symmetry

MDPI

Article

Improving the Performance and Explainability of Indoor Human Activity Recognition in the Internet of Things Environment

Ayse Betul Cengiz [1], Kokten Ulas Birant [2], Mehmet Cengiz [3], Derya Birant [2,*] and Kemal Baysari [1]

[1] Graduate School of Natural and Applied Sciences, Dokuz Eylul University, Izmir 35390, Turkey
[2] Department of Computer Engineering, Dokuz Eylul University, Izmir 35390, Turkey
[3] School of Computing, Newcastle University, Newcastle upon Tyne NE1 7RU, UK
* Correspondence: derya@cs.deu.edu.tr

Abstract: Traditional indoor human activity recognition (HAR) has been defined as a time-series data classification problem and requires feature extraction. The current indoor HAR systems still lack transparent, interpretable, and explainable approaches that can generate human-understandable information. This paper proposes a new approach, called Human Activity Recognition on Signal Images (HARSI), which defines the HAR problem as an image classification problem to improve both explainability and recognition accuracy. The proposed HARSI method collects sensor data from the Internet of Things (IoT) environment and transforms the raw signal data into some visual understandable images to take advantage of the strengths of convolutional neural networks (CNNs) in handling image data. This study focuses on the recognition of symmetric human activities, including walking, jogging, moving downstairs, moving upstairs, standing, and sitting. The experimental results carried out on a real-world dataset showed that a significant improvement (13.72%) was achieved by the proposed HARSI model compared to the traditional machine learning models. The results also showed that our method (98%) outperformed the state-of-the-art methods (90.94%) in terms of classification accuracy.

Keywords: machine learning; image classification; human activity recognition; convolutional neural networks; Internet of Things

Citation: Cengiz, A.B.; Birant, K.U.; Cengiz, M.; Birant, D.; Baysari, K. Improving the Performance and Explainability of Indoor Human Activity Recognition in the Internet of Things Environment. *Symmetry* **2022**, *14*, 2022. https://doi.org/10.3390/sym14102022

Academic Editors: Tzu Chuen Lu and Dalibor Štys

Received: 21 August 2022
Accepted: 23 September 2022
Published: 26 September 2022

1. Introduction

Human activity recognition (HAR) is the task of correctly identifying human activities (i.e., walking, eating, standing, and working) by analyzing sensor data collected by Internet of Things (IoT) devices. It is useful for understanding the behavioral human patterns in an IoT system. Our work focuses on human activity recognition in indoor environments.

The indoor HAR systems are important in many domains, such as assisted living and healthcare [1,2], biometric user identification for security [3], wellbeing in smart homes [4], evaluating employee performances in smart factories for Industry 4.0 [5], body motion analysis in sports, and monitoring safety (falls, injuries, and collisions) [6,7] in an IoT environment. Activity recognition is a significant indicator of participation, quality of life, and lifestyle. Human activities carry a lot of information about the context (i.e., a person's identity, personality, and mental state) and help systems to achieve context-awareness. For example, patient activity recognition is critical in analyzing treatment progress and can provide context information for decision-making for better treatment and care. Similarly, rehabilitation specialists and therapists can remotely benefit from information on patient activities outside of a health center. Having detected the activities, a broad range of analyses (i.e., activities by age group, by gender, by location, by day of the week) can be performed to answer the questions of where and when users perform which types of activities. It can help detect the abnormalities in surveillance systems, therefore preventing any unfavorable

consequences. Using wearable sensors, HAR applications detect the actions of the users to provide them with intelligent personal assistance and recommendation. In the military, it is important to recognize the activities of the soldiers to provide feedback to their managers that assist them in real time. Consequently, there are numerous potential computing systems where recognizing human activities plays an important role.

One of the main problems associated with the current indoor HAR systems is that they have been considered as black-box systems, using nonunderstandable sensor data and providing predictions without being able to explain them. Without explainability and transparency, an HAR model is not trustworthy for making real-world decisions, especially the high-risk ones in the IoT systems. The main aim of this study is to provide an interpretable, basic, and reliable approach to indoor HAR problems.

The main contributions of this study can be summarized as sixfold. (i) It proposes a new approach, called *Human Activity Recognition on Signal Images* (HARSI), which converts the time-series data to signal images and feeds them into a CNN for the image classification task. (ii) It is the first attempt to combine four methodologies: signal image-based indoor HAR, IoT, explainable artificial intelligence (XAI), symmetry, and deep learning (DL). (iii) It provides an important contribution by improving human-level explainability for smart sensor data by using signal images in the field of indoor HAR. (iv) It takes into consideration symmetric human activities. (v) This study is also original in that it compares the performances of different nine CNN architectures on signal image data in terms of accuracy to determine the best one for indoor HAR. (vi) The proposed method outperformed both the classical machine learning methods (13.72% improvement) and the state-of-the-art methods (7.06% improvement) on the same dataset.

With quantitative evaluation in experiments, we demonstrated the effectiveness of the proposed HARSI approach on a real-world dataset. The experimental results showed that HARSI accurately recognized symmetric human activities, including walking, jogging, standing, sitting, moving downstairs, and moving upstairs. The results also showed that a significant improvement was achieved by the proposed HARSI method (98%) compared to the traditional machine learning methods (84.28%) and the state-of-the-art methods (90.94%) in terms of recognition accuracy.

The organization of the paper is as follows. Section 2 explains the recent previous studies on HAR that use a deep learning technique. Section 3 describes the proposed HARSI method in detail. Section 4 provides a brief description of the data and presents the experimental results. This section also gives debates on the subject and explains our solutions. Section 5 presents concluding remarks with the main findings and opportunities for further research.

2. Related Work

Research in the field of HAR is becoming increasingly important with the rapid development of smart sensor systems [8]. Especially, HAR is of great significance in the Internet of Things (IoT) applications, which include sensor and communication technologies. Different sensor types have been utilized in the HAR systems, including wearable sensors [9–13], vision-based sensors/cameras [14–16], health sensors [17], and environmental sensors [6,18,19]. The ambient sensor-based HAR applications detect activities from the sensors that are installed at fixed locations (i.e., home, factory) or placed on a fixed object (i.e., fridge, door, toilet flush). In this study, we focused on wearable sensors since they provide many advantages such as privacy protection, wide coverage area, and high robustness when modeling an activity classifier.

Wearable sensors are lightweight (few grams), small in size (few mm), easy to program, and low cost. Using either a strap or an adhesive, they can be easily attached to many different body parts (i.e., arm, waist, shoulder, wrist, or leg) depending on the human activities being studied [11,20]. A pair of sensors can be symmetrically located on a human body to collect synchronized measurements of them, allowing for the assessment of symmetric human behaviors. The accelerometer (A) [21–24] is one of the widely used

sensors to collect acceleration data related to human activity. Besides the accelerometer, gyroscope (G) and magnetometer (M) sensors are attached to the body in various ways for monitoring actions at a particular point in time. The combination of the sensors (A, G, M) can also provide useful information when analyzed by machine learning methods to recognize human activities [25]. Most HAR systems [26,27] have been currently developed by smartphones; even other types of smart devices such as smartwatches, wristbands, and smart glasses have also been successfully employed.

Deep learning (DL) is one of the most exciting technologies that implements symmetry in computer science. In the literature, deep learning techniques have been proven to be powerful in classification. The recent studies [1–32] on human activity recognition that use deep learning technology are given in Table 1. A variety of DL methods have been successfully used for HAR, such as recurrent neural networks (RNN) [24,26,28,29], long short-term memory (LSTM) [22,23,30], autoencoder (AE) [4,20], deep neural network (DNN) [1,9,13], and convolutional neural network (CNN) [31,32].

Table 1. Comparison of the recent HAR studies with our study.

Ref	Year	Method					Description	Sensor Types	Data	Sensor Location	Number of Activities	Sensor-Data-Based	Signal-Image-Based	XAI
		CNN	DNN	RNN	LSTM	AE								
[8]	2022	√		√	√		Channel state information (CSI) based HAR	Wireless signal	CSI	Room	6	√	X	X
[9]	2022	√	√		√		Gait pattern analysis	A, G, M	SG *	Center of mass	7	√	X	X
[10]	2022	√					Personalization in HAR	A, G, M	UniMiB SHAR	Pocket	17	√	X	X
									Motion Sense		6			
									MobiAct		15			
[11]	2022	√			√		HAR from piezoelectric-based kinetic energy signals	KEH transducers	KEH	Hand, waist	5	√	X	X
[12]	2022	√					Feature extraction-based approach	A	WISDM	Pocket	6	√	X	X
[13]	2022	√	√		√		Comparative study on classifying human activities	A, G	UCI-HAR	Waist	6	√	X	X
[14]	2022	√		√			Gesture recognition in videos	Camera	SG	Room	4	√	X	X
[1]	2021		√				HAR from highly sparse body sensor data	A, RFID	Roomset1	Chest	4	√	X	X
									Roomset2					
[2]	2021	√		√	√		Feature extraction-based approach	A, G, M	UniMiB-SHAR	Pocket	17	√	X	X
[3]	2021	√			√		Biometric user identification	A, G	UCI-HAR	Waist	6	√	X	X
									USC-HAD		12			
[4]	2021			√	√	√	HAR in smart homes	Env. sensors	Orange4Home	Room	24	√	X	X
[5]	2021	√			√		Industry 4.0-oriented approach	A	WISDM	Pocket	6	√	X	X
[6]	2021	√			√		Human pose and motion estimation	Camera	SG	Room	5	√	X	X
								Env. sensors			8			
[15]	2021	√			√		Hybrid deep-learning-based model	Motion Kinect sensor	SG	Room	12	√	X	X
[16]	2021	√					HAR using skeleton datasets	Camera	UTD-MHAD	Room	27	√	X	X
									MSR-Action3D		20			

Table 1. *Cont.*

Ref	Year	Method					Description	Sensor Types	Data	Sensor Location	Number of Activities	Sensor-Data-Based	Signal-Image-Based	XAI
		CNN	DNN	RNN	LSTM	AE								
[17]	2021	√					Feature fusion-based approach	A, G, M	MHEALTH	Ankle, arm, chest	12	√	X	X
[18]	2021	√			√		Causality feature extraction based approach	Env. sensors	Aruba	Room	10	√	X	X
									Milan		15			
									Cairo		13			
[19]	2021	√			√		HAR based on the Inception-ResNet model	A, G	UCI-HAR	Waist	6	√	X	X
								Env. and body sensors	Opportunity	Room, body	18			
								A	Daphnet	Legs, hip	2			
								A, G, M	PAMAP2	Chest, ankle	18			
[20]	2021				√	√	Multiple domain DL framework	A, G, M	SG	Head, wrist, leg	12	√	X	X
[21]	2021	√					Feature-fusion-based approach	A, G	SG	Waist	6	√	X	X
									UCI-HAR					
[22]	2021	√			√		Recognizing transitional activities	A, G	HAPT	Waist	12	√	X	X
									HAD		5			
[23]	2021	√			√		Optimal deep-learning-based approach	A, G	UCI-HAR	Waist	6	√	X	X
									USC-HAD		12			
[24]	2021			√	√		HAR using time-series data	A	UniMiB SHAR	Pocket	17	√	X	X
								A	WISDM	Pocket	6			
								A	UCI-HAR	Waist	6			
[25]	2021	√					HAR on microcontrollers	A, G, M	PAMAP2	Hand, chest, ankle	12	√	X	X
[26]	2021	√		√	√		Hybrid deep-learning-based approach	A, G	UCI-HAR	Waist	6	√	X	X
								A	WISDM	Pocket				
[27]	2021	√					Feature-fusion-based approach	A, G	SG	Pocket	6	√	X	X
									UCI-HAR	Waist				
[28]	2021	√		√	√		Attention-based mechanism	A, G	HHAR	Hand, chest, ankle	6	√	X	X
								A, G, M	PAMAP2		12			
								A, G	USC-HAD		12			
[29]	2021	√		√			HAR using multimodal sensors	Multimodal	CMU-MMAC	Room	11	√	X	X
[30]	2021				√		Multimodal complex HAR	A, G, M	Lifelog	Pocket, wrist, chest	9	√	X	X
									PAMAP2	Wrist, arm, chest,	18			
[31]	2021	√			√		Hierarchical hybrid deep-learning-based approach	A, G	UCI-HAPT	Waist	12	√	X	X
									MobiAct	Pocket	11			
[32]	2021	√					Resource-constrained HAR	EMG sensors	Myo-TL	Elbow, wrist	9	√	X	X
									Db5		18			
Our Approach		√					Human activity recognition on signal images (HARSI)	A	WISDM	Pocket	6		√	√

* SG: self-generated.

In the literature, most HAR systems [33–77] applied a supervised learning method; on the other hand, other types of machine learning, such as unsupervised or semisupervised learning, have also been investigated. The most widely used classification methods in HAR systems are decision trees [33–37,39,45,48,49], multilayer perceptron [33,34,41,46,48,49],

support vector machines [12,35,37–40,42,44,45], naive Bayes [37,45,46], logistic regression [33,34,39,48,49], k-nearest neighbors [35–37,42,45], AdaBoost [47], and random forest [12,35–39,43,50].

In the literature, most HAR studies [12,13,21] focused on the identification of daily living activities such as standing, sitting, and walking. However, some previous studies tried to detect more specific types of activities such as cheating activity in an exam [14], rope jumping [25], shopping [30], housekeeping [18], hand-oriented activities (i.e., eating, clapping, writing) [78], virtual reality (VR) users' activities (slash, thrust, guard) [79], and sports activities (i.e., basketball, bowling, boxing, and tennis) [16]. Besides these high-level activities, some works [2,22] focused on the transitions between the activities, such as sit-to-stand or stand-to-sit. In addition, recognizing group activities such as punching, kicking, and pushing were also investigated in previous studies [80]. In this study, we built a CNN model to recognize six different activities, including walking, jogging, standing, sitting, moving downstairs, and moving upstairs.

Typically, an HAR framework contains the following stages: data collection, data preprocessing, feature extraction, feature selection, training, performance evaluation, classification, and decision-making. Using raw sensor data directly in machine learning is usually not practical since it does not carry sufficient information to distinguish different human activities. In other words, only one particular value at a specific time instant of an action does not carry enough information to describe an activity itself. For this reason, the standard HAR studies involve the feature extraction phase to transform time-series data into samples that summarize the data over a particular time period. In general, frequency-domain features and time-domain features are extracted from the raw sensor data, such as max, min, median, mean, peak-to-peak value, standard deviation, the number of zero crossings, skewness, kurtosis, and signal entropy [10]. Among them, the skewness informs about the symmetry of the signal. In the literature, some studies [17,21,27] mainly concentrated on the feature extraction issue since it plays an important role in the final system performance.

HAR models can be considered in two categories: per-subject (personalized) and cross-subject (generalized) models. In per-subject models, both the training and testing data are all from the same individual, since each person has unique characteristics of movements, such as speed corresponding to its physical properties (i.e., age, gender, height, and weight) and habits. On the other hand, in cross-subject models, the training and testing data come from all the persons. They provide many advantages, such as working on a large amount of data to build a robust classifier and dealing with a single classifier instead of multiple ones.

Our study differs from the studies aforementioned here in several aspects. In our study, we do not use the features that were extracted from sensor data as most HAR systems do; rather, we transfer time-series data into signal images that reflect the properties of activities. Here, we present a detailed analysis of the performances of different CNN architectures on human activity signal image data for the first time. Our aim is to provide an explainable artificial intelligence (XAI) approach that can give human-understandable information and prediction in an IoT system. In other words, in this study, we provide human-level explainability for smart sensor data in the field of indoor HAR. To the best of our knowledge, five concepts together (signal image-based indoor HAR, XAI, IoT, symmetry, and DL) have not been studied so far.

3. Proposed Method

3.1. Problem Definition

Human activity recognition is a problem of classifying data obtained from sensors into well-defined actions performed by humans, e.g., walking, jogging, and sitting. Recognition of activities is a challenging time-dependent task since there is no single and precise way or formula to define specific movements. Machine learning methods have played a major role in the analysis of sensor data for providing real-time feedback in HAR applications.

In an HAR system, an accelerometer sensor embedded in an IoT device is placed at a specific location on the human body and is synchronized to emit data in an IoT environment. The accelerometer measures three-dimensional acceleration referenced with the Earth's gravity during dynamic states. The sensor generates a signal along the x-axis, y-axis, and z-axis at a time step $t \in \{1,2, \ldots ,T\}$. Accelerometer measurements are collected in a time interval for a person. For this sensor, the sensing data along time can be represented by a multidimensional time series $S = [s_1, s_2, \ldots , s_t, \ldots]$ where s_t is the sensing signal of the sensor placed at a body location at time t. Each signal record has six attributes {datetime, sensorID, x-axis, y-axis, z-axis, activity}.

A single measurement is not enough to classify an activity because of the time-dependent nature of activities. To deal with this problem, S is segmented into multiple frames, called windows, and then, each frame is mapped into a predefined activity label. In our study, temporal segmentation through the sliding window technique is necessary to define the boundaries of signal images. Each image is annotated by an activity from a label set, which is denoted by $a_i \in \{A_1, A_2, \ldots ., A_m\}$, where m is the number of potential activities to be recognized. For instance, the class labels can be as follows: A_1 = walking, A_2 = sitting, and A_3 = stairs.

The activity recognition problem can be described as follows. Given a recorded signal time series S, the task is to detect an activity (e.g., walking) that infers human behavior in a time period. In the proposed approach, time-series data are converted to images by drawing three lines according to the recorded x–y–z values, and then, the images are fed into a CNN for the image classification task.

The concept of symmetry has been considered in many topics; similarly, it can be also discussed related to indoor human activity recognition. Figure 1 shows the example of symmetric and asymmetric activities with respect to the y-axis or arms/legs positions. Human activity such as walking can be considered as a symmetric movement depending on the biped's parameters such as slope angles. Similarly, a jogging activity is also symmetric since the arm and legs are coordinated and moving together at the same frequency; i.e., the phase-plane cycles of the two legs are identical. Some group activities are also defined as symmetric, such as shaking hands and hugging. Similarly, the activity "WalkTogether" is symmetric because "I am walking together with you" is the same as "you are walking together with me". On the other hand, some activities, such as kicking, falling down, pushing, picking up, and punching, can be categorized as asymmetric activities since two legs or two arms do not move simultaneously at the same angle or at the regions on symmetric sides.

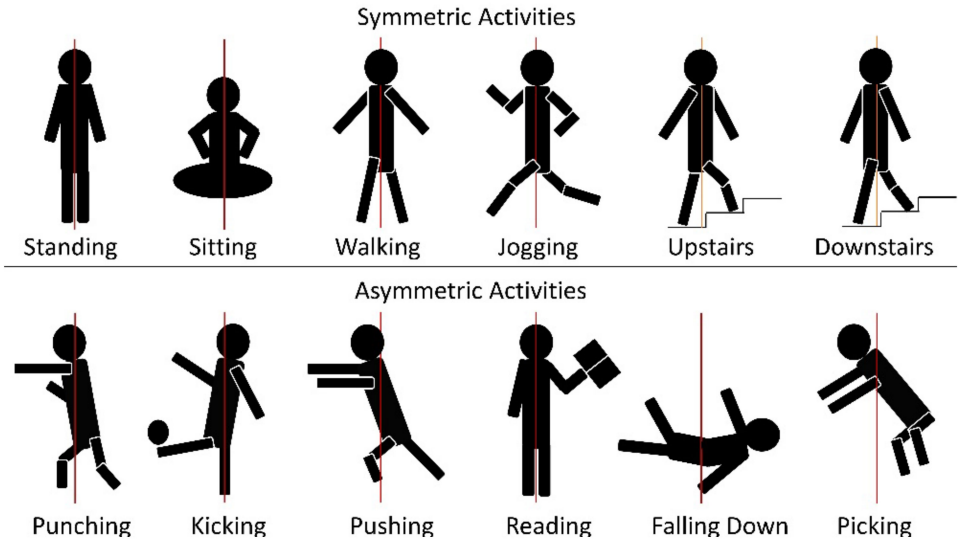


Figure 1. Examples of symmetric and asymmetric activities.

3.2. Proposed Approach

In a traditional indoor HAR system, features are commonly extracted from time-series sensor data by using statistical methods such as max, min, mean, peak-to-peak value, standard deviation, the number of zero crossings, skewness, kurtosis, and entropy. However, this input data cannot be interpretable by humans, as can be seen in Figure 2. For humans, the numerical values such as in Figure 2 cannot be matched with the activities. For instance, when the numeric feature vector [339, 27, 0.3, 0.4, 0.08, 0.06, 0.05, 0.07, . . .] is seen by a human, it cannot be directly associated with the "walking" activity since it lacks visual representation.

339,27,0.3,0.4,0.08,0.06,0.05,0.07,0,0,0,0,0,0.06,0.05,0.08,0.09,0.06,0.1,0.1,0.2,0,0.09,0.1,0.02,0.12,0.2,0.14,0.05,0.11,0.1,0,0,10,−0.23,1193.75,561.76,2625,2.39,4.45,2.72,3.2,5.13,3.2,11.14, Walking

1,6,0,0,0,0,0,0.02,0.04,0.1,0.06,0.01,0.01,0.02,0.07,0.08,0.02,0.02,0,0.01,0,0,0.02,0.01,0.01,0,0.05,0.07,0.02,0.01,0.03,0.01,0,0.23,−0.02,625,650,900,2.16,0.5,0.68,10.94,3.29,10.94,2.43, Jogging

3,15,0.1,0.1,0.1,0.1,0.09,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.09,0.1,0.09,0.1,0.1,0.1,0.09,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.08,0.12,0,8.92,−0.21,300,1683.33,2200,3.49,3.34,1.78,4.41,4.1,4.41,10.19, Upstairs

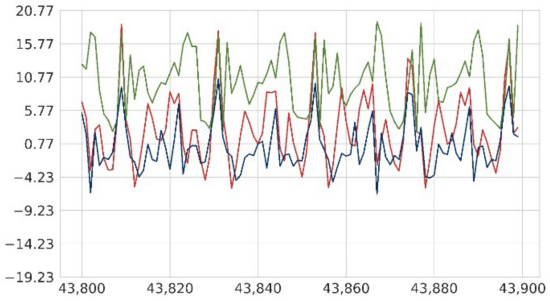
2,13,0.1,0.11,0.1,0.14,0.1,0.1,0.1,0.08,0.1,0,0.1,0.09,0.1,0.1,0.12,0.13,0.09,0.12,0.09,0,0.09,0.13,0.09,0.1,0.11,0.1,0.1,0.12,0.1,0,0,9.56,2.02,1000,850,3450,1.27,1.98,1.55,1.62,2.75,1.62,10.1, Downstairs

428,27,0.21,0.23,0.2,0,0.12,0.07,0,0.06,0.02,0.11,0.1,0.11,0,0,0.21,0,0,0.19,0.14,0.26,0.07,0.03,0,0.02,0.04,0,0.04,0.11,0.17,0.52,0,9.3,1.23,510.71,203.1,275,3.12,0.09,0.1,3.13,0.21,3.13,9.91, Sitting

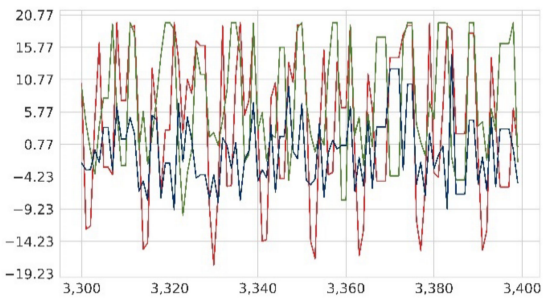
432,35,0.2,0.42,0.12,0.1,0,0.05,0,0.03,0.01,0.05,0.05,0.07,0,0.08,0,0,0.29,0,0,0.47,0.09,0.12,0.25,0,0,0.25,0,0.15,0,0.16,0,9.36,2.82,300,260.81,285.29,2.39,0.05,0.05,2.4,0.2,2.4,10.07, Standing

Figure 2. An example of sensor data, which cannot be interpretable by humans.

To provide human-level explainability for smart sensor data, we propose an approach that visualizes the data with charts, as can be seen in Figure 3. Generating signal images makes data understandable for humans. Each chart can be easily associated with activities by humans. For example, Figure 3b corresponds to the "jogging" activity since both the amplitude and frequency of the signal are high as a result of the high velocity and displacement of the person on the ground.



(**a**) Walking



(**b**) Jogging

Figure 3. *Cont.*

(**c**) Sitting (**d**) Standing

(**e**) Upstairs (**f**) Downstairs

Figure 3. Explainable and understandable sensor data for each human activity.

Figure 3 shows sample signal images that include the x, y, and z axes values of an accelerometer sensor (Ax (red), Ay (green), and Az (blue)) over time for each activity. The activity images can be easily understandable, interpretable, and explainable by humans since each one has different characteristics. For example, walking is more periodic than standing. Jogging requires higher effort and power than walking since it requires more intense muscle contractions. It can also be seen that very low x–y–z-values are observed for sitting. Moreover, a human may require different relaxation times when moving upstairs, and sometimes, he/she stops to have a rest, moves slowly, and spends more time because he/she feels tired. The acceleration curve of moving downstairs is similar to moving upstairs, but the cycle of motion is shorter. These observations are also consistent with the energy harvesting from human activities. Sitting and standing activities are nonperiodic since they are stable and straightforward postures, while other activities are mainly repetitive and quasi-periodic. The walking activity is symmetric since the human legs are coordinated and moving together at the same frequency and the phase-plane cycles of the two legs are the same. Signal data show a symmetric pattern for some human activities. For jogging activity, the signal amplitude is symmetric with the zero axis. By using the differences in the figures, a CNN algorithm can successfully distinguish the activities.

In this study, we do not extract features from raw sensor data as the traditional HAR studies do. Instead, we assemble x-axis, y-axis, and z-axis accelerometer signal sequences into an image to enable CNNs to learn the optimal features automatically from the signal image for the human activity classification task. In other words, we propose an approach, called HARSI, which transforms numerical sensor data into image format data and builds a CNN model that enables human activity recognition on these signal images. The main purpose of our study is to provide an interpretable and robust approach to indoor HAR problems.

Figure 4 shows a general overview of the proposed HARSI approach. The approach mainly includes the following stages: data collection, data transformation, training, testing, and classification. (i) The *data collection stage* comprises obtaining raw signal values via an accelerometer sensor available in an IoT environment when performing human actions.

After that, the collected raw data are transferred to a server through WiFi communication technology. (ii) In the *data transformation stage*, the time-series signal data are divided into fixed-size segments, called windows, by using a sliding-window method. After that, an image is generated for each window by drawing lines on sample values. Here, each signal image corresponds to a single activity such as standing, sitting, or walking. (iii) In the *training stage*, each signal image is fed to CNN as an input vector, and then CNN learns different features of the image through different layers. (iv) In the *classification stage*, the CNN model gives insight according to the features of the signal image. In other words, the CNN model makes a prediction for a given input image according to the class probabilities of activities, such as standing, sitting, jogging, walking, moving downstairs, and moving upstairs. (v) In the *testing stage*, the performance of the CNN model is assessed by using a test set to evaluate how well it recognizes human activities. If the prediction accuracy of the CNN model is at an acceptable level (i.e., >80%), it can be further used to recognize real-time human activities. Afterward, the final prediction can be considered in a decision support stage to provide guidance to the decision-maker. Since the indoor environment of HAR systems is dynamic and ever-evolving, it is required to update the model periodically by following the same stages to achieve high accuracy consistently.

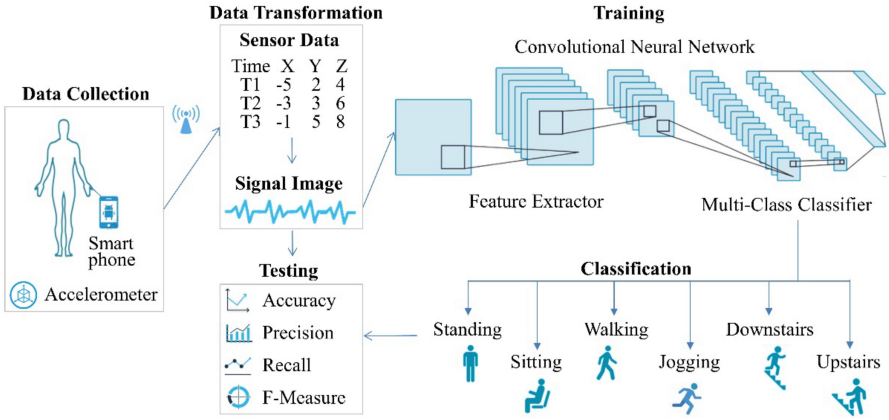


Figure 4. The general workflow of the proposed HARSI approach.

As seen in Figure 4, the CNN contains an input layer, multiple hidden layers, and an output layer. Signal image data are processed layer-by-layer, where the output of each layer becomes the input for the next layer. Each layer contains multiple units, which are denoted by U_i^l to indicate the ith unit in layer l. The hidden layers are composed of convolutional, pooling, and fully connected layers.

Convolutional layer (CL): These are used as feature extractors to automatically obtain high-level representations of input images. Formally, a feature map is extracted using a convolution procedure, as follows:

$$F_j^{l+1} = \alpha\left(\sum_{i=1}^{|F^l|} K_{j,i}^l F_j^l + b_j^l\right) \tag{1}$$

where F_j^l denotes the jth feature map in layer l, $\left| F^l \right|$ is the number of feature maps in layer l, $\alpha()$ is an activation function, b_j^l is a bias vector, and $K_{j,i}^l$ represents the kernel applied on feature map i in layer l to obtain jth feature map in layer $(l+1)$.

Pooling layer (PL): Pooling layers are used to reduce dimensionality, as well as the number of parameters. A PL is usually inserted between successive CLs in a CNN architecture. Formally, max pooling is given by:

$$v_i^{l+1} = \max_{1 \le k \le r}\left(v_{i+k}^l\right) \tag{2}$$

where r is the pooling size and v_i^l refers to the value of the ith unit in layer l.

Fully connected layer (FCL): After multiple CLs and PLs, the classification process is handled in a fully connected layer, which produces an output vector, as given in Equation (3).

$$\mathbf{z}^{l+1} = v_i^l \mathbf{w} \tag{3}$$

where $\mathbf{w}$ represents a weight vector and vector $\mathbf{z}$ includes nonnormalized log probabilities. The output of the FCL is fed into a softmax classifier, which predicts the activity label as follows:

$$Softmax(\mathbf{z}_i) = \mathrm{P}(o = a \mid \mathbf{z}_i) = \frac{e^{z_i}}{\sum_{j=1}^{m} e^{z_j}} \tag{4}$$

Where a is an activity label, o denotes the output of the classification model, $\mathbf{z}_j$ represents the jth element of log probability vector $\mathbf{z}$, and m is the number of class labels. The predicted activity label (al) for a given image is assigned to the one with the highest probability, as given in Equation (5).

$$al \leftarrow \mathrm{argmax}_{a=1}^{m} \mathrm{P}(o = a \mid image) \tag{5}$$

3.3. Formal Definition

Let the raw dataset D be a set of instances collected by an accelerometer sensor. Each instance in dataset D includes a set of pairs of x–y–z axis values and the corresponding activity label, which is denoted by $D = \{(x_1, y_1, z_1, a_1), (x_2, y_2, z_2, a_2), \ldots ., (x_n, y_n, z_n, a_n)\}$, where n is the number of instances. In other words, a_i is the activity (class label) belonging to the axes values of the sensor (x_i, y_i, z_i). The output attribute $O = \{a_1, a_2, \ldots , a_n\}$ has m different human activities, which is denoted by $a_i \in \{A_1, A_2, \ldots ., A_m\}$ for $i = 1,2, \ldots ,n$. For example, in a four-activity classification (sitting, standing, stairs, walking), the class labels of the instances are A_1 = sitting, A_2 = standing, A_3 = stairs, and A_4 = walking.

In the proposed HARSI approach, the raw sensor data D are transformed into signal images using a sliding window method. In this process, a large time-series dataset is split into fixed-sized chunks, referred to as windows, denoted as $W = (w_1, w_2, \ldots , w_{n/\mathrm{q}})$, where q is the window size.

Definition 1 (window). *A window is defined as a set of consecutive sensor measurements obtained within a time interval such that $w = \{s_r, s_{r+1}, \ldots , s_{r+q-1}\}$, where q refers to the window size and r corresponds to an arbitrary position, such as $1 \leq r \leq n\text{-}q+1$, where n is the data size.*

After generating windows, a single activity label $a_i \in \{A_1, A_2, \ldots ., A_m\}$ is assigned to each window $\{(w_1, a_i), (w_2, a_i), \ldots , (w_{n/\mathrm{q}}, a_i)\}$ such that all the samples within the window belong to the respective class. After that, an image is generated for each window by drawing lines on sample values. Each signal image is labeled with the corresponding activity a.

Definition 2 (activity). *An activity is a human movement characterized by a body action or posture, e.g., walking. An activity label $a_i \epsilon \{A_1, A_2, \ldots ., A_m\}$ is associated with an image that is generated from a window with fixed length (q) by segmenting the raw sensor data D, where m is the number of potential activities to be recognized.*

The problem studied in this work is to detect a corresponding activity implicated in a certain temporal sequence based on the classification. In other words, the aim of HAR is to build a model $M(image, \bullet)$ to infer the correct activity label for a given image, where $\bullet$ denotes all the parameters to be learned during the training process.

Definition 3 (activity recognition task). *Given a set of training images with their corresponding activity labels and a query image, the aim is to find a mapping functionf: image $\rightarrow$ activity that correctly infers the human behavior for the query image. The predicted activity label should be as*

similar as possible to the actual class label. Therefore, the task is to build a classification model M by minimizing total loss L(M).

It should be noted that a sliding-window method can be performed in either an overlapping or nonoverlapping way. A nonoverlapping method indicates that the values in one image do not intersect with the values of the other successive image, i.e., $w_1 \cap w_2 = \emptyset$. On the other hand, an overlapping method is defined by a particular percentage, which indicates how many samples from the previous image are repeated in the current image, i.e., $w_1 \cap w_2 \neq \emptyset$. In this study, we prefer to use the nonoverlapping image technique to prevent information duplication.

A crucial factor in a sliding window method is to select a suitable window size to achieve high recognition accuracy since the ideal window size varies in accordance with the characteristics of signals being processed. In general, a small window size can be useful to detect faster-changing activities better; however, using short windows may lead to misclassification because some vital information about a complex activity may not be captured by multiple windows. On the other hand, large windows can detect complex activities and semi-complex activities. However, a large window generates a signal image that belongs to more than one human activity, and this leads to a decrease in recognition accuracy. Considering this tradeoff, researchers usually determine the optimal window size by trying empirical values and assessing classification accuracy. In this study, the size of each window was set to 100 samples since the dataset was collected at a rate of 20 samples per second, and it is a sufficient sampling value to make a reasonable prediction for human activity.

Algorithm 1 shows the pseudocode of the proposed HARSI method. In the algorithm, first, a sliding window technique is used to split accelerometer sensor data (D) into windows with size q. In other words, it segments data streams into windows of equal length. For a dataset with n samples, the algorithm generates n/q windows, where each one ranges between $I \times q$ and $I \times q + q - 1$ for $i = 0, 1, \ldots, n/q$. After that, an image is generated for each window by drawing lines on sample values. Here, a small window is shifted along the continuous data stream, converting contiguous portions of sensor readings into images. Each image is labeled with the corresponding activity. A CNN classifier M is then trained on the signal image dataset. In the final step, the activity (class label) of each unseen image in the test set T is predicted by using the classifier.

Algorithm 1. Human Activity Recognition on Signal Images (HARSI)

Inputs: $D = \{(x_1, y_1, z_1, a_1), (x_2, y_2, z_2, a_2), \ldots ., (x_n, y_n, z_n, a_n)\}$
q: window size
T: Test set
Output: $O = \{o_1, o_2, \ldots, o_t\}$ a set of outputs for test images
Begin:
for $i = 0$ **to** n/q **do**
$W = \emptyset$
for $j = i * q$ **to** $i * q + q - 1$ **do**
$W = W \cup (x_j, y_j, z_j,)$
activity = a_j
end for
image = ConvertToImage(W)
$I = I \cup$ <image, activity>
end for
M = CNN(I)
foreach image i **in** T **do**
o = Classify(M, i)
$O = O \cup o$
end foreach
Return O
End

4. Experimental Studies

This section presents a detailed study that was carried out to evaluate the performance of the proposed HARSI method. The effectiveness of the method was demonstrated on a real-world dataset by using different CNN architectures, including AlexNet, ResNet, SqueezeNet, DenseNet, and VGG. The parameter settings and the number of parameters for

each CNN architecture are given in Table 2. The structures of CNNs are different from each other in several aspects, such as the number of parameters and the number of layers. For instance, ResNet34 consists of a residual network with 34 layers. Compared to ResNet50, the DenseNet121 has more layers, whilst VGG19 has fewer layers. Some parameter settings are common in all models, i.e., rectified linear unit (ReLU) was used as activation function and the output probability was calculated by softmax. As the nature of the models used, batch layers predate each ReLU. In all models, Adam and cross entropy were used as the optimizer and loss function, respectively. These techniques have lately gained popularity and performed promising outcomes for deep learning applications.

Table 2. Parameter settings and the number of parameters.

Model	Learning Rate	Activation Function	Optimizer	Loss Function	Total Parameters	Total Trainable Parameters	Total Nontrainable Parameters
HARSI-ResNet34	2×10^{-3}	ReLU (Rectified Linear Unit)	Adam (Adaptive Moment Estimation)	Cross Entropy	21,815,104	547,456	21,267,648
HARSI-ResNet50	6×10^{-4}				25,617,472	2,162,560	23,454,912
HARSI-ResNet101	1×10^{-3}				44,609,600	2,214,784	42,394,816
HARSI-AlexNet	2×10^{-3}				2,736,960	267,264	2,469,696
HARSI-DenseNet121	4×10^{-4}				8,010,624	1,140,416	6,870,208
HARSI-SqueezeNet_v1.0	2×10^{-3}				1,265,856	530,432	735,424
HARSI-SqueezeNet_v1.1	1×10^{-3}				1,252,928	530,432	722,496
HARSI-VGG16	1×10^{-3}				15,253,568	538,880	14,714,688
HARSI-VGG19	2×10^{-3}				20,565,824	541,440	20,024,384

In this study, optimal learning rate parameters were determined for each model separately to speed up the training process, adapt itself to the problem, and strengthen the generalization ability of the classifiers. While a low learning rate slows the convergence of the training process, a high learning rate can cause an unpleasant divergence in performance. Therefore, a suitable learning rate is vital for obtaining a satisfactory performance; however, finding an appropriate learning rate is both laborious and hard to decide. To solve this problem, we used the *lr_find()* method in *Fast.AI*, which is a deep learning library built on top of PyTorch. This method works on the principle of using a very low learning rate initially to train a minibatch and calculate the loss. In the next step, the method trains the next minibatch with a small-scale higher learning rate than the previous one until it finds a learning rate where the model diverges. The optimal learning rate values determined for each model are listed in Table 2.

One of the main differences between the CNN models is the number of parameters, which can reflect the computational complexity of the model. It may be noted that as the number of total parameters increases, the time required for training usually increases. There are two categories of parameters: one is *trainable parameters* (i.e., weights of connections between layers) that are continuously updated to reduce the loss, and the other one is *nontrainable parameters* (i.e., biases) that are not optimized during the training process. For instance, in VGG19 architecture, the number of trainable parameters is 541,440, while the number of nontrainable parameters is approximately 20 million. As seen in Table 2, SqueezeNet has the smallest total number of parameters, whilst ResNet101 has the largest. The architectures have approximately 2, 15, and 25 million parameters for AlexNet, VGG16, and ResNet50, respectively. These values may be associated with computational complexity, where the higher the number of parameters, the greater the computational load during the training process.

The method was implemented in Python by using the PyTorch framework and various libraries such as Fastai, NumPy, Pandas, Scikit-Learn, Matplotlib, and Seaborn. In this study, the CNN models were trained on a computer equipped with an Nvidia GTX 1060 graphics card using the Cuda toolkit in order to make use of the GPU computational capability and reduce implementation time through a rapid and simple design.

To demonstrate the superiority of the proposed approach (HARSI) over the previous approaches, we compared it with the classical machine learning methods (SVM, DT, NB, KNN, MLP, AdaBoost, and RF) [12,33–50] and also compared it with the state-of-the-art methods [5,12,13,24,26,34,51–77] on the same dataset.

In this study, we split the dataset into two subsets: 80% of the data were used for training and the remaining 20% were used for testing. This standard split approach was chosen since it is common in the previous studies [5,71,72,74] that used the same dataset. In order to provide comparability with the literature, the same split approach was preferred. In addition, the training part of the data was divided into training and validation sets as 80% and 20%, respectively. The test set contains 100 images from each category; therefore, it includes 600 images in total. Four different metrics were used to evaluate the performance of each CNN architecture: accuracy, recall, precision, and f-measure. Accuracy is the fraction of correct predictions of the model to total prediction. Equation (6) shows how the accuracy rate is calculated.

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \tag{6}$$

where TP is true positive, FP is false positive, TN is true negative, and FN is false negative. *Precision* describes how precise the model is out of the samples predicted positive, and how many of them are actually positive. *Recall* indicates how many of the actual positives the model captures through classifying them as positive. *F-measure* offers a single score that balances both the concerns of recall and precision values. Equations (7)–(9) show the calculations of precision, recall, and f-measure values, respectively.

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

$$Fmeasure = \frac{2 \times precision \times recall}{precision + recall} \tag{9}$$

4.1. Dataset Description

In order to show the effectiveness of the proposed approach, a publicly available dataset, named WISDM (Wireless Sensor Data Mining) dataset [33], was used in the experiments. The dataset is available at the website https://www.cis.fordham.edu/wisdm/dataset.php (accessed on 25 September 2022). It was released by the Laboratory at Fordham University in the United States. This dataset is one of the important and popular large-scale benchmark datasets in the field of HAR. It has been used in many studies [5,12,13,24,26,33–77], so it is suitable for making comparisons with previous works. The dataset is appropriate for detecting symmetric activities. Before collecting the data, the researchers obtained approval from the University Board since it involved research on human subjects and involved some risks, i.e., the subject could fall down while jogging. The data collection process was fully monitored and guided by researchers in the laboratory environment to ensure the quality of the data. The dataset contains routine motion patterns with a significant number of processable movement samples. The dataset has 1,098,207 samples that were collected from 36 different participants while performing six activities. Therefore, it could possibly be used to analyze the movement behaviors of different persons. The percentages of each activity in the dataset are as follows: jogging 31.2%, moving downstairs 9.1%, walking 38.6%, moving upstairs 11.2%, standing 4.4%, and sitting 5.5%. There is no missing value in the dataset. While collecting data, the participants are requested to carry an accelerometer sensor in their front pockets. They were asked to jog, walk, descend stairs, ascend stairs, stand, and sit for specific periods of time. With this experiment setup, accelerometer data were retrieved at every 50 ms, which means

20 samples per second, while participants were performing activities. The raw dataset consists of *x*-, *y*-, and *z*-axis values obtained by an accelerometer sensor embedded in a smart device.

In this study, image representations were first generated from the raw x, y, and z values in the accelerometer sensor data. In other words, we converted the time-series data into signal images by drawing three lines on sample x, y, and z values. Here, a small window was shifted along the continuous data stream converting contiguous portions of sensor readings into images. In the generated images, x, y, and z drawing lines are represented with different colors; red, green, and blue, respectively. While generating charts from sensor data values, attention was paid to choosing a fixed range for all the graphs. Accordingly, the max and min values were found by searching the x, y, and z axes values in the dataset. The vertical range of the graph in each image approximately lies between [–20, 20]. The horizontal range of the graph was set to 100 samples since the dataset was collected at a rate of 20 samples per second, and therefore it is a sufficient range to make a reasonable prediction for activity. Each image is labeled with the corresponding activity, such as standing, sitting, or walking. To create a balanced dataset, 400 images were generated for each activity; therefore, in total, 2400 images were generated. Figure 3 shows sample signal images for each activity.

By transforming numerical sensor data into image data, we aim to improve both explainability and recognition accuracy. The generated images provide human-level explainability for smart sensor data. Since each image reflects the properties of activities, they can be easily interpretable by humans. Rather than solving a time-series data classification problem, we define the HAR problem as an image classification problem. In this way, we provide an interpretable and robust approach to HAR problems.

4.2. Comparison of Different CNN Architectures

On the dataset described in the previous section, the effectiveness of the proposed approach (HARSI) was demonstrated by using different CNN architectures, including Alex Network (AlexNet), Residual Network (ResNet), Visual Geometry Group (VGG) Network, SqueezeNet, and Dense Convolutional Network (DenseNet). These CNN architectures were selected because of their popularity, high robustness, proven efficiency, and ability in image classification. They automatically extract features of images that are useful in the identification of human activities. They use a gradient descent algorithm to optimize the CNN parameters.

Table 3 shows the performance of the proposed HARSI method on different CNN architectures for the same dataset. Based on the accuracy rates, it is possible to say that all the CNN models have good classification ability. However, VGG19 is the most successful model among them with a 98% of success rate. Following this, ResNet34 has also a high accuracy rate (97.33%) in distinguishing human activities. This success is the result of the strengths of CNNs in classifying images. CNNs are capable of extracting key features directly and effectively from images, learning useful information layer-by-layer, and successfully classifying them into different classes.

In addition to accuracy, we also evaluated the performance of the proposed HARSI approach on different CNN architectures in terms of recall, precision, and f-measure metrics. The values of these metrics range between 0 and 1, where 1 is the best value. As can be seen in Table 3, the recall value obtained by the VGG19 model is closer to 1 than the others. This means that the VGG19 model often tends to give better predictions than the rest. As can be observed, the VGG16 model also outperformed the others in terms of precision and f-measure.

Table 3. The performance of the proposed HARSI method on different CNN architectures. The best values are highlighted in bold.

Model	Accuracy (%)	Precision	Recall	F-Measure
HARSI-ResNet34	97.33	0.97433	0.97333	0.97326
HARSI-ResNet50	96.00	0.96054	0.96000	0.96006
HARSI-ResNet101	97.17	0.97194	0.97166	0.97161
HARSI-AlexNet	89.17	0.89018	0.89166	0.89077
HARSI-DenseNet121	96.67	0.96695	0.96666	0.96671
HARSI-SqueezeNet_v1.0	89.67	0.90025	0.89666	0.89725
HARSI-SqueezeNet_v1.1	93.00	0.92915	0.93000	0.92937
HARSI-VGG16	96.83	0.96871	0.96833	0.96826
HARSI-VGG19	**98.00**	**0.97999**	**0.98000**	**0.97999**

In Figure 5, the loss values in both the training and validation processes are shown. While the vertical axis indicates the loss value, the horizontal axis represents the number of batches processed. In initial batches, the training loss is higher than the validation loss. As can be seen, both the training and validation losses reduce with the increase of the batches. The training loss and validation loss converged after approximately 200 batches were processed. When the minimum validation loss was obtained, the training process was stopped to avoid overfitting.

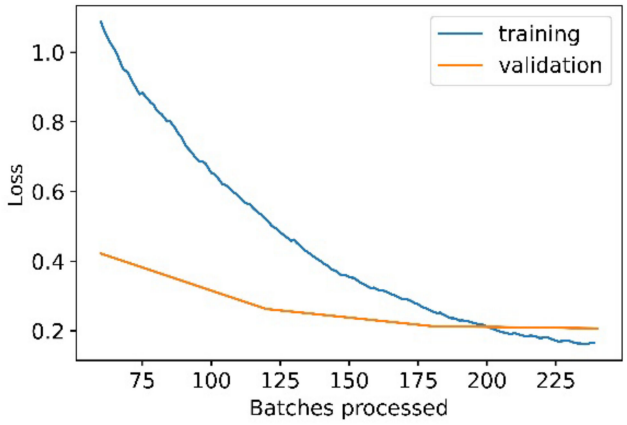


Figure 5. Loss values in the training and validation processes.

Figure 6 presents the confusion matrix to show the predictive performance of the proposed HARSI method on each human activity separately. The rows in the confusion matrix represent the predicted activity labels, whereas the columns represent the actual activity labels. Each cell in the matrix is a percentage value, indicating what percent of the data belongs to the column class but is incorrectly classified as the row class. All correctly classified samples are positioned on the diagonal of a confusion matrix, so, its diagonal should contain the highest values possible, and all the other elements should be close to zero. According to the matrix given in Figure 6, it is possible to say that the model usually had no difficulty in distinguishing human activities. For example, 98 out of 100 *walking* activities were predicted correctly; however, only two of the *walking* activities were misclassified by the classifier. Although each activity was recognized with a high accuracy rate, *downstairs* and *upstairs* activities were slightly confused with each other since they are similar activities. The algorithm produced an equal accuracy value (95%) for moving upstairs and moving downstairs activities since they have similar characteristics to others. It can be concluded from the confusion matrix that the best accuracies were achieved in the *sitting*, *standing*, and *jogging* activities.

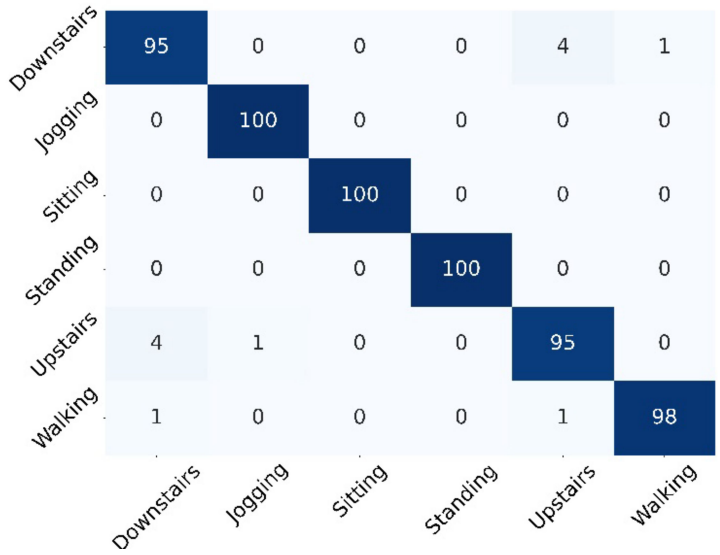


Figure 6. Confusion matrix obtained by the proposed HARSI method.

Figure 7 shows the execution times of the proposed HARSI method on different CNN architectures in minutes. Although all the times are close to each other, AlexNet and SqueezeNet are the fastest ones among their counterparts. They are followed by the ResNet34 model (1.11 min). The VGG models are also efficient in terms of training time (1.19 and 1.24 min). The DenseNet model may take a longer time (1.28 min), especially handling large image datasets. This is probably because of the higher number of layers in its architecture. Similarly, the required time for training ResNet101 is higher than others since it has a higher number of parameters to be assessed. The size and resolution of the images are also factors that affect computation time. When the sizes of the images are reduced by the resizing process, the time required for analyzing them decreases, and therefore, the performance of the HAR system is positively affected.

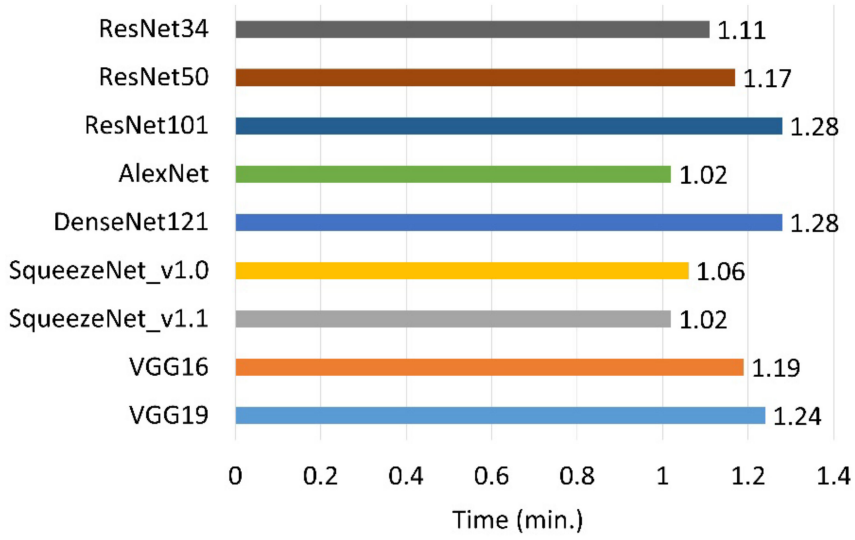


Figure 7. The execution time of the proposed method on different CNNs.

4.3. Comparison with the Classical Machine Learning Methods

In order to show the superiority of our method, we compared it with the classical machine learning methods such as multilayer perceptron (MLP), support vector machines (SVM), decision tree (DT), naive Bayes (NB), logistic regression (LR), k-nearest neighbors (KNN), AdaBoost, and random forest (RF). In order to make a plausible comparison, the

most important factor is to use the same data. Therefore, the results obtained by the classical machine learning methods on the same dataset [33] were used in the comparison. Table 4 lists the related studies with their methods and the corresponding accuracy rates. It can be seen from the table that the proposed HARSI method outperformed the other methods [12,33–50] with a 13.72% improvement on average. Employing HARSI achieved higher accuracy (98%) than the traditional machine learning models on the same dataset.

Table 4. Comparison of the proposed HARSI method against the classical machine learning methods on the same dataset.

Ref.	Year	Method	Accuracy (%)
[12]	2022	Support Vector Machines	87.40
		Random Forest	86.10
[34]	2021	Decision Tree	82.00
		Logistic Regression	68.00
		Multilayer Perceptron	80.00
		Neural Networks	94.00
[35]	2021	Decision Tree	89.76
		Linear Discriminant Analysis	86.64
		Gradients Boosting	89.65
		K-Nearest Neighbors	92.54
		Bagging	92.48
		Random Forest	92.71
		Linear Kernel SVM	78.55
		RBF Kernel SVM	89.07
		Polynomial Kernel SVM	92.48
[36]	2021	Random Forest	79.38
		K-Nearest Neighbors	75.04
		Decision Tree	77.60
		Gradient Boosting	74.80
[37]	2020	Random Forest	83.35
		Neural Networks	77.02
		Decision Tree (J48)	75.96
		Reduced-Error Pruning (REP) Tree	74.64
		K-Nearest Neighbors	72.08
		KStar	71.84
		Naive Bayes	63.89
		Support Vector Machines	55.45
[38]	2020	Random Forest	92.78
		Support Vector Machines	91.39
[39]	2020	Neural Networks	89.10
		Decision Tree	87.45
		Support Vector Machines	95.13
		Linear Support Vector Classifier	86.20
		Logistic Regression	81.10
		Random Forest	82.10
[40]	2020	Support Vector Machines	82.00
[41]	2020	Multilayer Perceptron	86.95
[42]	2019	K-Nearest Neighbors	92.00
		Support Vector Machines	93.50
		Bagging	93.80
[43]	2018	Random Forest	82.66
		K-Nearest Neighbors	66.19
[44]	2018	Support Vector Machines	82.27

Table 4. *Cont.*

Ref.	Year	Method	Accuracy (%)
[45]	2018	Naive Bayes	80.12
		Decision Tree	81.02
		K-Nearest Neighbors	77.58
		Support Vector Machines	80.93
[46]	2017	Naive Bayes Tree	87.70
		Multilayer Perceptron	77.52
		DT + LR + MLP	91.62
		NB Tree + MLP	96.35
[47]	2016	AdaBoost + J48	97.83
		AdaBoost + REP Tree	97.33
		AdaBoost + Random Tree	95.69
		AdaBoost + Random Forest	94.44
		AdaBoost + Hoeffding Tree	87.84
		AdaBoost + Decision Stump	57.31
[48]	2015	Decision Tree (J48)	86.08
		Logistic Regression	77.52
		Multilayer Perceptron	88.81
		J48 + LR + MLP	91.62
[49]	2015	Decision Tree (J48)	92.40
		Logistic Regression	84.30
		Multilayer Perceptron	91.70
		J48 + LR + MLP	93.00
[50]	2015	Neural Networks with Dropout	85.36
		Random Forest	83.46
[33]	2010	Logistic Regression	78.10
		Decision Tree (J48)	85.10
		Multilayer Perceptron	91.70
Average			84.28
Our Approach		**Human Activity Recognition on Signal Images (HARSI)**	**98.00**

4.4. Comparison with the State-of-the-Art Methods

This section presents comparative results which highlight the performance of the proposed method over the state-of-the-art methods in the literature. Table 5 shows the performance improvement of our method over the state-of-the-art methods [5,12,13,24,26,34,51–77]. The results were taken directly from the referenced studies since the researchers used the same dataset [33] as our study. It can be seen from the table that the proposed HARSI method outperformed the other methods with a 7.06% improvement on average.

Table 5. Comparison of the proposed HARSI method against the state-of-the-art methods on the same dataset.

Ref.	Year	Method	Accuracy(%)
[12]	2022	CNN—Transfer Learning	90.40
		Convolutional Neural Networks	88.20
[5]	2021	CNN + Long Short-Term Memory	97.76
		Long Short-Term Memory	96.61
		Convolutional Neural Networks	94.51
[13]	2021	Deep Neural Networks	93.00
[24]	2021	Vanilla RNN + LSTM + GRU	97.13

Table 5. *Cont.*

Ref.	Year	Method	Accuracy(%)
[26]	2021	CNN + Random Forest	97.77
		Deep Neural Networks	74.00
		Deep Neural Networks + LSTM	81.00
		Deep Neural Networks + Gated Recurrent Unit (GRU)	80.00
		Convolutional Neural Networks	88.00
		Convolutional Neural Networks + LSTM	94.00
		Convolutional Neural Networks + GRU	82.00
[34]	2021	Deep Neural Networks	95.00
[51]	2021	Residual Network	95.66
		Convolutional Neural Networks	92.19
[52]	2021	Deep Convolutional Neural Networks	91.25
[53]	2021	1D Convolutional Neural Networks	91.12
		1D CNN + Fuzzy Neural Network	92.96
[54]	2021	Ensemble of Autoencoders (EAE)	82.00
		KNN + Very Fast Decision Tree + Naive Bayes (EkVN)	73.00
[55]	2021	NOvelty discrete data stream for Human Activity Recognition (NOHAR)	93.00
[56]	2021	Deep Convolutional Neural Networks Ensemble	89.01
[57]	2021	Convolutional AutoEncoder (CAE)	95.60
[58]	2021	Convolutional Neural Networks	95.00
		Long Short-Term Memory	97.50
[59]	2020	Convolutional Neural Networks	93.25
[60]	2020	Deep Convolutional Neural Networks	94.18
		Region-based CNN	93.68
[61]	2020	CNN—DenseNet	94.65
[62]	2020	Bidirectional Long Short-Term Memory	94.10
[38]	2020	Genetic algorithm-based classifier	95.37
[39]	2020	Convolutional Neural Networks	83.98
		Long Short-Term Memory	95.45
[63]	2020	LSTM–Convolutional Neural Networks	95.75
[64]	2020	Lightweight Recurrent Neural Network—LSTM	95.78
[65]	2020	Multihead Convolutional Attention	95.40
[66]	2020	Two-Stage End-to-end CNN with data augmentation (TSE + CNN + Aug)	95.70
[41]	2020	Gramian Angular Field + Multidilated Kernel Residual Network	96.83
		Long Short-Term Memory	87.53
		1D Convolutional Neural Network	93.66
[6]	2020	EnsemConvNet (CNN-Net + Encoded-Net + CNN-LSTM)	97.20
[68]	2020	Convolutional Neural Networks	97.51
[69]	2020	Convolutional Neural Networks	94.11
		Residual Network	95.72
		Residual Network of Residual Network	96.73
[40]	2020	Convolutional Neural Networks	81.70
		Recurrent Convolutional Network (RCN)	94.00
		Recurrent Convolutional Network + SVM	91.50

Table 5. *Cont.*

Ref.	Year	Method	Accuracy(%)
[70]	2019	U-Net	96.40
		Mask Region-based CNN (R-CNN)	86.20
		SegNet: A Deep Convolutional Encoder-Decoder Arch.	95.70
		Full Convolutional Network (FCN)	87.90
		Deep Convolutional and LSTM	94.80
		Long Short-Term Memory	93.80
		Convolutional Neural Networks	94.10
[71]	2019	LSTM–Recurrent Neural Networks	93.81
[42]	2019	Supervised Regularization-based Robust Subspace (SRRS)	93.50
		Robust Principal Component Analysis	85.70
		Latent Low-Rank Representation (LLRR)	91.90
		Joint Embedding Learning and Sparse Regression (JELSR)	73.40
		Principal Component Analysis (PCA)	92.30
		Linear Discriminant Analysis (LDA)	71.50
[72]	2019	Long Short-Term Memory	97.00
[44]	2018	Convolutional Neural Networks	91.97
[45]	2018	Multivariate Bag-Of-SFA-Symbols	83.35
[73]	2018	Deep Autoencoder-Set Network	94.90
[74]	2018	Long Short-Term Memory	97.00
[43]	2018	Convolutional Neural Networks	93.32
[75]	2017	Impersonal Smartphone-based Activity Recognition (ISAR)	75.21
[76]	2016	Long Short-Term Memory	92.10
[77]	2015	STream learning for mobile Activity Recognition (STAR)	71.20
		Average	90.94
OurApproach		**Human Activity Recognition on Signal Images (HARSI)**	**98.00**

4.5. Discussion

The main debates in the field of HAR and our solutions can be summarized as follows.

- In HAR, the ideal input data format is still a subject of much debate and there are various ongoing works for improving the accuracy of the models. Traditional HAR has been defined as a time-series data classification problem and requires feature extraction. In contrast, we transfer time-series data into signal images that reflect the properties of human activities. It avoids the need to perform an explicit feature generation and selection stage. We improved accuracy by working on signal image data, instead of numerical time-series data.
- Many applications in HAR [33–50] have used classical machine learning methods such as DT, SVM, MLP, NB, LR, KNN, and RF. However, the performance of these methods is still highly debated. In this study, we take advantage of the strengths of deep learning approaches.
- Another debate is how to design CNN architecture to be able to obtain good performance. For example, the number of layers and parameter settings are still subjects of much debate. In this study, we compared nine different CNN architectures to determine the best suitable one.
- In the activity recognition community, there is an open debate on providing explainability in the HAR systems. The main problem is how to increase the transparency

and interpretability of the models. In this study, to increase human-level explainability, we visualize the data with charts since generating signal images makes data understandable for humans.

- Another ongoing debate is which activities can be predicted more precisely. This study showed that the best accuracies were achieved in the sitting, standing, and jogging activities due to their diverse natures.
- The proposed HAR model can be connected to many different fields of study such as health monitoring, fitness tracking, home and work automation, and self-managing system. With the rapid technological developments in smartphones, the model can enable new opportunities for developing informative systems on a large scale to perceive and act on what users (i.e., your children, elderly mother, or sick family member) are doing. Recognizing human activities is important for the treatment of patients and can provide useful feedback to the clinicians since the activity is associated with health. For example, it can be used to monitor patients in rehabilitation since the functional status of a person is an important parameter in this area. In addition, it could be used to offer activity-aware services to smartphone users, such as movement recommendations. A number of lifestyle diseases and movement disorders are associated with inactivity; therefore, our model can be used to give information to prevent diseases. The users can participate in the tracking of their activities for the sake of health, fitness, or other purposes due to its strength in providing personalized support.

5. Conclusions and Future Works

Classical HAR has been defined as a standard data classification problem and extracts statistical features (i.e., min, max, skewness, kurtosis) from data, which cannot be readable and interpretable by humans. Transparent and explainable indoor HAR systems are required to generate human-understandable information. For this purpose, an approach, called Human Activity Recognition on Signal Images (HARSI), is proposed in this study. The proposed approach creates image representations of the time-series sensor data to improve both explainability and recognition accuracy. This is the first attempt to combine five methodologies: signal image-based indoor HAR, XAI, IoT, symmetry, and DL. It takes advantage of the strengths of CNNs in handling signal image data. In the experimental studies, we demonstrated the effectiveness of the proposed HARSI approach compared to the previous studies on a real-world dataset.

The main findings of the study can be concluded as follows:

- The proposed approach improves human-level explainability for smart sensor data by using signal images in the field of HAR.
- The proposed HARSI approach improves the recognition accuracy in the HAR problems by converting time-series data to image data.
- The experimental results showed that HARSI successfully (98%) recognized six symmetric human activities, including walking, jogging, standing, sitting, moving downstairs, and moving upstairs.
- According to the experimental results, it can be concluded that the best suitable and consistent CNN model for the WISDM dataset is VGG19. It achieved the best results on all the metrics (accuracy, precision, recall, and f-measure). Therefore, this model can be successfully used to identify human activities.
- The prediction accuracy changes according to human activities. Among the activities, sitting, standing, and jogging were correctly predicted by the proposed method. On the other hand, the model had a little difficulty in classifying downstairs and upstairs activities with an accuracy of 95% for the WISDM dataset.
- The number of layers and number of parameters of a CNN model may be associated with computational complexity, where the higher the number of layers and parameters, the greater the computational load during the training process.

- A significant improvement (13.72% on average) was achieved by the proposed HARSI model compared to the classical machine learning methods such as KNN, DT, SVM, NB, LR, MLP, AdaBoost, and RF.
- Our approach achieved higher classification accuracy than the state-of-the-art approaches. It outperformed them by 7.06% on average on the same dataset.
- The proposed HARSI approach has the potential to expand the application of machine learning in many different sectors, thanks to its advantages.

One limitation of this study is related to sensors such as signal delays, noises, damages, battery capacity, and shelf-life. However, this limitation is also valid for all other wearable sensor-based HAR applications. It can be overcome in the future with developments in sensor technology. Another limitation is that it focuses on single-person activity detection. In the future, we plan to adapt it for recognizing group activities such as handshaking and hugging.

Author Contributions: Conceptualization, K.U.B. and M.C.; methodology, K.B.; software, A.B.C.; validation, A.B.C.; formal analysis, D.B.; investigation, K.U.B., M.C. and K.B.; data curation, A.B.C.; writing—original draft preparation, D.B.; writing—review and editing, A.B.C.; visualization, A.B.C.; supervision, D.B.; funding acquisition, K.U.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The "WISDM (Wireless Sensor Data Mining)" dataset [33] is publicly available at the following website: https://www.cis.fordham.edu/wisdm/dataset.php (accessed on 20 August 2022).

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

References

1. Hassan, M.M.; Ullah, S.; Hossain, M.S.; Alelaiwi, A. An end-to-end deep learning model for human activity recognition from highly sparse body sensor data in internet of medical things environment. *J. Supercomput.* **2021**, *77*, 2237–2250. [CrossRef]
2. Kanjilal, R.; Uysal, I. The future of human activity recognition: Deep learning or feature engineering? *Neural Process. Lett.* **2021**, *53*, 561–579. [CrossRef]
3. Mekruksavanich, S.; Jitpattanakul, A. Biometric user identification based on human activity recognition using wearable sensors: An experiment using deep learning models. *Electronics* **2021**, *10*, 308. [CrossRef]
4. Mihoub, A. A deep learning-based framework for human activity recognition in smart homes. *Mob. Inf. Syst.* **2021**, *11*, 6961343. [CrossRef]
5. Mohsen, S.; Elkaseer, A.; Scholz, S.G. Industry 4.0-oriented deep learning models for human activity recognition. *IEEE Access* **2021**, *9*, 150508–150521. [CrossRef]
6. Madokoro, H.; Nix, S.; Woo, H.; Sato, K. A mini-survey and feasibility study of deep-learning-based human activity recognition from slight feature signals obtained using privacy-aware environmental sensors. *Appl. Sci.* **2021**, *11*, 11807. [CrossRef]
7. Casilari, E.; Álvarez-Marco, M.; García-Lagos, F. A study of the use of gyroscope measurements in wearable fall detection systems. *Symmetry* **2020**, *12*, 649. [CrossRef]
8. Shalaby, E.; ElShennawy, N.; Sarhan, A. Utilizing deep learning models in CSI-based human activity recognition. *Neural Comput. Appl.* **2022**, *34*, 5993–6010. [CrossRef]
9. Bijalwan, V.; Semwal, V.B.; Gupta, V. Wearable sensor-based pattern mining for human activity recognition: Deep learning approach. *Ind. Robot-Int. J. Robot Res. Appl.* **2021**, *49*, 21–33. [CrossRef]
10. Ferrari, A.; Micucci, D.; Mobilio, M.; Napoletano, P. Deep learning and model personalization in sensor-based human activity recognition. *J. Reliab. Intell. Environ.* **2022**, *2022*, 1–13. [CrossRef]
11. Manjarrés, J.; Lan, G.; Gorlatova, M.; Hassan, M.; Pardo, M. Deep learning for detecting human activities from piezoelectric-based kinetic energy signals. *IEEE Internet Things J.* **2022**, *9*, 7545–7558. [CrossRef]
12. Bhat, O.; Khan, D.A. Evaluation of deep learning model for human activity recognition. *Evol. Syst.* **2022**, *13*, 159–168. [CrossRef]
13. Bozkurt, F. A comparative study on classifying human activities using classical machine and deep learning methods. *Arab. J. Sci. Eng.* **2022**, *47*, 1507–1521. [CrossRef]
14. Khan, A.R.; Saba, T.; Khan, M.Z.; Fati, S.M.; Khan, M.U.G. Classification of human's activities from gesture recognition in live videos using deep learning. *Concurr. Computat. Pract. Exper.* **2022**, *34*, e6825. [CrossRef]

15. Khan, I.U.; Afzal, S.; Lee, J.W. Human activity recognition via hybrid deep learning based model. *Sensors* **2022**, *22*, 323. [CrossRef]
16. Tasnim, N.; Islam, M.K.; Baek, J.-H. Deep learning based human activity recognition using spatio-temporal image formation of skeleton joints. *Appl. Sci.* **2021**, *11*, 2675. [CrossRef]
17. Maitre, J.; Bouchard, K.; Gaboury, S. Alternative deep learning architectures for feature-level fusion in human activity recognition. *Mob. Netw. Appl.* **2021**, *26*, 2076–2086. [CrossRef]
18. Hwang, Y.M.; Park, S.; Lee, H.O.; Ko, S.-K.; Lee, B.-T. Deep Learning for human activity recognition based on causality feature extraction. *IEEE Access* **2021**, *9*, 112257–112275. [CrossRef]
19. Ronald, M.; Poulose, A.; Han, D.S. iSPLInception: An inception-ResNet deep learning architecture for human activity recognition. *IEEE Access* **2021**, *9*, 68985–69001. [CrossRef]
20. Pei, L.; Xia, S.; Chu, L.; Xiao, F.; Wu, Q.; Yu, W.; Qiu, R. MARS: Mixed virtual and real wearable sensors for human activity recognition with multidomain deep learning model. *IEEE Internet Things J.* **2021**, *8*, 9383–9396. [CrossRef]
21. Yen, C.-T.; Liao, J.-X.; Huang, Y.-K. Feature fusion of a deep-learning algorithm into wearable sensor devices for human activity recognition. *Sensors* **2021**, *21*, 8294. [CrossRef]
22. Irfan, S.; Anjum, N.; Masood, N.; Khattak, A.S.; Ramzan, N. A novel hybrid deep learning model for human activity recognition based on transitional activities. *Sensors* **2021**, *21*, 8227. [CrossRef]
23. Al-Wesabi, F.N.; Albraikan, A.A.; Hilal, A.M.; Al-Shargabi, A.A.; Alhazbi, S.; Duhayyim, M.A.; Rizwanullah, M.; Hamza, M.A. Design of optimal deep learning based human activity recognition on sensor enabled internet of things environment. *IEEE Access* **2021**, *9*, 143988–143996. [CrossRef]
24. Alawneh, L.; Alsarhan, T.; Al-Zinati, M.; Al-Ayyoub, M.; Jararweh, Y.; Lu, H. Enhancing human activity recognition using deep learning and time series augmented data. *J. Ambient Intell. Humaniz. Comput.* **2021**, *12*, 10565–10580. [CrossRef]
25. Elsts, A.; McConvill, R. Are microcontrollers ready for deep learning-based human activity recognition? *Electronics* **2021**, *10*, 2640. [CrossRef]
26. Ghate, V.; Hemalatha, S.C. Hybrid deep learning approaches for smartphone sensor-based human activity recognition. *Multimed. Tools Appl.* **2021**, *80*, 35585–35604. [CrossRef]
27. Thakur, D.; Biswas, S. Feature fusion using deep learning for smartphone based human activity recognition. *Int. J. Inf. Tecnol.* **2021**, *13*, 1615–1624. [CrossRef]
28. Buffelli, D.; Vandin, F. Attention-based deep learning framework for human activity recognition with user adaptation. *IEEE Sens. J.* **2021**, *21*, 13474–13483. [CrossRef]
29. Alhersh, T.; Stuckenschmidt, H.; Rehman, A.U.; Belhaouari, S.B. Learning human activity from visual data using deep learning. *IEEE Access* **2021**, *9*, 106245–106253. [CrossRef]
30. Chen, L.; Liu, X.; Peng, L.; Wu, M. Deep learning based multimodal complex human activity recognition using wearable devices. *Appl. Intell.* **2021**, *51*, 4029–4042. [CrossRef]
31. Thu, N.H.T.; Han, D.S. HiHAR: A hierarchical hybrid deep learning architecture for wearable sensor-based human activity recognition. *IEEE Access* **2021**, *9*, 145271–145281. [CrossRef]
32. Stuart, M.; Manic, M. Deep learning shared bandpass filters for resource-constrained human activity recognition. *IEEE Access* **2021**, *9*, 39089–39097. [CrossRef]
33. Kwapisz, J.R.; Weiss, G.M.; Moore, S.A. Activity recognition using cell phone accelerometers. *ACM SIGKDD Explor. Newsl.* **2010**, *12*, 74–82. [CrossRef]
34. Suwannarat, K.; Kurdthongmee, W. Optimization of deep neural network-based human activity recognition for a wearable device. *Heliyon* **2021**, *7*, e07797. [CrossRef]
35. Vijayvargiya, A.; Kumari, N.; Gupta, P.; Kumar, R. Implementation of machine learning algorithms for human activity recognition. In Proceedings of the 3rd International Conference on Signal Processing and Communication (ICPSC), Coimbatore, India, 13–14 May 2021; pp. 440–444.
36. Semwal, V.B.; Lalwani, P.; Mishra, M.K.; Bijalwan, V.; Chadha, J.S. An optimized feature selection using bio-geography optimization technique for human walking activities recognition. *Computing* **2021**, *103*, 2893–2914. [CrossRef]
37. Kee, Y.J.; Zainudin, M.S.; Idris, M.I.; Ramlee, R.H.; Kamarudin, M.R. Activity recognition on subject independent using machine learning. *Cybern. Inf. Technol.* **2020**, *20*, 64–74. [CrossRef]
38. Jalal, A.; Quaid, M.A.K.; Kim, K. A study of accelerometer and gyroscope measurements in physical life-log activities detection systems. *Sensors* **2020**, *20*, 6670. [CrossRef]
39. Khare, S.; Sarkar, S.; Totaro, M. Comparison of sensor-based datasets for human activity recognition in wearable IoT. In Proceedings of the IEEE 6th World Forum on Internet of Things (WF-IoT), New Orleans, LA, USA, 2–16 June 2020; pp. 1–6.
40. Arigbabu, O.A. Entropy decision fusion for smartphone sensor based human activity recognition. *arXiv* **2021**, arXiv:2006.00367v1.
41. Xu, H.; Li, J.; Yuan, H.; Liu, Q.; Fan, S.; Li, T.; Sun, X. Human activity recognition based on gramian angular field and deep convolutional neural network. *IEEE Access* **2022**, *8*, 199393–199405. [CrossRef]
42. Lu, W.; Fan, F.; Chu, J.; Jing, P.; Yuting, S. Wearable computing for internet of things: A discriminant approach for human activity recognition. *IEEE Internet Things J.* **2019**, *6*, 2749–2759. [CrossRef]
43. Ignatov, A. Real-time human activity recognition from accelerometer data using convolutional neural networks. *Appl. Soft. Comput.* **2018**, *62*, 915–922. [CrossRef]

44. Xu, W.; Pang, Y.; Yang, Y.; Liu, Y. Human activity recognition based on convolutional neural network. In Proceedings of the 24th International Conference on Pattern Recognition, Beijing, China, 20–24 August 2018; pp. 165–170.
45. Quispe, K.G.M.; Lima, W.S.; Batista, D.M.; Souto, E. MBOSS: A symbolic representation of human activity recognition using mobile sensors. *Sensors* **2018**, *18*, 4354. [CrossRef] [PubMed]
46. Azmi, M.S.M.; Sulaiman, M.N. Accelerator-Based human activity recognition using voting technique with NBTREE and MLP classifiers. *Int. J. Adv. Sci. Eng. Inf. Technol.* **2017**, *7*, 146–152. [CrossRef]
47. Walse, K.H.; Dharaskar, R.V.; Thakare, V.M. A study of human activity recognition using adaboost classifiers on WISDM dataset. *Inst. Integr. Omics Appl. Biotechnol. J.* **2016**, *7*, 68–76.
48. Catal, C.; Tufekci, S.; Pirmit, E.; Kocabag, G. On the use of ensemble of classifiers for accelerometer-based activity recognition. *Appl. Soft. Comput.* **2015**, *37*, 1018–1022. [CrossRef]
49. Zainudin, M.S.; Sulaiman, M.N.; Mustapha, N.; Perumal, T. Activity recognition based on accelerometer sensor using combinational classifiers. In Proceedings of the IEEE Conference on Open Systems (ICOS), Melaka, Malaysia, 24–26 August 2015; pp. 68–73.
50. Kolosnjaji, B.; Eckert, C. Neural network-based user-independent physical activity recognition for mobile devices. In *Lecture Notes in Computer Science*; Jackowski, K., Burduk, R., Walkowiak, K., Wozniak, M., Yin, H., Eds.; Springer: Cham, Switzerland, 2015; pp. 378–386.
51. Zhang, J.; Qiao, S.; Lin, Z.; Zhou, Y. Human activity recognition based on residual network. In Proceedings of the 8th Annual International Conference on Geo-Spatial Knowledge and Intelligence, Xian, China, 18–19 December 2020; pp. 1–6.
52. Lin, S.B.; Wang, K.; Wang, Y.; Zhou, D.X. Universal consistency of deep convolutional neural networks. *arXiv* **2021**, arXiv:2106.12498. [CrossRef]
53. Zihao, Z.; Geng, J.; Jiang, W. A time series classification method based on 1DCNN-FNN. In Proceedings of the 33rd Chinese Control and Decision Conference (CCDC), Kunming, China, 22–24 May 2021; pp. 1566–1571.
54. Garcia, K.D.; de Sá, C.R.; Poel, M.; Carvalho, T.; Mendes-Moreira, J.; Cardoso, J.M.; de Carvalho, A.C.P.L.F.; Kok, J.N. An ensemble of autonomous auto-encoders for human activity recognition. *Neurocomputing* **2021**, *439*, 271–280. [CrossRef]
55. Lima, W.S.; Bragança, H.L.; Souto, E.J. NOHAR-NOvelty discrete data stream for human activity recognition based on smartphones with inertial sensors. *Expert Syst. Appl.* **2021**, *166*, 114093. [CrossRef]
56. Sena, J.; Barreto, J.; Caetano, C.; Cramer, G.; Schwartz, W.R. Human activity recognition based on smartphone and wearable sensors using multiscale DCNN ensemble. *Neurocomputing* **2021**, *444*, 226–243. [CrossRef]
57. Ramesh, A.K.; Gajjala, K.S.; Nakano, K.; Chakraborty, B. Person authentication by gait data from smartphone sensors using convolutional autoencoder. In Proceedings of the International Conference on Intelligence Science, Durgapur, India, 24–27 February 2021; pp. 149–158.
58. Dhammi, L.; Tewari, P. Classification of human activities using data captured through a smartphone using deep learning techniques. In Proceedings of the 3rd International Conference on Signal Processing and Communication (ICPSC), Coimbatore, India, 13–14 May 2021; pp. 1–6.
59. Wenzheng, Z. Human activity recognition based on acceleration sensor and neural network. In Proceedings of the 8th International Conference on Orange Technology (ICOT), Daegu, Korea, 18–21 December 2020; pp. 1–5.
60. Peppas, K.; Tsolakis, A.C.; Krinidis, S.; Tzovaras, D. Real-time physical activity recognition on smart mobile devices using convolutional neural networks. *Appl. Sci.* **2020**, *10*, 8482. [CrossRef]
61. Mehmood, K.; Imran, H.A.; Latif, U. HARDenseNet: A 1D DenseNet inspired convolutional neural network for human activity recognition with inertial sensors. In Proceedings of the 23rd International Multitopic Conference (INMIC), Bahawalpur, Pakistan, 5–7 November 2020; pp. 1–6.
62. Aswal, V.; Sreeram, V.; Kuchik, A.; Ahuja, S.; Patel, H. Real-time human activity generation using bidirectional long short term memory networks. In Proceedings of the 4th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 13–15 May 2020; pp. 775–780.
63. Xia, K.; Huang, J.; Wang, H. LSTM-CNN architecture for human activity recognition. *IEEE Access* **2020**, *8*, 56855–56866. [CrossRef]
64. Agarwal, P.; Alam, M. A lightweight deep learning model for human activity recognition on edge devices. *Procedia Comput. Sci.* **2020**, *167*, 2364–2373. [CrossRef]
65. Zhang, H.; Xiao, Z.; Wang, J.; Li, F.; Szczerbicki, E. A novel IoT-perceptive human activity recognition (HAR) approach using multihead convolutional attention. *IEEE Internet Things J.* **2019**, *7*, 1072–1080. [CrossRef]
66. Huang, J.; Lin, S.; Wang, N.; Dai, G.; Xie, Y.; Zhou, J. TSE-CNN: A two-stage end-to-end CNN for human activity recognition. *IEEE J. Biomed. Health Inf.* **2019**, *24*, 292–299. [CrossRef] [PubMed]
67. Mukherjee, D.; Mondal, R.; Singh, P.K.; Sarkar, R.; Bhattacharjee, D. EnsemConvNet: A deep learning approach for human activity recognition using smartphone sensors for healthcare applications. *Multimed. Tools Appl.* **2020**, *79*, 31663–31690. [CrossRef]
68. Tang, Y.; Teng, Q.; Zhang, L.; Min, F.; He, J. Efficient convolutional neural networks with smaller filters for human activity recognition using wearable sensors. *arXiv* **2020**, arXiv:2005.03948v1.
69. Beirami, M.J.; Shojaedini, S.V. Residual network of residual network: A new deep learning modality to improve human activity recognition by using smart sensors exposed to unwanted shocks. *J. Health Manag. Inf.* **2020**, *7*, 228–239.
70. Zhang, Y.; Zhang, Z.; Zhang, Y.; Bao, J.; Zhang, Y.; Deng, H. Human activity recognition based on motion sensor using U-Net. *IEEE Access* **2019**, *7*, 75213–75226. [CrossRef]

71. Pienaar, S.W.; Malekian, R. Human activity recognition using LSTM-RNN deep neural network architecture. In Proceedings of the IEEE 2nd Wireless Africa Conference, Pretoria, South Africa, 18–20 August 2019; pp. 1–5.
72. Manu, R.D.; Kumar, S.; Snehashish, S.; Rekha, K.S. Smart home automation using IoT and deep learning. *Int. Res. J. Eng. Technol.* **2019**, *6*, 1–4.
73. Varamin, A.A.; Abbasnejad, E.; Shi, Q.; Ranasinghe, D.C.; Rezatofighi, 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.
74. Chandini, U. A Machine learning based activity recognition for ambient assisted living. *Int. J. Future Revolut. Comput. Sci. Commun. Eng.* **2018**, *4*, 323–326.
75. Dungkaew, T.; Suksawatchon, J.; Suksawatchon, U. Impersonal smartphone-based activity recognition using the accelerometer sensory data. In Proceedings of the 2nd International Conference on Information Technology (INCIT), Nakhonpathom, Thailand, 2–3 November 2017; pp. 1–6.
76. Chen, Y.; Zhong, K.; Zhang, J.; Sun, Q.; Zhao, X. LSTM networks for mobile human activity recognition. In Proceedings of the International Conference on Artificial Intelligence: Technologies and Applications, Bangkok, Thailand, 24–25 January 2016; pp. 50–53.
77. Abdallah, Z.S.; Gaber, M.M.; Srinivasan, B.; Krishnaswamy, S. Adaptive mobile activity recognition system with evolving data streams. *Neurocomputing* **2015**, *150*, 304–317. [CrossRef]
78. Mekruksavanich, S.; Jitpattanakul, A.; Youplao, P.; Yupapin, P. Enhanced hand-oriented activity recognition based on smartwatch sensor data using LSTMs. *Symmetry* **2020**, *12*, 1570. [CrossRef]
79. Han, D.; Lee, C.; Kang, H. Gravity control-based data augmentation technique for improving VR user activity recognition. *Symmetry* **2021**, *13*, 845. [CrossRef]
80. Su, J.-Y.; Cheng, S.-C.; Huang, D.-K. Unsupervised object modeling and segmentation with symmetry detection for human activity recognition. *Symmetry* **2015**, *7*, 427–449. [CrossRef]