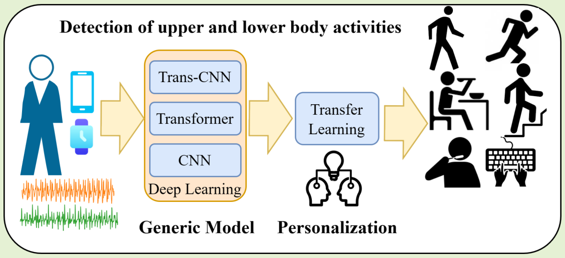
Deep Transfer Learning for Detection of Upper and Lower Body Movements: Transformer with Convolutional Neural Network

Kyle Lacroix, Davoud Gholamiangonabadi, Ana Luisa Trejos *Senior Member, IEEE*, Katarina Grolinger *Senior Member, IEEE*

***Abstract*—When humans repeat the same motion, the tendons, muscles, and nerves can be damaged, causing Repetitive Stress Injuries (RSI). If the repetitive motions that lead to RSI are recognized early, actions can be taken to prevent these injuries. As Human Activity Recognition (HAR) aims to identify activities employing wearable or environment sensors, HAR is the first step toward identifying repetitive motions. Deep learning models, such as Convolutional Neural Networks (CNNs), have seen great success in recognizing activities for participants whose data are used in the model training; however, their accuracy drops for new participants as people move in different ways. Moreover, most studies focus on lower body movement, while upper body movements are the main cause of RSI. On the other hand, in recent years, transformers have been dominating natural language processing, and have the potential to improve modelling in other domains involving sequential data such as HAR. Consequently, this paper combines a Transformer and CNN (Trans-CNN) for the recognition of upper and lower body movements. Transfer learning was employed to personalize the generic model for the target participant. The experiments demonstrate that the generic Trans-CNN outperforms the standalone transformer and CNN. The accuracy of the generic Trans-CNN for both upper and lower body movements improved from 69.6% to 92.4% when personalization was introduced. All models, irrespective of the algorithm, have more difficulty recognizing upper body than lower body movements. Nevertheless, the proposed personalized approach for the detection of upper and lower body movements represents significant progress toward RSI prevention.**



***Index Terms*—human activity recognition, personalized models, convolutional neural network, transformer model, transfer learning, deep learning**

I. Introduction

HUMANS have a tendency to repeat the same motion over and over, which can lead to Repetitive Stress Injuries (RSI) [1]. According to the Occupational Safety and Health Administration, RSI affects about 1.8 million workers per year [2], and in the USA, RSI costs $15 to $20 billion a year in workers' compensation [3].

Workers can perform the same job every day for years leading to an increased risk of developing RSI in the lower body, or even more commonly, in the upper body. Due to the gradual development of RSI, the warning signs are often ignored. As a result, if the symptoms are not treated, they may eventually become chronic and be detrimental to the worker's job performance, or even to their ability to carry out activities of daily living.

If a sensor system with a Machine Learning (ML) model could watch over a person and monitor their activities, the employee could be given advice as to how to adjust their behaviour to prevent injury when they are repeating motions that can lead to RSI. Human Activity Recognition (HAR) models attempt to detect the activity that a human is carrying out based on raw data from sensors [4], [5]. There are two types of HAR: vision-based and sensor-based [6]. Vision-based HAR uses video or image data, while sensor-based HAR uses time series data collected from sensors, such as accelerometers and/or gyroscopes [7]. Sensor-based HAR is

Manuscript received xxxx xx, 2023; accepted xxxx xx, 2023. Date of publication xxxx xx, 202x; date of current version xxxx xx, 202x. This work was supported in part by the Western's Industry 4.0 Network, in part by the Bell-Western 5G Research Centre, in part by the Vector Scholarship in Artificial Intelligence, provided through the Vector Institute (K. Lacroix), and in part by the Ontario Graduate Scholarship (K. Lacroix), and in part by the Canada Research Chairs Program under File 950-233069 (A.L. Trejos) and CRC-2022-00078 (K. Grolinger). Computation was enabled in part by the Digital Research Alliance of Canada. (Corresponding author Katarina Grolinger.)

K. Lacroix was with the Department of Electrical and Computer Engineering, Western University, London, ON N6A 5B9, Canada. (e-mail: klacroi6@uwo.ca)

D. Gholamiangonabadi is with the Department of Electrical and Computer Engineering, Western University, London, ON N6A 5B9, Canada. (e-mail: dgholam@uwo.ca)

A.L. Trejos is with the Department of Electrical and Computer Engineering and with the School of Biomedical Engineering, Western University, London, ON N6A 5B9, Canada (e-mail: atrejos@uwo.ca)

K. Grolinger is with the Department of Electrical and Computer Engineering, Western University, London, ON N6A 5B9, Canada (e-mail: kgroling@uwo.ca, Tel: 519-661-2111 ext. 81407)

more common, as vision-based methods require installation of cameras, limiting the area that can be monitored and raising concerns about privacy and intrusiveness.

Recognizing human activities has been done successfully with different ML models [5] such as Hidden Markov models [8], support vector machines [9], and deep learning [10]. However, these models rely on having all of the data upfront, which means that they cannot adapt to new participants without being retrained [11]. In general, these models perform well when they are applied to participants on which they have been trained, but their performance drops dramatically for new participants. This limitation arises from the fact that people move differently due to differences in body size, gender, age, and other physiological characteristics. Model personalization can address this problem by tailoring the model to a specific participant [12].

While HAR approaches have been successful with personalization techniques by customizing the model for the target subjects, they have not examined the recognition of upper body movements. They have either used data from only the lower body or data from both the upper and lower body together. Since the majority of RSI cases are caused by upper body movements, without a model that can accurately discriminate between upper and lower body movements, it will be difficult to prevent RSI using these methods.

Recently, transformers [13] have been dominating the Natural Language Processing (NLP) field, with applications such as ChatGPT. With the use of a self-attention mechanism, the transformers are capable of differentiating the significance of each part from the input sequence, leading to state-of-the-art results in NLP applications. In recent years, transformers have been adapted for use with time series data in applications such as fault detection [14], and therefore, they present a promising opportunity for advancing HAR.

Consequently, this paper proposes a personalized hybrid model, combining a Transformer with a Convolutional Neural Network (Trans-CNN), for the recognition of upper and lower body movements. The Convolutional Neural Network (CNN) model brings the ability to capture high-level spatial–temporal features, while the transformer contributes by efficiently capturing temporal dependencies. Personalization was added to customize the model for the target person, and the personalized model was examined on upper and lower body movements. The main contributions of this work are (i) combining transformer and CNN for HAR and personalizing the model for new participants, (ii) examining the ability of the HAR models to distinguish between various upper and lower body movements, and (iii) demonstrating that Trans-CNN is capable of recognizing upper and lower body activities.

The paper is organized as follows: Section II provides background information and discusses related work, Section III presents the methodology, Section IV explains the experiments, and Section V discusses results. Finally, Section VII concludes the paper.

II. Background and Related Work

This section first provides an overview of CNNs and transformers. Then, recent work in sensor-based human activity recognition and personalization approaches for HAR are reviewed.

A. Convolutional Neural Network and Transformer

Convolutional Neural Networks are a type of neural network optimized for processing grid-like data, such as images [15]. CNNs are able to learn features or internal representations of the input data (feature learning) automatically. In addition, CNNs can process one-dimensional (1D) sequence data, such as data from a gyroscope or an accelerometer. For sequence classification tasks, CNNs have the advantage of learning features directly from raw time series data, eliminating the need for manually engineering features [16].

A CNN model consists of three types of layers: convolution, pooling, and fully connected. The convolution layer has a matrix of learnable parameters, known as kernels or filters, which move across the input and perform the dot product with the input matrix to create the activation maps. The pooling layer reduces the computational complexity and dimensionality by downsampling. The data can pass through multiple convolution and pooling layer pairs as required before going to the fully connected layers. The final output of the pooling layer is then flattened such that all of the nodes of the fully connected layers are connected to the previous layer. The predictions are produced in the last classification layer. Lastly, backpropagation with gradient descent updates the kernels and weights in the convolution and fully connected layers.

The transformer model is an alternative approach for processing sequential data, such as those present in NLP tasks [13]. The self-attention mechanism allows the transformer to understand the relationships between words and to learn the importance of each word in the sequence, making it a dominant technique in NLP. As seen in Figure 1, the model consists of an encoder and a decoder, each one including multi-head self-attention, normalization, and feed-forward layers. In contrast to the encoder, the decoder starts with a masked multi-head attention layer, which ensures that the prediction for the current position only depends on the outputs of the previous positions, allowing the transformer to adopt a teacher-forcing learning procedure. Finally, a linear transformation with a $softmax$ function creates the output probabilities.

In many NLP applications, transformers have been outperforming Recurrent Neural Networks (RNNs) and other techniques in terms of accuracy and effectiveness [17]. In recent years, models that have been successful in NLP have also shown success in time series data [14]; therefore, the transformer model has great potential in HAR.

B. Related Work in Sensor-based HAR

In the past, HAR required custom hardware to collect sensor data. In comparison to other wearable devices, smartphones and smartwatches have the advantage of non-intrusive and convenient data collection [18]. This section presents an overview of ML approaches for sensor-based, smartphone and smartwatch HAR.

He et al. [19] used the discrete cosine transform, Principal Component Analysis (PCA), and Support Vector Machine

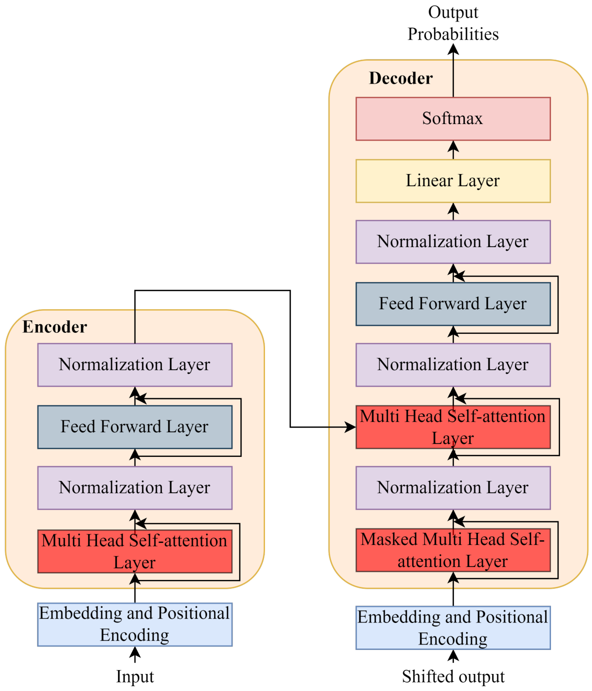


Fig. 1. Original transformer model for NLP tasks [13] consists of an encoder and a decoder, each one relying on the self-attention mechanism.

(SVM) to recognize human activity from a tri-axial accelerometer sensor. The participants had the tri-axial accelerometer sensor in their pocket and performed four activities for a minute each: walking, jumping, standing still, and running. After the data were collected, the discrete cosine transform extracted features from the data while PCA reduced the dimension of the features. The SVM model performed well for different activities. Also, a study by Alsheikh et al. [20] examined the differences between traditional models and deep models for HAR using tri-axial accelerometers. They used three datasets, which involved lower body activities, and proved that deep learning models are more accurate than traditional models, such as K-nearest neighbors and logistic regression.

Several studies investigated deep learning models, including CNNs and Long Short-Term Memory (LSTM). Chen et al. [21] placed a single-axis accelerometer on an individual to investigate HAR with CNN. The individuals performed seven common activities and the CNN-based HAR model achieved an accuracy of 93.8% without any feature extraction methods. A study by Pienaar et al. [22] looked at LSTM deep neural architecture for HAR. They used raw sensor data, and the network was capable of detecting all activities with an accuracy of 94%. Another study by Hendry et al. [23] employed a CNN model to take a closer look at the relationship between a dancer's training time and the dancer's pain. Dancing involves a lot of training such as jumping and landing, which can cause injuries such as foot/ankle, knee, and lower back pain if performed incorrectly. The authors proposed a CNN-based model with six ActiGraph Link wearable sensors, and their experiments showed that the approach can accurately detect different movements (jumping and lifting the leg).

Staczkiewicz et al. [24] reviewed over 100 articles to determine whether smartphone sensors were suitable for HAR, and found that smartphones are well-suited for such research in the health sciences. A study by Ronao et al. [25] looked into using CNN and a smartphone for HAR. In their study, the participants held a smartphone in their hand while performing the following activities: standing, walking, going upstairs, going downstairs, and running. For moving activities, CNN was impressively accurate and achieved a performance of 94.79% on the test set of raw sensor data.

Mekruksavanich et al. [26] examined the benefits of using a hybrid of LSTM and CNN for HAR. The participants carried a smartphone and a smartwatch and performed 18 activities of daily living. All of the models had their hyperparameters tuned with Bayesian optimization. The hybrid deep learning model outperformed other baseline models (only CNN and only LSTM) with an accuracy of 96.2%.

Luptáková et al. [27] took a closer look at adapting the transformer model for HAR. They used data from internal sensors (accelerometer and gyroscope) from smartphones. The activities considered in their study involved mainly lower body movements. The study concluded that the transformer model could identify the difference between the 18 activities with an accuracy of 99.2%.

Wang et al. [28] proposed a deep multi-feature extraction framework for recognizing human activities. Two feature extraction layers were used: the Channel and Spatial Attention Feature Extraction Layer (CSAFEL) and the Temporal Attention Feature Extraction Layer (TAFEL). CSAFEL, comprised of the convolutional block attention module and the residual network (ResNet-18), extracts channel and spatial features, while TAFEL, consisting of a bidirectional gated recurrent unit and self-attention mechanism, captures temporal features. These extracted features are fused and, after the fully connected layer, a SoftMax layer recognizes human activities. Their experiments show that the combination of deep learning neural networks increases the diversity of the extracted features and improves accuracy.

Similarly, Zhang et al. [29] proposed a combination of networks, specifically CNN and LSTM (DeepConvLSTM), for HAR. Their approach integrates a Squeeze and Excitation (SE) module and group convolutions to improve the extraction of both temporal and spatial features. The SE module focuses on recalibrating features by emphasizing informative ones and suppressing less useful ones, while group convolutions reduce the model's parameter count and computational complexity. The evaluation demonstrated improved accuracy and computational efficiency.

Finally, Imran et al. [30] integrated CNN and a Bidirectional Gated Recurrent Unit (BiGRU) to classify human activities based on inertial sensor data from smartwatches and smartphones. Their work emphasizes using the magnitude of three-dimensional (3D) acceleration for minimizing input space and enhancing computational efficiency. Evaluation on the WISDM dataset demonstrated that employing the magnitude of the signal, rather than the 3D signal itself, yields a reduction in computational intensity with minimal impact on accuracy.

While the reviewed studies made great strides towards accurate HAR, they produce a generic model that exhibits degradation when used by people not included in model training. Moreover, these studies focus primarily on lower body movement while RSI mainly results from repeating upper body movements. Also, a review of deep learning techniques for HAR [5] recognized the need to consider more complex and diverse activities. To address this gap, our work proposes the personalized Trans-CNN model for HAR by customizing the generic model for the target participant, and examines the ability of the model to distinguish upper body movements.

C. Related Work in Personalized Models for HAR

Personalization of the model is necessary since HAR techniques currently rely on user-independent models, which have difficulties in generalizing to new users. In other words, generic or user-independent models only work well on the participants on which they were trained, but their performance decreases greatly with new participants. The first phase in creating a personalized model is to create a generic model; this generic model is trained on many different participants to create a general understanding of the human motions that are being detected. Once the generic model is created, the personalization phase customizes the generic model to a given participant using different techniques.

Amrani et al. [31] investigated the use of incremental learning to create personalized models. The procedure included three phases: data preparation, training the generic model, and personalizing the model. The models examined included Learn++, ResNet, and CNN. They found that across all tested models, the accuracy increased from the generic model to the personalized model.

A study by Rokni et al. [11] looked at transfer learning for personalizing a CNN model. They initially trained the model on a group of participants to create the generic model. Once the generic model was created, they fixed the weights in all of the layers except the classification layer and further trained the model with three labelled instances per activity from the target participant. The evaluation was carried out on two different datasets, consisting of common lower body movements. Across both datasets, the personalized model achieved higher accuracy as compared to models trained using the traditional method.

Gholamiangonabadi and Grolinger [12] personalized a trained CNN for a target participant by selecting the best-suited model using a small fragment of the target participant's data. In their study, the frequency and time-domain features were extracted with linear and non-linear signal decomposition techniques. Personalization increased the accuracy from 85.2% to 91.2%.

The studies discussed in this subsection demonstrated different ways of personalizing ML models; however, they still focused on lower body movements without examining upper body movements. Also, our Trans-CNN technique takes advantage of transformers to improve the quality of the model.

III. Methodology

This paper proposes a personalized Trans-CNN model, a combination of a transformer and CNN, for detection of upper and lower body movements. The selection of the transformer is due to its self-attention mechanism that enables learning to focus on different segments of the input sequence [14]. Meanwhile, the CNN model was chosen for its exceptional generalization capabilities, feature extraction abilities, and recent accomplishments in HAR [32].

Figure 2 presents the overview of the complete process, while the remainder of this section provides details on the three main components: data preparation, model structuring, and model training.

A. Data Preparation

Data preparation is the process of transforming the raw data into data that can be used for ML algorithms, helping the ML model make better predictions. Here, data preparation consists of applying a sliding window technique and normalization.

In activity recognition, the sliding window technique is a widely employed technique to segment accelerometer or gyroscope data [33]. With this technique, sensor data are partitioned into fixed time slots [34]. The sliding window technique transforms time series data into data windows of $w \times f$ size, where w is the number of time steps, and f is the number of features. The first window starts at the beginning of the data and has a size of $w \times f$. The window then slides s time steps to create the next window, and so on. In other words, the sliding window technique is applied to help the model capture time dependencies.

After the sliding window technique, a standardization process scales data to have a mean of 0 and a standard deviation of 1. Standardization was selected to normalize the features because, in contrast to min–max normalization, this technique is not sensitive to outliers. Features are scaled as follows:

$$z = \frac{x - \alpha}{\sigma} \tag{1}$$

where x is the original feature value, α is the mean and σ is the standard deviation of the features, and z is the normalized value. Note that the normalization is carried out on the per-subject level.

As seen from Figure 2, the sliding window technique was applied first, followed by the separation of subjects: one subject (Subject x) was reserved for personalization and testing, while the remaining $M - 1$ subjects were used for training. Data from $M - 1$ subjects underwent normalization before being passed to the Model Structuring component for the training of generic deep learning models. On the other hand, data from Subject x underwent another split into two portions, $D1$ and $D2$. The split was $\frac{1}{3}$ of each class in $D1$, and the remaining $\frac{2}{3}$ in $D2$. Model Training used $D1$ to personalize the model by further training the model, and $D2$ to assess the performance of each model. Each part, $D1$ and $D2$, underwent normalization separately. For $D1$, the mean and standard deviation were calculated with $D1$ data, and then Equation 1 was applied. Normalization for $D2$ was carried

This article has been accepted for publication in IEEE Sensors Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2024.3451291

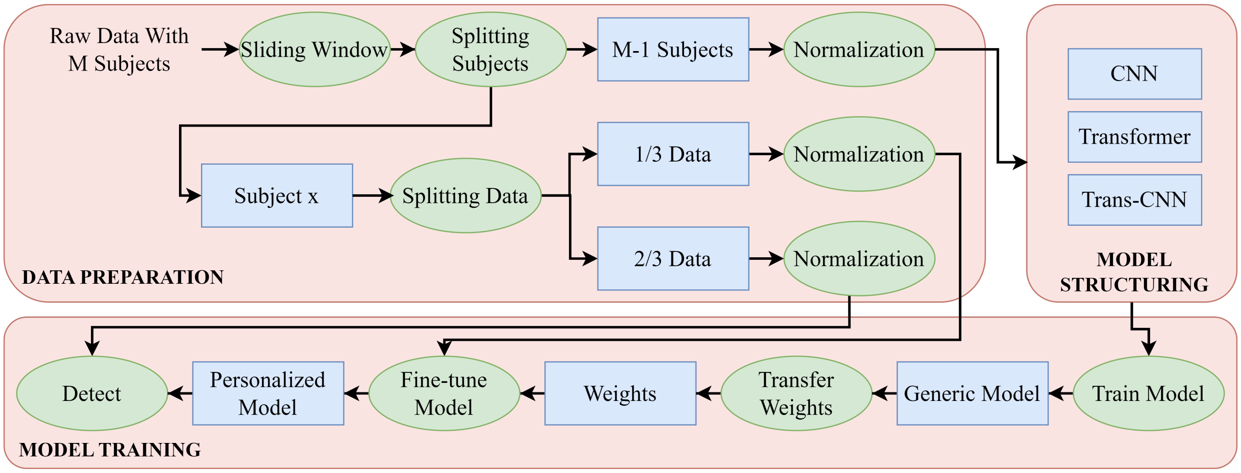


Fig. 2. The process of creating personalized ML models for HAR involves three main components: data preparation, model structuring, and model training. The same process is followed for each model: CNN, transformer, and Trans-CNN.

out with the mean and standard deviation calculated on $D1$ to avoid data leakage. Additional details on data preparation for the dataset used in the evaluation are provided in Section IV.

B. Model Structuring

The three models, CNN, transformer, and Trans-CNN, were all examined for their potential use as personalized models for HAR. While this section described the models, the tuned hyperparameters for each of the models are provided in Table I, Section IV.

1) CNN Structure: The CNN structure consists of different layers as shown in Figure 3. The input layer is followed by the convolutional block, which contains a convolutional layer, a max pooling layer, and a dropout layer. For CNN models, convolutional blocks are commonly stacked to ensure that the model has a hierarchical decomposition of the input. The convolutional layers have weights/kernels that are trained, while the max pooling layers reduce the dimensions of the feature maps. The dropout layer minimizes overfitting and the generalization error. After the last convolutional block, a flattening layer transforms the current output into a one-dimensional vector. Next, three fully connected layers were added to the CNN model to help interpret the features that were learned in convolutional blocks. The output of the last fully connected layer goes to the output layer. The output layer, which is a fully connected layer, outputs the predictions using $softmax$ as the activation function.

2) Transformer Structure: The modified transformer architecture for HAR can be seen in Figure 4. While the original transformer consists of an encoder and decoder, here, only the encoder is used in order to learn latent semantic representations and temporal dependencies. The traditional transformer commonly used in NLP consists of an encoder and a decoder. The encoder's role is to understand and capture temporal relationships within the input data while the decoder leverages the information provided by the encoder to generate the output sequence, such as a sentence in language translation and text generation. In HAR, data from sensors inherently contain temporal information, i.e., samples recorded over time, and spatial information such as multiple features recorded by the same or different sensors; for example, an accelerometer records x, y, and z acceleration over time. Therefore, the input to the transformer is a multi-feature time series $f \times w$, as prepared with the sliding window technique, where f is the number of features and w is the number of time steps in the sliding window. The encoder learns to extract and interpret temporal and spatial relationships from sensor readings, providing information for activity classification. The decoder is not necessary for HAR, as the nature of the classification task differs greatly from typical NLP tasks, such as translation or text generation, which involve generating sequences from representations provided by the encoder. In HAR, the encoder extracts information from multi-feature temporal data, and subsequently, fully connected layers can be used for the final classification.

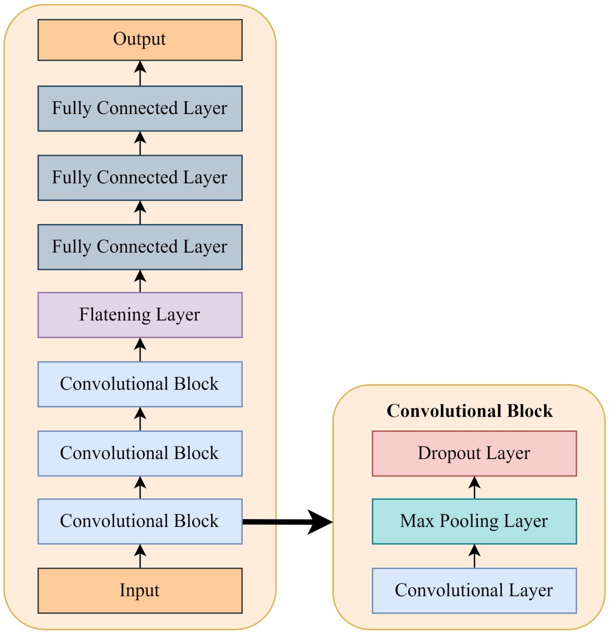


Fig. 3. The CNN model structure consists of three convolutional blocks followed by flattening and fully connected layers.

The modified architecture starts with an input layer followed by two stacked encoder blocks, each one consisting of

This article has been accepted for publication in IEEE Sensors Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2024.3451291

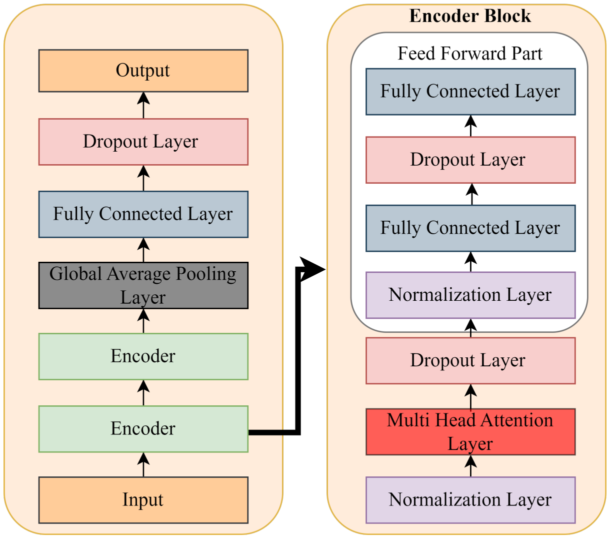


Fig. 4. The transformer model structure consists of two encoder blocks, followed by global average pooling, fully connected, and dropout layers.

normalization, multi head attention, dropout, normalization, fully connected, dropout, and fully connected layers, as seen in Figure 4. Following the mapping of the input to a higher dimensional space by the encoder blocks, a global average pooling layer is employed. Next, a fully connected layer, similar to the one in the CNN model, serves as a buffer from the learned features to the predictions. The output of the fully connected layer is passed to a dropout layer, which is added to help reduce overfitting. Lastly, the dropout layer is connected to a fully connected layer, which uses the $softmax$ function as an activation function for predictions.

3) Trans-CNN Structure: Trans-CNN is a hybrid model specifically designed for HAR in order to capitalize on the advantages of both models. CNN models have the ability to capture high-level spatial–temporal features, while transformers are efficient at capturing latent semantics and global dependencies [35]. By combing the two models, the Trans-CNN structure contains the major components of both models: the encoder block and the convolutional block.

The hybrid structure starts with an input layer, as seen in Figure 5. Similar to the transformer model structure, Trans-CNN has encoder blocks after the input layer. The output from the last encoder block is the input to the first convolutional block. Similar to the CNN model structure (Figure 3), in the convolutional block, after each convolutional layer, there is a max pooling layer. Unlike the CNN model structure, there is no dropout layer after the max pooling layer. The dropout layer was omitted since the encoders have two dropout layers inside of them. Next, the output from the convolutional block goes to a global average pooling layer. Similar to the transformer structure, only one fully connected layer follows the global average pooling layer. Lastly, a dropout layer and output layer generate the network output.

C. Model Training

After determining the structure of the three models, the models were trained. In traditional ML model training for HAR, some of the data from each participant are in the training dataset. However, when this traditional model is used with new participants, the performance of this model decreases, despite the fact that they work well for participants on which they were previously trained [32]. The decrease in performance with new participants comes from the diversity in humans. People can differ in body size, gender, age, and other physiological properties, which leads to the same activity being carried out by two participants in two different ways. As a result, the model cannot be easily adapted to new participants without retraining. In order to address this problem, ML models can be personalized to a particular participant; here, personalization involves learning how a particular participant moves. The creation of the personalized models was split into two parts: creating a general model and then personalizing that model to a certain participant; therefore, the model training consisted of two phases: the Generalization Phase, and the Personalization Phase.

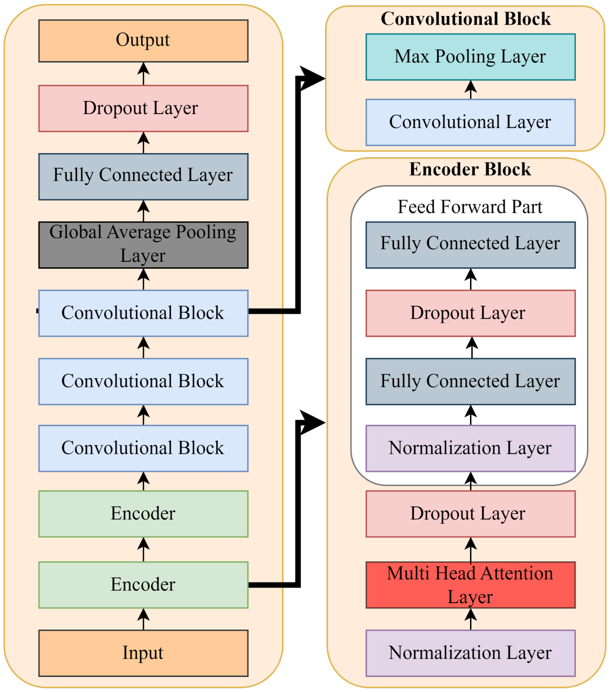


Fig. 5. The proposed hybrid Trans-CNN model structure consists of encoder and convolutional blocks followed by global average pooling, fully connected, and dropout layers.

1) Generalization Phase: The Generalization Phase consisted of training the three models, CNN, transformer, and Trans-CNN, as generic models. For a given dataset with M participants, the generic model was created by training the model using data from all $M - 1$ participants (excluding the target participant). This procedure was repeated with each of the M participants as the target. After training, the models became general models for human motion that provide a broad understanding of movement patterns in all participants but are less precise when it comes to a specific target participant.

2) Personalization Phase: Personalization provides a way to achieve better results for a particular participant. An individual's movement can be more easily detected when the model is personalized to that individual. Other studies such as Wu et al. [36] have successfully used transfer learning to personalize models. One way of performing transfer learning involves transferring the weights from the generic model to a new model that will be personalized. Here, CNN, transformer, and Trans-CNN are used to create the generic models, and these generic models provide the initial weights for the personalized model. Next, all of the layers are frozen except for the classification layer (last layer) to make sure that the knowledge gained from the other participants is preserved. Freezing layers during training prevents their weights from being modified; hence, the knowledge inside the frozen layers is untouched. Since the latter layers are typically learning task-specific features, the classification layer is the only layer that is not frozen. As described in Subsection III-A, $\frac{1}{3}$ of each class makes the $D1$ dataset, which is used to train the models after the layers are frozen. During this training, only the weights of the classification layer change, to help improve the precision of the model for the target participant. The remaining $\frac{2}{3}$ of the data from the target participant makes the $D2$ dataset, which is used to test each model. The complete process is repeated M times for each of the M participants acting as the target.

IV. Evaluation

In order to evaluate the proposed model, and to compare it to standard models, this section introduces the dataset and experiments.

A. Dataset

Several datasets [37], [38] are available for HAR; however, these datasets contain only lower body movements, and therefore are not suitable for this study, as the goal is to examine the detection of lower and upper body movements. Therefore, the WISDM 2019 [39] open-source dataset was chosen for evaluation, since it contains both upper and lower body movement data.

The data in this dataset have been collected using both a smartphone in the participant's pocket and a smartwatch on their dominant hand. Each device has a built-in gyroscope and accelerometer, which were used to collect data from the participants' movements while carrying out various activities. The data were collected from 51 participants who performed 18 different activities, including walking, sitting, and eating, for a period of three minutes for each activity. Three readings were collected from each sensor: the phone gyroscope, the phone accelerometer, the watch gyroscope, and the watch accelerometer. The three readings collected by each sensor are x, y, and z axis coordinates; therefore, a total of 12 readings (2 devices × sensors × 3 axes = 12) were available. The label for each activity was identified by a letter from A–S (no ‘N’) and each sensor collected the data at a rate of 20 Hz.

B. Experiments

The dataset was processed according to the methodology described in Section III. This section provides details of the data preparation as applied to the WISDM dataset, together with the model tuning process for each CNN, transformer, and Trans-CNN.

1) Data Preparation: Of 51 participants, 12 did not have recordings from all of the sensors for all 18 activities. For example, Participant 1607 had 18 activities recorded using the watch and only 17 using the phone, while Participant 1642 had recordings for only 16 activities. Those 12 participants were removed from the dataset, leaving 39 participants for the analysis presented herein.

The width of the window was chosen to be 10 seconds, since a human can carry out the activities present in the dataset multiple times in that time period. The data were sampled at a rate of 20 Hz; therefore, the width of the window was 200 samples. Since there were 12 features in this dataset, the dimension of the window was 12×200. A 75% overlap was chosen, which indicates that the window moved 50 time steps each time it slid. After the sliding window, the standardization method was applied to all of the participants' features separately.

2) Model Structuring: The three ML models (CNN, Transformer, and Trans-CNN) discussed in Section III-B were examined with respect to their use as personalized models for HAR. For the three models, the hyperparameters, parameters that are selected before the model is trained, were tuned using grid search with 5-fold cross-validation. Since the dataset is balanced, accuracy was chosen as the performance metric to select the hyperparameters.

Table I shows the hyperparameters that were tuned for each of the three models, together with the hyperparameter values that were considered in the grid search. The hyperparameter values that were selected with the grid search are shown in the 'Selected' column. For the kernel sizes, for example [3,3,3], the first number represents the kernel size for the first convolutional layer, the second number is for the second layer and the last number is for the third layer.

3) Model Training: The Generalization Phase involved training with data from all 38 participants (removing the 39th, the target participant) with the three models: CNN, transformer, and Trans-CNN. For training the CNN model, 150 epochs were used, as that was sufficient for the algorithm to converge. For both the transformer and Trans-CNN model, 100 epochs were sufficient. During the Personalization Phase, the algorithms converged after 100 epochs for all three types of models. $D2$ ($\frac{2}{3}$ of the target data) was used after each phase to evaluate each model with different data: all movement data, only upper body, and only lower body. For all participants, both phases were conducted with each of the three models.

The algorithms were implemented in Python with the Keras and TensorFlow deep learning libraries. The experiments were performed on a computer with Windows 10 OS, Intel(R) Core(TM) i9 CPU, 32 GB RAM, and an NVIDIA GeForce RTX 2070 graphics card.

V. Results

The results are presented in three parts considering all movements, upper body movements, and lower body movements. For each part, comparison between generic and personalized models is conducted first. Next, the three generic models, CNN, transformer, and Trans-CNN are compared, and finally, the three personalized models are compared.

TABLE I
HYPERPARAMETER TUNING FOR CNN, TRANSFORMER, AND TRANS-CNN MODELS.

	Hyperparameters	Considered	Selected
CNN	Dropout Rate	0.2, 0.25, 0.3	0.25
	Filter Sizes	32, 64, 128	64
	Kernel Sizes	[3, 3, 3], [5, 5, 5], [11, 11, 11], [3, 5, 11]	[3, 5, 11]
	Optimizer	Adam, SGD	Adam
Transformer	Dropout Rate	0.2, 0.25, 0.3	0.25
	Number of Heads	1, 2, 4, 8	4
	Head Size	16, 32, 64	32
	Number of Neurons	512, 1024, 2048	1024
Trans-CNN	Dropout Rate	0.2, 0.25, 0.3	0.2
	Filter Sizes	32, 64, 128	128
	Kernel Sizes	[3,3,3], [5,5,5], [11,11,11], [3,5,11]	[3, 5, 11]
	Optimizer	Adam, SGD	Adam
	Number of Neurons	512, 1024, 2048	2048

Note: For the kernel sizes, for example [3,3,3], the first number represents the kernel size for the first convolutional layer, the second number is for the second layer and the last number is for the third layer.

TABLE II
ALL MOVEMENTS: AVERAGE ACCURACY, PRECISION, RECALL, AND F1 SCORE, EXPRESSED AS PERCENTAGES WITH CORRESPONDING STANDARD DEVIATIONS, ACROSS ALL PARTICIPANTS FOR EACH OF THE THREE MODELS.

	CNN	Transformer	Trans-CNN
Accuracy			
Generic Model	41.2±8.9%	49.2±15.1%	69.6±15.1%
Personalized Model	94.1±4.7%	92.1±5.4%	92.4±4.8%
Precision			
Generic Model	46.3±10.3%	48.4±16.4%	70.5±15.1%
Personalized Model	94.8±4.2%	93.3±4.6%	93.1±4.6%
Recall			
Generic Model	41.2±8.9%	49.2±15.1%	69.6±15.1%
Personalized Model	94.1±4.7%	92.1±5.4%	92.4±4.8%
F1-Score			
Generic Model	37.6±8.9%	45.5±15.7%	67.3±15.7%
Personalized Model	93.8±4.9%	91.6±5.7%	92.0±5.1%

A. All Movements

In the evaluation of all movements, all of the activities were included, regardless of whether the activities were related to upper or lower body movements.

1) Comparison of Generic and Personalized Models for All Movements: In order to compare the results of generic and personalized models, the analysis focuses on accuracy as the dataset is balanced; nevertheless, precision, recall, and F1 scores are also reported. To determine if there was a statistically significant difference between the models, statistical tests were conducted.

Table II shows the average accuracy, precision, recall, and F1 scores, with corresponding standard deviations calculated across all participants for generic and personalized models, for each of the three ML models. The generic Trans-CNN model achieved an accuracy of 69.6%, which is the highest average value among generic models; the transformer and CNN achieved accuracies of 49.2% and 41.2%, respectively. Similarly, in terms of precision, recall, and F1 score, Trans-CNN also achieved better results than the remaining models. Note that recall is equivalent to accuracy because this is a balanced dataset. These results suggest that Trans-CNN is able to take advantage of CNN and transformer models to capture generic patterns better.

On the other hand, all personalized models performed much better, with the accuracy in the low 90%. While all three models greatly benefited from the personalization, the CNN model benefited the most, achieving the highest accuracy of 94.1%. The remaining metrics – precision, recall, and F1 score – exhibit the same pattern, with CNN achieving the best values.

To examine if the performance difference between the models is statistically significant, a Shapiro–Wilk test was conducted first to determine whether the data followed a normal distribution. Since personalized models were not normally distributed, the Mann-Whitney (MW) test was used to compare the generic and personalized models. Figures 6, 7, and 8 show the box plots of the generic and personalized CNN, Transformer, and Trans-CNN. The MW test p values for all CNN ($3.075e-14$), transformer ($4.189e-14$), and Trans-CNN ($7.206e-13$) models were less than 0.05, meaning that the difference between generic and personalized models is statistically significant for each of the three models. Consequently, we can conclude that the personalization improves the generic model irrelevant of the ML technique. Note that '$*$' in the figures indicates that the difference between the methods is statistically significant.

2) Comparison of Generic Models for All Movements: Next, the generic models were compared, and statistical tests were employed to find whether the difference among generic models is significant. The Kruskal-Wallis test was chosen because the comparison is done among three models and the data from the generic Trans-CNN model are nonparametric. Figure 9 shows the performance metric for all three generic models. Since the resulting p value ($3.74e-11$) is under 0.05, the differences between the groups are statistically significant. To determine where the difference was between the models, a Dunn test with Bonferroni adjusted p value was performed. The p value for (Trans-CNN, CNN) pair was $4.24e-11$ and for (Trans-CNN, transformer) pair was $7.42e-06$. Since both values were less than the significant level 0.05, it can be concluded that the Trans-CNN performs better than the other two models. The CNN model and the transformer model were not statistically different (p value of 0.122).

3) Comparison of Personalized Models for All Movements: While the previous subsection compared the generic model, this section does the same for the personalized models. Figure 10 shows a comparison of the three personalized models. Based on the Kruskal-Wallis test, there was no significant difference between the three personalized models ($p = 0.109$).

B. Upper Body Movements

The evaluation of upper body movements examined the performance of the three models on movements involving the upper body: typing, brushing teeth, eating soup, eating chips, eating pasta, drinking from a cup, eating a sandwich, playing catch w/tennis ball, dribbling (basketball), writing, clapping, and folding clothes. The evaluation process is the same as for the combined upper and lower body study (Subsection V-A) and the models are the same, but, here, the analysis is carried out on test set $D2$ with removed lower body movements.

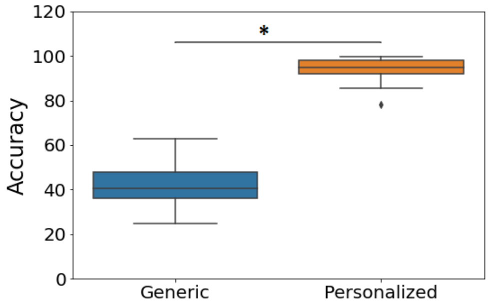


Fig. 6. All movements: comparison of generic and personalized CNN. Here, ∗ indicates significance at the 5% level.

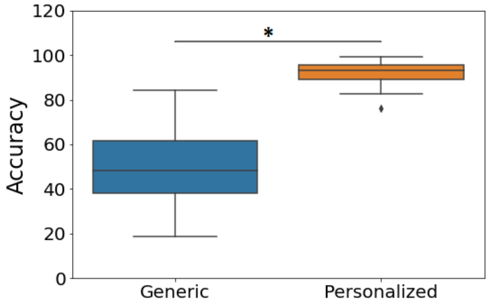


Fig. 7. All movements: comparison of generic and personalized transformer. Here, ∗ indicates significance at the 5% level.

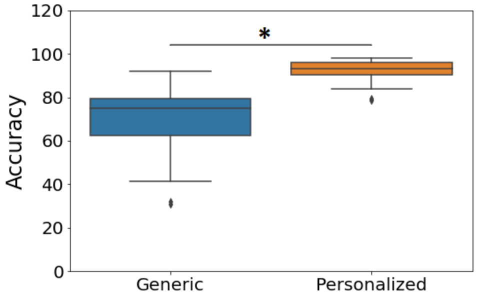


Fig. 8. All movements: comparison of generic and personalized Trans-CNN. Here, ∗ indicates significance at the 5% level.

1) Comparison of Generic and Personalized Models for Upper Body Movements: Considering only upper body movements, Table III shows the average performance metrics and standard deviations across all participants for the generic and the personalized models. Among the three models, the Trans-CNN model as a generic model achieved the highest average value in terms of all four metrics. As with All Movements, all personalized models performed much better, with CNN benefiting the most from personalization.

Figure 11 shows the boxplot of the generic and personalized Trans-CNN models. Boxplots for CNN and transformer are omitted, as they follow the same pattern. Since the data for both personalized models were nonparametric, the MW test was chosen. As the p value $(1.985e-12)$ for this test is below 0.05, there is a statistically significant difference between the generic and personalized Trans-CNN models.

Comparing to when all movements were evaluated, there is a drop in accuracy when considering upper body movements

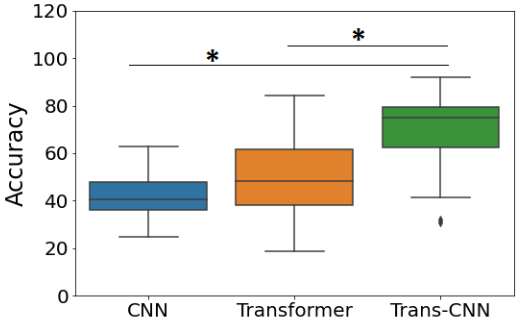


Fig. 9. All movements: comparison of generic models. Here, ∗ indicates significance at the 5% level.

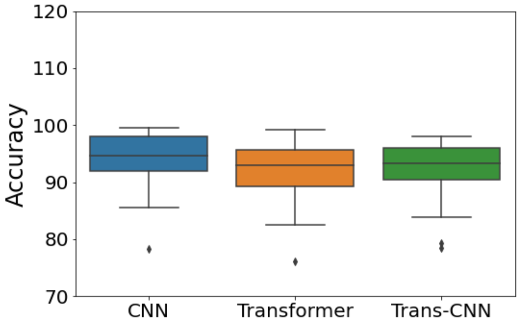


Fig. 10. All movements: comparison of personalized models.

TABLE III
UPPER BODY MOVEMENTS: AVERAGE ACCURACY, PRECISION, RECALL, AND F1 SCORE, EXPRESSED AS PERCENTAGES WITH CORRESPONDING STANDARD DEVIATIONS, ACROSS ALL PARTICIPANTS FOR EACH OF THE THREE MODELS.

	CNN	Transformer	Trans-CNN
Accuracy			
Generic Model	36.6±11.7%	42.5±17.9%	65.9±18.4%
Personalized Model	93.7±5.5%	92.0±6.6%	91.3±6.0%
Precision			
Generic Model	47.5±13.2%	48.7±18.1%	72.4±15.1%
Personalized Model	95.5±3.8%	94.5±4.6%	93.5±4.6%
Recall			
Generic Model	36.6±11.7%	42.5±17.9%	65.9±18.4%
Personalized Model	93.7±5.5%	92.0±6.6%	91.3±6.0%
F1-Score			
Generic Model	35.6±11.5%	41.7±17.3%	65.9±17.7%
Personalized Model	93.7±5.6%	92.2±6.6%	91.4±6.0%

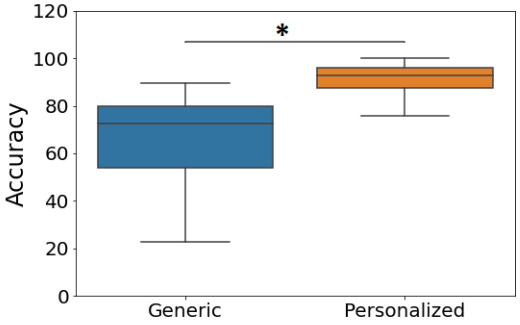
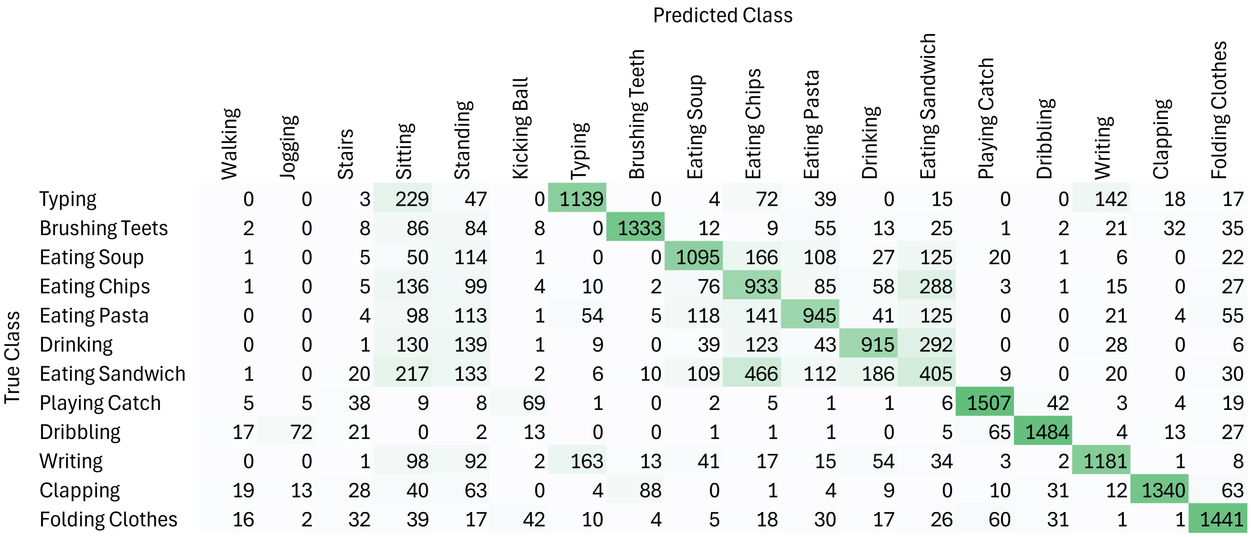
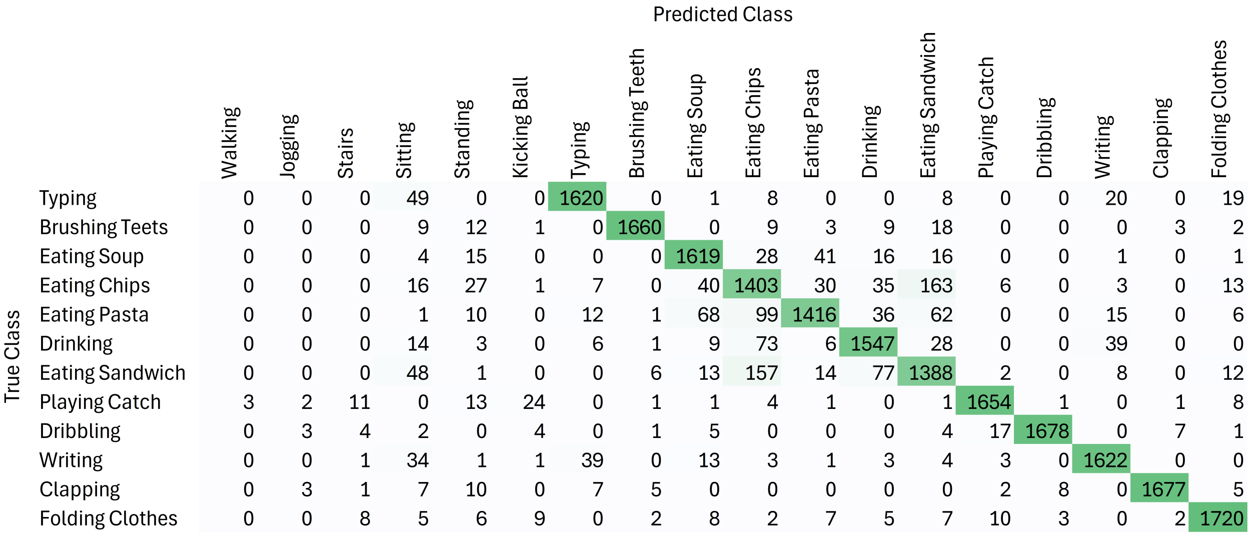


Fig. 11. Upper body movements: comparison of generic and personalized Trans-CNN. Here, ∗ indicates significance at the 5% level.


Predicted Class
True Class

	Walking	Jogging	Stairs	Sitting	Standing	Kicking Ball	Typing	Brushing Teeth	Eating Soup	Eating Chips	Eating Pasta	Drinking	Eating Sandwich	Playing Catch	Dribbling	Writing	Clapping	Folding Clothes
Typing	0	0	3	229	47	0	1139	0	4	72	39	0	15	0	0	142	18	17
Brushing Teets	2	0	8	86	84	8	0	1333	12	9	55	13	25	1	2	21	32	35
Eating Soup	1	0	5	50	114	1	0	0	1095	166	108	27	125	20	1	6	0	22
Eating Chips	1	0	5	136	99	4	10	2	76	933	85	58	288	3	1	15	0	27
Eating Pasta	0	0	4	98	113	1	54	5	118	141	945	41	125	0	0	21	4	55
Drinking	0	0	1	130	139	1	9	0	39	123	43	915	292	0	0	28	0	6
Eating Sandwich	1	0	20	217	133	2	6	10	109	466	112	186	405	9	0	20	0	30
Playing Catch	5	5	38	9	8	69	1	0	2	5	1	1	6	1507	42	3	4	19
Dribbling	17	72	21	0	2	13	0	0	1	1	1	0	5	65	1484	4	13	27
Writing	0	0	1	98	92	2	163	13	41	17	15	54	34	3	2	1181	1	8
Clapping	19	13	28	40	63	0	4	88	0	1	4	9	0	10	31	12	1340	63
Folding Clothes	16	2	32	39	17	42	10	4	5	18	30	17	26	60	31	1	1	1441

Fig. 12. Upper body movements: the confusion matrix displays the results from the generic Trans-CNN model.


Predicted Class
True Class

	Walking	Jogging	Stairs	Sitting	Standing	Kicking Ball	Typing	Brushing Teeth	Eating Soup	Eating Chips	Eating Pasta	Drinking	Eating Sandwich	Playing Catch	Dribbling	Writing	Clapping	Folding Clothes
Typing	0	0	0	49	0	0	1620	0	1	8	0	0	8	0	0	20	0	19
Brushing Teets	0	0	0	9	12	1	0	1660	0	9	3	9	18	0	0	0	3	2
Eating Soup	0	0	0	4	15	0	0	0	1619	28	41	16	16	0	0	1	0	1
Eating Chips	0	0	0	16	27	1	7	0	40	1403	30	35	163	6	0	3	0	13
Eating Pasta	0	0	0	1	10	0	12	1	68	99	1416	36	62	0	0	15	0	6
Drinking	0	0	0	14	3	0	6	1	9	73	6	1547	28	0	0	39	0	0
Eating Sandwich	0	0	0	48	1	0	0	6	13	157	14	77	1388	2	0	8	0	12
Playing Catch	3	2	11	0	13	24	0	1	1	4	1	0	1	1654	1	0	1	8
Dribbling	0	3	4	2	0	4	0	1	5	0	0	0	4	17	1678	0	7	1
Writing	0	0	1	34	1	1	39	0	13	3	1	3	4	3	0	1622	0	0
Clapping	0	3	1	7	10	0	7	5	0	0	0	0	0	2	8	0	1677	5
Folding Clothes	0	0	8	5	6	9	0	2	8	2	7	5	7	10	3	0	2	1720

Fig. 13. Upper body movements: the confusion matrix displays the results from the personalized Trans-CNN model.

only. To identify which activities were confused with other activities, Figures 12 and 13 show confusion matrices for the generic and personalized Trans-CNN, respectively. It can be observed that the generic model, Figure 12, often confuses many upper body movements with other lower body movements as well as with upper body movements. For example, many upper body movements are confused with Eating Sandwich or Sitting.

In contrast, the personalized Trans-CNN, Figure 13, performs much better and does not confuse upper body movements as often as the generic Trans-CNN model. However, the personalized Trans-CNN still has difficulties distinguishing between different eating activities.

2) Comparison of Generic Models for Upper Body Movements: Figure 14 compares the three generic models for upper body movements. A Kruskal-Wallis test p value of $3.35e-9$ indicates that there is a significant difference between the three generic models. Next, a Dunn test with Bonferroni adjusted p value was performed to find where the difference was. The Trans-CNN model compared to CNN and transformer achieved p values of $1.18e-08$ and $5.99e-6$, respectively. Therefore, the Trans-CNN is significantly better than the remaining two models, as indicated with * in Figure 14. Results also showed that there is no significant difference between the generic CNN and the transformer (p = 0.771).

3) Comparison of Personalized Models for Upper Body Movements: While Figure 14 compared generic models for upper body movements, Figure 15 compares personalized models. For these three models, the p value for the Kruskal-Wallis test was 0.771, meaning that there is no statistically significant difference among them.

C. Lower Body Movements

The evaluation of lower body movements included six activities that involve only lower body movements: walking, jogging, stairs, sitting, standing, and kicking (soccer ball). The process is the same as in the upper body movements evaluation but including only the lower body movements in $D2$.

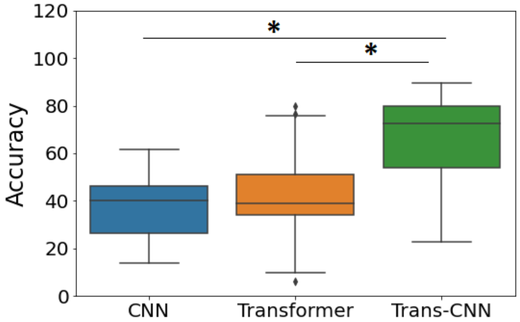


Fig. 14. Upper body movements: comparison of the three generic models. Here, $*$ indicates significance at the 5% level.

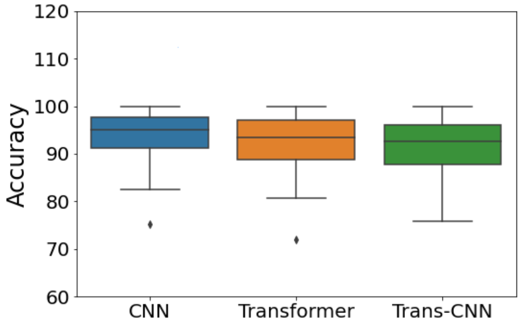


Fig. 15. Upper body movements: comparison of the three personalized models.

1) Comparison of Generic and Personalized Models for Lower Body Movements: Table IV shows the average performance metrics and standard deviations across all participants for the generic and personalized models, considering only lower body movements.

Again, generic Trans-CNN with an accuracy of 76.9% was better than the remaining generic models, while all three personalized models achieved an accuracy of over 90%. In terms of precision, recall, and F1 score, generic Trans-CNN was also better than the remaining generic models. Examining the normality of the data using the Shapiro-Wilk test, showed that the generic models follow the normal distribution while the other models do not. Therefore, the MW test was used to compare between generic and personalized models for each of the three algorithms. The test results confirmed that the difference between generic and personalized models are significant: comparison of the generic and personalized CNN models had a p value of $3.02e-14$, for generic and personalized transformer the p value was $3.587e-12$, and, finally, the p value was $2.06e-8$ for Trans-CNN.

Figure 16 shows the boxplot for the generic and personalized Trans-CNN models, while the remaining two are omitted as they follow the same pattern. The accuracy of the generic Trans-CNN model was 76.9%, while the personalized Trans-CNN model achieved around 92.9%, and the p value $(2.061e-08)$ for the MW test confirmed that the difference is significant.

2) Comparison of Generic Models for Lower Body Movements: Figure 17 compares the three generic models for lower body movements. A Kruskal-Wallis test with a p value of $8e-11$, showed that there was a significant difference within the group. Again, as when evaluating all movements and lower body movements, a Dunn test with a Bonferroni adjusted p value confirmed that the Trans-CNN model outperformed the CNN ($p = 2.76e-11$) and transformer ($p = 1.58e-03$) models. With lower body movements, there was also a significant difference between the CNN and the transformer models ($p = 0.00241$) with the transformer model achieving better accuracy than CNN.

TABLE IV

LOWER BODY MOVEMENTS: AVERAGE ACCURACY, PRECISION, RECALL, AND F1 SCORE, EXPRESSED AS PERCENTAGES WITH CORRESPONDING STANDARD DEVIATIONS, ACROSS ALL PARTICIPANTS FOR EACH OF THE THREE MODELS.

	CNN	Transformer	Trans-CNN
Accuracy			
Generic Model	50.4±10.9%	62.8±17.6%	76.9±13.0%
Personalized Model	93.5±7.6%	92.7±7.3%	92.9±8.0%
Precision			
Generic Model	62.9±12.4%	73.3±17.5%	84.7±11.7%
Personalized Model	97.2±4.4%	97.0±4.0%	97.1±3.9%
Recall			
Generic Model	50.4±10.9%	62.8.2±17.6%	76.9±13.0%
Personalized Model	93.5±7.6%	92.7±7.3%	92.9±8.0%
F1-Score			
Generic Model	49.4±11.0%	63.5±18.1%	77.8±13.0%
Personalized Model	94.3±7.0%	93.9±6.2%	94.1±6.8%

3) Comparison of Personalized Models for Lower Body Movements: Finally, Figure 18 compares the personalized models for lower body movements. Based on the Kruskal-Wallis test, there was no significant difference between the three models since the p value of 0.743 was over 0.05. Thus, all three models perform similarly on lower body movements.

VI. DISCUSSION

In this study, combined data (both upper and lower movements) were used to train the model, since the activity labels in a real-world application are unknown beforehand. The same models were evaluated with all movements, upper body movements, and lower body movements. When evaluating all movements, Table II demonstrated the value of personalization, since the performance metric increased for all three models: for example, from the 40s to the 90s for the CNN model. The same trend can be seen for the upper (Table III) and lower (Table IV) body movements. The accuracy increased for all three models when personalization was applied.

For the generic models, the Trans-CNN model outperformed the other two models in each of the three analyses, as seen in Figures 9, 17, and 14. Statistical tests confirmed that the results from the generic Trans-CNN are significantly better than the remaining two generic models. These results show that the Trans-CNN is capable of providing better generic models than the CNN model, which has been the dominant HAR model in recent years.

Figures 10, 15, and 18 show that after personalization the three models achieved an accuracy in the 90s for all three analyses. Moreover, there was no statistical difference between the three models. The CNN and transformer models benefited more from the personalization than Trans-CNN. Based on this finding, any of the three models can be used for personalizing the models.

All generic models had a harder time distinguishing upper body movement, as observed from the drop in accuracy in

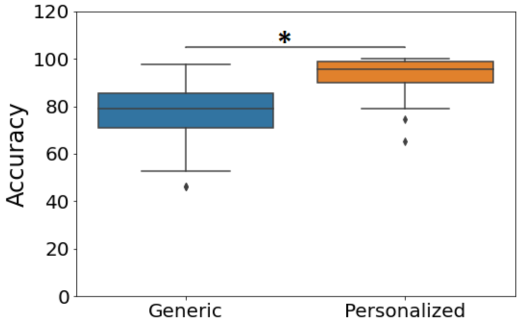


Fig. 16. Lower body movements: comparison of generic and personalized Trans-CNN. Here, ∗ indicates significance at the 5% level.

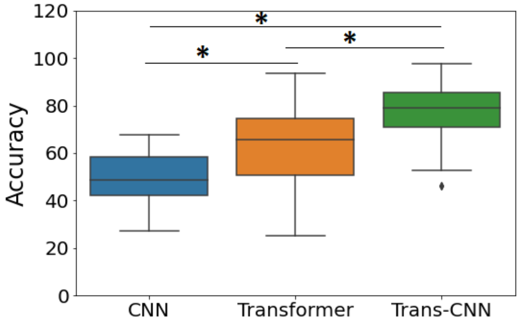


Fig. 17. Lower body movements: comparison of the three generic models. Here, ∗ indicates significance at the 5% level.

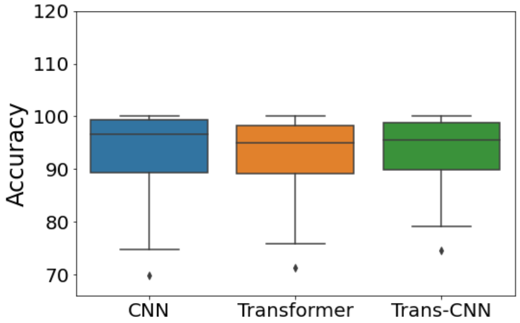


Fig. 18. Lower body movements: comparison of the three personalized models.

Table Tables III compared to II and IV. As seen from the same tables, when models are personalized, the accuracy of detecting lower body movements comes closer to the accuracy of all or upper body movements. As long as the model was personalized, the model performed similarly well with any type of data, irrelevant of the base model, CNN, transformer, or Trans-CNN.

VII. Conclusion

This paper proposed a hybrid ML method, Trans-CNN, combining a transformer and CNN, for recognition of human activities, including both lower and upper body movements. CNN brings the strengths of spatial–temporal modelling, while the transformer enhances temporal-dependency modelling through self-attention. The personalization was added to customize the generic model to the target person. In contrast to other studies, the behaviour of the models on upper and lower body movements was examined. Experiments using the WISDM data show that the proposed Trans-CNN achieves much better generic model accuracy results than the transformer or CNN alone, demonstrating the benefits of merging the two techniques. Personalization results in much better models for all three, CNN, transformer, and Trans-CNN, and after the personalization, all three models achieve similar accuracy. Moreover, after personalization, the accuracy of detecting upper body movements becomes closer to the accuracy of detecting all movements. Examining upper body movements showed that occasionally those movements are confused with lower body movements, but mostly, various eating and drinking activities are confused.

While this study presents steps toward recognition of upper and lower body movements, it is important to recognize the challenges. The scarcity of datasets that include both upper and lower body movements, together with the limited diversity of upper body movements in existing datasets, limit research progress in this domain. The practical use of personalized models requires obtaining some data from the target person, which necessitates the implementation of data acquisition and labeling on the wearable device. Moreover, continuous data acquisition and monitoring will be needed to adapt the model to the changes in the participants' movement over time.

Consequently, as new datasets become available, future work will evaluate the presented approaches with different data sets and with data from different wearable sensors instead of smartphones and watches from the WISDM dataset. Moreover, the minimal amount of data needed from the target participant for personalization will be explored, the scalability of the training for a large number of participants will be investigated, and the continuous adaptation of the model will be examined.

References

[1] B. A. O'Neil, M. E. Forsythe, and W. D. Stanish, "Chronic occupational repetitive strain injury." *Canadian Family Physician*, vol. 47, no. 2, pp. 311–316, 2001.

[2] P. Bierma, "Repetitive stress injury," Aug 2022. [Online]. Available: https://consumer.healthday.com/encyclopedia/pain-management-30/pain-health-news-520/repetitive-stress-injury-rsi-646236.

[3] Microsoft, "Reducing the incidence and cost of work-related musculoskeletal." [Online]. Available: https://webobjects.cdw.com/webobjects/media/pdf/CDWCA/Ergo_Whitepaper_June-2017.pdf

[4] M. M. Hassan, M. Z. Uddin, A. Mohamed, and A. Almogren, "A robust human activity recognition system using smartphone sensors and deep learning," *Future Generation Computer Systems*, vol. 81, pp. 307–313, 2018.

[5] E. Ramanujam, T. Perumal, and S. Padmavathi, "Human activity recognition with smartphone and wearable sensors using deep learning techniques: A review," *IEEE Sensors Journal*, vol. 21, no. 12, pp. 13 029–13 040, 2021.

[6] L. M. Dang, K. Min, H. Wang, M. J. Piran, C. H. Lee, and H. Moon, "Sensor-based and vision-based human activity recognition: A comprehensive survey," *Pattern Recognition*, vol. 108, p. 107561, 2020.

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

[8] X. Cheng and B. Huang, "CSI-Based human continuous activity recognition using GMM–HMM," *IEEE Sensors Journal*, vol. 22, no. 19, pp. 18 709–18 717, 2022.

[9] Z. Ahmad and N. Khan, "Human action recognition using deep multilevel multimodal (M^2) fusion of depth and inertial sensors," *IEEE Sensors Journal*, vol. 20, no. 3, pp. 1445–1455, 2019.

[10] H. Li, A. Shrestha, H. Heidari, J. Le Kernec, and F. Fioranelli, "Bi-LSTM network for multimodal continuous human activity recognition and fall detection," *IEEE Sensors Journal*, vol. 20, no. 3, pp. 1191–1201, 2019.

This article has been accepted for publication in IEEE Sensors Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2024.3451291

[11] S. A. Rokni, M. Nourollahi, and H. Ghasemzadeh, "Personalized human activity recognition using convolutional neural networks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, New Orleans, USA, February 2–7, 2018.

[12] D. Gholamiangonabadi and K. Grolinger, "Personalized models for human activity recognition with wearable sensors: deep neural networks and signal processing," *Applied Intelligence*, vol. 53, no. 5, pp. 6041–6061, 2023.

[13] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *31st Conference on Neural Information Processing Systems*, Long Beach, USA, December 4-9, 2017.

[14] K. Rai, F. Hojatpanah, F. B. Ajaei, J. M. Guerrero, and K. Grolinger, "Deep learning for high-impedance fault detection and classification: transformer-CNN," *Neural Computing and Applications*, vol. 34, no. 16, pp. 14 067–14 084, 2022.

[15] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, "A survey of the recent architectures of deep convolutional neural networks," *Artificial Intelligence Review*, vol. 53, no. 8, pp. 5455–5516, 2020.

[16] Y. Xu and T. T. Qiu, "Human activity recognition and embedded application based on convolutional neural network," *Journal of Artificial Intelligence and Technology*, vol. 1, no. 1, pp. 51–60, 2021.

[17] A. Gillioz, J. Casas, E. Mugellini, and O. Abou Khaled, "Overview of the transformer-based models for NLP tasks," in *15th Conference on Computer Science and Information Systems*. Sofia, Bulgaria: IEEE, September 6-9, 2020.

[18] W. Sousa Lima, E. Souto, K. El-Khatib, R. Jalali, and J. Gama, "Human activity recognition using inertial sensors in a smartphone: An overview," *Sensors*, vol. 19, no. 14, p. 3213, 2019.

[19] Z.-Y. He and L.-W. Jin, "Activity recognition from acceleration data using AR model representation and SVM," in *International Conference on Machine Learning and Cybernetics*, vol. 4, Kunming, China, July 12-15, 2008.

[20] M. A. Alsheikh, A. Selim, D. Niyato, L. Doyle, S. Lin, and H.-P. Tan, "Deep activity recognition models with triaxial accelerometers," in *Workshops at the Thirtieth AAAI Conference on Artificial Intelligence*, Phoenix, USA, February 12-17, 2016.

[21] Y. Chen and Y. Xue, "A deep learning approach to human activity recognition based on single accelerometer," in *IEEE International Conference on Systems, Man, and Cybernetics*, Hong Kong, October 9-12, 2015.

[22] S. W. Pienaar and R. Malekian, "Human activity recognition using LSTM-RNN deep neural network architecture," in *IEEE 2nd Wireless Africa Conference*, Pretoria, South Africa, August 18-20, 2019.

[23] D. Hendry, K. Chai, A. Campbell, L. Hopper, P. O'Sullivan, and L. Straker, "Development of a human activity recognition system for ballet tasks," *Sports Medicine-Open*, vol. 6, no. 1, pp. 1–10, 2020.

[24] M. Straczkiewicz, P. James, and J.-P. Onnela, "A systematic review of smartphone-based human activity recognition methods for health research," *NPJ Digital Medicine*, vol. 4, no. 1, pp. 1–15, 2021.

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

[26] S. Mekruksavanich and A. Jitpattanakul, "Smartwatch-based human activity recognition using hybrid LSTM network," in *IEEE Sensors Conference*, Rotterdam, Netherlands, October 25-28, 2020.

[27] I. Dirgová Luptáková, M. Kubovčík, and J. Pospíchal, "Wearable sensor-based human activity recognition with transformer model," *Sensors*, vol. 22, no. 5, p. 1911, 2022.

[28] Y. Wang, H. Xu, Y. Liu, M. Wang, Y. Wang, Y. Yang, S. Zhou, J. Zeng, J. Xu, S. Li *et al.*, "A novel deep multifeature extraction framework based on attention mechanism using wearable sensor data for human activity recognition," *IEEE Sensors Journal*, vol. 23, no. 7, pp. 7188–7198, 2023.

[29] N. Zhang, Y. Song, D. Fang, Z. Gao, and Y. Yan, "An improved deep convolutional lstm for human activity recognition using wearable sensors," *IEEE Sensors Journal*, 2023.

[30] H. A. Imran, Q. Riaz, M. Hussain, H. Tahir, and R. Arshad, "Smart-wearable sensors and cnn-bigru model: A powerful combination for human activity recognition," *IEEE Sensors Journal*, 2023.

[31] H. Amrani, D. Micucci, and P. Napoletano, "Personalized models in human activity recognition using deep learning," in *25th International Conference on Pattern Recognition*, Milano, Italy, January 10-15, 2021.

[32] D. Gholamiangonabadi, N. Kiselov, and K. Grolinger, "Deep neural networks for human activity recognition with wearable sensors: Leave-one-subject-out cross-validation for model selection," *IEEE Access*, vol. 8, pp. 133 982–133 994, 2020.

[33] A. Dehghani, O. Sarbishei, T. Glatard, and E. Shihab, "A quantitative comparison of overlapping and non-overlapping sliding windows for human activity recognition using inertial sensors," *Sensors*, vol. 19, no. 22, p. 5026, 2019.

[34] A. Ignatov, "Real-time human activity recognition from accelerometer data using convolutional neural networks," *Applied Soft Computing*, vol. 62, pp. 915–922, 2018.

[35] M. C. Leong, H. Zhang, H. L. Tan, L. Li, and J. H. Lim, "Combined CNN transformer encoder for enhanced fine-grained human action recognition," *arXiv preprint arXiv:2208.01897*, 2022.

[36] D. Wu, X. Han, Z. Yang, and R. Wang, "Exploiting transfer learning for emotion recognition under cloud-edge-client collaborations," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 2, pp. 479–490, 2020.

[37] A. D. G. A. O. L. Reyes-Ortiz, Jorge and X. Parra, "Human Activity Recognition Using Smartphones," UCI Machine Learning Repository, 2012, DOI: https://doi.org/10.24432/C54S4K.

[38] D. Anguita, A. Ghio, L. Oneto, X. Parra, J. L. Reyes-Ortiz *et al.*, "A public domain dataset for human activity recognition using smartphones." in *Esann*, vol. 3, 2013, p. 3.

[39] WISDM-2019, https://archive.ics.uci.edu/ml/datasets/WISDM+Smartphone+and+Smartwatch+Activity+and+Biometrics+Dataset+.

Kyle Lacroix received the BESc degree in mechatronic systems engineering and the MESc degree in software engineering degrees from Western University. He is the co-founder and CTO at Blue Guardian, a mental health early warning system that uses emotional analysis of social media usage to detect mental health problems. His research interests include machine learning for sensor data analysis and natural language processing.

Davoud Gholamiangonabai received the B.Sc. degree in applied math from Ferdowsi University, and the M.Sc. degree in Industrial Engineering-System and Productivity Management from Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran. After getting his second master in software engineering at Western University, he is currently pursuing his Ph.D. in software engineering. His current research interests include machine learning, deep learning, federated learning, and natural language processing.

Ana Luisa Trejos (S'08–M'12–SM'16) is a Full Professor with the Department of Electrical and Computer Engineering and the School of Biomedical Engineering at Western University. She has expertise in the design, development and testing of medical mechatronic systems. This experience led her to establish the Wearable Biomechatronics Laboratory in 2013, dedicated to the design of wearable mechatronic devices for upper body rehabilitation and motion assistance, including wearable devices for tremor suppression and smart orthotic devices. Her research focuses on designing novel sensing/actuation components, creating models based on sensed biosignals, and developing intelligent control systems.

Katarina Grolinger (M'11-SM'24) is an Associate Professor of software engineering in the Department of Electrical and Computer Engineering at Western University, Canada, Canada Research Chair in Engineering Applications of Machine Learning, and a faculty affiliate at Vector institute for AI. Dr. Grolinger received the BSc and MSc degrees in mechanical engineering from the University of Zagreb, Croatia, and the M.Eng. and PhD degrees in software engineering from Western University, London, Canada. She has been involved in the software engineering area in academia and industry, for over 20 years. Her research interests include machine learning, sensor data analytics, data management, and IoT.