

Article

BSTCA-HAR: Human Activity Recognition Model Based on Wearable Mobile Sensors

Yan Yuan, Lidong Huang *, Xuwen Tan, Fanchang Yang and Shiwei Yang

School of Mathematics and Computer Science, Yunnan Minzu University, Kunming 650031, China; yuanyan_2022@163.com (Y.Y.); tanxw0910@ymu.edu.cn (X.T.); yfc2936060726@gmail.com (F.Y.); yangshiwei05@163.com (S.Y.)

* Correspondence: hldwinter@163.com

Abstract: Sensor-based human activity recognition has been widely used in various fields; however, there are still challenges involving recognition of daily complex human activities using sensors. In order to solve the problem of timeliness and homogeneity of recognition functions in human activity recognition models, we propose a human activity recognition model called 'BSTCA-HAR' based on a long short-term memory (LSTM) network. The approach proposed in this paper combines an attention mechanism and a temporal convolutional network (TCN). The learning and prediction units in the model can efficiently learn important action data while capturing long time-dependent information as well as features at different time scales. Our series of experiments on three public datasets (WISDM, UCI-HAR, and ISLD) with different data features confirm the feasibility of the proposed method. This method excels in dynamically capturing action features while maintaining a low number of parameters and achieving a remarkable average accuracy of 93%, proving that the model has good recognition performance.

Keywords: humanactivity recognition; sensor data; LSTM; deep learning



Citation: Yuan, Y.; Huang, L.; Tan, X.; Yang, F.; Yang, S. BSTCA-HAR: Human Activity Recognition Model Based on Wearable Mobile Sensors. *Appl. Sci.* **2024**, *14*, 6981. <https://doi.org/10.3390/app14166981>

Academic Editor: Junseop Lee

Received: 8 July 2024

Revised: 4 August 2024

Accepted: 5 August 2024

Published: 9 August 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Human activity recognition (HAR) has numerous emerging uses that significantly enhance human wellbeing, such as health monitoring [1], daily activities [2], and telemedicine [3]. The two primary methods for tracking human activity are computer vision and wearable sensors. Computer vision-based methods typically use cameras to monitor and record activities [4], requiring multi-angle adjustments and deep learning techniques to process the captured images. This complexity makes decoding human activities challenging. Moreover, these methods often only analyze pixel information in the RGB streams, neglecting the detailed understanding of human motion and privacy issues, which limits their applicability in certain scenarios.

Comparatively, sensor data preserve privacy, as these data are not affected by light intensity and do not record facial features. As such, sensor-based HAR has attracted a great deal of interest [5]. Because it can automatically identify human behaviors from data acquired by various sensing devices and achieve adequate performance at a relatively cheap computing cost, it is more palatable to both consumers and researchers [6]. There are two types of HAR techniques: fixed sensor-based and mobile sensor-based [7]. Continuous monitoring of a given region is made possible by fixed sensor techniques, which also usually have cheap deployment and maintenance costs while guaranteeing steady and dependable data collecting. Unfortunately, their applicability is limited to a small number of indoor locations due to their susceptibility to ambient occlusion, backdrop, and light fluctuations [8]. On the other hand, micro-electronic or biochemical-based wearable mobile sensors are usually small, lightweight, and reasonably priced for individuals to wear. They can be positioned on various body areas or in the subject's pocket as needed. We can now monitor and comprehend human behavior and health like never before thanks to

the tremendous amount of data created by the recent rapid development of wearable sensors [9]. Wide-ranging real-world uses for wearable-sensor-based health analytics include surveillance systems, telemedicine, smart healthcare, assisted living facilities, and predictive health diagnostics [10]. In addition to increasing the precision and effectiveness of human activity detection, wearable sensor technology is portable, low-power, and effective. It also offers creative and useful solutions for safety monitoring and health management.

Despite these benefits, modeling and analysis are significantly hampered by the complexity and volume of data produced by wearable sensors. In order to successfully analyze and interpret the data, a variety of techniques must be applied, including deep learning and time series modeling. It is possible to identify human behavior more precisely by analyzing, modeling, comprehending, and producing data from wearable sensors. Consequently, the research presented in this paper explores activity recognition using human gesture recognition algorithms based on wearable mobile sensors. This research is of great significance and value in advancing smart health monitoring, smart wearable devices, human–computer interaction technologies, and other related fields.

This study introduces a human activity recognition (HAR) model that combines bidirectional long short-term memory (BiLSTM) with a temporal convolutional network (TCN) and attention mechanism. The purpose of this integration is to overcome existing challenges in HAR by increasing the model's ability to capture temporal patterns and pull out relevant features from the data, thereby enhancing both the accuracy and dependability of the system. The pivotal contributions of this endeavor can be summarized as follows:

- The BSTCA-HAR architecture is proposed for accepting raw unlabeled sensor data as input and then autonomously extracting features. The adaptability of the model is enhanced through crafting the LSTM and convolutional layers with minimal parameters, which enables the capture of long-term correlations within intricate time series data while mitigating the risk of model overfitting.
- We conducted detailed experimental tests on three publicly available mobile sensor datasets in order to demonstrate the effectiveness of the BSTCA-HAR network in human activity recognition research. Specifically, we utilized the WISDM, UCI-HAR, and ISLD datasets to experimentally validate the performance of our model.
- To demonstrate the superiority of the introduced architecture, we conducted detailed comparative evaluations, comparing BSTCA-HAR with various techniques described recently in the literature. These comparative experiments using the same datasets and evaluation metrics clearly show the advantages of the new architecture in terms of both performance and effectiveness.

2. Related Work

In recent years, HAR systems based on machine learning (ML) and deep learning (DL) have become increasingly common [11]. Traditional ML methods emphasize the feature engineering stage, in which key attributes are pinpointed and distilled to distinguish between diverse activity profiles and minimize reliance on manual preprocessing. Feature extraction from primitive data is crucial for the effective operation of HAR systems. These features are then used as inputs to a classifier for recognizing human behavior. Earlier studies used the discrete wavelet and Fourier transforms [12] to identify frequency-based characteristics, then ML methods such as decision trees, logistic regression, and random forest (RF) to classify the extracted features [13,14]. While these methods can obtain better results on the HAR problem, they rely heavily on manual feature extraction. This process is marked by its time-consuming and labor-intensive nature while also being crucially contingent upon the excellence of feature extraction, factors which can easily lead to unstable model performance.

DL techniques have been extensively employed within the realm of HAR. Their attractiveness mainly lies in their ability to automatically extract features without human intervention. Studies have shown that recurrent neural network (RNN)-based models can be successfully applied to the problem of human behavior and activity recognition [15],

effectively recognizing various human activities such as walking and running. However, the high complexity of RNN models makes it difficult to effectively capture long-term dependencies. In order to increase the convergence speed of the loss function, scholars have proposed many RNN improvement methods to optimize model performance [16].

Subsequently, convolutional neural networks (CNN) have excelled in feature extraction. Mohi ud Din Dar et al. [17] achieved remarkable success in medical image analysis using a CNN, particularly in the classification of Alzheimer's disease. Analogous to our approach in the HAR system, where a CNN is utilized for feature extraction and classification, converting raw sensor data into high-level abstract sequences addresses the issue of manual feature extraction in traditional human activity recognition [18]. Zheng et al. [19] converted trivial accelerometer data into an "image" format, with the convolutional layer extracting deep spatial features from action video frames; however, its ability to capture temporal information was limited, and it had difficulty dealing with complex temporal dependencies. In addition, some previous models have utilized transformer-like methods to handle irregularly sampled time series [20], as in Cho et al. [21], who utilized a self-attention mechanism in their model for improved human behavior recognition accuracy obtained better performance than traditional methods. However, these methods are deficient in capturing temporal information, and cannot effectively handle time series data. To enhance the model's ability to capture features at different time scales, researchers have proposed methods that incorporate multiple DL techniques. Pravanya et al. [22] demonstrated that an architecture combining CNN and LSTM with an attention mechanism excels in handling HAR tasks. Their model performed well in multi-scale feature extraction and key feature recognition. Thakur et al. [23] constructed an end-to-end model that integrates the CNN, autoencoder (AE), and LSTM architectures, enabling efficient HAR operations without the need for intricate data preprocessing. Furthermore, in [24] they proposed a method for hemiplegic gait prediction based on smartphone sensor data, combining both manually designed and automatically extracted features. However, when dealing with multi-class tasks and detecting intricately detailed movements, their system still faces challenges such as structural complexity and high computational costs.

To address the aforementioned limitations, we propose a novel human activity recognition model called BSTCA-HAR. This model deeply integrates the advantages of TCN, attention mechanisms, and BiLSTM to overcome the constraints of existing solutions in multiple dimensions. First, in terms of temporal information processing, BSTCA-HAR employs BiLSTM as one of its core components. Through its unique bidirectional structure, BiLSTM captures both forward and backward dependencies in sequential data, effectively enhancing the ability to model long-term temporal dependencies. This feature makes the model more accurate in parsing complex activity patterns, efficiently bolstering its capacity to capture prolonged temporal correlations. Second, to meet the need for automated and efficient feature extraction, BSTCA-HAR integrates a TCN module. TCN, with its unique causal convolution mechanism, can effectively expand the receptive field without increasing the model's complexity, thereby automatically extracting features at different temporal scales. Furthermore, the model incorporates an attention mechanism to further enhance its feature extraction capabilities. This enables the model to adaptively assess the significance of features across varying temporal intervals, increasing its sensitivity to key features and enhancing both feature extraction ability and overall model performance.

In summary, the BSTCA-HAR model constructs an efficient end-to-end HAR system that not only simplifies the data preprocessing process but also significantly improves the model's recognition performance and generalization ability in complex scenarios. It effectively addresses the shortcomings of existing solutions in feature extraction, temporal dependency modeling, and feature importance adjustment. Experimental results show that the BSTCA-HAR model performs excellently on multiple publicly recognized datasets with good generalization capability, verifying its effectiveness and advancement in addressing current HAR issues.

3. Methods

3.1. BiLSTM

LSTM is an RNN that solves the problem of the difficulty RNN models face in handling long-term dependent data [25]. RNN networks outperform CNNs in feature extraction from sequential data by using gated units to control memory forgetting and updating while selectively retaining long-term and short-term information. In our proposed model, we employ bidirectional LSTM for enhancing temporal pattern recognition from sequential data. Each layer comprises 64 memory units, with the input disseminated to distinct gates—the input gate, the forget gate, and the output gate—that orchestrate the functioning of each memory cell, governing the flow of information. By leveraging its ability to consider both past and future information simultaneously, bidirectional LSTM has better ability to understand the contextual information in sequential data. BiLSTM consists of two independent LSTM networks; one processes the sequence in chronological order, while the other processes it in reverse chronological order. Their outputs are then combined and passed to subsequent layers to capture bidirectional dependencies in the sequential data.

LSTM consists of a set of recurrently connected memory units, each of which contains one or more cell structures. Each cell structure retains or forgets data information through different gates. The activation formulas for each LSTM unit are described as follows:

1. Historical information deemed unnecessary or irrelevant is discarded by the forget gate, as shown in Equation (1).

$$f_t = \sigma(W_f^* h_{t-1} + W_f * x_t + b_f) \quad (1)$$

2. The information retained from the previous time step along with the current input information jointly serve as the updated state of the input gate, as shown in Equations (2) and (3).

$$i_t = \sigma(W_i^* h_{t-1} + W_i * x_t + b_i) \quad (2)$$

$$\tilde{C} = W_i^* h_{t-1} + W_c * x_t + b_c; C_t = f_t^* C_{t-1} + i_t^* \tilde{C} \quad (3)$$

3. The current state information is output by the output gate, as shown in Equations (4) and (5):

$$o_t = \sigma(W_o^* h_{t-1} + W_o * x_t + b_o) \quad (4)$$

$$h_t = o_t * \tanh(C_t) \quad (5)$$

where W_i , W_c , W_f , and W_o are the corresponding connection weights, while b_i , b_c , b_f , and b_o are the corresponding biases. The variable f represents the forget factor determined by the forget gate at time t , σ is the Sigmoid function, C_t corresponds to the updated value of the cellular state at the moment t , and h_t corresponds to the output value of the current neural unit.

3.2. Temporal Convolutional Network

TCN is a network structure proposed by Lea et al. [26] that is more suitable for solving time series data problems. It consists of stacked one-dimensional convolutional layers, each with a unique inflated causal convolutional structure; for the causal convolution, the result at any given time point solely depends on the inputs at that time point and those preceding it. For a one-dimensional input sequence, the convolution kernel can expand the sensory field using filter coefficients k and expansion coefficients d . The expansion convolution operation is shown as follows:

$$F(s) = \sum_{i=0}^{k-1} f(i)x_{s-di} \quad (6)$$

where x is the input, f is the filter function, d is the dilation factor, k is the convolution kernel size, and $s - di$ determines the convolution operation to be performed only on past input data.

In addition, the training stability of the TCN is crucial for overall model performance, which is achieved by incorporating residual connections in the TCN network. The proposed residual connection consists of two layers of causal convolution, ReLU nonlinear activation function layers, and batch normalization layers. To prevent overfitting, dropout layers are also included. To accommodate different input and output dimensions, a 1×1 convolution is used to ensure that the elements maintain the same tensor shape after computation. The batch normalization layer normalizes the feature vectors and parameters from the previous layer, accelerating neural network training and improving convergence speed and stability. The expression for the residual connection is

$$o = \text{Activation}(x + F(x)), \quad (7)$$

where x represents the input signal to the residual block, o denotes the output generated by the residual block, and *Activation* is the activation function.

3.3. Attention Mechanism

The attention mechanism strategically allocates varying degrees of importance (weights) to input information, enabling the model to prioritize focus on segments that exhibit similarity to the input elements, thereby efficiently distilling the salient features within the sequence.

The attention mechanism determines the hidden state y_i generated at each specific time point by taking a vector x_n as the weighted average of the state sequence y (extracted key features). In this paper, the output sequence of the BiLSTM is passed into the TCN. The TCN processes long sequence information through stacked one-dimensional convolutional layers and residual connections, extracting important features from the sequence. The hidden state sequence $H = [h_1, h_2, \dots, h_T]$ output from the TCN layer is then fed into the attention layer. The attention mechanism performs a weighted summation of these hidden states to highlight key features. The attention mechanism first calculates the weight α_{ni} for each time step, which represents the importance of each hidden state for the current task. The formula can be expressed as follows.

$$e_{ni} = \alpha(M_{n-1}, y_i)\alpha_{ni} = \frac{\exp(e_{ni})}{\sum_{i=1}^n \exp(e_{ni})} \quad (8)$$

The calculated weights α_{ni} are used to perform a weighted summation of the hidden states from the TCN layer, generating a weighted average x_n . The formula can be expressed as follows:

$$x_n = \sum_{i=1}^N \alpha_{ni} y_i \quad (9)$$

where N is the total number of time steps of the input sequence and α_{ni} is the weight calculated for each state y_i . After the vectors are computed, the vectors can be used to compute a new state sequence M .

The aggregated context vector x_n is passed to the output layer, where the final output $M = \tanh(W_x[x_i : y_i])$ is calculated using the tanh function to generate the final prediction results. In this way, the attention mechanism plays the role of filtering and enhancing key features, allowing the model to better capture important information in the action sequences and improving its recognition accuracy.

3.4. BSTCA-HAR Prediction Modeling

Based on the proposed improvement of the LSTM neural network, a BSTCA-HAR action recognition model combining attention mechanism and TCN parts is proposed. BiLSTM is used as a network layer for feature extraction and preprocessing of the input sequence data, which is a three-dimensional tensor, (i.e., Samples, Time_Steps, and Features), with one processing the sequence front-to-back and the other processing it back-to-front, leading to a better understanding of the context and coherence of the actions and capturing the bidirectional information in the action sequences during action recognition.

With the expansion of network depth and iterations, the weights change too quickly for a network degradation effect to appear. We introduce the TCN to deal with time series that become too long by using the reusable convolutional layer module and residual connection to solve the problem of training difficulties' however, it is easy for the model to ignore key time points or features. The key time points or features can be automatically identified incorporation of the attention layer, thereby enhancing the precision of the HAR framework. Figure 1 depicts the BSTCA-HAR model's network structure.

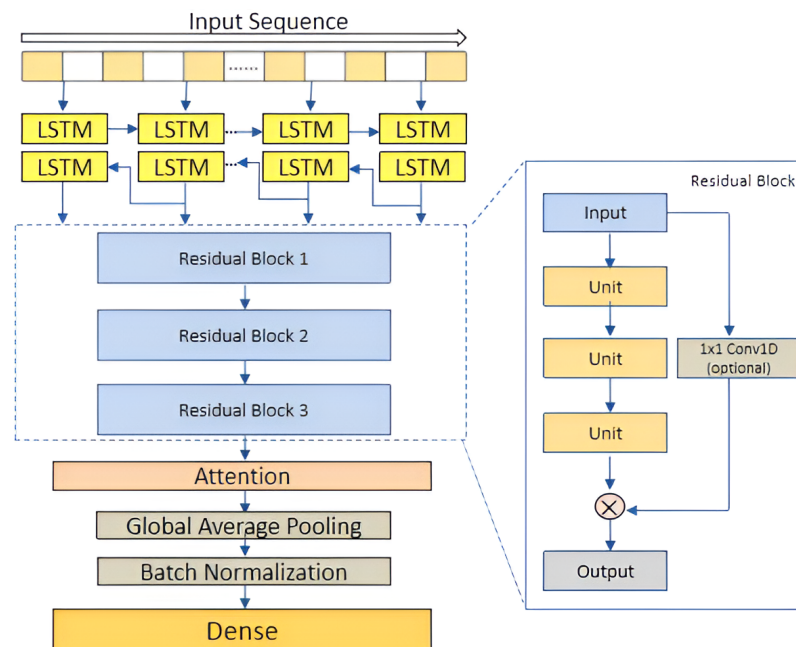


Figure 1. Framework diagram of the BSTCA-HAR model.

The bidirectional LSTM layer in the BSTCA-HAR model receives a three-dimensional tensor as its input. To align with the expected input dimensions of the TCN layer, the input samples introduced for the TCN layer are similarly structured as a three-dimensional tensor. As a result, there is no need to resize or scale the output of the BiLSTM layer to fit these dimensions, which streamlines the model's architecture. The causal convolution process within the TCN layer performs advanced feature refinement on the feature representation from the preceding layer, facilitating a deeper level of feature extraction.

It is worth noting that the architecture outlined in this study substitutes for the traditional fully connected layer with a global average pooling (GAP) layer subsequent to a 1D convolutional layer. The GAP layer efficiently conducts a global average pooling operation across each feature map, ensuring a streamlined process with no additional optimization parameters required within the layer itself. This improvement reduces the number of parameters for model training from 81,382 to 34,631, which substantially reduces the consumed memory and computational cost and somewhat simplifies the model structure while maintaining the model performance. In addition, we incorporate a batch normalization (BN) layer subsequent to the GAP layer, aiming to expedite the model's

convergence and solve the problem of network training difficulty due to the changing input data distribution in each layer. Finally, the computational results of the attention network layer are output through the dense layer, and the output serves as an indication of the probability of the present sample being part of a specific class. The equation is stated as follows:

$$S_j = \frac{e^{a_j}}{\sum_{k=1}^N e^{a_k}} \quad (10)$$

where N represents the total number of categories and a_j specifically refers to the j -th value within that output vector.

4. Experiment and Results

4.1. Datasets

We evaluated our model using the WISDM [27], UCI-HAR [28], and ISLD [29] datasets. The WISDM dataset comprises accelerometer data from 36 subjects performing six specific daily activities, as detailed in Table 1. The UCI-HAR dataset includes waist-embedded sensor data from 30 subjects recording six activities, as detailed in Table 2. The ISLD dataset consists of data from five sensors recording 18 activities from 10 subjects captured from five different perspectives, with the category information presented in Table 3. The data were split based on subject IDs, with the specific training and test sets detailed in Table 4.

Table 1. Activities included in the WISDM Dataset.

Activities	Samples	Percentage
Walking	424,400	38.6%
Jogging	342,177	31.2%
Upstairs	122,869	11.2%
Downstairs	100,427	9.1%
Sitting	59,939	5.5%
Standing	48,397	4.4%

Table 2. Activities included in the UCI-HAR Dataset.

Activities	Samples	Percentage
Walking	122,091	16.3%
Upstairs	116,707	15.6%
Downstairs	107,961	14.4%
Sitting	126,677	16.9%
Standing	138,105	18.5%
Laying	136,865	18.3%

Table 3. Activities included in the ISLD Dataset.

Gestures	
Bending (Bd)	jogging (jg)
Boxing (Bx)	Jumping (Jm)
watch-checking (wc)	Kicking (Kc)
Arms-crossing (Ac)	Jumping-in-place (Jp)
Getting-up (Gu)	Pointing (Pt)
Hands-clapping (Hc)	Running (rn)
One-hand-waving (Ow)	Head-scratching (Hs)
Two-hands-waving (Tw)	Sitting-down (Sd)
Standing (St)	Walking (Wk)

Table 4. Sample sizes for the three public datasets.

	WISDM	UCI-HAR	ISLD
Training set	13,654	7319	600
Test set	3036	2947	76

4.2. Experimental Setup

Utilizing an RTX 3090 GPU for effective computation, experiments were carried out on a Windows 10 operating system and implemented using the TensorFlow and Keras deep learning frameworks. Python 3.8 was utilized as the programming language. All layers commenced their training with random weights and biases during the fully supervised training phase of the model. To calculate the error between the expected and actual numbers, the cross-entropy function was utilized. The network used the Tanh function specifically for its attention layer while utilizing the ReLU nonlinear activation function for the network; the Adam optimizer was chosen for training. During the data preparation phase, signal interference or equipment malfunction during data collecting led to missing values, which were first filled using linear interpolation. To lessen the training bias, the data characteristics were then standardized to a range of 0 to 1. Ultimately, a sliding window with a 50% overlap was utilized to partition the data, ensuring continuity across the sequential elements. For the WISDM and UCI-HAR datasets, the sliding window length was set to 128, while for the ISLD dataset it was set to 28. The user ID of the subject served as the basis for data segmentation. To achieve optimal performance, extensive hyperparameter tuning was conducted, including a grid search to determine the optimal learning rate of 0.001, batch size of 192, and dropout rate of 0.2. The number of training epochs was set to 100, with a learning rate scheduler employed to gradually reduce the learning rate of training, ensuring the accuracy and stability of the model.

4.3. Evaluation Metrics

The performance of classifiers in HAR can be measured by a variety of performance measures; in this paper, we have selected four performance metrics: accuracy, precision, recall, and F₁-Score. Accuracy is defined as the proportion of correctly predicted samples out of the total number of samples. If the data are balanced, using the accuracy rate to measure the classification performance is recommended. The accuracy rate is defined as follows: $\frac{TP+TN}{TP+TN+FP+FN}$. However, accuracy metrics do not adequately account for the bias that arises from unbalanced datasets; thus, the F₁-Score is often more useful as a performance metrics than the accuracy. The F₁-Score is the weighted average of precision and recall, combining the performance of both. Especially in the case of unbalanced data (for instance, both WISDM and ISLD are unbalanced datasets), the F₁-Score offers a more balanced evaluation of model performance compared to precision or recall individually. The precision rate is defined as $\frac{TP}{TP+FP}$, recall is defined as $\frac{TP}{TP+FN}$, and the F₁-Score is calculated using the formula

$$F_1 = \sum_i 2 \cdot w_i \frac{\text{precision}_i \cdot \text{recall}_i}{\text{precision}_i + \text{recall}_i}, \quad (11)$$

where TP represents positive examples of correct classification, TN indicates cases where negative examples are accurately classified, FP signifies occasions where positive examples are mistakenly labeled, and FN covers situations where negative examples are erroneously mislabeled. $w_i = n_i / N$ is the proportion of samples of class i , n_i is the number of samples of class i , and N is the total number of samples. The F1-score is used to assess the effectiveness of the classification model, taking a value in the range between 0 and 1. When the F₁-Score is 1, this indicates that both the precision and recall rates of the model have reached the optimal level.

4.4. Experimental Results

4.4.1. Evaluation Study of Datasets

In this study, to undertake a rigorous validation of the BSTCA-HAR model's performance, we trained and then tested it on datasets with different characteristics. As shown in Table 5, the BSTCA-HAR model performed excellently across all three datasets, particularly achieving an accuracy of 96.1% on the ISLD dataset. This demonstrates the model's high generalization capability in handling tasks with a wide range of activity types, including both static and dynamic activities. Overall, the model maintained high levels of accuracy, recall, and precision, all above 92%, indicating its ability to accurately recognize user activities. These results are based on data evaluation in the final testing phase, reflecting the model's performance on independent test sets after the training process.

Table 5. Action recognition accuracy on the datasets.

Evaluation Metric	WISDM	UCI-HAR	ISLD
Accuracy	93.2%	91.9%	96.1%
Recall	92.0%	92.0%	96.0%
Precision	93.0%	92.0%	96.0%

Figures 2–4 show the confusion matrices of the model on the test datasets, using heatmaps to clearly reveal the specific performance of the model in classification tasks. To distinguish the data representation, the horizontal axis depicts the forecasted outcomes, whereas the vertical axis mirrors the observed or genuine values. On the WISDM dataset, the model correctly classified 6612 instances. To highlight the concern regarding imbalanced datasets, it is crucial to recognize that they can potentially skew the model's predictions in favor of the prevalent class, resulting in a bias towards the majority group and leading to decreased performance on some minority classes (e.g., “Downstairs” and “Upstairs”). Despite these confusions, the overall performance of the model remains quite impressive, with an average accuracy of around 93%, indicating that the model can accurately classify activities into the correct categories in most cases.

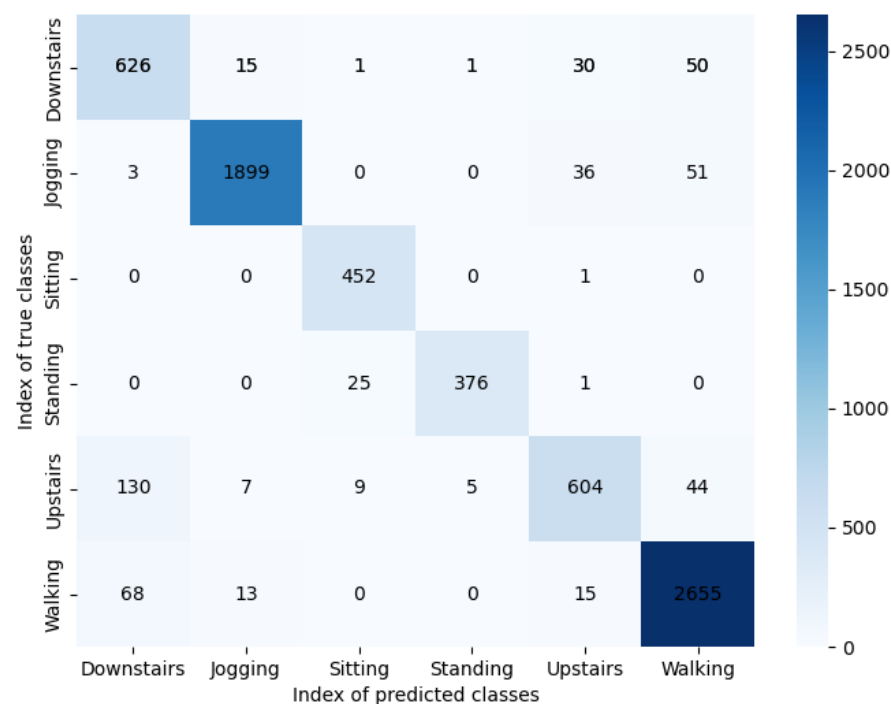


Figure 2. Classification confusion matrix on WISDM.

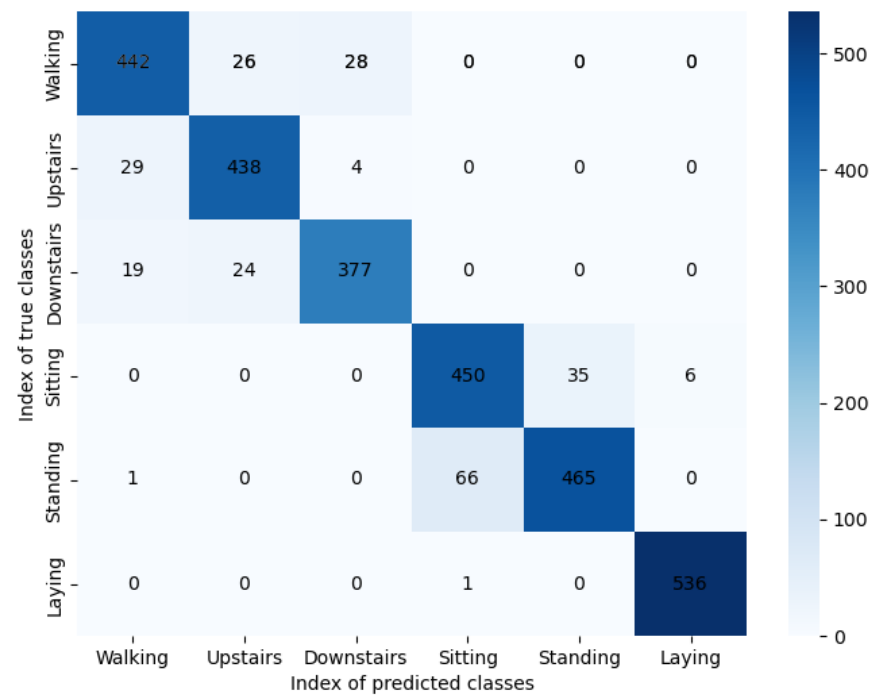


Figure 3. Classification confusion matrix on UCI-HAR.

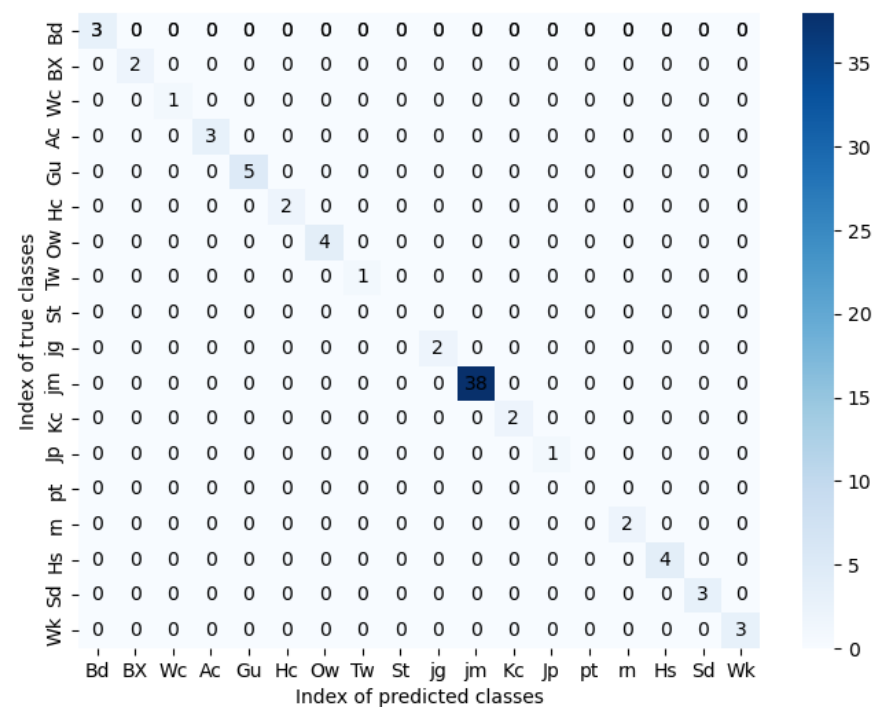


Figure 4. Classification confusion matrix on ISLD. Each activity is represented by an abbreviation, with its full name as follows: Bd, Bending; Jg, Jogging; Bx, Boxing; Jm, Jumping; Wc, Watch-checking; Kc, Kicking; Ac, Crossing the arms; Jp, Jumping in place; Gu, Getting up; Pt, Pointing; Hc, Clapping hands; Rn, Running; Ow, Waving one hand; Hs, Head scratching; Tw, Waving two hands; Sd, Sitting down; St, Standing; Wk, Walking.

When evaluated on the UCI-HAR dataset, which includes approximately 2947 new instances, the model achieved an overall accuracy of around 92%, indicating a commendable performance. This demonstrates that the model can accurately classify human activities in most cases. However, despite the overall excellent performance, the model still shows

some tendency for misclassification between certain activities. For instance, there may be misclassifications between the activities “Walking” and “Downstairs” as these activities share similarities in human movement patterns, particularly in terms of changes in acceleration and angular velocity. When an individual walks, especially on uneven surfaces such as stairs or slopes, the rhythm of their steps and the tilt of their body can create periodic fluctuations similar to those observed when descending stairs. This similarity can cause the data recorded by accelerometers and gyroscopes to overlap between the two activities, increasing the difficulty for the model in distinguishing between them. Additionally, the model’s performance on the “Standing” activity may also be influenced by other similar static activities such as “Sitting”, as both are relatively static and lack significant dynamic changes. The sensor data might not have enough distinction to separate them, although this error is not very prominent; In the case of ISLD gesture recognition, upon excluding the Null class from the classification tasks, our approach achieved an overall accuracy of around 96% on the task test. Notably, the ISLD model achieved 100% accuracy in recognizing 14 specific behaviors with significant Doppler effects, such as bending and crossing the arms. This accomplishment not only demonstrates our model’s ability to sensitively capture subtle differences in movements but also further confirms its exceptional performance in handling complex temporal dependencies and varying human gesture recognition tasks.

4.4.2. Experimental Results of the Suggested Model Using LOSO Cross-Validation

To further validate the model’s performance, we conducted a more comprehensive evaluation using LOSO (Leave-One-Subject-Out) cross-validation. This method involves using data from one subject for testing while the remaining subjects’ data are used for training. This approach simulates a realistic application scenario, providing a more rigorous evaluation. Table 6 presents the performance of various models using LOSO cross-validation on the WISDM dataset, which includes data from 36 subjects performing six activities. The training set comprises data from 30 subjects, while the test set includes data from the remaining six subjects.

Table 6. LOSO cross-validation for model performance evaluation on the WISDM dataset.

Models	95% Confidence Interval	Accuracy
RF [30]	(0.8191, 0.8228)	86.23%
SVM [31]	(0.8592, 0.8647)	89.08%
Ensemble of AE [32]	(0.7407, 0.7511)	88.01%
Bidirectional LSTM [33]	(0.8896, 0.8997)	89.08%
CNN-LSTM [34]	(0.8703, 0.8845)	88.01%
LSTM-CNN [8]	(0.9095, 0.9154)	91.34%
Convolutional AE [35]	(0.8997, 0.9058)	90.28%
CNN + Handcrafted features[36]	(0.8609, 0.8712)	86.75%
Proposed	(0.9245, 0.9391)	93.18%

We compared the proposed model with the traditional HAR model and examined the accuracy of activity recognition. In this comparative evaluation, we consider the following approaches: “Ensemble of AE”; “Bidirectional LSTM”; “CNN-LSTM”; “LSTM-CNN”; “Convolutional AE”; “CNN + Handcrafted features”; and two shallow machine learning methods, RF and SVM. We performed a five-fold cross-validation by taking the mean and standard deviation of all accuracy measures to assess the accuracy of the model at a 95% confidence level.

The following conclusions can be drawn. Regarding accuracy, although each baseline model has its merits, the proposed method demonstrates certain advantages. Specifically, while Ensemble of AE (88.01%) and Bidirectional LSTM (89.08%) effectively processed the data and exhibited competitiveness, their average accuracy confidence intervals were relatively wide, indicating potential instability in the results and a limited upper accuracy bound. The hybrid CNN and LSTM models and the Convolutional AE model, although

theoretically strong in feature extraction, had wider confidence intervals, suggesting slightly lower performance stability and reliability. The CNN model combined with handcrafted features (86.75%) was also limited by the quality and generalization ability of feature engineering, with its confidence interval reflecting dependency on specific datasets. While shallow machine Learning methods such as Random Forest (86.23%) and SVM (89.08%) showed good stability and reliability, their wide confidence intervals in terms of accuracy indicate potential limitations when dealing with highly variable datasets.

In contrast, the proposed method not only achieved an average accuracy of 93.18% but also had a confidence interval of [0.9245, 0.9391], demonstrating high consistency and stability. In summary, the proposed method not only achieves a significant improvement in accuracy, it also demonstrates superior performance in terms of stability and generalization ability.

4.4.3. Comparative Evaluation Using Various Network Architectures

As shown in Table 7, using six model structures (A, B, C, D, E, and BSTCA-HAR) devised specifically for the purpose of experimental analysis and comparisons, we examined the effects of various network structures on the overall efficacy and functionality of the model. This evaluation of the classification outcomes focuses on two crucial aspects: first, the calibration and optimization of model parameters; and second, computation of the F₁-Score on the designated test set, offering an alternative and thorough assessment of model performance. These evaluations were conducted on the WISDM test set.

Table 7. Evaluating the efficacy of different network architectures on the WISDM dataset.

Models		Building Blocks						Params	F ₁ -Score
A		Conv_1	Max_pooling	Conv_2	Flatten	Dense	Output	502,726	82.10%
B	LSTM_1	Conv_1	Max_pooling	Conv_2	GAP		Output	34,438	82.48%
C	LSTM_1	Conv_1	Max_pooling	Conv_2	GAP	BN	Output	41,286	84.05%
D	LSTM_1	Conv_1	Residual_Block		GAP	BN	Output	34,598	88.82%
E	LSTM_1;LSTM_2	Conv_1	Residual_Block	ATT	GAP	BN	Output	35,527	92.84%
Proposed	Bi-LSTM	Conv_1	Residual_Block	ATT	GAP	BN	Output	34,631	93.17%

Using the conventional CNN architecture, Model A achieved an F₁-Score of 82.10% on the WISDM test set. With almost 502,000 parameters, primarily in the dense layer, its performance is expensive and needs 1681 milliseconds of training time per round. Notwithstanding its correctness, Model A's quantity of parameters and training duration are significant drawbacks. In Model B, time series functionality is introduced. The data are initially fed into a two-layer LSTM and subsequently processed by a convolutional component for the purpose of extracting salient features. This approach is necessitated by the fact that activity recordings captured by mobile sensors inherently consist of time series data where actions evolve dynamically over time; the LSTM architecture adeptly captures these temporal nuances. The fully connected layer that comes after the convolutional layers in Model B is swapped out with a GAP layer. By applying the GAP layer to every feature map, this replacement normalizes the overall network topology and streamlines the model, resolving the overfitting problem. Notably, Model B has 34,438 parameters, which is a considerable drop of 93.15% compared to Model A. Despite this substantial parameter reduction, the model's efficiency remains virtually unchanged, suggesting that the combination of LSTM for capturing temporal information and the GAP layer is both feasible and effective.

Model C incorporates a BN layer subsequent to the GAP layer, aiming to normalize the output from a previous layer and accelerate model convergence. Furthermore, in order to lessen the computational and memory load, the classic max-pooling layer is employed to shrink the spatial dimensions of the feature maps; nevertheless, this results in the loss of some information, particularly significant features that could be lost when working with time series data. For more effective feature extraction and temporal modeling, Model D

substitutes a TCN layer with three residual blocks for the max-pooling layer. In addition to preserving more time series data and features, replacing the max-pooling layer with the TCN layer greatly enhances the model's accuracy and performance, with Model D surpassing Model C by an average of 4.77%. Model E builds on the foundation of Model D by adding another LSTM layer and an attention mechanism, further enhancing the model's feature extraction capabilities, dynamic weight allocation, and ability to capture long-term dependencies. In this paper, our model is based on Model E, utilizing the characteristics of BiLSTM. By replacing the two LSTM layers with a BiLSTM, we enhanced the feature extraction and contextual understanding capabilities, allowing for better handling of complex time series data. As a result, our model achieved an F₁-Score of 93.17% on the test set with only 34,631 parameters, highlighting its efficiency and effectiveness compared to other models.

In summary, the performance differences between the models mainly stem from their ability to handle time series data, feature extraction mechanisms, and the complexity of the model structures. By replacing fully connected layers with BN and GAP layers and incorporating advanced methods such as LSTM, TCN, attention mechanisms, and bidirectional LSTM, our model makes significant progress in capturing complex temporal dependencies and improving recognition accuracy. At the same time, it maintains a low number of parameters and efficient computational capability.

4.4.4. Exploration of Hyperparameter Tuning on Model Efficacy

The profound influence of hyperparameters on model performance necessitates a rigorous examination. Here, we conduct a thorough analysis of the influence of various hyperparameters on the predictive ability of the model, including the number of units and the size of the batch, among others. To assess the model's performance in a methodical manner, We embarked on a suite of tests utilizing the WISDM dataset, utilizing the F₁-Score as a reliable metric to quantify and compare the performance outcomes.

(1) Effect of the number of LSTM cells

Selection of LSTM unit count per layer is pivotal, as a higher number facilitates the extraction of more intricate and profound features, enhancing the model's learning capabilities. However, this augmentation in feature complexity comes at the cost of increased model parameters and potential exacerbation of overfitting issues. Figure 5 succinctly illustrates the trade-off between accuracy gains and parameter growth in the BSTCA-HAR model, emphasizing the strategic importance of optimizing the LSTM unit count. With the increasing number of network parameters, the accuracy of the model does not increase; thus, the F₁-Score effect is the best when the number of Bi-LSTM units is 32.

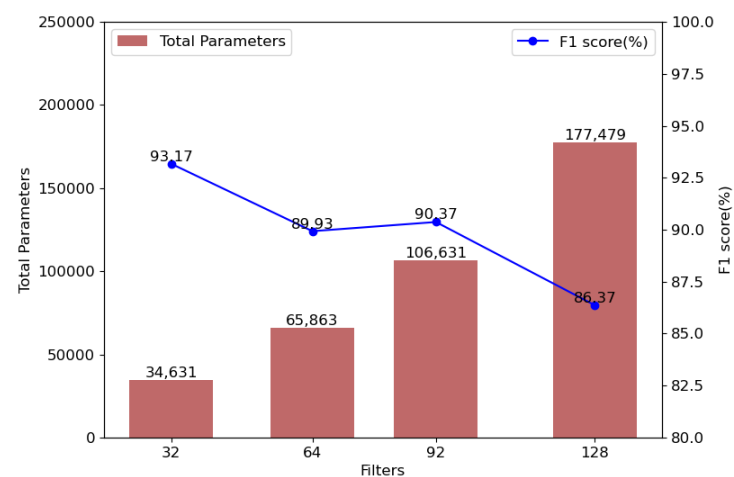


Figure 5. Effect of increasing the number of LSTM layer cells on model performance.

(2) Effect of Batch Size Variations

Batch processing with smaller batches is a commonly used data processing technique in deep learning and other machine learning tasks. In this method, the dataset is divided into multiple smaller batches (or subsets), each containing a certain number of samples. In each iteration, the model calculates the gradient of the current batch of data and uses this gradient to update the model's weights. Because batch data are used instead of the entire dataset, this update is typically faster than batch gradient descent using the entire dataset and more stable than stochastic gradient descent using a single sample. Figure 6 demonstrates the trend of model accuracy with five different batch sizes. The results show that the model has the highest accuracy when the batch size is set to 192.

Additionally, we conducted experiments with other hyperparameters, such as different learning rates and optimizers. The findings indicated that the influence of these hyperparameters on the model's execution was relatively marginal. For instance, we experimented with several optimizers (Adam, SGD, RMSprop) and learning rates (0.001, 0.01, 0.1), discovering that these changes had no discernible effect on the F_1 -Score. Therefore, for the sake of simplicity and emphasis, this paper primarily discusses the impact of the number of filters and batch size on model performance.

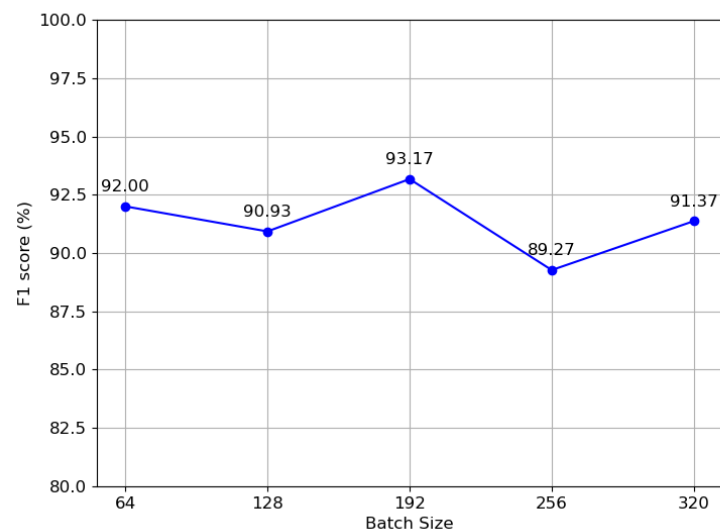


Figure 6. Effect of batch size on model performance.

5. Conclusions and Future Work

This paper proposes a HAR recognition model called BSTCA-HAR based on an LSTM neural network. In addition, the proposed model integrates an attention mechanism and TCN network. The model inputs data collected by mobile sensors into the BiLSTM layer; the bidirectional structure of the LSTM considers both past and future information, providing a better understanding of the context within sequential data. Subsequently, a TCN is utilized to process long time series information and an attention mechanism is introduced to learn the associations and dependencies between different time steps in the action sequences, effectively improving the model's recognition accuracy. After feature extraction, GAP and BN layers are added to significantly reduce the number of model parameters and accelerate the model's convergence speed. We tested the proposed model's efficacy and generalizability using three publicly available datasets (WISDM, UCI-HAR, and ISLD), each with varying data features. With accuracy rates of 93.2%, 91.9%, and 96.1%, respectively, in human activity recognition, the model demonstrated strong performance. Multiple baseline models were used in comparison studies, which showed that the BSTCA-HAR model has higher training efficiency and improved generalization ability. Our model possesses the capability to dynamically extract salient activity features while exhibiting the benefits of a reduced parameter count and enhanced prediction precision.

Lastly, more investigation into the cooperative learning of visual and non-visual modalities may be conducted in future studies. This includes using multimodal data fusion techniques to further increase the accuracy and resilience of action identification by merging visual modalities such as video frames and skeletal data with non-visual modalities such as inertial sensor data.

Author Contributions: Conceptualization, S.Y.; methodology, Y.Y.; validation, Y.Y. and F.Y.; formal analysis, X.T.; investigation, S.Y.; data curation, L.H. and F.Y.; writing—original draft preparation, Y.Y.; writing—review and editing, L.H. and F.Y.; visualization, L.H. and Y.Y.; supervision, Y.Y.; funding acquisition, L.H. and X.T. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by National Natural Science Foundation of China (Nos. 11361104 and 12261104), the Youth Talent Program of Xingdian Talent Support Plan (No. XDYC-QNRC-2022-0514), and the Yunnan Provincial Basic Research Program Project (Nos. 202301AT070016 and 202401AT070036).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data generated and analyzed during this study are included in this article.

Acknowledgments: We would like to thank the editor and the anonymous referees for their valuable comments and suggestions that greatly improved the presentation of this work. This work was supported by various funding sources, as detailed in the Funding section.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Gul, M.A.; Anwar, S.M.; Majid, M.; Alnowami, M. Patient monitoring by abnormal human activity recognition based on CNN architecture. *Electronics* **2020**, *9*, 1993. [\[CrossRef\]](#)
2. Qi, W.; Aliverti, A. A multimodal wearable system for continuous and real-time breathing pattern monitoring during daily activity. *IEEE J. Biomed. Health Inform.* **2019**, *24*, 2199–2207. [\[CrossRef\]](#) [\[PubMed\]](#)
3. Lauraitis, A.; Maskeliūnas, R.; Damaševičius, R.; Wozniak, M. A smartphone application for automated decision support in cognitive task-based evaluation of central nervous system motor disorders. *IEEE J. Biomed. Health Inform.* **2019**, *23*, 1865–1876. [\[CrossRef\]](#)
4. Poppe, R. A survey on vision-based human action recognition. *Image Vis. Comput.* **2010**, *28*, 976–990. [\[CrossRef\]](#)
5. Ramanujam, E.; Perumal, T.; Padmavathi, S. The title of the cited article. *IEEE Sens. J.* **2021**, *21*, 13029–13040. [\[CrossRef\]](#)
6. Hamad, R.A.; Woo, W.L.; Wei, B.; Yang, L. Overview of Human Activity Recognition Using Sensor Data. *arXiv* **2023**, arXiv:2309.07170.
7. Jung, I.Y. A review of privacy-preserving human and human activity recognition. *Int. J. Smart Sens. Intell. Syst.* **2020**, *13*, 1–13. [\[CrossRef\]](#)
8. Odhiambo, C.O.; Saha, S.; Martin, C.K.; Valafar, H. Human activity recognition on time series accelerometer sensor data using LSTM recurrent neural networks. *arXiv* **2022**, arXiv:2206.07654.
9. Xia, K.; Huang, J.; Wang, H. LSTM-CNN architecture for human activity recognition. *IEEE Access* **2020**, *8*, 56855–56866. [\[CrossRef\]](#)
10. Tan, M.; Ni, G.; Liu, X.; Zhang, S.; Wu, X.; Wang, Y.; Zeng, R. Bidirectional posture-appearance interaction network for driver behavior recognition. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 13242–13254. [\[CrossRef\]](#)
11. Xiao, F.; Pei, L.; Chu, L.; Zou, D.; Yu, W.; Zhu, Y.; Li, T. A Deep Learning Method for Complex Human Activity Recognition Using Virtual Wearable Sensors. *arXiv* **2021**, arXiv:2105.02782.
12. Li, M.; Chen, W. FFT-based deep feature learning method for EEG classification. *Biomed. Signal Process. Control* **2021**, *66*, 102492. [\[CrossRef\]](#)
13. Jain, A.; Kanhangad, V. Human activity classification in smartphones using accelerometer and gyroscope sensors. *IEEE Sens. J.* **2017**, *18*, 1169–1177. [\[CrossRef\]](#)
14. Fullerton, E.; Heller, B.; Munoz-Organero, M. Recognizing human activity in free-living using multiple body-worn accelerometers. *IEEE Sens. J.* **2017**, *17*, 5290–5297. [\[CrossRef\]](#)
15. Singh, D.; Merdivan, E.; Psychoula, I.; Kropf, J.; Hanke, S.; Geist, M.; Holzinger, A. Human activity recognition using recurrent neural networks. In *Machine Learning and Knowledge Extraction, Proceedings of the First IFIP TC 5, WG 8.4, 8.9, 12.9 International Cross-Domain Conference, CD-MAKE 2017, Reggio, Italy, 29 August–1 September 2017*; Springer International Publishing: Cham, Switzerland, 2017; pp. 267–274.

16. Zhao, Y.; Yang, R.; Chevalier, G.; Xu, X.; Zhang, Z. Deep residual bidirLSTM for human activity recognition using wearable sensors. *Math. Probl. Eng.* **2018**, *2018*, 1–13.
17. Mohi ud Din Dar, G.; Bhat, G.M.; Ahmad, S.R.; Reshi, J.A.; Ahmad, S.R.; Bhardwaj, S. A novel framework for classification of different Alzheimer's disease stages using CNN model. *Electronics* **2023**, *12*, 469. [\[CrossRef\]](#)
18. Feichtenhofer, C.; Pinz, A.; Zisserman, A. Convolutional two-stream network fusion for video action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1933–1941.
19. Zheng, Y.; Liu, Q.; Chen, E.; Ge, Y.; Zhao, J. Time series classification using multi-channels deep convolutional neural networks. In *Proceedings of the International Conference on Web-Age Information Management*; Springer International Publishing: Cham, Switzerland, 2014; pp. 298–310.
20. Chen, T.Y. Research on action recognition based on deep learning. *Inf. Technol. Inform.* **2023**, *8*, 172–175.
21. Cho, S.; Maqbool, M.; Liu, F.; Foroosh, H. Self-attention network for skeleton-based human action recognition. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Snowmass Village, CO, USA, 1–5 March 2020; pp. 635–644.
22. Pravanya, P.; Priya, K.L.; Khamarjaha, S.K.; Shanthini, D. Human Activity Recognition Using CNN-Attention-Based LSTM Neural Network. In *Intelligent Communication Technologies and Virtual Mobile Networks*; Springer Nature Singapore: Singapore, 2023; pp. 593–605.
23. Thakur, D.; Biswas, S. Online change point detection in application with transition-aware activity recognition. *IEEE Trans. Hum.-Mach. Syst.* **2022**, *52*, 1176–1185. [\[CrossRef\]](#)
24. Thakur, D.; Biswas, S. Attention-based deep learning framework for hemiplegic gait prediction with smartphone sensors. *IEEE Sens. J.* **2022**, *22*, 11979–11988. [\[CrossRef\]](#)
25. Smagulova, K.; James, A.P. A survey on LSTM memristive neural network architectures and applications. *Eur. Phys. J. Spec. Top.* **2019**, *228*, 2313–2324. [\[CrossRef\]](#)
26. Lea, C.; Vidal, R.; Reiter, A.; Hager, G.D. *Temporal Convolutional Networks: A Unified Approach to Action Segmentation*; Springer International Publishing: Cham, Switzerland, 2016.
27. Kwapisz, J.R.; Weiss, G.M.; Moore, S.A. Activity recognition using cell phone accelerometers. *ACM SigKDD Explor. Newsl.* **2011**, *12*, 74–82. [\[CrossRef\]](#)
28. Reyes-Ortiz, J.; Anguita, D.; Ghio, A.; Oneto, L.; Parra, X. Human Activity Recognition Using Smartphones. UCI Machine Learning Repository. Available online: <https://archive.ics.uci.edu/dataset/240/human+activity+recognition+using+smartphones> (accessed on 22 July 2024).
29. Angelini, F.; Naqvi, S.M. Intelligent Sensing Lab Dataset (ISLD) for Posture-based Human Action Recognition based on Pose Data. Dataset. Ph.D. Dissertation, Newcastle University, Newcastle upon Tyne, UK, 2021.
30. Lee, J.; Park, S.; Shin, H. Detection of hemiplegic walking using a wearable inertia sensing device. *Sensors* **2018**, *18*, 1736. [\[CrossRef\]](#) [\[PubMed\]](#)
31. Thang, H.M.; Viet, V.Q.; Thuc, N.D.; Choi, D. Gait identification using accelerometer on mobile phone. In Proceedings of the 2012 International Conference on Control, Automation and Information Sciences (ICCAIS), Saigon, Vietnam, 26–29 November 2012; pp. 344–348.
32. Garcia, K.D.; de Sá, C.R.; Poel, M.; Carvalho, T.; Mendes-Moreira, J.; Cardoso, J.M.; de Carvalho, A.C.; Kok, J.N. An ensemble of autonomous auto-encoders for human activity recognition. *Neurocomputing* **2021**, *439*, 271–280. [\[CrossRef\]](#)
33. Yu, S.; Qin, L. Human activity recognition with smartphone inertial sensors using bidir-LSTM networks. In Proceedings of the 2018 3rd International Conference on Mechanical, Control and Computer Engineering (ICMCCE), Huhhot, China, 14–16 September 2018; pp. 219–224.
34. Deep, S.; Zheng, X. Hybrid model featuring CNN and LSTM architecture for human activity recognition on smartphone sensor data. In Proceedings of the 20th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT), Gold Coast, QLD, Australia, 5–7 December 2019; pp. 259–264.
35. Varamin, A.A.; Abbasnejad, E.; Shi, Q.; Ranasinghe, D.C.; Rezaatofighi, H. Deep auto-set: A deep auto-encoder-set network for activity recognition using wearables. In Proceedings of the 15th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, New York, NY, USA, 5–7 November 2018; pp. 246–253.
36. Ignatov, A. Real-time human activity recognition from accelerometer data using convolutional neural networks. *Appl. Soft Comput.* **2018**, *62*, 915–922. [\[CrossRef\]](#)

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.