network (with spectrograms given as input) start becoming domain specific after the second layer. On the other hand, the LSTM branch extracts more generalizable features well into the deeper layers of the network. We attain the highest accuracy with the features extracted after the three LSTM layers.

Rows and columns of Table II(b), respectively, represent the 2D CNN and LSTM branch layers of our DL model developed by processing the DSA Dataset. Note that in the rows of Table II(b), we omit the 2D CNN layer because the 2D output of the 2D CNN layer is not compatible with the 1D output of the LSTM branch for concatenation and needs

TABLE II
UIR ACCURACY PLUS/MINUS ONE STANDARD DEVIATION (%) WITH FEATURES EXTRACTED FROM EACH LAYER COMBINATION OF THE MODEL'S TWO BRANCHES, BASED ON PROCESSING THE (A) UCI HAR DATASET (B) DSA DATASET

(a)

Branch Layers	LSTM 1	dropout 4	LSTM 2	dropout 5	LSTM 3	dropout 6	dense 3
2D CNN 1	91.06 ± 0.51	90.49 ± 0.42	89.97 ± 0.67	90.40 ± 1.12	93.11 ± 0.94	92.31 ± 0.79	93.07 ± 0.34
dropout 1	91.34 ± 1.02	91.09 ± 0.39	90.20 ± 0.74	90.30 ± 1.19	93.14 ± 1.14	**93.31 ± 0.82**	92.58 ± 0.54
2D CNN 2	89.02 ± 1.03	90.19 ± 1.07	88.84 ± 0.61	89.46 ± 0.73	89.42 ± 0.68	89.49 ± 1.09	87.93 ± 0.86
dropout 2	89.89 ± 1.36	90.84 ± 1.74	89.37 ± 0.98	89.53 ± 0.84	89.17 ± 1.59	89.14 ± 0.50	88.18 ± 0.69
flatten	90.56 ± 0.39	91.11 ± 0.71	89.13 ± 1.22	89.49 ± 0.66	89.54 ± 0.59	89.00 ± 1.09	88.82 ± 0.92
dropout 3	91.31 ± 0.87	91.23 ± 0.11	88.43 ± 1.29	89.53 ± 0.91	89.24 ± 0.36	89.71 ± 0.98	88.62 ± 0.62
dense 1	63.11 ± 1.16	64.95 ± 1.13	52.51 ± 0.66	52.44 ± 0.71	27.29 ± 1.02	25.64 ± 0.91	26.33 ± 1.24
dense 2	63.15 ± 1.17	62.52 ± 1.18	50.78 ± 0.98	51.28 ± 1.12	19.07 ± 0.96	19.46 ± 0.82	17.47 ± 1.06

(b)

Branch Layers	LSTM	batch 2	dense 2
flatten	95.85 ± 0.68	95.75 ± 0.48	**95.91 ± 0.33**
batch 1	95.15 ± 0.56	94.83 ± 0.19	95.27 ± 0.26
dense 1	56.71 ± 0.91	59.75 ± 0.87	57.62 ± 1.14

Main Layers	Accuracy ± std (%)
concatenation	59.80 ± 0.65
dropout	59.14 ± 1.55
dense 3	61.55 ± 1.52

to be converted to a 1D vector first. Since the flatten layer that follows the 2D CNN layer handles this conversion, the rows of the subtable start with the flatten layer. As in Table II(a), the top left corner of the table corresponds to the initial layers of the model that extract rather elementary features and the bottom right corner displays the results from the deepest layers that extract more complex features. Fig. 4(b) illustrates the feature transfer for the DSA Dataset. We achieve the highest accuracy with the features extracted after the flatten layer of the 2D CNN branch and the dense layer 2 of the LSTM branch. Thus, the results from both datasets indicate that the shallow layers of the 2D CNN branch and the deep layers of the LSTM branch extract informative and useful features for TL from the HAR domain to the UIR domain. The subtable on the right side of Table II(b) displays the UIR accuracies starting with the concatenation layer for the DSA Dataset. Note that these results correspond to the DL model on the left side of Fig. 4(b). We observe that concatenated features, when transferred to the UIR domain, cannot provide high accuracies, indicating that the extracted features become too domain specific after concatenation. Therefore, features need to be transferred prior to concatenation.

Based on the presented results, we have used the appropriate initial layers of the DL models from each branch [see the scissors cuts in dashed lines on the left sides of Fig. 4(a) and (b)] which have already been trained for the HAR task, as the shallow layers that will function as feature extractors for the UIR task. We freeze the connection weights of these layers and do not modify them during the training process for UIR. We only need to train the weights of the newly added layers.

Since the previously trained (frozen) layers operate as feature extractors, the newly added layers effectively function as classifiers. Therefore, we have designed the newly added layers in the simplest possible form by including one dropout layer (as regularizer) and one dense layer in each branch, and one softmax layer after concatenation of features from the two branches. The only purpose of the additional flatten layer on the right side of Fig. 4(a) on the 2D CNN branch is to convert the tensor shape to make it compatible with the next layer.

TABLE III
UIR ACCURACY PLUS/MINUS ONE STANDARD DEVIATION (%) AS THE NUMBER OF DLNS AND THE DROPOUT PROBABILITY ARE VARIED, BASED ON PROCESSING THE (A) UCI HAR DATASET (B) DSA DATASET

(a)

	0.1	0.3	0.5	0.7	0.9
32	90.83 ± 1.01	91.37 ± 0.63	91.20 ± 0.61	89.97 ± 1.14	85.32 ± 0.93
64	92.67 ± 0.53	92.25 ± 0.56	92.75 ± 0.20	92.47 ± 0.30	89.03 ± 0.74
128	92.87 ± 0.32	93.68 ± 0.46	**93.71 ± 0.56**	93.65 ± 0.93	90.60 ± 0.78
256	93.67 ± 0.75	**93.88 ± 0.67**	**93.89 ± 0.52**	93.62 ± 0.60	92.65 ± 0.67
512	93.78 ± 0.72	93.81 ± 0.95	93.61 ± 0.58	92.77 ± 0.25	91.77 ± 0.62
1024	93.50 ± 0.55	90.78 ± 3.31	92.80 ± 1.04	93.25 ± 1.10	91.63 ± 0.30
2048	88.89 ± 9.43	91.73 ± 2.08	92.38 ± 1.30	93.18 ± 0.54	92.91 ± 1.19
4096	92.56 ± 0.48	92.21 ± 1.87	92.21 ± 0.71	92.77 ± 0.57	91.26 ± 1.07

(b)

	0.1	0.3	0.5	0.7	0.9
8	88.60 ± 0.85	88.08 ± 0.74	86.45 ± 0.45	83.73 ± 0.79	72.89 ± 2.51
16	93.54 ± 0.64	93.10 ± 0.09	93.24 ± 0.76	92.88 ± 0.64	83.96 ± 0.79
32	94.55 ± 0.49	94.74 ± 0.57	94.82 ± 0.34	94.45 ± 0.10	90.62 ± 0.24
64	95.05 ± 0.65	95.12 ± 0.35	95.71 ± 0.31	95.95 ± 0.53	93.89 ± 0.41
128	96.03 ± 0.32	96.02 ± 0.18	96.26 ± 0.21	**96.32 ± 0.29**	94.56 ± 0.42
256	96.21 ± 0.40	96.33 ± 0.45	96.63 ± 0.27	96.65 ± 0.21	95.41 ± 0.64
512	96.29 ± 0.44	96.15 ± 0.63	96.75 ± 0.25	96.81 ± 0.29	95.62 ± 0.37
1024	96.18 ± 1.77	**96.95 ± 0.23**	96.72 ± 0.45	96.54 ± 0.39	95.38 ± 0.54
2048	96.83 ± 0.15	96.29 ± 0.57	96.70 ± 0.42	96.67 ± 0.41	95.37 ± 0.25
4096	96.91 ± 0.33	96.70 ± 0.23	96.56 ± 0.37	96.68 ± 0.41	95.39 ± 0.53
8192	96.91 ± 0.26	96.31 ± 0.46	96.73 ± 0.22	96.63 ± 0.30	94.94 ± 0.43

While training the weights of the newly added layers of the transfer model for UIR, we label the activity recordings with the corresponding user labels and provide them as input to the network. In doing this, we have only processed the data recorded from the dynamic activities of the UCI HAR Dataset. When training the transfer model developed for the DSA Dataset, we have used data from all the activity types of that dataset.

2) Hyperparameter Selection of the Transfer Model: The newly added layers of the DL network in the UIR domain have two hyperparameters to be selected which are the number of dense layer neurons (DLNs) and the dropout probability. Since there is a small number of hyperparameters (only two), we have conducted a grid search to select suitable values for these hyperparameters. Table III displays the UIR accuracies as we vary the values of the two hyperparameters while processing each dataset. For the UCI HAR Dataset [Table III (a)], DLN numbers of 128 and 256 result in the highest accuracy values that do not differ much. Increasing the number of DLNs beyond 512 causes accuracy degradation. We achieve the

TABLE IV
NUMBER OF PARAMETERS, TRAINING TIME, AND THE UIR ACCURACY OF THE FULLY TRAINED VERSUS TRANSFER NETWORK FROM THE HAR DOMAIN TO THE UIR DOMAIN, BASED ON PROCESSING THE (A) UCI HAR DATASET (B) DSA DATASET

(a)

	fully trained	transfer
Number of Parameters	1,367,966	1,277,726
Training Time (s)	1,295	88
UIR accuracy ± std (%)	96.83 ± 0.31	94.38 ± 0.09

(b)

	fully trained	transfer
Number of Parameters	1,533,764	950,408
Training Time (s)	4,329	126
UIR accuracy ± std (%)	94.59 ± 0.16	96.15 ± 0.62

maximum accuracy of 93.89% with 256 DLNs and a dropout probability of 0.5. Considering the small differences between the high accuracy values, we have set the number of DLNs equal to 128 to keep the network complexity low. As for the DSA Dataset [Table III (b)], DLN numbers greater than or equal to 128 result in the best accuracy values, again with rather small differences between them. With the same considerations as for the UCI HAR Dataset, we have once again set the number of DLNs equal to 128. Table III also indicates that the dropout probability values of 0.5 and 0.7 are suitable choices for the UCI HAR and the DSA Datasets, respectively.

3) Comparison of the Fully Trained and Transfer Models: After identifying the salient features to be transferred to the UIR domain and selecting suitable hyperparameter values for the newly added layers of the transfer model, next, we evaluate the performance of the UIR model that exploits the transferred features.

Table IV shows the comparison between the fully trained and transfer models for each dataset. Transfer models diminish the number of parameters of the DL models by 6.60% for the UCI HAR Dataset and by 38.03% for the DSA Dataset. More importantly, using transfer models results in considerably reduced training times. For the UCI HAR Dataset, the training time of the transfer model is 14.7 times less than that of the fully trained one, while for the DSA Dataset, it is 34.4 times lower. The main reason for the reduced training times is, obviously, the elimination of the need to train the transfer layers whose weights are frozen.

To provide a more quantitative result on the training times, we have measured the time spent for training each layer of the transfer networks approximately. To do this, we have excluded the layers of each transfer network one at a time and measured the time for training the remainder of the network. The difference between the training time of the complete transfer network and the measured value approximately corresponds to the time required for training the excluded layer. The results displayed in Table V indicate that bulk of the training time is spent on the LSTM layers. This is because LSTM cells require a large number of matrix operations to compute the cell outputs. Besides, they operate recurrently which multiplies the amount of computation time with the number of time steps in the input sequences. Therefore, we can state that the transfer networks achieve reduction in the training times mainly by eliminating the need to train the LSTM layers via freezing their weights.

TABLE V
TRAINING TIME OF EACH LAYER OF THE TRANSFER NETWORKS, BASED ON PROCESSING THE (A) UCI HAR DATASET (B) DSA DATASET

(a)

Layers	Training Time (s)
2D CNN	128
dense 1	61
LSTM 1	425
LSTM 2	430
LSTM 3	452
dense 2	22

(b)

Layers	Training Time (s)
2D CNN	285
dense 1	24
LSTM	3,917
dense 2	19
dense 3	25

Referring to Table IV again, we observe that besides the lower number of parameters and reduced training times, the differences between the classification accuracies of the fully trained and transfer networks are quite small. While the accuracy of the transfer model is lower than that of the fully trained model by 2.45% in processing the UCI HAR Dataset, it is even higher for the DSA Dataset by 1.56%. This could be for the same two reasons given in Section IV-A for the higher UIR accuracies obtained with the DSA Dataset in Table I(b).

Parts (a) and (c) of Fig. 5 display the accuracy versus epoch number plots during the training process of the fully trained and transfer networks for the two datasets. Both parts of the figure illustrate that the transfer models converge to maximum accuracy values in earlier epochs compared to the fully trained ones. This is because there is a smaller number of layers and parameters to be trained in the transfer networks. Consequently, the Adam optimizer can reach the optimal loss point in earlier epochs.

Parts (b) and (d) of Fig. 5 depict the accuracy versus time plots for the two datasets. Since the time-consuming task of training the LSTM layers of the fully trained models is eliminated, the training time spent on each epoch is reduced considerably. As a result, the transfer models reach the maximum accuracy level much faster compared to the fully trained ones.

4) Binary Classifier for Selecting Activities Informative for UIR: We have achieved accurate UIR with fully trained and transfer networks where the transfer networks require significantly less training time and lower number of parameters. However, in the real world, users do not necessarily perform activities that are discerning for UIR. Therefore, UIR algorithms should be able to distinguish between activity signals that are informative and noninformative for the UIR task. Based on the results in Table I(a), we have labeled the dynamic (walking-related) activities of the UCI HAR Dataset as informative and the remaining three activities (which are static) as noninformative for UIR. Referring to the results in Table I (b), we have labeled 16 out of 19 activities of the DSA Dataset as informative and considered the activities of moving around in an elevator, rowing, jumping, and playing basketball as noninformative. With activities labeled as such, we train a binary classifier model to categorize activities into two classes as informative and noninformative.

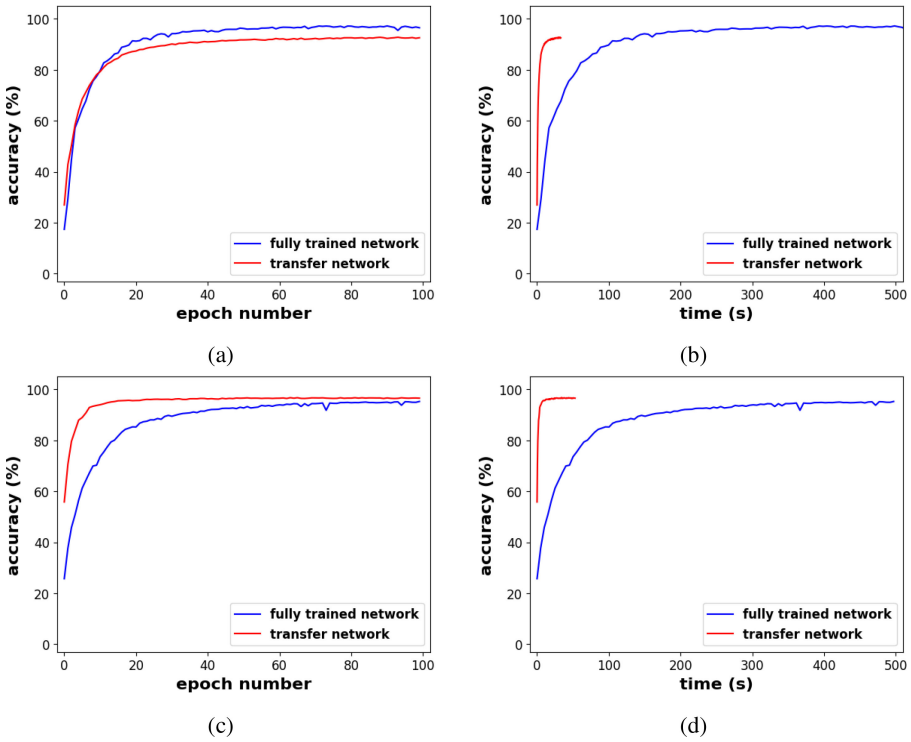

Fig. 5. Convergence of UIR accuracy with respect to epoch numbers and time in TL from HAR to UIR domain. Parts (a), (b): UCI HAR Dataset results; (c), (d): DSA Dataset results.

TABLE VI
NUMBER OF PARAMETERS, TRAINING TIME, AND THE CLASSIFICATION ACCURACY OF THE BINARY CLASSIFIER OBTAINED BY PROCESSING EACH OF THE TWO DATASETS

	UCI HAR Dataset	DSA Dataset
Number of Parameters	79,410	59,306
Training Time (s)	13	11
Accuracy ± std (%)	100.00 ± 0.00	98.92 ± 0.15

The proposed binary classifier employs the transferred features that the frozen parts of the models in Fig. 4 have extracted. In processing both datasets, we have used the same structure, shown in Fig. 6. We have kept the layers of the binary classifier simple by including only a single dense layer in each branch. Each dense layer contains eight neurons and the binary models trained with the UCI HAR and the DSA Datasets have a total of 79,410 and 59,306 parameters, respectively. We have classified the concatenated features with a softmax layer. Table VI displays the number of parameters, training time, and the accuracies of the binary classifers trained using the two datasets. We have achieved the accuracy figures of 100.00% and 98.92% by processing the UCI HAR and the DSA Datasets, respectively. If the recorded activity signal is classified as informative for UIR, then the proposed UIR module is activated. If not, the activity signal is considered noninformative and discarded. We illustrate the UIR scheme in Fig. 7.

5) Comparison with a Recent Study: As reviewed in Section II-A, a limited number of studies have addressed the UIR problem by processing data acquired from wearable motion sensors. Reference [50] presents a novel framework for

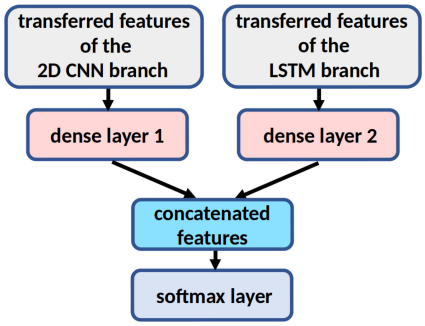

Fig. 6. Structure of the binary classifier designed to identify activities that are informative for UIR.

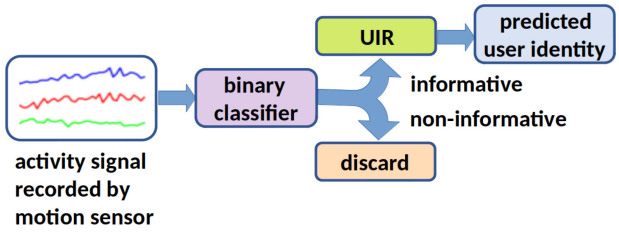

Fig. 7. UIR scheme with wearable motion sensor data.

multiclass UIR based on recognizing daily activities through the use of wearables and DL models. The study employs a hierarchical ensemble of classifiers as a DL model for UIR and tests it using the UCI HAR and the USC HAD Datasets (USC HAD: University of Southern California Human Activity Dataset). This allows us to compare our results based on the

TABLE VII
COMPARISON OF THE UIR ACCURACY PLUS/MINUS ONE STANDARD DEVIATION RESULTS OF THE FULLY TRAINED AND TRANSFER MODELS WITH THOSE IN [50], BASED ON PROCESSING THE UCI HAR DATASET

Activity Category	Activity Name	Our Fully Trained Model Accuracy ± std (%)	Our Transfer Model Accuracy ± std (%)	1D CNN-LSTM Model [50] Accuracy ± std (%)
static	sitting	48.28 ± 2.53	30.71 ± 1.74	55.74 ± 2.28
	standing	51.26 ± 3.32	41.68 ± 1.30	62.18 ± 2.02
	laying down	73.84 ± 3.26	63.94 ± 1.86	70.44 ± 2.01
dynamic	walking	98.84 ± 0.32	98.49 ± 0.50	94.57 ± 3.26
	walking upstairs	99.05 ± 0.22	95.52 ± 0.42	92.98 ± 0.89
	walking downstairs	95.59 ± 0.82	91.28 ± 0.63	89.78 ± 1.67
dynamic	walking-related activities combined	**96.83 ± 0.31**	94.38 ± 0.09	92.44 ± 1.94

TABLE VIII
HAR ACCURACY PLUS/MINUS ONE STANDARD DEVIATION (%) WITH FEATURES EXTRACTED FROM EACH LAYER COMBINATION OF THE MODEL'S TWO BRANCHES, BASED ON PROCESSING THE (A) UCI HAR DATASET (B) DSA DATASET

(a)

Branch Layers	LSTM 1	dropout 4	LSTM 2	dropout 5	LSTM 3	dropout 6	dense 3
2D CNN 1	92.33 ± 0.30	92.29 ± 0.71	90.74 ± 0.67	90.69 ± 0.41	87.82 ± 0.58	87.77 ± 0.82	88.21 ± 0.70
dropout 1	92.37 ± 0.44	**92.39 ± 0.32**	91.15 ± 0.33	90.46 ± 0.75	87.60 ± 0.75	88.57 ± 0.67	87.73 ± 1.03
2D CNN 2	91.42 ± 0.75	91.57 ± 0.97	88.62 ± 0.50	88.52 ± 0.68	79.20 ± 0.97	79.28 ± 0.86	79.75 ± 0.61
dropout 2	91.66 ± 0.49	91.70 ± 0.55	88.48 ± 0.41	88.83 ± 1.16	78.22 ± 0.61	78.84 ± 0.39	79.47 ± 1.14
flatten	91.39 ± 0.60	91.72 ± 0.60	88.38 ± 0.94	88.13 ± 0.50	79.35 ± 1.10	78.18 ± 0.95	78.55 ± 0.82
dropout 3	91.86 ± 0.80	91.27 ± 0.66	89.15 ± 0.66	88.79 ± 0.93	79.77 ± 1.30	78.70 ± 0.51	79.65 ± 0.69
dense 1	88.60 ± 0.58	88.57 ± 0.64	83.28 ± 0.53	82.45 ± 0.89	67.06 ± 1.22	67.76 ± 0.74	67.47 ± 1.69
dense 2	83.76 ± 0.40	84.19 ± 0.53	77.10 ± 1.07	77.73 ± 0.90	56.47 ± 0.84	56.99 ± 1.66	55.28 ± 2.18

(b)

Branch Layers	LSTM	batch 2	dense 2
flatten	**88.67 ± 0.34**	88.24 ± 0.20	88.40 ± 0.21
batch 1	88.40 ± 0.48	88.62 ± 0.30	88.30 ± 0.54
dense 1	47.58 ± 0.40	47.44 ± 0.77	45.53 ± 1.37

Main Layers	Accuracy ± std (%)
concatenation	44.77 ± 0.35
dropout	51.59 ± 0.65
dense 3	28.29 ± 0.65

UCI HAR Dataset with those reported in [50] for the same dataset.

Activities listed in the first three rows of Table VII are static activities while those in the next three rows are dynamic (walking-related) activities. Our hybrid DL model exhibits superior performance with the dynamic activities both when their signals are used individually and collectively. These are the activities that achieve UIR with accuracies above 95.5% with our fully trained DL model and above 91.2% with our transfer model. Signals of the static activities are not as useful for UIR and are discarded, as depicted in Fig. 7.

The best performing model in [50] is the cascaded (series) 1D CNN-LSTM architecture which achieves 92.44% accuracy with the combination of the three walking-related dynamic activities (see the last row of Table VII). Our proposed model surpasses this score and achieves 96.83% and 94.38% accuracy, with the fully trained and transfer networks, respectively.

B. Feature Transfer from UIR Domain to HAR Domain

To this point, we have shown that the features extracted by our DL models in the HAR domain are transferable to the UIR domain. Thus, UIR models can directly use the shallow layers of our DL models, developed for the HAR task, as feature extractors in the UIR domain without further training. Now, we consider feature transfer in the reverse direction and investigate to what extent the features extracted by the DL models trained for the UIR task can be employed for HAR.

1) Selecting the Features to be Transferred: Fig. 8 illustrates the feature transfer from the UIR domain to the HAR domain. Note that the 2D CNN-LSTM models illustrated on the left sides of the figure are the same two models as on the left sides of Fig. 4(a) and (b). These models are taken to be the base models for both directions of TL to implement bidirectional feature transfer. Weights of the layers transferred from the UIR domain are frozen and the weights of the rest of the network are trained for the HAR task. Table VIII(a) shows the resulting accuracies for the features extracted at each layer in processing the UCI HAR Dataset. As in Table II, rows and columns, respectively, correspond to the 2D CNN and LSTM branches and the bottom right corner of the table represents the deepest layers. We achieve the maximum HAR accuracy of 92.39% when we employ the features extracted after dropout layers 1 and 4, which are the second shallowest layer of each branch of the network. This is unlike the case for feature transfer in the opposite direction (HAR to UIR). Recall that, more advanced features, extracted by the deeper layers of the LSTM branch, were more useful in knowledge transfer from the HAR domain to the UIR domain. This indicates that, compared to the HAR features, UIR features are getting to be domain specific faster, especially along the LSTM branch.

Table VIII(b) displays the HAR accuracy results with the features transferred from the UIR domain in processing the DSA Dataset. The highest accuracy of 88.67% is obtained by employing the features extracted by the flatten and LSTM layers which are the shallowest. The results in Table VIII(b) also support that the features extracted in the UIR domain are more domain specific while those in the HAR domain are more generalizable.

Comparing the maximum accuracy values in Tables II and VIII, we also note that the maximum UIR accuracies obtained in transferring features from the HAR

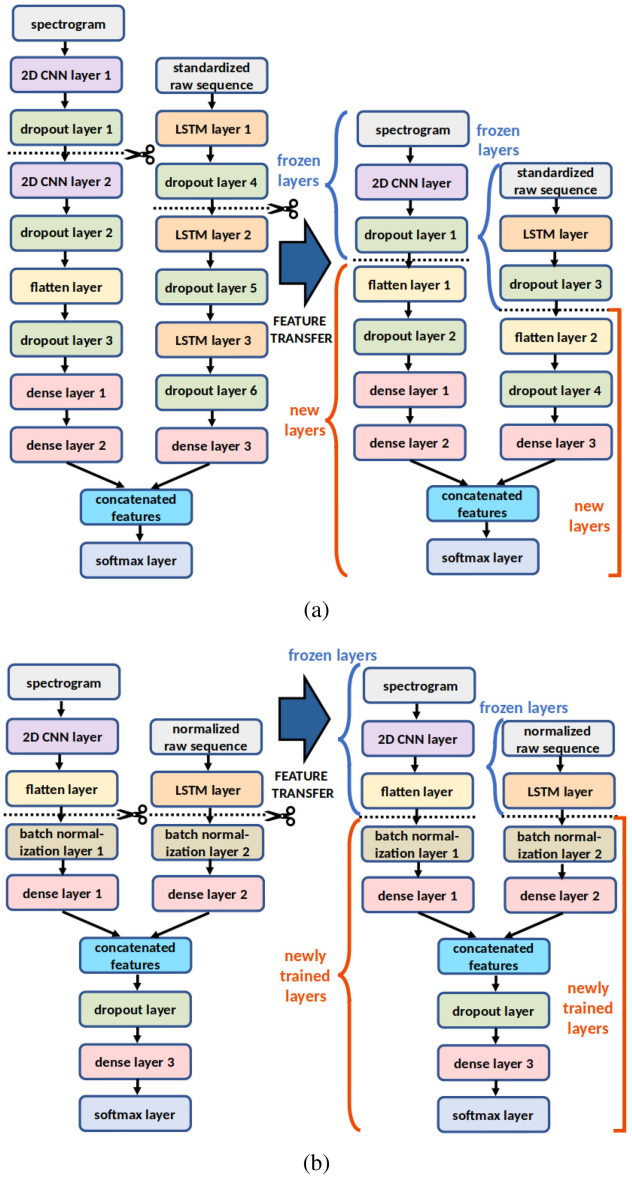

Fig. 8. TL from the UIR domain to the HAR domain for the (a) UCI HAR Dataset (b) DSA Dataset.

domain to the UIR domain (93.31% and 95.91%) are larger than those from the UIR domain to the HAR domain (92.39% and 88.67%), indicating that feature transfer in the former direction is more favorable and advantageous.

2) Hyperparameter Selection of the Transfer Model: To select suitable values for the two hyperparameters in the newly added layers of the DL models in the HAR domain, we have conducted another grid search. Table IX displays the HAR accuracies as the number of DLNs and the dropout probability are varied for the two datasets. For both datasets, we observe that the accuracy does not change much with the number of DLNs. This indicates that the learning capacities of the models are sufficient to learn the data features and are not improved by further increase in the number of DLNs. We selected the number of DLNs as 64 and 128 for the UCI HAR and the DSA Datasets, respectively. Although these are not the DLN numbers that correspond to the maximum accuracies, since

TABLE IX
HAR ACCURACY PLUS/MINUS ONE STANDARD DEVIATION (%) AS THE NUMBER OF DLNS AND THE DROPOUT PROBABILITY ARE VARIED, BASED ON PROCESSING THE (A) UCI HAR DATASET (B) DSA DATASET

(a)

	0.1	0.3	0.5	0.7	0.9
16	91.70 ± 0.51	92.05 ± 0.48	92.41 ± 0.13	91.97 ± 0.70	91.32 ± 0.81
32	92.08 ± 0.28	92.23 ± 0.24	92.45 ± 0.49	92.00 ± 0.49	91.78 ± 0.57
64	92.06 ± 0.23	92.40 ± 0.20	**92.46 ± 0.41**	92.20 ± 0.29	92.21 ± 0.40
128	92.52 ± 0.11	92.83 ± 0.24	92.32 ± 0.62	92.33 ± 0.68	91.87 ± 0.34
256	92.81 ± 0.18	92.43 ± 0.64	92.40 ± 0.27	92.17 ± 0.37	91.54 ± 0.19

(b)

	0.1	0.3	0.5	0.7	0.9
32	83.48 ± 0.16	84.09 ± 1.11	82.84 ± 0.85	79.62 ± 0.76	72.17 ± 1.14
64	86.77 ± 0.77	86.66 ± 0.69	86.92 ± 0.21	86.28 ± 1.12	82.77 ± 0.62
128	87.14 ± 0.31	87.12 ± 0.30	**87.69 ± 0.31**	87.65 ± 0.38	86.64 ± 0.45
256	87.66 ± 0.29	87.66 ± 0.24	87.78 ± 0.17	88.30 ± 0.37	87.82 ± 0.23
512	87.11 ± 0.33	87.22 ± 0.15	87.64 ± 0.15	87.91 ± 0.25	88.44 ± 0.20
1024	86.87 ± 0.42	86.81 ± 0.37	87.13 ± 0.27	87.78 ± 0.26	88.15 ± 0.41

TABLE X
NUMBER OF PARAMETERS, TRAINING TIME, AND THE HAR ACCURACY OF THE FULLY TRAINED VERSUS TRANSFER NETWORK FROM THE UIR DOMAIN TO THE HAR DOMAIN, BASED ON PROCESSING THE (A) UCI HAR DATASET (B) DSA DATASET

(a)

	fully trained	transfer
Number of Parameters	767,110	1,160,134
Training Time (s)	2,968	207
HAR accuracy ± std (%)	95.66 ± 0.63	93.67 ± 0.35

(b)

	fully trained	transfer
Number of Parameters	1,794,638	1,026,527
Training Time (s)	1,069	38
HAR accuracy ± std (%)	92.95 ± 0.47	87.57 ± 0.50

their resulting accuracies are quite close to the maximum, we have chosen them to keep the model complexity low. We have selected the dropout probability as 0.5 for both datasets since this value provides the highest accuracy values as presented in Table IX.

3) Comparison of the Fully Trained and Transfer Models: Table X shows the comparison between the fully trained and the transfer models in terms of the number of parameters, training time, and the HAR accuracy. Training times of the transfer networks are 15.7 and 8.7 times less than those of the fully trained networks based on processing the UCI HAR and the DSA Datasets, respectively. This considerable reduction in the training time is accompanied by accuracy values degraded by small margins (1.99% and 5.38%, respectively).

Parts (a) and (c) of Fig. 9 display the accuracy versus epoch number plots during the training process of the fully trained and transfer networks for the two datasets. Transfer networks enable higher accuracies in the initial epochs and faster and smoother convergence to the maximum accuracy values. We achieve this through TL by eliminating the need to train the shallow layers of the networks which are responsible for most of the computational cost.

Parts (b) and (d) of Fig. 9 depict the accuracy versus time plots during the training process of the two networks based on the two datasets. Again, the transfer networks exhibit larger accuracies, especially in the initial epochs, mainly because of their faster converge rate compared to the fully trained networks.

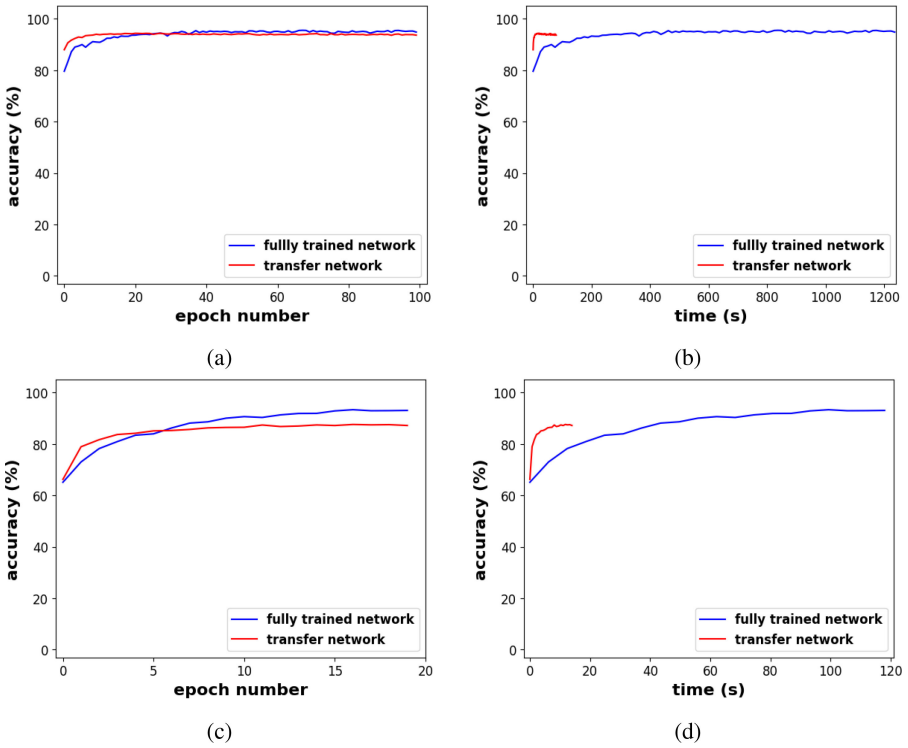

Fig. 9. Convergence of HAR accuracy with respect to epoch number and time in TL from UIR to HAR domain. Parts (a), (b): UCI HAR Dataset results; (c), (d): DSA Dataset results.

Considering the significant reduction in the training time and the tolerable difference between the accuracies we obtained with the fully trained and transfer networks, we can conclude that the features extracted in the UIR domain can be used in the HAR domain with satisfactory performance, especially if the computational resources are limited.

V. Comparison with SoTA Models

A. Layer Selection

In this section, we compare the performance of our proposed model with those of four of the six basic SoTA models that we have considered and implemented in our earlier work for the HAR task [14]. These are the 1D CNN, 2D CNN, LSTM, which are single models, and a 1D CNN-LSTM which is a hybrid model. In this standard hybrid model, the 1D CNN and LSTM networks are connected in series and are provided with raw time sequences as input. Note that all four comparison models are made up of only a single branch with a series combination of layers [14].

As before, we have conducted experiments to determine the optimal layer depth for feature transfer for each of the four comparison models by using the output of each layer as the features for the transfer network. We have selected the model layers whose features result in the maximum classification accuracy, as the feature extraction layers. For the proposed model, we have already provided the accuracy values at each layer combination in Tables II and VIII. Nevertheless, based on our experiments with all five models, we tabulate the indices of the best feature extractor layers in Table XI.

B. Hyperparameter Selection of the Transfer Model

We have conducted a grid search to tune the three hyperparameters of the transfer models, which are the learning rate, the number of DLNs, and the dropout probability, over the following sets of values.

1) Learning Rate: [0.01, 0.05, 0.001, 0.005, 0.0001, 0.0005].
2) Number of DLNs: $[2^5, 2^6, 2^7, 2^8, 2^9, 2^{10}, 2^{11}, 2^{12}]$.
3) Dropout Probability: [0.1, 0.3, 0.5, 0.7, 0.9].

We present the tuned hyperparameter values in Table XII. Whenever the accuracy values were similar, we have selected the smaller number of DLNs to avoid an unnecessary increase in the number of model parameters. Note that, in the last column of Table XII, we have also included the hyperparameter values used in the 2D CNN-LSTM model for completeness. The DLN and dropout probability hyperparameters in this column are previously optimized in Section IV-A2 (see Table III) and Section IV-B2 (see Table IX). For the learning rate hyperparameter of the HAR models, we have used the values optimized in [14] while we used 0.001 for the UIR models.

C. Results

The results in Tables XIII and XIV indicate that our DL model consistently provides more generalizable features compared to existing SoTA architectures, especially in processing the UCI HAR Dataset. In the tables, we have highlighted the largest two accuracy values using boldface fonts. In our earlier work on HAR, the second-ranking model was the 2D CNN [14], [15]. Consistent with that, in the present TL study, 2D CNN is again the second-ranking model, based on

TABLE XI
INDICES OF THE SELECTED FEATURE EXTRACTOR LAYERS OF THE MODELS IN TL [INDEXING STARTS WITH THE INPUT LAYER(S)]

Transfer Direction / Dataset	1D CNN	2D CNN	LSTM	1D CNN-LSTM	2D CNN-LSTM
HAR-to-UIR / UCI HAR	3	3	1	1	2 - 6
UIR-to-HAR / UCI HAR	3	1	1	1	2 - 2
HAR-to-UIR / DSA	1	1	1	1	2 - 3
UIR-to-HAR / DSA	1	1	1	1	2 - 1

TABLE XII
TUNED HYPERPARAMETER VALUES IN TL (LEARNING RATE, NUMBER OF DLNS, AND DROPOUT PROBABILITY)

Transfer Direction / Dataset	1D CNN	2D CNN	LSTM	1D CNN-LSTM	2D CNN-LSTM
HAR-to-UIR / UCI HAR	0.001, 64, 0.1	0.005, 64, 0.5	0.0005, 64, 0.1	0.0005, 64, 0.3	0.001, 128, 0.5
UIR-to-HAR / UCI HAR	0.005, 64, 0.7	0.001, 32, 0.1	0.0001, 32, 0.7	0.0001, 64, 0.3	0.001, 64, 0.5
HAR-to-UIR / DSA	0.001, 256, 0.7	0.0005, 64, 0.3	0.01, 512, 0.1	0.001, 128, 0.7	0.001, 128, 0.7
UIR-to-HAR / DSA	0.0005, 32, 0.7	0.0005, 64, 0.1	0.01, 512, 0.1	0.001, 32, 0.7	0.000041614, 128, 0.5

TABLE XIII
NUMBER OF PARAMETERS, TRAINING TIME, AND THE UIR ACCURACY PLUS/MINUS ONE STANDARD DEVIATION OF THE FULLY TRAINED VERSUS TRANSFER NETWORKS FROM THE HAR DOMAIN TO THE UIR DOMAIN, BASED ON PROCESSING THE (A) UCI HAR DATASET (B) DSA DATASET

(a)

Model Name	Number of Parameters	Training Time (s)	Accuracy $\pm$ std (%)
1D CNN fully trained	1,032,611	536	82.56 $\pm$ 1.61
1D CNN transfer	616,414	61	54.19 $\pm$ 1.37
2D CNN fully trained	314,382	300	**95.48 $\pm$ 0.59**
2D CNN transfer	228,702	75	86.26 $\pm$ 0.65
LSTM fully trained	2,504,951	10,752	94.33 $\pm$ 1.83
LSTM transfer	3,885,022	300	78.05 $\pm$ 1.01
1D CNN-LSTM fully trained	1,638,356	1,450	92.33 $\pm$ 1.11
1D CNN-LSTM transfer	1,476,574	119	49.02 $\pm$ 1.44
proposed 2D CNN-LSTM fully trained	1,367,966	1,295	**96.83 $\pm$ 0.31**
proposed 2D CNN-LSTM transfer	1,277,726	88	94.38 $\pm$ 0.09

(b)

Model Name	Number of Parameters	Training Time (s)	Accuracy $\pm$ std (%)
1D CNN fully trained	5,379,651	915	91.68 $\pm$ 0.55
1D CNN transfer	2,658,312	310	72.00 $\pm$ 0.90
2D CNN fully trained	219,899	174	**95.46 $\pm$ 0.40**
2D CNN transfer	187,208	60	95.27 $\pm$ 0.38
LSTM fully trained	198,303	1,909	91.48 $\pm$ 1.03
LSTM transfer	85,000	42	65.73 $\pm$ 0.79
1D CNN-LSTM fully trained	753,969	486	90.80 $\pm$ 1.12
1D CNN-LSTM transfer	1,057,160	152	70.12 $\pm$ 1.46
proposed 2D CNN-LSTM fully trained	1,533,764	4,329	94.59 $\pm$ 0.16
proposed 2D CNN-LSTM transfer	950,408	126	**96.15 $\pm$ 0.62**

the accuracy results. The proposed model and the 2D CNN are superior in generalizable feature extraction. The effect of TL is, in general, to reduce the training time considerably. In most cases, we also observe lower number of model parameters.

VI. IMPLEMENTATION AND RESOURCES

Some of the hyperparameters of the model layers, such as the optimizer, loss and activation functions can be selected through common practice. We have chosen to use the optimizer *Adam* and the loss function *Categorical Cross Entropy* in developing our hybrid DL model. While processing the UCI HAR Dataset, we have employed the sigmoid and ReLU activation functions for the dense and 2D CNN layers, respectively. However, in processing the DSA Dataset, since 2D CNN layers without any activation functions yielded better classification results, we have not used any activation functions in those layers.

We have implemented both the fully trained and transfer models using Python's Keras and Tensorflow libraries. Because hyperparameter tuning, training multiple models, and processing the datasets are all computationally demanding tasks, we have employed the T4 GPU with 15-GB RAM provided by Google Colab platform. Our findings indicate that the proposed DL model and feature transfer approach enable rapid model training within a few minutes (for a single model) on a standard CPU, eliminating the need for a GPU. For time measurements, we have used the CPU provided by the Amazon web services (AWS) platform with an EC2 p2.xlarge instance, as this platform offers more consistent time measurements.

VII. DISCUSSION AND CONCLUSION

We have implemented bidirectional TL between the HAR and UIR domains employing our recently proposed 2D CNN-LSTM hybrid DL model. The high-accuracy HAR and UIR results that we have achieved demonstrate the feasibility of effective feature transfer between these two domains

TABLE XIV
NUMBER OF PARAMETERS, TRAINING TIME, AND THE HAR ACCURACY PLUS/MINUS ONE STANDARD DEVIATION OF THE FULLY TRAINED VERSUS TRANSFER NETWORKS FROM THE UIR DOMAIN TO THE HAR DOMAIN, BASED ON PROCESSING THE (A) UCI HAR DATASET (B) DSA DATASET

(a)

Model Name	Number of Parameters	Training Time (s)	Accuracy ± std (%)
1D CNN fully trained	5,402,340	1,182	90.93 ± 0.50
1D CNN transfer	614,854	124	76.43 ± 0.45
2D CNN fully trained	302,976	346	93.21 ± 1.60
2D CNN transfer	34,726	44	83.35 ± 0.74
LSTM fully trained	2,498,327	23,307	90.03 ± 0.68
LSTM transfer	1,941,734	356	81.71 ± 0.53
1D CNN-LSTM fully trained	1,630,916	3,079	91.21 ± 0.57
1D CNN-LSTM transfer	1,475,014	201	81.54 ± 0.36
proposed 2D CNN-LSTM fully trained	767,110	2,968	**95.66 ± 0.63**
proposed 2D CNN-LSTM transfer	1,160,134	207	**93.67 ± 0.35**

(b)

Model Name	Number of Parameters	Training Time (s)	Accuracy ± std (%)
1D CNN fully trained	5,404,163	229	85.75 ± 0.75
1D CNN transfer	332,659	22	65.06 ± 0.39
2D CNN fully trained	227,293	37	**89.77 ± 0.97**
2D CNN transfer	187,923	14	84.54 ± 0.67
LSTM fully trained	202,929	413	78.72 ± 1.43
LSTM transfer	90,643	10	58.20 ± 0.54
1D CNN-LSTM fully trained	754,419	106	84.28 ± 1.76
1D CNN-LSTM transfer	264,659	19	63.43 ± 0.63
proposed 2D CNN-LSTM fully trained	1,794,638	1,069	**92.95 ± 0.47**
proposed 2D CNN-LSTM transfer	1,026,527	38	87.57 ± 0.50

in both directions. This work is the first to attain successful, high-accuracy TL between the HAR and UIR domains, making a notable contribution in wearable sensing research.

One of the key achievements of this work is illustrating how a model pretrained in one domain can support training in another domain, leveraging existing models to reduce the need for training from scratch. This capability is particularly valuable when a pretrained model does not exist for one task but available for another, demonstrating a practical approach to resource efficiency in model training.

To pave the way for TL between the HAR and UIR domains, we have effectively developed a unified encoder that can extract features jointly for the HAR and UIR tasks. This approach eliminates the need to use two separate encoders for feature extraction and the need to retrain the whole model from scratch. As a result, the model architecture is simplified, training times, and required memory space and energy are considerably reduced, while an acceptable level of accuracy is maintained. Thus, the overall complexity of the model is lower at the expense of a small degradation in classification accuracy, pointing to the tradeoff between complexity and accuracy. Such a lightweight memory-efficient model that consumes less energy is well-suited for deployment on resource-constrained edge devices. This is the main motivation and reason for prioritizing training and memory efficiency over accuracy in this study. Thus, we stress the practical benefits of our proposed method in resource-limited environments. Besides the use of a shared encoder being a tangible contribution of our work, our results provide foundation for future studies to explore additional practical advantages—such as online learning on edge devices for retraining the model on new users aiming better adaptation—of such a unified learning framework.

Bidirectional transferability of features between two distinct domains (HAR and UIR) further validates the superiority of the 2D CNN-LSTM model that we have originally proposed in [14] and [15] and employed in this study for feature extraction and transfer. We have compared the performances of the best five out of the seven SoTA models that we have previously implemented and evaluated [14]. Our recently proposed hybrid DL model, which was among them, demonstrates superior performance in bidirectional TL as well. Unlike existing basic DL models which struggle to extract transferable features for the HAR and UIR tasks, our model successfully extracts and transfers robust features, exploitable across both domains.

The main benefit of feature transfer is the significant reduction in the training times of the DL models. Besides, we also observe a moderate reduction in the number of model parameters. Feature transfer from the HAR domain to the UIR domain shortens the training process by a factor of 14.7 and 34.4 for the UCI HAR and DSA Datasets, respectively, while maintaining the accuracy at acceptable levels. Respective factors for feature transfer from the UIR domain to the HAR domain are 14.3 and 28.1, with a small accuracy degradation compared to the fully trained models. Overall performance is superior and performance gaps between the fully trained and transfer networks are smaller for feature transfer from the HAR domain to the UIR domain. This indicates that it is possible to extract more generalizable features in the HAR domain compared to the UIR domain, making feature transfer in this direction more favorable and advantageous.

The key technical challenge of this work has been maintaining an acceptable level of accuracy in activity/user recognition while benefiting from the advantages of TL.

An additional contribution of this work is the binary classifier module that we have integrated in our end-to-end

UIR system to identify the activity signals that are particularly informative for UIR. Inclusion of such a binary classifier further enhances the UIR accuracy. We underline that we have trained separate UIR models with different feature subsets only as an off-line, one-time feasibility analysis task to identify which features are most suitable for transfer. This procedure is not intended to be part of the practical application pipeline and does not need to be repeated. Once the informative features and activity types are identified, the results can be directly used in future applications without any additional computation.

The work presented in this article can be extended in several directions. One future research topic could be investigating whether the features extracted by our proposed DL network architecture have sufficient representation capacity for problem domains besides HAR and UIR or not. Features can be transferred across different domains to achieve cross-body part, cross-sensor, cross-user, cross-dataset, and cross-activity knowledge transfer. DL models need to be energy efficient for deployment on wearable edge devices that typically have limited battery life. Energy consumption of the DL models on embedded devices can be measured to develop models with lower energy requirements. Our current work is focused on reducing the complexity of the transfer models while maintaining an acceptable level of accuracy.

REFERENCES

[1] N. Niknejad, W. B. Ismail, A. Mardani, H. Liao, and I. Ghani, "A comprehensive overview of smart wearables: The state of the art literature, recent advances, and future challenges," *Eng. Appl. Artif. Intell.*, vol. 90, Art. no. 103529, Apr. 2020. [Online]. Available: https://doi.org/10.1016/j.engappai.2020.103529

[2] E. Ramanujam, T. Perumal, and S. Padmavathi, "Human activity recognition with smartphone and wearable sensors using deep learning techniques: A review," *IEEE Sensors J.*, vol. 21, no. 12, pp. 13029–13040, Jun. 2021. [Online]. Available: https://doi.org/10.1109/jsen.2021.3069927

[3] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: A survey," *Pattern Recognit. Lett.*, vol. 119, pp. 3–11, Mar. 2019. [Online]. Available: https://doi.org/10.1016/j.patrec.2018.02.010

[4] S. Zhang et al., "Deep learning in human activity recognition with wearable sensors: A review on advances," *Sensors*, vol. 22, no. 4, Art. no. 1476, Feb. 2022. [Online]. Available: https://doi.org/10.3390/s22041476

[5] L. Alawneh, M. Al-Zinati, and M. Al-Ayyoub, "User identification using deep learning and human activity mobile sensor data," *Int. J. Inf. Security*, vol. 22, pp. 289–301, Feb. 2023. [Online]. Available: https://doi.org/10.1007/s10207-022-00640-4

[6] J. A. Miranda et al., "Bindi: Affective Internet of Things to combat gender-based violence," *IEEE Internet Things J.*, vol. 9, no. 21, pp. 21174–21193, Nov. 2022. [Online]. Available: https://doi.org/10.1109/jiot.2022.3177256

[7] A. I. Middya, S. Roy, S. Mandal, and R. Talukdar, "Privacy protected user identification using deep learning for smartphone-based participatory sensing applications," *Neural Comput. Appl.*, vol. 33, pp. 17303–17313, Dec. 2021. [Online]. Available: https://doi.org/10.1007/s00521-021-06319-6

[8] E. Maiorana, "A survey on biometric recognition using wearable devices," *Pattern Recognit. Lett.*, vol. 156, pp. 29–37, Apr. 2022. [Online]. Available: https://doi.org/10.1016/j.patrec.2022.03.002

[9] S. Bianco, P. Napoletano, A. Raimondi, and M. Rima, "U-WeAr: User recognition on wearable devices through arm gesture," *IEEE Trans. Human-Mach. Syst.*, vol. 52, no. 4, pp. 713–724, Aug. 2022. [Online]. Available: https://doi.org/10.1109/thms.2022.3170829

[10] M. Senbekov et al., "The recent progress and applications of digital technologies in healthcare: A review," *Int. J. Telemed. Appl.*, vol. 2020, Art. no. 8830200, Dec. 2020. [Online]. Available: https://doi.org/10.1155/2020/8830200

[11] K. Fang et al., "IDRes: Identity-based respiration monitoring system for digital twins enabled healthcare," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 10, pp. 3333–3348, Oct. 2023. [Online]. Available: https://doi.org/10.1109/jsac.2023.3310095

[12] M. L. Suwandy, B. B. Zaidan, A. A. Zaidan, and A. S. Albahri, "Sensor-based mHealth authentication for real-time remote healthcare monitoring system: A multilayer systematic review," *J. Med. Syst.*, vol. 43, Art. no. 33, Jan. 2019. [Online]. Available: https://doi.org/10.1007/s10916-018-1149-5

[13] K. Chen, D. Zhang, L. Yao, B. Guo, Z. Yu, and Y. Liu, "Deep learning for sensor-based human activity recognition: Overview, challenges, and opportunities," *ACM Comput. Surveys*, vol. 54, no. 4, Art. no. 77, May 2021. [Online]. Available: https://doi.org/10.1145/3447744

[14] E. Koşar and B. Barshan, "A new CNN-LSTM architecture for activity recognition employing wearable motion sensor data: Enabling diverse feature extraction," *Eng. Appl. Artif. Intell.*, vol. 124, Art. no. 106529, Sep. 2023. [Online]. Available: https://doi.org/10.1016/j.engappai.2023.106529

[15] E. Koşar, "A memory efficient novel deep learning architecture enabling diverse feature extraction on wearable motion sensor data," M.S. thesis, Dept. Electr. Electron. Eng., Bilkent Univ., Ankara, Türkiye, Sep. 2022. [Online]. Available: Bilkent University Institutional Repository (BUIR) https://repository.bilkent.edu.tr/

[16] Z. Qin, P. Zhao, T. Zhuang, F. Deng, Y. Ding, and D. Chen, "A survey of identity recognition via data fusion and feature learning," *Inf. Fusion*, vol. 91, pp. 694–712, Mar. 2023. [Online]. Available: https://doi.org/10.1016/j.inffus.2022.10.032

[17] S. Liu, W. Shao, T. Li, W. Xu, and L. Song, "Recent advances in biometrics-based user authentication for wearable devices: A contemporary survey," *Digit. Signal Process.*, vol. 125, Art. no. 103120, Jun. 2022. [Online]. Available: https://doi.org/10.1016/j.dsp.2021.103120

[18] Y. Su, Y. Li, and Z. Cao, "Gait-based privacy protection for smart wearable devices," *IEEE Internet Things J.*, vol. 11, no. 2, pp. 3497–3509, Jan. 2024. [Online]. Available: https://doi.org/10.1109/jiot.2023.3296650

[19] A. I. Middya, S. Roy, and S. Mandal, "User recognition in participatory sensing systems using deep learning based on spectro-temporal representation of accelerometer signals," *Knowl.-Based Syst.*, vol. 258, Art. no. 110046, Dec. 2022. [Online]. Available: https://doi.org/10.1016/j.knosys.2022.110046

[20] C. Benegui and R. T. Ionescu, "Convolutional neural networks for user identification based on motion sensors represented as images," *IEEE Access*, vol. 8, pp. 61255–61266, Mar. 2020. [Online]. Available: https://doi.org/10.1109/access.2020.2984214

[21] S. Yi, Z. Mei, K. Ivanov, Z. Mei, T. He, and H. Zeng, "Gait-based identification using wearable multimodal sensing and attention neural networks," *Sens. Actuators A, Phys.*, vol. 374, Art. no. 115478, Aug. 2024. [Online]. Available: https://doi.org/10.1016/j.sna.2024.115478

[22] F. Luo, S. Khan, Y. Huang, and K. Wu, "Activity-based person identification using multimodal wearable sensor data," *IEEE Internet Things J.*, vol. 10, no. 2, pp. 1711–1723, Jan. 2023. [Online]. Available: https://doi.org/10.1109/jiot.2022.3209084

[23] S. R. V. Sudhakar, N. Kayastha, and K. Sha, "ActID: An efficient framework for activity sensor based user identification," *Comput. Security*, vol. 108, Art. no. 102319, Sep. 2021. [Online]. Available: https://doi.org/10.1016/j.cose.2021.102319

[24] Q. Li, Z. Yu, L. Yao, and B. Guo, "RLTIR: Activity-based interactive person identification via reinforcement learning tree," *IEEE Internet Things J.*, vol. 9, no. 6, pp. 4464–4475, Mar. 2022. [Online]. Available: https://doi.org/10.1109/jiot.2021.3104024

[25] A. S. Guinea, S. Heinrich, and M. Mühlhäuser, "Activity-free user identification using wearables based on vision techniques," *Sensors*, vol. 22, no. 19, Art. no. 7368, Sep. 2022. [Online]. Available: https://doi.org/10.3390/s22197368

[26] Z. Ahmad and N. Khan, "Inertial sensor data to image encoding for human action recognition," *IEEE Sensors J.*, vol. 21, no. 9, pp. 10978–10988, May 2021. [Online]. Available: https://doi.org/10.1109/jsen.2021.3062261

[27] F. J. Ordóñez and D. Roggen, "Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition," *Sensors*, vol. 16, no. 1, Art. no. 115, Jan. 2016. [Online]. Available: https://doi.org/10.3390/s16010115

[28] R. A. Hamad, L. Yang, W. L. Woo, and B. Wei, "Joint learning of temporal models to handle imbalanced data for human activity recognition," *Appl. Sci.*, vol. 10, no. 15, Art. no. 5293, Jul. 2020. [Online]. Available: https://doi.org/10.3390/app10155293

[29] N. Hernandez, J. Lundström, J. Favela, I. McChesney, and B. Arnrich, "Literature review on transfer learning for human activity recognition using mobile and wearable devices with environmental technology," *SN Comput. Sci.*, vol. 1, Art. no. 66, Feb. 2020. [Online]. Available: https://doi.org/10.1007/s42979-020-0070-4

[30] N. Agarwal, A. Sondhi, K. Chopra, and G. Singh, "Transfer learning: Survey and classification," in *Smart Innovations in Communication and Computational Sciences* (Advances in Intelligent Systems and Computing 1168), S. Tiwari, M. C. Trivedi, K. K. Mishra, A. K. Misra, K. K. Kumar, and E. Suryani, Eds. Singapore: Springer Nat., 2021, pp. 145–155. [Online]. Available: https://doi.org/10.1007/978-981-15-5345-5_13

[31] D. Cook, K. D. Feuz, and N. C. Krishnan, "Transfer learning for activity recognition: A survey," *Knowl. Inf. Syst.*, vol. 36, pp. 537–556, Sep. 2013. [Online]. Available: https://doi.org/10.1007/s10115-013-0665-3

[32] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. 13th Eur. Conf. Comput. Vis. (ECCV), Part I* (Lecture Notes in Computer Science 8689), D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., Cham, Switzerland: Springer, 2014, pp. 818–833. [Online]. Available: https://doi.org/10.1007/978-3-319-10590-1_53

[33] X. Du, K. Farrahi, and M. Niranjan, "Transfer learning across human activities using a cascade neural network architecture," in *Proc. 23rd ACM Int. Symp. Wearable Comput. (ISWC)*, London, U.K., Sep. 2019, pp. 35–44. [Online]. Available: https://doi.org/10.1145/3341163.3347730

[34] F. Zhuang et al., "A comprehensive survey on transfer learning," *Proc. IEEE*, vol. 109, no. 1, pp. 43–76, Jan. 2021. [Online]. Available: https://doi.org/10.1109/jproc.2020.3004555

[35] J. Wang, V. W. Zheng, Y. Chen, and M. Huang, "Deep transfer learning for cross-domain activity recognition," in *Proc. 3rd Int. Conf. Crowd Sci. Eng. (ICCSE)*, Singapore, Art. no. 16, Jul. 2018. [Online]. Available: https://doi.org/10.1145/3265689.3265705

[36] P. Alinia, I. Mirzadeh, and H. Ghasemzadeh, "ActiLabel: A combinatorial transfer learning framework for activity recognition," Mar. 2020, *arXiv:2003.07415v1*. [Online]. Available: https://doi.org/10.48550/arXiv.2003.07415

[37] A. Yurtman and B. Barshan, "Activity recognition invariant to sensor orientation with wearable motion sensors," *Sensors*, vol. 17, no. 8, Art. no. 1838, Aug. 2017. [Online]. Available: https://doi.org/10.3390/s17081838

[38] A. Yurtman, B. Barshan, and B. Fidan, "Activity recognition invariant to wearable sensor unit orientation using differential rotational transformations represented by quaternions," *Sensors*, vol. 18, no. 8, Art. no. 2725, Aug. 2018. [Online]. Available: https://doi.org/10.3390/s18082725

[39] B. Barshan and A. Yurtman, "Classifying daily and sports activities invariantly to the positioning of wearable motion sensor units," *IEEE Internet Things J.*, vol. 7, no. 6, pp. 4801–4815, Jun. 2020. [Online]. Available: https://doi.org/10.1109/jiot.2020.2969840

[40] A. Yurtman, B. Barshan, and S. Redif, "Position invariance for wearables: Interchangeability and single-unit usage via machine learning," *IEEE Internet Things J.*, vol. 8, no. 10, pp. 8328–8342, May 2021. [Online]. Available: https://doi.org/10.1109/jiot.2020.3044754

[41] A. Ray and M. H. Kolekar, "Transfer learning and its extensive appositeness in human activity recognition: A survey," *Expert Syst. Appl.*, vol. 240, Art. no. 122538, Apr. 2024. [Online]. Available: https://doi.org/10.1016/j.eswa.2023.122538

[42] B. Barshan and A. Yurtman, "Investigating inter-subject and inter-activity variations in activity recognition using wearable motion sensors," *Comput. J.*, vol. 59, no. 9, pp. 1345–1362, Sep. 2016. [Online]. Available: https://doi.org/10.1093/comjnl/bxv093

[43] Z. Fu, X. He, E. Wang, J. Huo, J. Huang, and D. Wu, "Personalized human activity recognition based on integrated wearable sensor and transfer learning," *Sensors*, vol. 21, no. 3, Art. no. 885, Jan. 2021. [Online]. Available: https://doi.org/10.3390/s21030885

[44] S. O. Bursa, O. D. Incel, and G. I. Alptekin, "Personalized and motion-based human activity recognition with transfer learning and compressed deep learning models," *Comput. Elect. Eng.*, vol. 109, Art. no. 108777, Jul. 2023. [Online]. Available: https://doi.org/10.1016/j.compeleceng.2023.108777

[45] Q. Jia, J. Guo, P. Yang, and Y. Yang, "A holistic multi-source transfer learning approach using wearable sensors for personalized daily activity recognition," *Complex Intell. Syst.*, vol. 10, pp. 1459–1471, Feb. 2024. [Online]. Available: https://doi.org/10.1007/s40747-023-01218-w

[46] J.-L. Reyes-Ortiz, D. Anguita, L. Oneto, and X. Parra, Jul. 2015, "Smartphone-based recognition of human activities and postural transitions," Dataset, UC Irvine Machine Learning Repository, School Inf. Comput. Sci., Univ. California at Irvine, Irvine, CA, USA. [Online]. Available: https://doi.org/10.24432/C54G7M

[47] J.-L. Reyes-Ortiz, L. Oneto, A. Samà, X. Parra, and D. Anguita, "Transition-aware human activity recognition using smartphones," *Neurocomputing*, vol. 171, pp. 754–767, Jan. 2016. [Online]. Available: https://doi.org/10.1016/j.neucom.2015.07.085

[48] K. Altun and B. Barshan, Feb. 2019, "Daily and sports activities data set," IEEE Data Port. [Online]. Available: https://doi.org/10.21227/at1v-6f84

[49] B. Barshan and K. Altun, Jul. 2013, "Daily and sports activities," Dataset, UC Irvine Machine Learning Repository, School Inf. Comput. Sci., Univ. California at Irvine, Irvine, CA, USA. [Online]. Available: https://doi.org/10.24432/C5C59F

[50] S. Mekruksavanich and A. Jitpattanakul, "Biometric user identification based on human activity recognition using wearable sensors: An experiment using deep learning models," *Electronics*, vol. 10, no. 3, Art. no. 308, Jan. 2021. [Online]. Available: https://doi.org/10.3390/electronics10030308

Billur Barshan received the B.S. degrees in electrical engineering and in physics from Boğaziçi University, Istanbul, Türkiye, in 1986, and the M.S., M.Phil., and Ph.D. degrees all in electrical engineering from Yale University, New Haven, CT, USA, in 1988, 1988, and 1991, respectively.

After working as a Postdoctoral Researcher with the Robotics Research Group, University of Oxford, Oxford, U.K., she joined Bilkent University, Ankara, Türkiye, as a Faculty Member where she is currently a Professor with the Department of Electrical and Electronics Engineering. Her current research interests include wearable sensing, wearable robots and mechanisms, motion capture and analysis, detection and classification of falls, machine/deep learning, pattern analysis and classification, and multisensor data fusion.

Dr. Barshan received the TÜBİTAK Incentive Award in 1998, the METU Mustafa Parlar Foundation Research Incentive Award in 1999, and two best paper awards. She serves on the Editorial Board of *Digital Signal Processing* journal (Elsevier) and recently served as a Guest Editor for IEEE SELECTED TOPICS IN SIGNAL PROCESSING.

Enes Koşar received the B.S. degrees in electrical and electronics engineering and in computer engineering from Koç University, Istanbul, Türkiye, in 2019 and 2020, respectively, and the M.S. degree in electrical and electronics engineering from Bilkent University, Ankara, Türkiye, in 2022, where he is currently pursuing the Ph.D. degree with the same department, conducting research on developing robust and computationally efficient machine/deep learning models for time-series signals.

Mr. Koşar is the recipient of graduate study fellowships from TÜBİTAK for Ph.D. in 2022 and for M.S. in 2019, and 5G and Beyond Graduate Support Program in 2020, as well as a merit-based full scholarship awarded by Koç University in 2013.