electronics

MDPI

Article

Improved Human Activity Recognition Using Majority Combining of Reduced-Complexity Sensor Branch Classifiers

Julian Webber [1,*], Abolfazl Mehbodniya [2,*], Ahmed Arafa [2] and Ahmed Alwakeel [2]

[1] Graduate School of Engineering Science, Osaka University, Toyonaka 560-8531, Japan
[2] Department of Electronics & Communications Engineering, Kuwait College of Science and Technology, 7th Ring Road, Doha 20185145, Kuwait; a.arafa@kcst.edu.kw (A.A.); a.alwakeel@kcst.edu.kw (A.A.)
* Correspondence: jwebber@ieee.org or webber@ee.es.osaka-u.ac.jp (J.W.); a.niya@kcst.edu.kw (A.M.)

Abstract: Human activity recognition (HAR) employs machine learning for the automated recognition of motion and has widespread applications across healthcare, daily-life and security spaces. High performances have especially been demonstrated using video cameras and intensive signal processing such as the convolutional neural network (CNN). However, lower complexity algorithms operating on low-rate inertial data is a promising approach for portable use-cases such as pairing with smart wearables. This work considers the performance benefits from combining HAR classification estimates from multiple sensors each with lower-complexity processing compared with a higher-complexity single-sensor classifier. We show that while the highest single-sensor classification accuracy of 91% can be achieved for seven activities with optimized number of hidden units and sample rate, the classification accuracy is reduced to 56% with a reduced-complexity 50-neuron classifier. However, by majority combining the predictions of three and four low-complexity classifiers, the average classification accuracy increased to 82.5% and 94.4%, respectively, demonstrating the efficacy of this approach.

Keywords: human activity recognition; LSTM; machine-learning; majority combining; smartphone; sensors; sensor-fusion

Citation: Webber, J.; Mehbodniya, A.; Arafa, A.; Alwakeel, A. Improved Human Activity Recognition Using Majority Combining of Reduced-Complexity Sensor Branch Classifiers. *Electronics* **2022**, *11*, 392. https://doi.org/10.3390/electronics11030392

Academic Editor: Giovanni Andrea Casula

Received: 5 December 2021
Accepted: 25 January 2022
Published: 28 January 2022

1. Introduction

Situation-aware technology facilitates comfortable living by augmenting human activities with contextual information and has many applications including in transportation, health, communications sports, forecasting and security [1]. Knowing the physical state or orientation of a driver or pilot can save vital seconds and enable machines to make optimized decisions for evasive action. Devices can monitor the technique of sports players and provide real-time feedback. Smart wearables incorporate sensors embedded in the fabric and are one of the emerging and efficient means to enable situation awareness. There are, however, challenges to optimally manage the large amount of sensor information in a timely and efficient manner.

Ambient assisted living (AAL) is the application of technology to facilitate and enable elderly and infirm persons to live comfortable and safe lives [2]. Care can be brought to elderly or infirm persons if a fall is detected. Information can be sent to a care-provider when an activity outside a normal routine or a dangerous condition is encountered. Examples of the technology include informing an owner if a fire-heater is left on, protecting against burglary, setting air-conditioning controls, turning on devices automatically or contextually when a person is deemed to need them. An infirm person does not need to make a specific command or select a particular service: sensors provide the relevant information to a machine learning 'brain' that decides on the most likely activity and can select the most appropriate action. Assisted living systems can apply human activity recognition (HAR) for improving life and maintaining a healthy lifestyle [2]. Sensors in smartphones have been used to monitor the severity of nervous system disorders [3].

Human activity recognition can be applied to determine worker presence, meta-tagging and human counting in order to restrict number of persons under pandemic conditions. The reflected and blocked wireless local area network (WLAN) signals can be used to predict human presence and activity in mainly indoor environments. Through analyzing received WLAN signals accuracy approaching 100% was achieved using the multi-class support vector machines (SVM) algorithm to separate user activities [4]. Employing wearable sensors on the other hand enables HAR in both indoors and outdoors with less restrictions. Common sensors for HAR include accelerometer, temperature, radar, pressure, stress, and magnetic field [5]. One challenge is processing the large amounts of sensing data with low-latency and machine learning algorithms are particularly suited for this processing.

Various studies have been conducted on HAR using smartphone sensors. Authors in [6] were able to increase the classification precision by augmenting sensor information with position-aware context, and placing sensors at specific positions on the human body with division into dominant and non-dominant limbs. It was shown in [7] to be possible to identify smartphone position using accelerometer data only. An accuracy of 77.3%, was achieved using accelerometer data only and increased to 85% when combined with angular and orientation data. Using accelerometer and gyroscope signals, authors in [8] extracted features through training with a deep belief network achieving 89.6% accuracy for twelve physical activities compared with 82.0% using SVM and 65.3% using an artificial neural network (ANN). A classification accuracy of 95.6% was achieved using the IMU smartphone sensors and a convolutional neural network (CNN) that could differentiate going upstairs and walking in [9]. Authors in [10] reported a 99.5% accuracy for real-time classification using a CNN operating on the discrete Fourier transform (DFT) of 14×60 images constructed from IMU data. In [11], the authors detect three human activities by utilizing four different machine learning approaches (SVM, multi-layer perceptron, J48 and naive Bayes). They concluded that the J48 statistical classifier that generates decision trees is the most efficient algorithm that results in a simple IF-THEN rule implementation. Single versus multiple voting systems for HAR were compared in [12]. It was shown that Random Forest alone achieved highest validation accuracy of 73% but performance was not improved combining with Gradient Boosting and Gaussian Naive Bayes. Using data from nineteen smartphone data sensors it was possible to identify the individual with an average 80% accuracy [13]. Cited challenges to achieving higher success rates were dealing with missing and insufficient data samples. By considering the symmetry of motion and energy distribution between left and right limbs though using the discrete wavelet transform of accelerometer data an accuracy of 91.0% was reported in [14]. Self-supervised learning from unlabeled multi-sensor data through recognizing features of the scalogram computed from the wavelet transform was considered in [15]. By combining average probabilities from three classifiers a 91% classification accuracy was achieved for six activities (dancing, stairs-down, low-walk, running, stairs-up and fast-walk) with the smartphone placed in either the hand or in trouser pocket [16]. A similar approach for HAR deploying the CNN for recognizing features in the time-frequency scalogram of received signal strength data produced as humans interact with the wireless channel was reported in [17].

Human gait recognition is a biometric approach that often employs a stick-figure for the human and bio-mechanical model for movement using an ellipse fitting technique to track body parts. The CNN was pre-trained on extracting Gait features in [18] with an average 90% accuracy reported for three viewing angles using the CASIA B dataset [19]. Performance was the dependent on training with a sufficient amount of data. Human gait recognition using a model-based approach is conducted in [20] by employing the ResNet-V2 and NASNEt that had been trained on a CASIA B gait data-set. The extracted features were optimized using a whale optimization algorithm. By fusing the best features based on the near absolute deviation an average accuracy of 89% was achieved. A method to learn the common feature subspaces from feature sets including skeleton, depth and

optical flow was considered in [21]. The CNN is a deep learning algorithm with many layers for extracting features, activation and down-sampling. Extraneous features still exist and a feature selection stage often follows to reduce the dimensionality to the single most important feature often using genetic optimization algorithms. These steps further increase the complexity. Using the Kinect interaction dataset [22] a 90% accuracy was achieved for optical flow-depth applying a technique to learn the common subspace from two sets. In the case of classifiers intended for portable devices it is important to develop models that offer optimized performance-to-complexity trade-offs. Although a camera can be embedded in clothing and the CNN provides good classification results, the high computational complexity, cost and weight make this approach less desirable. Light-weight smart clothing typically employing inertial sensors are ideally paired with low processing complexity, low-power, compact units.

The long short-term memory (LSTM) algorithm is a class of recurrent neural network with strong ability for detection and prediction of time varying patterns. An application for sleep motion detection using infra-red was proposed in [23]. Rough classification is computed on micro-Doppler data and the LSTM is used to classify the body movements. Although the feedback architecture of the LSTM increases the complexity that scales with the number of neurons and sample-rate many researches have addressed complexity reduction. Reduced complexity algorithms have been proposed for voice activity detection in [24], brain electroencephalogram (EEG) signal detection in [25], and for energy-efficient speech recognition in [26]. Performance enhancement has been made by acquiring the covariance matrix to improve the update factors for real-time processing [27]. An approach of limiting the weight data length was proposed in [28]. A two-stage activity classification for indoor positioning is proposed in [29]. Authors use inertial sensors for sensing three activities (walking, running, and stopping) and seven action units (i.e., three-step types, two-turn types, abnormal activity, and stop). A first-stage LSTM determines the number of steps and turns while a second-stage LSTM determines the action units. A moving distance estimator determines the length of each step, and the final position of pedestrian's indoor trajectory is obtained. An average position error of 0.782 m was computed. The proposed network correctly recognized the activities and action units with 97.9% and 95.5% accuracy, respectively.

Improved performance can be obtained by combining the classifier predictions from differing modal sensors. Fusing shape and texture data of human motions with features extracted by optical flow enabled accuracies of 94.5% to 100% to be achieved over different activities [30]. By fusion of Kinect vision data with inertial sensor data a 97.2% accuracy was achieved in [31] and compared with about 88% without fusion. Authors in [32] proposed a hybrid data fusion technique to estimate activities pertaining to meeting, walking or driving using accelerometer and gyroscope data on a smartwatch. A 98.2% accuracy was achieved for meeting-walking and meeting-motorized transportation activity using accelerometer data. Data-fusion uses a matrix time-series method, and a modified Better-than-the-Best Fusion (BB-Fus) algorithm. An optimized combination of sensors for validation was obtained from the confusion matrix after training. Authors in [33] apply fusion of RGB images, optical flow and audio achieving an accuracy of 82.4%. Meanwhile authors in [21] proposed the fusion of RGB depth and optical flow collected from the Kinect camera to achieve a classification accuracy of 91.4%. Fusion was achieved by finding a common subspace representation of extracted features.

Combining classifier outputs produces best performance when they each contribute complementary information [34] through use of multiple sensors. Performance improvements can also be gained through a spatial distribution of same-type sensors and sensor is uncorrelated. Researchers have considered various rule-based combining methodologies including based on min, max, median, mode, range, correlation, root mean square and variance of branch classifier [16]. Combining the sensor outputs based on the lowest variance is one of the techniques that has been proposed to improve the performance under controlled experiments [35]. The variance is dependent on the orientation of a sensor

and can change when the sensor slips or is orientated differently than during training. This strategy therefore requires regular calibration which increases the complexity and latency. For a consumer smart wearable use-case where there is increased potential for misalignment a robust solution with limited complexity is desirable. Table 1 provides a summary of current research on sensor fusion using machine learning.

Table 1. Multi-classifier combining using machine learning.

Algorithm	Rx Sensor	Fusion	Accuracy (%)	Reference
M-SVM	Camera	Shape and Texture	94.5 to 100	[30]
CNN	Kinect and Accel.	Depth and Inertial	97.2	[31]
kNN, SVM	IMU	Accel. and Gyro.	98.3	[32]
CNN	Camera	RGB, Flow and Audio	82.4	[33]
Subspace	Kinect	Flow and Depth	90.1	[21]
Multiple	Accel.	MP, LogicBoost and SVM	91.2	[16]
J48, k-NN, LR	IMU	Accel. and Gyro.	94	[35]
LSTM	IMU	Accel, MF, Ang. Vel and Orient.	94.4	This

Despite the progress in classifier fusion there still remain a number of challenges with the existing fusion approaches:

- Training and evaluation times should be minimized.
- Individuals' motion modes are highly variable.
- Human actions can be arbitrary and complicated making specific models restrictive and ineffective.
- Confusion can occur between similar motions such as walking on level ground and upstairs.
- Complexity should be limited.

One strategy to solve a number of these issues is to deploy a complex model such as the CNN with fusion of 2D images and 3D optical flow. Processing video data in real-time with a typically over one hundred layer CNN requires significant computing resources and energy. The availability of a camera constantly focused directly at a subject is also not very practical. Therefore we consider a non optical-flow approach which combines multiple classifier predictors each having lower complexity.

We first conduct a measurement campaign to collect unique data from a set of sensors contained in smartphones placed on the lower-limbs of participants as they conduct seven different physical activities. We propose the majority-combiner that predicts based on a consensus of classifier predictions. The solution has limited complexity and is less affected by outliers and/or a poorly positioned sensor. After a review of the recent literature on multi-classifier combining for HAR we make the following contributions:

- Propose a majority combiner for improving the multi-classifier performance with reduced-complexity.
- Demonstrate that reduced complexity processing can be compensated by combining estimates from multiple sensor predictors.
- Study the complexity versus performance trade-offs with different sensor sample rates, number of hidden units and solver types.

The layout of this paper is as follows. Section 2 introduces the sensor measurement and LSTM algorithm. Section 3 describes the experiment set-up and procedure. Section 4 presents the performance results including as a function of sensor type, number of hidden units, solver type and sample rate. Performance results resulting from combining multiple classifier outputs are then discussed. Future work, a conclusion and discussion are then presented in Sections 5–7, respectively.

2. Hardware and Software

In this section, we describe the sensor measurement and LSTM algorithm for motion classification.

2.1. Sensor Measurement

Smartphones contain dozens of sensors enabling their precise state in space to be determined. Common sensors includes: image, sound, proximity, motion (accelerometer), ambient-light, moisture, gyroscope, barometer, position (GPS), compass (magnetometer), fingerprint, light (LiDAR) and Soli (radar).

The accelerometer, gyroscope and magnetometer sensors are combined in the smartphone inertial measurement unit (IMU). The accelerometer measures the rate of change of movement (m/s^2). The gyroscope measures the angular velocity, which is the change in rotational angle per unit of time measured in ($^\circ/s$). The magnetometer measures the magnetic field (μT). In addition orientation is reported by the smartphone application as shown in Figure 1. Orientation records the azimuth, pitch and roll of the smartphone ($^\circ$). In this work, we refer to 'IMU sensors' to include the accelerometer, gyroscope and magnetometer sensors and 'set of four sensor data' to include the three IMU sensor data and the orientation data.

An example of an IMU is the Bosch Sensortec found in the iPhone X, which is an evolution of the BMI055 [36] IMU that samples at a maximum rate of 1 kHz. The BMI055 accelerometer has 0.98 mg accuracy with 12-bit digital resolution, while the gyroscope has 0.004° accuracy and 16-bit digital resolution. A GPS sensor additionally provides latitude, longitude, speed, coarse and altitude information. Although this data could be used to enhance accuracy and provide useful contextual background it was not used in this work. The set of four sensor data is uploaded to a cloud server on activity completion and downloaded to a computer for post-processing.

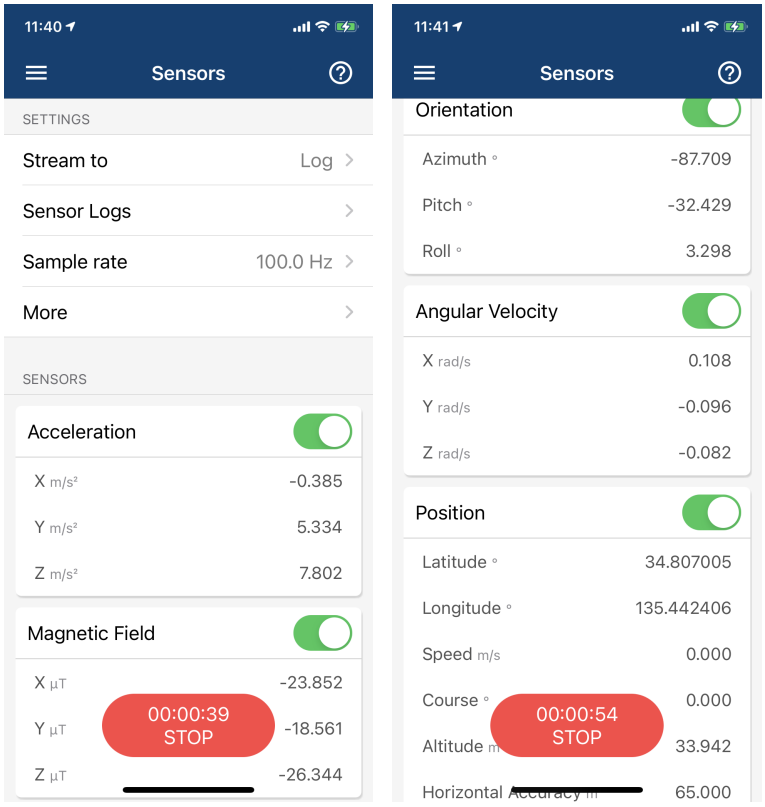

Figure 1. Smartphone "Sensors" application screenshots.

In this work, we specify seven human activities and aim to recognize the particular motion from test data having previously trained the neural network. The specific activities are:

- Activity 1: Standing;
- Activity 2: Walking;
- Activity 3: Stepping machine;
- Activity 4: Cycling indoors;
- Activity 5: Jogging on the spot;
- Activity 6: Jogging outdoors;
- Activity 7: Spinning (turning in a circle on the spot).

Example data from the X-axis acceleration sensor for each activity is plotted in Figure 2 and from all 3 axes in Figure 3. Sitting is additionally shown as a reference signal. The response of each sensor is clearly dependent on the particular activity undertaken. It can be seen that the sensor output for Standing has slightly higher variance as well as an offset compared with the Sitting data. A higher signal variance can be seen for the activities undertaken outdoors such as Walking and Jogging (out).

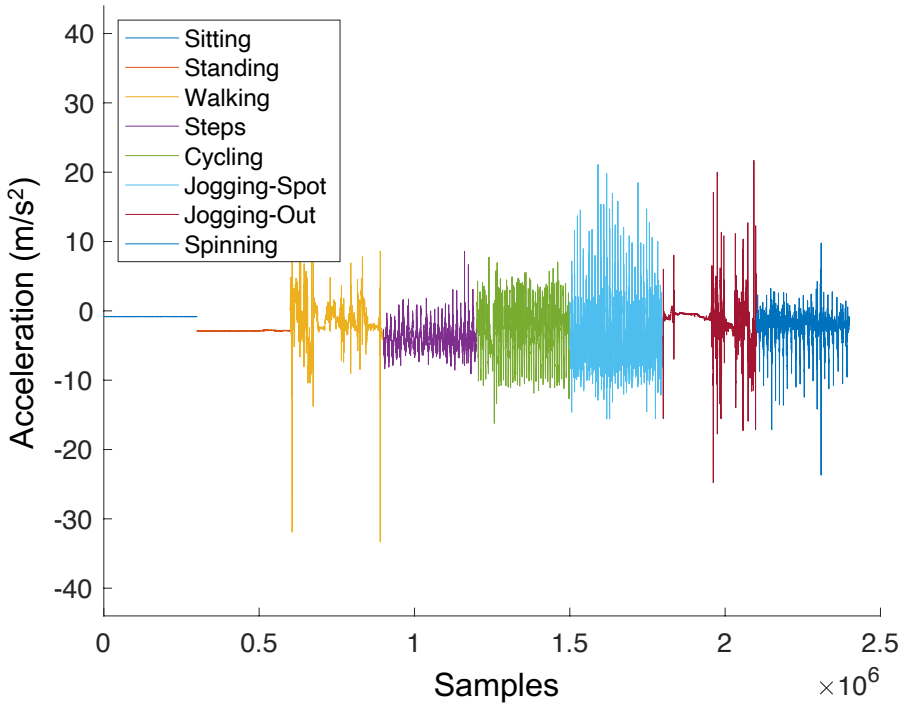

Figure 2. Accelerometer x-axis sensor data corresponding to each of the activities.

The accuracy of a sensor depends on factors including the digital quantization, transducer quality, signal conditioning fidelity as well as environmental conditions. If two or more sensors are sufficiently separated the noise will be uncorrelated and improvements made by averaging. Accelerometer noise is inversely proportional to the square root of the number of sensors. Therefore noise is reduced by a factor of two by averaging over four sensors. Measures such as applying a low-pass filter can also reduce the noise but can add lag and reduce the responsiveness. Combining information from two different types of sensor such as angular-velocity and magnetic-field can be beneficial as the sensor noise will be uncorrelated.

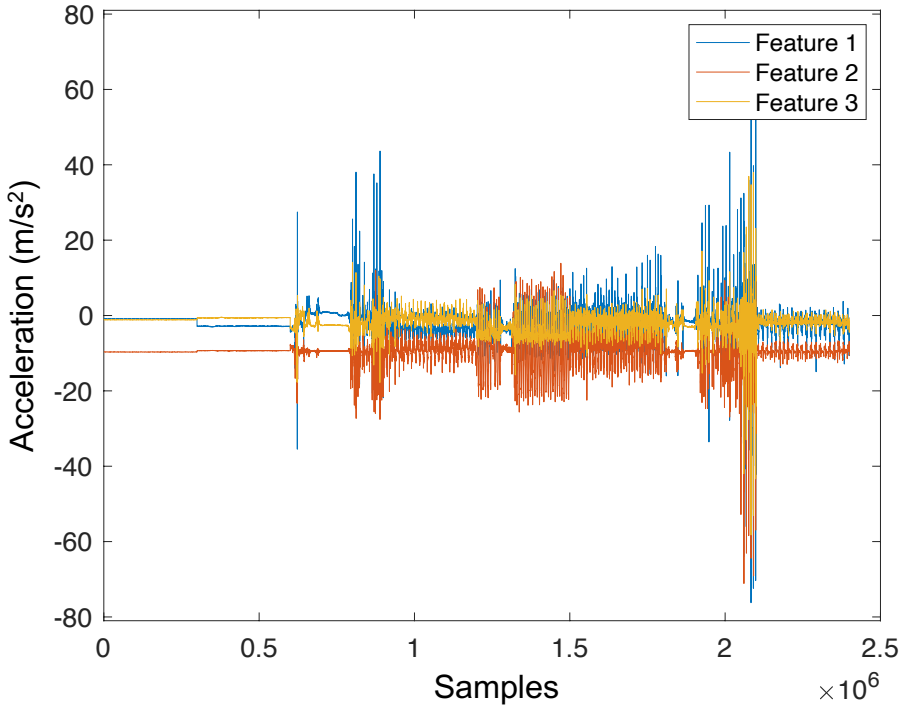

Figure 3. Accelerometer sensor features on 3 axes for each of the activities.

2.2. Long Short-Term Memory (LSTM) Network

LSTM is a form of recurrent neural network (RNN) that can learn the long-term inter-dependencies in time-series data and was first proposed by Hochreiter and Schmidhuber [37]. The network used in this work comprises a sequence layer for handling the series input data, a LSTM layer which computes the learning, a fully-connected layer, a softmax layer and finally a classification layer. The LSTM is configured to predict the next time-step in a sequence on a sample-by-sample basis. The fully-connected layer's dimension affects how well the network learns dependencies, but it should not be too large to avoid over-fitting and excessive complexity. The hidden-state and cell-states at time t are termed $\mathbf{h}_t$ and $\mathbf{c}_t$, respectively. The current state and the next sequence data samples will determine the output and updated cell state. The cell state is given by Equation (1).

$$\mathbf{c}_t = f_t \odot \mathbf{c}_{t\text{-}1} + i_t \odot g_t \tag{1}$$

The hidden-state is given by Equation (2)

$$\mathbf{h}_t = \mathrm{o}_t \odot \sigma_c(\mathbf{c}_t), \tag{2}$$

where σ_c represents the state activation function.

The cell-candidate, input and output-states at time step t are expressed as:

$$g_t = \sigma_c(W_g \mathbf{x}_t + R_g \mathbf{h}_{t-1} + b_g), \tag{3}$$
$$i_t = \sigma_c(W_i \mathbf{x}_t + R_i \mathbf{h}_{t-1} + b_i), \tag{4}$$
$$o_t = \sigma_c(W_o \mathbf{x}_t + R_o \mathbf{h}_{t-1} + b_o), \tag{5}$$

where W_g, W_i and W_o represent the cell-candidate and input and output weights. R_g, R_i and R_o are the respective recurrent weights. b_g, b_i and b_o are the respective biases. Gating permits data to be discarded or kept at every iteration.

3. Experiment Set-Up and Procedure

The system performance is evaluated through a software simulation using the set of four sensor data collected from four smartphones, namely: (i) Apple iPhone X, (ii) Motorola Moto-G6 plus (Android), (iii) Apple iPhone 6s-Plus and (iv) Apple iPod Touch 6.

Classification performance can depend on the orientation and sensor attachment position relative to the body and hence they were fixed in a consistent position for each measurement. Smartphones i and ii were attached firmly to the left leg at the shin-level while smartphones iii and iv were firmly attached at exactly the same height on the right leg. The devices were held securely in place using dedicated smartphone holders for sport activities and this reduces occurrences of slowly time-varying signals and outliers that can occur when a smartphone slips. This promotes consistency and repeatability of results. Legs have restricted angles of movement and attached sensors generally track a more natural human motion. When the smartphone is held in the hand sensor signal generally has a greater variance as it depends not only on arm-swing but also on a quite variable wrist-rotation. The sensing performance should also be indicative of that using sensors placed in sports shoes. Sensors in footwear enable direct and accurate feedback on running or sports technique and products are available from the major sport-shoe manufacturers [38]. Note that the placement is in contrast to traditional ways of holding smartphones such as when listening to calls with the smartphone placed at the ears, holding the device in a shirt pocket, or keeping it in the hand while walking. The results obtained under the two configurations are likely to be different. The use of four smartphones to measure the movements is also not very comfortable and extra set-up time is required as the number of sensors increase. As a consumer wearable solution it is envisaged that the smartphones would be replaced by IMU sensors placed in the shoes or in a pocket at the base of the trousers. Data from the sensors (maximum rate 100 Hz) would ideally be sent by wireless to a central computer for processing. Note that while the raw data corresponding to when the smartphone is in a hand or in a trouser pocket will be different, as a result of experiments in [16] it was reported that the machine learning can classify the activities with similar levels of reliability.

The sensors were set to start sampling at approximately the same time under the Application control software. A consistent start-time for all smartphones was later obtained from the timestamp data. The participant then started the particular activity. On completion of the measurement the set of four sampled data is uploaded to a Mathworks web-server and then downloaded to a personal computer running Matlab software for processing. In a practical system, algorithms running in the cloud can process the data from the multiple sensors or smartphones and return classification results directly to the smartphone. Each smartphone contains four sensors and enable the collection of four times as much data per measurement and with an increase in type of devices. Data was collected for four volunteers each wearing the four smartphones. Participant activity may vary in the intensity and regularity of movements and hence performance results are averaged over four users. The data was sampled at the maximum rate of 100 Hz and subsequently down-sampled to the rates of {1, 2, 4, 10, 25, 50, 100} Hz.

The LSTM algorithm was trained using Matlab software. The sequence-to-sequence classification mode was set to provide an output at each time step. A drop-out rate of 0.2 was set to randomly remove 20% of neural node-connections during training in efforts to avoid over-fitting. The learning rate was set to decrease at 30% to avoid over-fitting when the loss function stops decreasing. The number of features on the LSTM was set to 3 for the accelerometer corresponding to the three axis. The initial learning rate was 0.001, the squared gradient decay was 0.999, and decay factor was 0.9.

To ensure the measurement data corresponded exactly to the labeled activity, the first 90 s of recorded data was discarded. This also helped ensure a more constant and regular speed had been achieved in each activity. The training data for LSTM algorithm was collected over the duration of the next 90 s and data for validation from a subsequent 90 s on different smartphones. We applied the K-folds cross-validation technique for maximizing

use of the collected data and improving accuracy in the averaged results. K-fold cross-validation is a resampling technique that is used to assess machine learning models using a small sample of data. The algorithm takes a parameter, *K*, which specifies the number of groups into which a given data sample should be divided. The test and validation sets were each split into three groups of 30 s duration and one section was randomly set to train the model and a remaining section for evaluation. The process is repeated *K*-times rotating the test data. The sampled data is first normalized to the maximum value on all branches, such that the highest value is one. Outliers beyond two standard deviations from the mean value are considered as outliers and removed from the data set. The measurement procedure was repeated to collect data for four volunteers.

The LSTM was trained using a 30-s segment of the training data. The network weights and biases are updated using a stochastic gradient descent for minimizing the loss function through optimized step-sizes at each time-step. A subset of the training data or mini-batch is selected at each iteration. The epoch number is a measure of the number of times all of the training vectors are used once to update the weights and was set to 75. The number is chosen not too small to avoid underfitting but not too large to avoid overfitting and was selected after trials to find a suitable value. The recurrent neural network can suffer from the vanishing and exploding gradient where the last-step value is significantly reduced or increased as it reaches the initial time step. The gradient threshold was set to 2.0 to avoid the gradients from exploding. Three different solvers were evaluated for updating the gradient descent namely the momentum (SGDM), adaptive moment estimation (ADAM) [39] and root mean square propagation (RMSProp) [40]. ADAM is an optimization algorithm using stochastic gradient descent which incorporates properties of the AdaGrad and RMSProp algorithms to improve performance under sparse gradients in noisy conditions. SGD is a gradient descent version which rather than conducting calculations on the whole dataset, performs computations on a subset or random selection of data samples [41].

The classification accuracy generally increases as the number of possible activities decreases. We therefore evaluate the performance as a function of the total number of activities to classify, i.e., ΣCateg. = {3, 4, 5, 6, 7}. The number of LSTM hidden-units was varied from 25 to 250 in increments of 25. The sample rate was evaluated from a minimum of 1 Hz to a maximum of 100 Hz. The simulation settings are summarized in Table 2. The classification accuracy is computed from the ratio of total correct to total number of classifications.

Table 2. Experimental Settings.

Feature	Value
Sensor data	Acceleration, Angular-Velocity, Magnetic-Field and Orientation
Solvers	SGDM, ADAM and RMSProp
No. of hidden-units	25, 50, 75, 100, 125, 150, 175, 200, 225 and 250
Sample rate	1, 2, 4, 10, 25, 50 and 100 Hz

The predicted versus actual activity for accelerometer data is plotted in Figure 4 for 25 hidden units and 2 Hz sensor sampling rate. Note in this example the evaluation was made with a sub-optimal number of hidden-units and sample-rate in order to reduce the accuracy and highlight the error distributions. It can be seen that there is generally good agreement between the trained and evaluation sequences with a stair-case shape clearly visible. There are some errors particularly in the border region between the activities. Spinning and Steps show a higher proportion of errors and this is considered to be due to the greater difficulty in maintaining a steady cadence in these activities compared to Cycling for example.

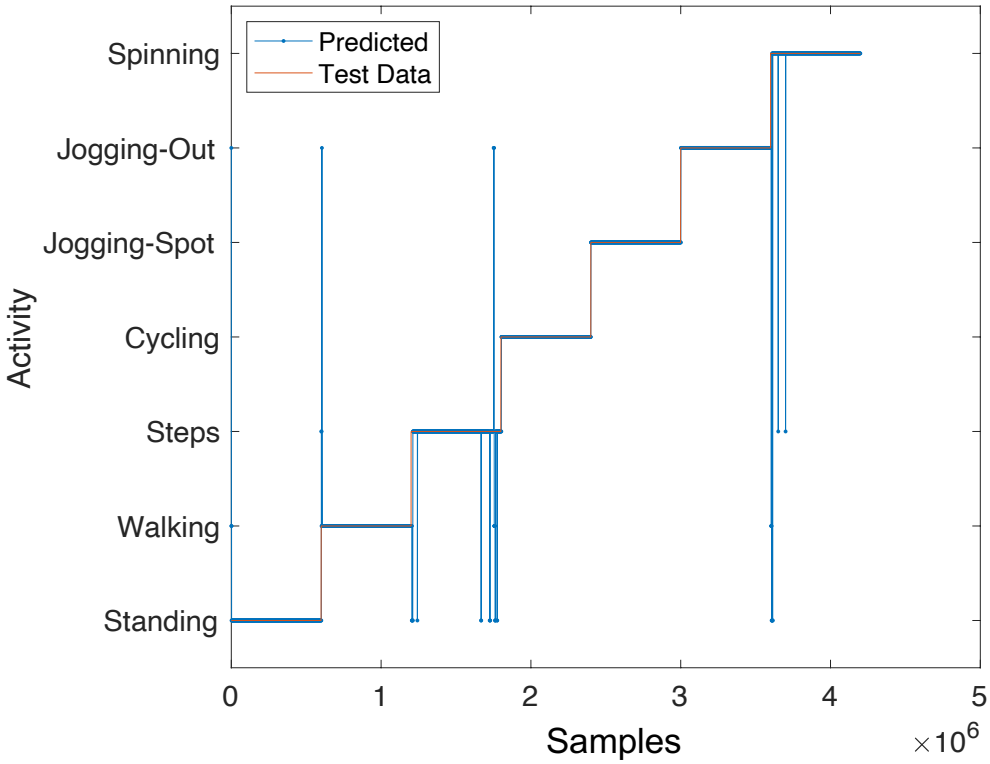

Figure 4. Predicted versus actual activity with time for each sensor type.

Classification accuracy was evaluated using three cross-validation folds. The results at each cross-validation run is tabulated in Table 3. An average accuracy and confidence level are computed. The accuracies presented in Section 4 are the averaged results.

Table 3. Accuracy at each cross-validation fold for Acceleration sensor.

Run	ΣCat. = 3	ΣCat. = 4	ΣCat. = 5	ΣCat. = 6	ΣCat. = 7
1	0.93	0.92	0.96	0.89	0.92
2	0.94	0.92	0.93	0.88	0.88
3	0.87	0.94	0.98	0.96	0.92
Average	0.91	0.93	0.96	0.91	0.91

Multi-Classifier Combining

In this work, we combine hard classifications at the output. Let the posterior class probability from the n-th classifier to the m-th combiner at sample time t be represented by $d_{n,m}(t)$. The decisions profile matrix is then expressed as [42]:

$$A_{m,n} = \begin{pmatrix} d_{1,1}(t) & d_{1,2}(t) & \cdots & d_{1,M}(t) \\ d_{2,1}(t) & d_{2,2}(t) & \cdots & d_{2,M}(t) \\ \vdots & \vdots & \ddots & \vdots \\ d_{N,1}(t) & d_{N,2}(t) & \cdots & d_{N,M}(t) \end{pmatrix},$$

where the columns represent the vote from the N classifiers to a particular class.

The majority combiner makes a decision to select class c_j if

$$\sum_{i=1}^{N} d_{i,j} = \max_{j=1}^{M} \sum_{i=1}^{N} d_{i,j} \tag{6}$$

where N is the number of classifiers, M is the number of classes and $d_{i,j} \in \{0,1\}$ is the decision of the ith classifier and jth class as shown in Figure 5. As the number of classifiers tends to infinity the misclassification approaches zero assuming each classifier makes an estimate with error probability less than half [43]. The performance of a combining scheme is dependent on the constituent classifiers and works best when each sensor provides the same average accuracy.

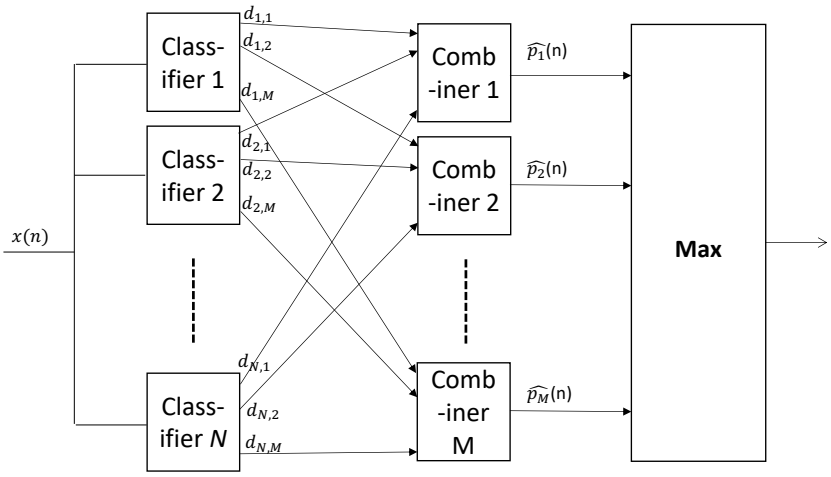

Figure 5. Block diagram of the multi-class combining function.

The classification reliability depends not only the particular activity data but also on the presence of training-test mismatches, sensor calibration and interference [44]. The majority operation will discard an unlikely classification when two or more predictors confer on an alternative activity. The majority combining will improve performance in the case a sensor slips or fails for any reason. As an example if the accelerometer and angular-velocity sensor predictors estimate the activity is Walking while the magnetometer and orientation sensor predictor estimates the activity as Jogging and Sitting, respectively, then Walking is finally selected as the most likely activity.

Difference in sensor reliability can be managed by applying a weighting scheme of which finding optimum weights is an application-specific and open problem. As the same sensor types are used in all smartphones and hence all have the same average reliability, we give equal weighting to each branch in order to limit the complexity and to provide a clear baseline performance from the combining.

4. Results

In this section, we provide an analysis of the measured sensor data followed by the classification performance results. The variance of the sensor output is first assessed by calculation of the Allen variance. The single-sensor classification accuracies are computed first for each sensor type, number of hidden units, and sample-rate. The performance benefits through obtained multiple sensor combining is then presented.

4.1. Allan Variance

The noise component of inertial sensors is crucial and influences accuracy of inertial navigation systems in real time. As the inertial measurement unit comprises an accelerometer and gyroscope, all sensor faults will affect the position determination accuracy. The signal variance can differ between each axis on the same sensor. Modern mass-produced sensors in smartphones have high but finite accuracies and the analogue components have manufacturing tolerances which mean the same movement may result in a different reading between two sensors in the same position. Finite-precision analogue-to-digital converters as well as measurement and thermal noise contribute to error at the sensor output. Each sensor will generate a bias producing an output value for no activity. A

common measurement of sensor performance is provided by the Allan variance computed as in [45]. The sensor bias is first estimated as Equation (7)

$$x_{bias} = \frac{1}{N}\sum_{i=0}^{N-1} x(i), \tag{7}$$

where, N is the total number of samples.

The successive error estimates of sensor bias are then computed as Equation (8)

$$e(m) = \frac{1}{N}\sum_{i=0}^{N-1} x(i+mN). \tag{8}$$

The variance, $\sigma^2(N)$, is computed as Equation (9)

$$\sigma^2(N) = \frac{1}{2(M-1)}\sum_{m=1}^{M} (e(m) - e(m-1))^2. \tag{9}$$

The computed Allan variance is shown for the IMU sensor outputs in Figure 6 (top) accelerometer (Ac) and (bottom) Magnetic force (Mf). Results are labeled for two sensors (1–2) on each cardinal axis (x, y, z). It can be seen that the variance depends on the particular axis and values are consistent between sensors with greatest variation for the Magnetic Force output on the y-axis.

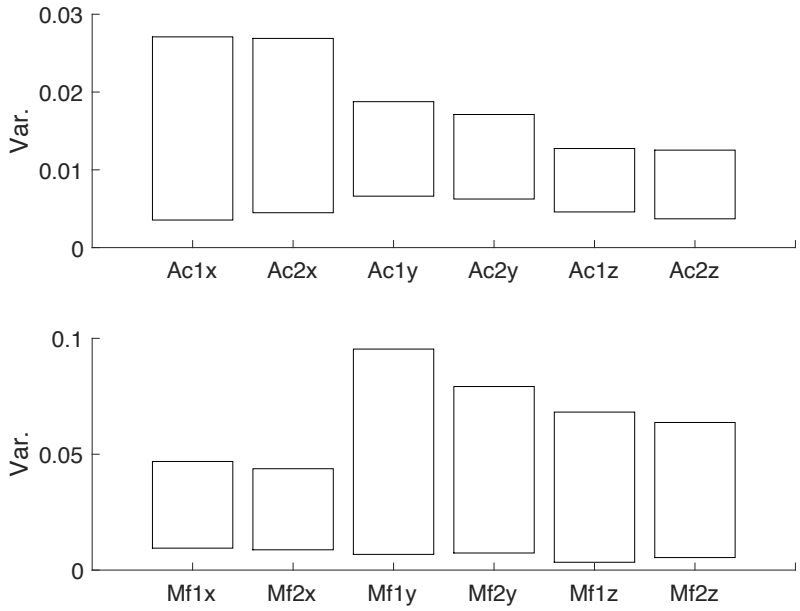

Figure 6. Plots of the Allen variance on IMU sensors (**top**) Accelerometer (Ac) and (**bottom**): Magnetic Force (Mf). Results are shown for two sensors (1–2), and on each cardinal axis (x, y, z).

4.2. Classification Accuracy by Sensor-Type

Performance versus type of sensor for increasing number of motion activities is plotted in Figure 7. For the range of activities considered in this work we found that the performance was best using the angular-velocity followed by acceleration sensor. SGDM achieved prediction accuracies of {0.91, 0.93, 0.96, 0.91 and 0.91} as the respective total number of activities increased from 3 up to 7. Meanwhile ADAM and RMSProp solvers achieved respective accuracies of {0.80, 0.80, 0.63, 0.69 and 0.62} and {0.71, 0.80, 0.63, 0.69 and 0.55}. SGDM was also the most consistent solver performing well regardless of the number of categories. Studies in [41] also showed that the SGDM solutions generalization better than

the adaptive methods even when the latter performed best in training. Hussain et al. were able to detect jogging, walking and standing/sitting with accuracies of 84%, 96% and 100% respectively corresponding to an average of 93% using the inertial sensor using SGDM solver [29]. We cannot make a direct comparison as their results contained three activities which are slightly different to our three-activity set (Standing, Walking and Stepping). However, if we consider, Stepping motions as similar to Jogging, our performance was an average of 91% (first bar in Figure 7) which is comparable with the 93% in [29].

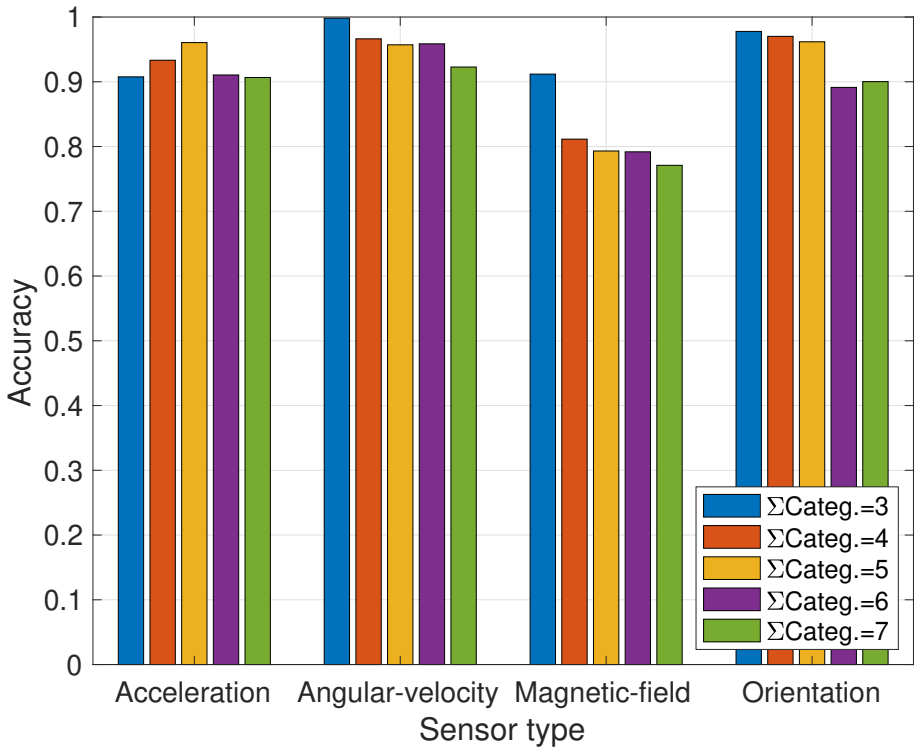

Figure 7. Classification accuracy versus sensor-type for accelerometer sensor as a function of the number of motion activities. No. of hidden units = 150, solver = ADAM, sample-rate = 10 Hz.

4.3. Effect of Increasing the Number of Activities

Classification accuracy versus type of solver algorithm for increasing number of motion activities is plotted in Figure 8. It was found that the SGDM solver performed best followed by ADAM solver.

4.4. Effect of Increasing Neuron-Complexity

Classification accuracy using the acceleration sensor for an increasing number of hidden-units with ADAM solver and a sample-rate of 10 Hz is shown in Figure 9. The performance was generally poorer when the number of hidden-units was 25 or 50 and increases as the units increase until about 125 units. There was little benefit from increasing the number of units beyond 125. We can observe that the average accuracy for ΣCateg. = 4 is greater than ΣCateg. = 3 despite the number of activities to categorize increasing. This can be explained by the knowledge that Categ. = 4 corresponds to Cycling. Cycling generates a stronger response on the sensors and can therefore be more accurately categorized compared to the Steps or Jogging (Spot) activities for example. Therefore the average accuracy improves when Cycling is included. It is noted that the average accuracy decreased from ΣCateg. = 4 to ΣCateg. = 5. Categ. = 5 corresponds to Jogging on the spot which has an acceleration motion similar to that of Steps and therefore miss-categorization between these activities can occur. Similarly the average accuracy increased when ΣCateg. = 6 i.e., jogging outdoors was included. Compared to the other walking activities there is a higher acceleration signal generated when running at speed which permits a more accurate cate-

gorization for this activity. The overall accuracy decreased when the 7th activity, Spinning, was included.

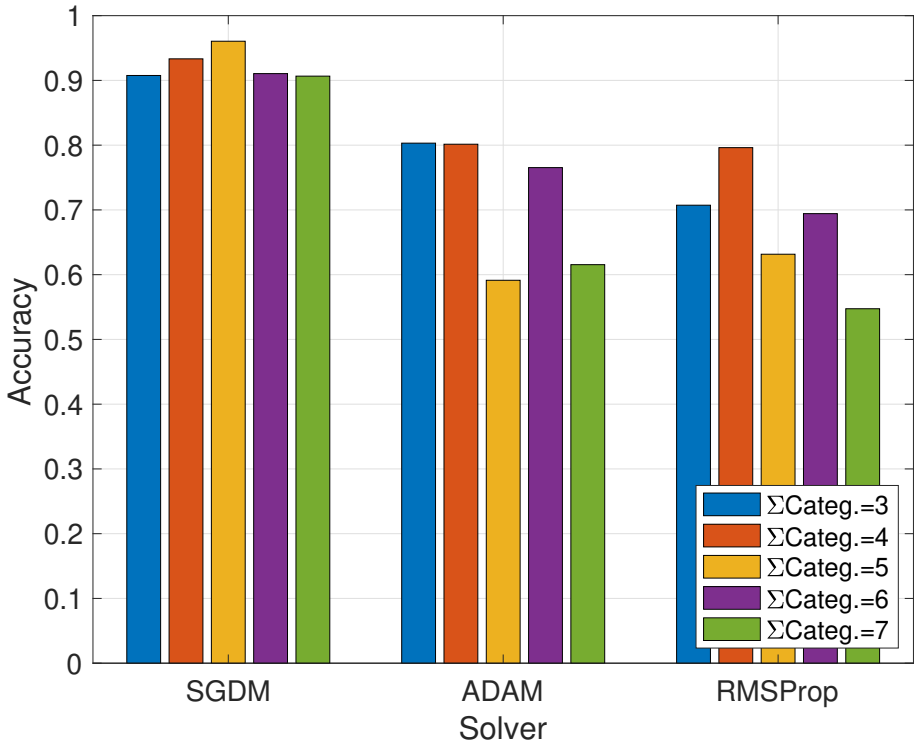

Figure 8. Classification accuracy versus solver-type for acceleration sensor. No. of hidden units = 150, sample rate = 10 Hz.

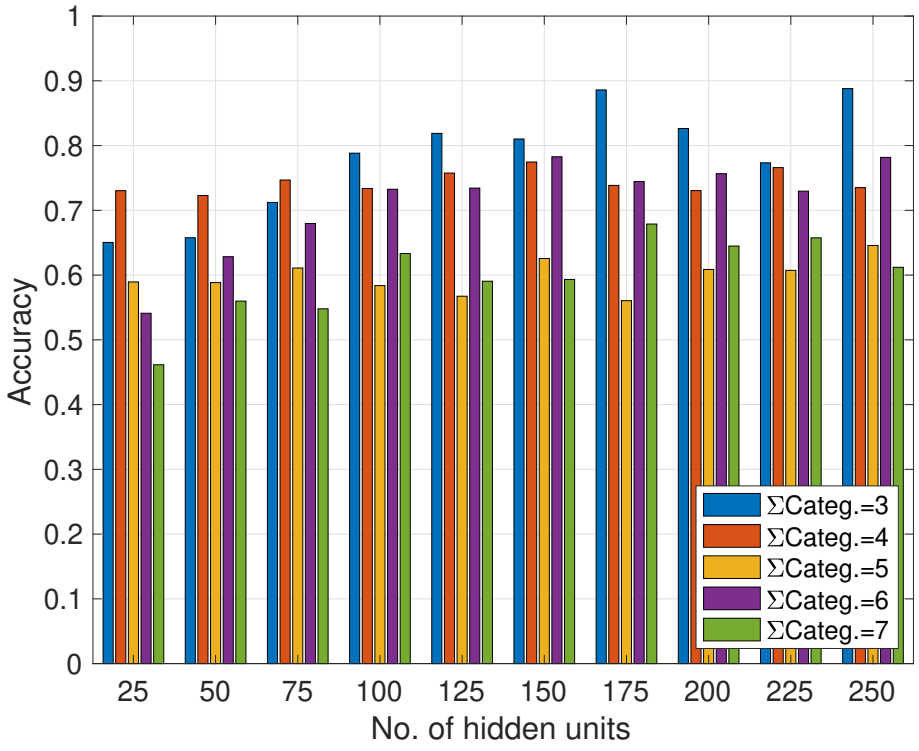

Figure 9. Classification accuracy versus number of hidden-units. Sensor = acceleration, solver = ADAM, sample-rate = 10 Hz.

4.5. Effect of Sampling Rate

Classification accuracy versus sample-rate for increasing number of motion activities is plotted in Figure 10. The performance steadily increases as the number of Categories to test is reduced from 7 to 3. Moreover this order is maintained across all of the different Sample-rates measured although at certain rates the performance difference was less than at others and could be exploited. For example, there is a smaller performance degradation in increasing the number of Categories from 3 to 7 at 22.5 Hz compared to 50 Hz. Good performance was achieved even with the relatively low-sample rate of 4 Hz. Considering that the repetition rate for each of the human activities (e.g., Walking and Cycling) is below about 2 Hz, a frequency of 4 Hz is sufficiently high to capture the signal components. However, it can be advantageous to sample at a slightly higher rate such as 10 Hz to improve performance by averaging in the presence of high noise. The default sampling-rate for the Sensors application is also 10 Hz and our results reaffirm that this is a suitable rate for sampling.

Sampling above the default 10 Hz rate does not significantly affect the performance within each sum of Category group setting. t The rate should be kept low as the computational complexity in terms of LSTM training time is almost linearly proportional to the number of samples and hence sample-rate as shown in Figure 11. The training time could be reduced further almost in proportion to the number of parallel processors available. A higher sample rate of 50 Hz has been suggested in some studies for fall detection [46]. It is possible that new activities particularly those with slower movement or rates of rotation could benefit from a higher sample rate than 10 Hz. If the processor has the capacity then it is recommended to set a higher sample rate for training untested activities. The total training and classification time was measured for a single-core on an Intel-i7 8th generation (8700B) processor with clock-rate 3.2 GHz. When the sample-rate was 100 Hz it took 1.95 h to compute the LSTM network training for all five configurations with an increasing number of activities (i.e., ΣCateg. = {3,4,5,6,7}).

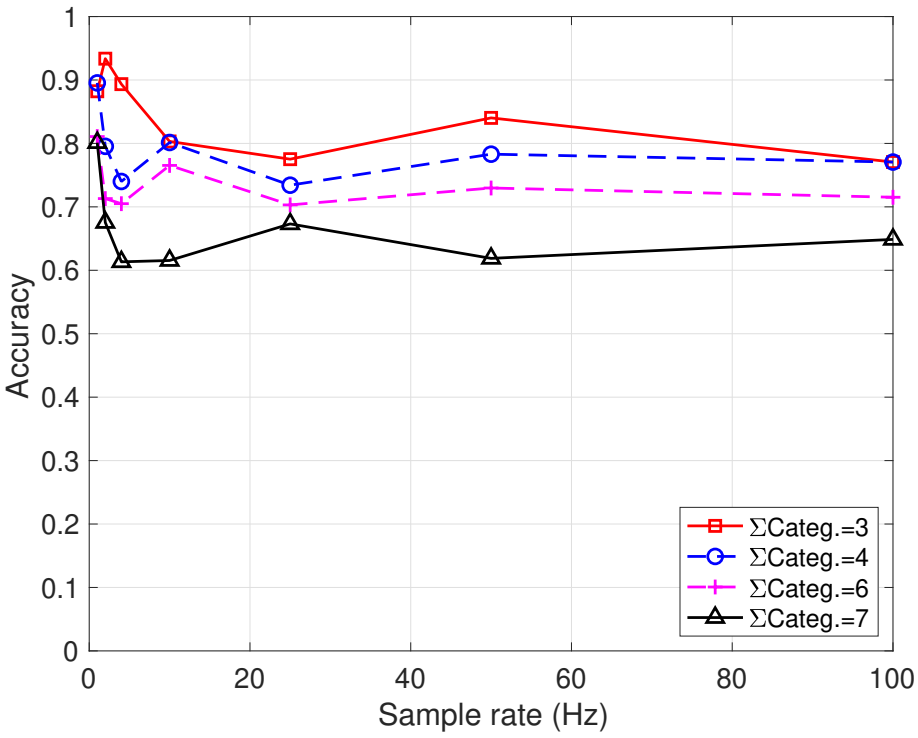

Figure 10. Classification accuracy versus sensor sample-rate. Sensor = acceleration, no. of hidden units = 150, solver = ADAM.

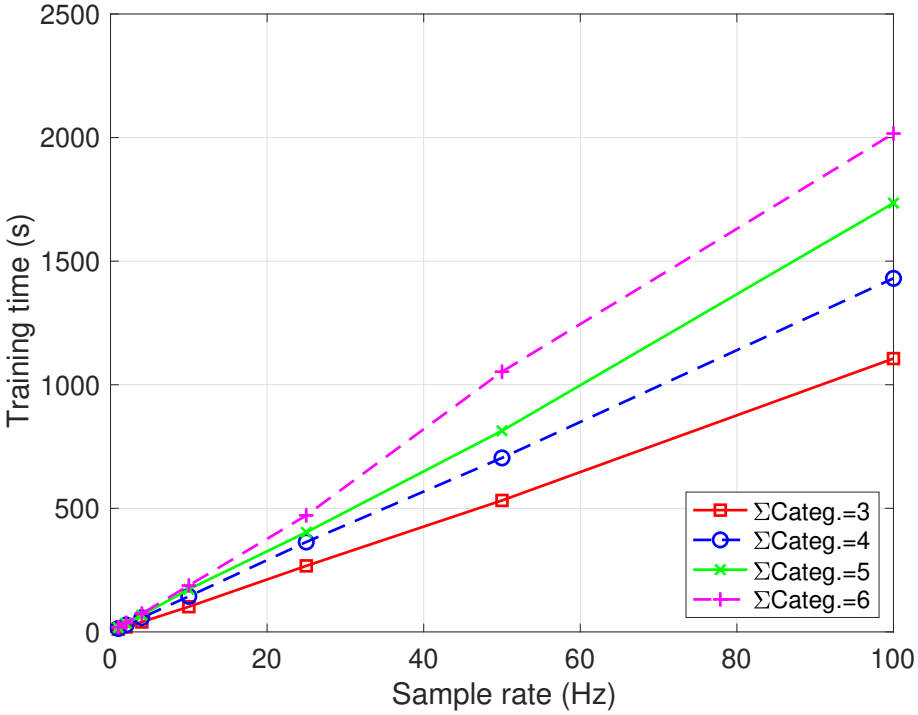

Figure 11. LSTM training time versus sample-rate. Sensor = acceleration, no. of hidden units = 150, solver = ADAM.

4.6. Classification Accuracy Combining Sensors

The classification accuracies for the case of combining three sensor (acceleration, angular-velocity and magnetic-field) predictors is shown in the confusion chart of Figure 12. The chart shows the True class versus Predicted class and the number of simulation results in each group for all activity combinations in a 7-by-7 grid. The non-zero off-diagonal elements indicate the cases where there are estimation errors. In the sensor fusion experiment, the number of hidden units was set low at 50 units, the sample-rate was set at 25 Hz and ΣCateg. = 7.

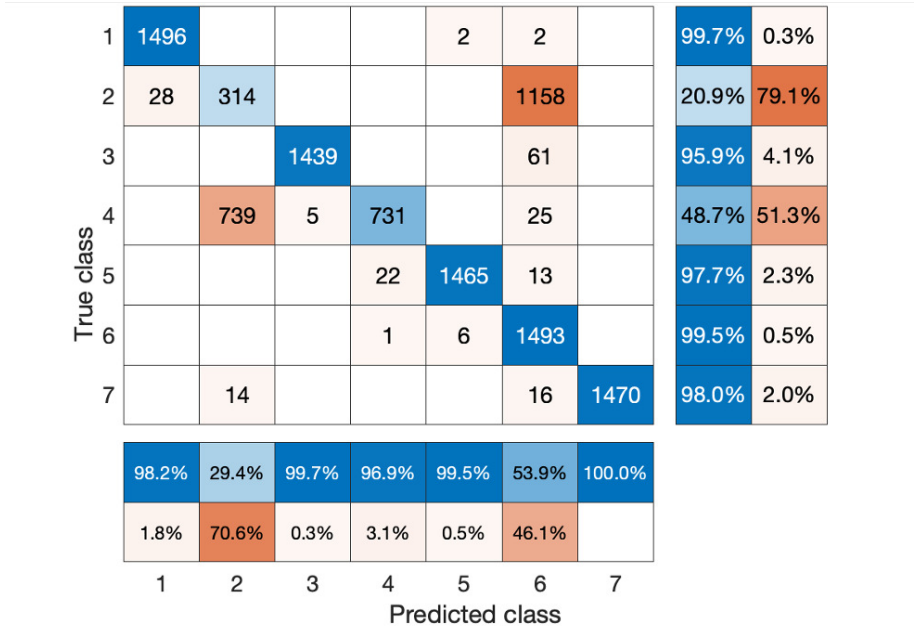

Figure 12. Confusion matrix for combining 3 sensor outputs, no. of hidden units = 50, and ΣCateg. = 7.

The average accuracy of 83% was achieved despite the relatively low number of hidden units. By comparison an average accuracy of 56% was achieved for a single acceleration sensor with the same experiment settings. Misclassification of Class-6 occurred with relatively high frequency when predicting Class number 2. Similarly, misclassification into Class-2 occurred with lower frequency when predicting Class number 4. This is considered due to the similarity in sensor responses for these two classes and for best accuracy the sensors should produce complementary information. It can be seen that Cycling was confused with Walking and also Walking was confused with Jogging in 46.1% of the particular cases.

It has been shown that fusing Kinect depth with inertial sensor data enabled a 100% accuracy to be obtained for three activities [29]. We cannot make a direct comparison due to the different sensor type, our increased number of activities and use of the lower-neuron count LSTM to demonstrate the gain from combining. However, we would be able to also achieve near 100% fusion accuracy by starting with a higher complexity single sensor classifier as achieved in Figure 7. We note that the depth information from a Kinect image sensor is very useful providing independent information from the IMU and it would be interesting for future work to see if it can eliminate the misclassifications between Jogging and Walking when using a lower-neuron count classifier.

We next consider the majority combining of four sensor-branch predictions. In theory we expect the overall accuracy to increase with the rise in number of sensors and increased computational complexity. Figure 13 shows the prediction accuracy for the case of combining estimates from the set of four sensor data. By combining the outputs of four classifiers the overall average accuracy rose to 94% with main errors caused only by confusing Walking with Jogging and on a much reduced 24.8% of occasions.

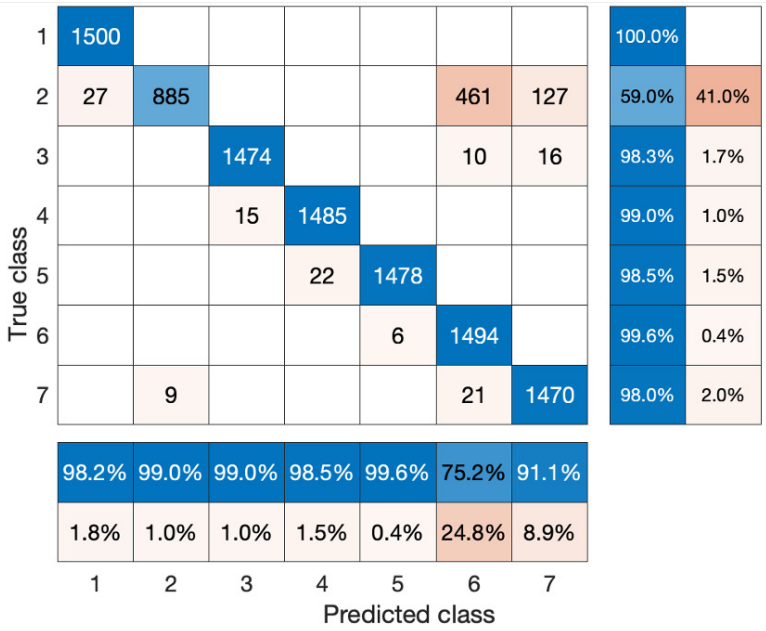

Figure 13. Confusion matrix for combining 4 sensor outputs, no. of hidden units = 50, and ΣCateg. = 7.

5. Further Work

This research can be extended in a number of directions.

- Optimized combining weights could be computed depending on the activity and environment.
- The number of sensor processing chains and complexity per sensor can be optimized at run-time to meet the required performance with minimum battery drain.
- The multiple low-complexity classifier combiner could be applied to recognizing the person undertaking an activity as well as the action being undertaken by the person.

Research could confirm whether a similar combining architecture can be used and measure the performance benefits.
- The performance of the LSTM can be compared with other algorithms such as SVM and k-means clustering.
- A comparison can be made by combining predictions from spatially separated sensors of the same type.
- The use of smartphones can be replaced with IMU units embedded in clothing and shoes.
- The hardware implementation and complexity of the classifier and should also be considered as part of our further work.
- It is desirable to evaluate the algorithm with a greater number of activities and collect data from more subjects.
- We will record additional activities representative of everyday living such as eating, drinking, cooking, clapping, waving, writing, going up/down stairs as well as sports activities such as tennis and football.

6. Conclusions

This work has considered the performance benefits from combining HAR classification estimates from multiple sensors each with lower-complexity compared with a higher-complexity single-sensor classifier. We first showed that the highest single-sensor classification accuracy of 91% was achieved using the SGDM computational solver for seven activities with optimized number of hidden units and sample rate. It was shown that the processing on the angular-velocity sensor provided the best results of the single sensors followed by that on the acceleration sensor. We then reduced the classifier neuron-complexities to just 50 units and reduced the sample-rate and found that the single-sensor classification accuracy was reduced to 56%. By majority combining the predictions of three sensors, the average classification accuracy increased to 82.5% using the same reduced-complexity settings. This accuracy was further improved to 94.4% when combining predictions from the set of four sensor data.

It was also shown that the classification errors were not evenly distributed but typically occurred when confusing a small number of similar motions. For example, Walking with Jogging activities. The classification accuracy improved as the neuron-complexity increases until about 125 units and adding further units do not significantly improve the performance. It was found that a sensor sampling rate of about 10 Hz is sufficient to reliably classify the activity.

7. Discussion

This paper has demonstrated a HAR methodology enabling increased classification accuracy through the deployment of multiple sensors branches each with limited processing complexity. The majority combiner is a good structure for fusing multiple sensor classifiers to achieve a required level of accuracy. The method promotes scalability-additional sensor classifier outputs can be added with minimal change to the architecture and the accuracy will approach 100% as the number of independent sensor samples increases. The fusion approach is robust to sensors failure and the problem of outlier samples. In this work, we utilized four independent smartphones (i.e., two placed on each leg) and therefore the number of subjects paired with independent smartphones is sixteen. Although the number of subjects is relatively small the averaged accuracy using this sixteen subject-data combinations will be higher than using a single smartphone.

7.1. Limitations

There are, however, limitations in the proposed methodology.

- The sensor placement in the lower leg is in contrast to traditional ways of holding smartphones, such as in the hand or trouser pocket. Some studies have indicated that while the raw data is different the classification accuracy can be similar, e.g., [16]. As a

consumer wearable solution it is envisaged that the smartphones would be replaced by IMU sensors placed in the shoes or in a pocket at the base of the trousers.

- Sensor slippage can cause misclassification. One solution is to restrict the movement of the sensor in clothing by making the surrounding fabric stiffer. Removal of outliers and majority combining of multiple sensor classifications can be effective when errors are caused by a sudden short-term slippage.
- The IMU unit contains a limited number of sensors and hence combining more than four branches will require multiple IMU. As the IMU has lower-cost and processing requirements compared to optical flow processing this may not be an issue. Spatial gains can also be achieved using multiple sensors of the same type.
- Only 3% improvement was obtained by employing information from the fourth sensor. The additional complexity may not warrant this modest performance increase.
- It can be cumbersome collecting data from multiple sensors of the same type with multiple smartphones. We employed smartphones as they are compact and reliable units for collecting and storing the information. For a practical system the sensors can be embedded in clothing or shoes and relayed to a central control unit by Bluetooth using an IoT-type transceiver such as the Intel Edison.

The approximate 1 mg performance of the accelerometer is an example of a mid-range IMU where performances range from 100 mg to a highly accurate 10 µg device. It is considered that the classification accuracy would not substantially change by deploying a more accurate IMU for the activities undertaken using human power, e.g., walking or running. An important step is to calibrate the sensors which can improve the raw performance by up to two decades.

7.2. Complexity

The learning time for the standard LSTM architecture using stochastic gradient decent optimization is of Order-1 with time complexity per step proportional to the total number of parameters. The learning time for a network is dominated by the factor $n_c \times (n_c + n_o)$ where n_c is the number of memory cells and n_o is the number of output units [47]. With a modest number of activities to classify (e.g., under 50) it is considered processing would be conducted by a field programmable gate array (FPGA) or applications specific integrated circuit (ASIC) and classification data sent to a communications device using Bluetooth or Wireless LAN. This would also be practical as a smart wearable solution. The implementation complexity is architecture specific, dependent on the type of hardware (e.g., FPGA or DSP), the manufacturer and family of device. We would like to consider this as part of our further work.

Author Contributions: All authors contributed to the paper. Conceptualization & methodology, J.W., and A.M.; software, J.W.; validation,investigation, formal analysis, all authors; writing—original draft preparation, J.W., and A.M.; writing—review and editing, J.W., A.M., A.A. (Ahmed Arafa) and A.A. (Ahmed Alwakeel); funding acquisition, A.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by the Kuwait Foundation for Advancement of Sciences (KFAS) under Grant PR19-13NH-04.

Acknowledgments: The authors would like to thank anonymous reviewers for their constructive comments, which helped in improving this manuscript.

Conflicts of Interest: The authors declare that there are no conflict of interest regarding the publication of this paper.

References

1. Forkan, A.R.M.; Khalil, I.; Ibaida, A.; Tari, Z. BDCaM: Big data for context-aware monitoring—A personalized knowledge discovery framework for assisted healthcare. *IEEE Trans. Cloud Comput.* **2015**, *25*, 628–641. [CrossRef]
2. Maskeliūnas, R.; Damaševičius, R.; Segal, S. A review of internet of things technologies for ambient assisted living environments. *Future Internet* **2019**, *11*, 259. [CrossRef]

3. Lauraitis, A.; Maskeliūnas, R.; Damaševičius, R.; Połap, D.; Woźniak, 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. Bhat, S.; Mehbodniya, A.; Al Wakeel A.; Webber, J.; Al Begain, K. Human Motion Patterns Recognition based on RSS and Support Vector Machines. In Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC20), Seoul, Korea, 25–28 May 2020.
5. Qiu, S.; Zhao, H.; Jiang, N.; Wang, Z.; Liu, L.; An, Y.; Zhao, H.; Liu, R.; Fortino, G. Multi-sensor information fusion based on machine learning for real applications in human activity recognition: State-of-the-art and research challenges. *Inf. Fusion* **2022**, *80*, 241–265. [CrossRef]
6. Esfahani, P.; Malazi, H. PAMS: A new position-aware multi-sensor dataset for human activity recognition using smartphones. In Proceedings of the 2017 19th International Symposium on Computer Architecture and Digital Systems (CADS), Kish Island, Iran, 21–22 December 2017; pp. 1–7.
7. Coskun, D.; Incel, O.D.; Ozgovde, A. Phone position/placement detection using accelerometer: Impact on activity recognition. In Proceedings of the 2015 IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), Singapore, 7–9 April 2015; pp. 1–6.
8. Hassan, M.; Uddin, Z.; Mohamed, A.; Almogren, A. A robust human activity recognition system using smartphone sensors and deep learning. *Future Gener. Comput. Syst.* **2018**, *81*, 307–313. [CrossRef]
9. Zhu, R.; Xiao, Z.; Cheng, M.; Zhou, L.; Yan, B.; Lin, S.; Wen, H. Deep Ensemble Learning for Human Activity Recognition Using Smartphone. In Proceedings of the 2018 IEEE 23rd International Conference on Digital Signal Processing (DSP), Shanghai, China, 19–21 November 2018; pp. 1–5.
10. Alemayoh, T.; Lee, J.H.; Okamoto, S. Deep Learning Based Real-time Daily Human Activity Recognition and Its Implementation in a Smartphone. In Proceedings of the 16th International Conference on Ubiquitous Robots (UR), Jeju, Korea, 24–27 June 2019; pp. 179–182.
11. Yin, X.; Shen, W.; Samarabandu, J.; Wang, X. Human activity detection based on multiple smart phone sensors and machine learning algorithms. In Proceedings of the IEEE 19th International Conference on Computer Supported Cooperative Work in Design (CSCWD), Calabria, Italy, 6–8 May 2015; pp. 582–587.
12. Dogan, G.; Cay, I.; Ertas, S.; Keskin, S.; Alotaibi, N.; Sahin, E. Where Are You? Human Activity Recognition with Smartphone Sensor Data. In Adjunct Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2020 ACM International Symposium on Wearable Computers, Virtual Event, Mexico, 12–16 September 2020; pp. 301–304.
13. Mafrur, R.; Nugraha, I.; Choi, D. Modeling and discovering human behavior from smartphone sensing life-log data for identification purpose. *Hum.-Centric Comput. Inf. Sci.* **2015**, *5*, 1–18. [CrossRef]
14. Procházka, A.; Vyšata, O.; Charvátová, H.; Vališ, M. Motion Symmetry Evaluation Using Accelerometers and Energy Distribution. *Symmetry* **2019**, *11*, 871. [CrossRef]
15. Saeed, A.; Salim, F.; Ozcelebi, T.; Lukkien, J. Federated Self-Supervised Learning of Multi-Sensor Representations for Embedded Intelligence. *IEEE Internet Things J.* **2020**, *8*, 1030–1040. [CrossRef]
16. Bayat, A.; Pomplun, M.; Tran, D.A. A study on human activity recognition using accelerometer data from smartphones. *Procedia Comput Sci.* **2014**, *34*, 450–457. [CrossRef]
17. Webber, J.; Mehbodniya A.; Fahmy, G. Human Motion Identity using Machine Learning on Spectral Analysis of RSS Signals. In Proceedings of the IEEE International Conference on Computer and Communications (ICCC'20), Chengdu, China, 11–14 December 2020; pp. 1405–1409.
18. Mehmood, A.; Khan, M.A.; Sharif, M.; Khan, S.A.; Shaheen, M.; Saba, T.; Riaz, N.; Ashraf, I. Prosperous human gait recognition: An end-to-end system based on pre-trained CNN features selection. *Multimed. Tools Appl.* **2020**, *79*, 1–21. [CrossRef]
19. Zheng, S.; Zhang, J.; Huang, K.; He, R.; Tan, T. Robust view transformation model for gait recognition. In Proceedings of the 2011 18th IEEE International Conference on Image Processing, Brussels, Belgium, 11–14 September 2011; pp. 2073–2076.
20. Saleem, F.; Khan, M.A.; Alhaisoni, M.; Tariq, U.; Armghan, A.; Alenezi, F.; Choi, J.I.; Kadry, S. Human gait recognition: A single stream optimal deep learning features fusion. *Sensors* **2021**, *21*, 7584. [CrossRef]
21. Elmadany, N.E.D.; He, Y.; Guan, L. Information fusion for human action recognition via biset/multiset globality locality preserving canonical correlation analysis. *IEEE Trans. Image Process.* **2018**, *27*, 5275–5287. [CrossRef]
22. Yun, K.; Honorio, J.; Chattopadhyay, D.; Berg, T.L.; Samaras, D. Two-person interaction detection using body-pose features and multiple instance learning. In Proceedings of the 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Providence, RI, USA, 16–21 June 2012; pp. 28–35.
23. Lai, J.; Yang, Z.; Guo, B. A Two-Stage Low-Complexity Human Sleep Motion Classification Method Using IR-UWB. *IEEE Sens. J.* **2021**, *21*, 20740–20749. [CrossRef]
24. Yang, R.; Liu, J.; Deng, X.; Zheng, Z. A Low Complexity Long Short-Term Memory Based Voice Activity Detection. In Proceedings of the IEEE 22nd International Workshop on Multimedia Signal Processing (MMSP), Tampere, Finland, 21–24 September 2020; pp. 1–6.

25. Nazari, N.; Mirsalari, S.; Sinaei, S.; Salehi, M.; Daneshtalab, M. Multi-level Binarized LSTM in EEG Classification for Wearable Devices. In Proceedings of the 28th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP), Västerås, Sweden, 11–13 March 2020; pp. 175–181.
26. Jo, J.; Kung, J.; Lee, Y. Approximate LSTM Computing for Energy-Efficient Speech Recognition. *Electronics* **2020**, *9*, 2004. [CrossRef]
27. Mirza, A.; Kerpicci, M.; Kozat, S.S. Efficient Online Learning with Improved LSTM Neural Networks. *Digit. Signal Process.* **2020**, *102*, 102742. [CrossRef]
28. Liu, Y.; Chiueh, T. Low-Complexity LSTM Training and Inference with FloatSD8 Weight Representation. *arXiv* **2020**, arXiv:2001.08450.
29. Hussain, G.; Jabbar, M.S.; Cho, J.D.; Bae, S. Indoor positioning system: A new approach based on lstm and two stage activity classification. *Electronics* **2019**, *8*, 375. [CrossRef]
30. Afza, F.; Khan, M.A.; Sharif, M.; Kadry, S.; Manogaran, G.; Saba, T.; Ashraf, I.; Damaševičius, R. A framework of human action recognition using length control features fusion and weighted entropy-variances based feature selection. *Image Vis. Comput.* **2021**, *106*, 104090. [CrossRef]
31. Chen, C.; Jafari, R.; Kehtarnavaz, N. A real-time human action recognition system using depth and inertial sensor fusion. *IEEE Sens. J.* **2015**, *16*, 773–781. [CrossRef]
32. Şengül, G.; Ozcelik, E.; Misra, S.; Damaševičius, R.; Maskeliūnas, R. Fusion of smartphone sensor data for classification of daily user activities. *Multimed. Tools Appl.* **2021**, *80*, 33527–33546. [CrossRef]
33. He, D.; Li, F.; Zhao, Q.; Long, X.; Fu, Y.; Wen, S. Exploiting spatial-temporal modelling and multi-modal fusion for human action recognition. *arXiv* **2018**, arXiv:1806.10319.
34. Duin, R.P.; Tax, D.M. Experiments with classifier combining rules. In *International Workshop on Multiple Classifier Systems*; Springer: Berlin/Heidelberg, Germany, 2000; pp. 16–29.
35. Saha, J.; Chowdhury, C.; Roy Chowdhury, I.; Biswas, S.; Aslam, N. An ensemble of condition based classifiers for device independent detailed human activity recognition using smartphones. *Information* **2018**, *9*, 94. [CrossRef]
36. Bosch Sensortec GmbH. BMI055-Small, Versatile 6DoF sensor module. *Doc. BST-BMI0555-ds000 Data sheet.* November 2021, v1.4, pp. 1–151. Available online: https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bmi055-ds000.pdf (accessed on 3 January 2022)
37. Hochreiter, S.; Schmidhuber, J. LSTM can solve hard long time lag problems. In *Advances in Neural Information Processing Systems*; A Bradford Book; The MIT Press: Cambridge, MA, USA, 1997, pp. 473–479.
38. Gokalgandhi, D.; Kamdar, L.; Shah, N.; Mehendale, N. A Review of Smart Technologies Embedded in Shoes. *J. Med. Syst.* **2020**, *44*, 1–9. [CrossRef]
39. Kingma, D.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
40. Wichrowska, O.; Maheswaranathan, N.; Hoffman, M.; Colmenarejo, S.; Denil, M.; Freitas, N.; Sohl-Dickstein, J. Learned optimizers that scale and generalize. In Proceedings of the International Conference on Machine Learning (PMLR), Sydney, Australia, 6–11 August 2017; pp. 3751–3760.
41. Wilson, A.; Roelofs, R.; Stern, M.; Srebro, N.; Recht, B. The marginal value of adaptive gradient methods in machine learning. *arXiv* **2017**, arXiv:1705.0829.
42. Hassan, M.F.; Abdel-Qader, I. Performance analysis of majority vote combiner for multiple classifier systems. In Proceedings of IEEE 14th International Conference on Machine Learning and Applications (ICMLA), Miami, FL, USA, 9–11 December 2015; pp. 89–95.
43. Polikar, R. Ensemble based systems in decision making. *IEEE Circuits Syst. Mag.* **2006**, *6*, 21–45. [CrossRef]
44. Webber, J.; Suga, N.; Mehbodniya, A.; Yano, K.; Suzuki, Y. Study on channel prediction for automated guided vehicle using a probabilistic neural network. *IEICE Commun. Express* **2019**, *8*, 311–317. [CrossRef]
45. El-Sheimy, N.; Hou, H.; Niu, X. Analysis and modeling of inertial sensors using Allan variance. *IEEE Trans. Instrum. Meas.* **2007**, *57*, 140–149. [CrossRef]
46. González-Cañete, F.J.; Casilari, E. Consumption Analysis of Smartphone based Fall Detection Systems with Multiple External Wireless Sensors. *Sensors* **2020**, *20*, 622. [CrossRef]
47. Sak, H.; Senior, A.; Beaufays, F. Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition. *arXiv* **2014**, arXiv:1402.1128.