applied sciences

MDPI

Article

A Pilot Study of the Efficiency of LSTM-Based Motion Classification Algorithms Using a Single Accelerometer

Kyu-Young Kang [1,†], Seul-Gi Lee [1,†], Hyeon Kang [1], Jung-Gil Kim [1], Gye-Rae Tack [1,2] and Jin-Seung Choi [1,2,*]

[1] Department of Biomedical Engineering, Konkuk University, Chungju 27478, Korea; best335@hanmail.net (K.-Y.K.); delicament@nate.com (S.-G.L.); jindokang@kku.ac.kr (H.K.); junggill0219@gmail.com (J.-G.K.); grtack@kku.ac.kr (G.-R.T.)
[2] Research Institute of Biomedical Engineering, Konkuk University, Chungju 27478, Korea
* Correspondence: jschoi98@kku.ac.kr
† These authors contributed equally to this work.

Abstract: Inertial sensors are widely used for classifying the motions of daily activities. Although hierarchical classification algorithms were commonly used for defined motions, deep-learning models have been used recently to classify a greater diversity of motions. In addition, ongoing studies are actively investigating algorithm efficiency (e.g., training time and accuracy). Thus, a deep-learning model was constructed in this study for the classification of a given motion based on the raw data of inertial sensors. Furthermore, the number of epochs (150, 300, 500, 750, and 900) and hidden units (100, 150, and 200) were varied in the model to determine its efficiency based on training time and accuracy, and the optimum accuracy and training time was determined. Using a basic long short-term memory (LSTM), which is a neural network known to be suitable for sequential data, the data classification training was conducted on a common desktop PC with typical specifications. The results show that the accuracy was the highest (99.82%) with 150 hidden units and 300 epochs, while the training time was also relatively short (78.15 min). In addition, the model accuracy did not always increase even when the model complexity was increased (by increasing the number of epochs and hidden units) and the training time increased as a consequence. Hence, through suitable combinations of the two factors that constitute deep-learning models according to the data, the potential development and use of efficient models have been verified. From the perspective of training optimization, this study is significant in having determined the importance of the conditions for hidden units and epochs that are suitable for the given data and the adverse effects of overtraining.

Keywords: inertial sensor; long short-term memory; motion classification; efficiency

check for updates

Citation: Kang, K.-Y.; Lee, S.-G.; Kang, H.; Kim, J.-G.; Tack, G.-R.; Choi, J.-S. A Pilot Study of the Efficiency of LSTM-Based Motion Classification Algorithms Using a Single Accelerometer. *Appl. Sci.* **2022**, *12*, 7243. https://doi.org/10.3390/app12147243

Academic Editors: Hanatsu Nagano and Claudio Belvedere

Received: 11 April 2022
Accepted: 17 July 2022
Published: 19 July 2022

1. Introduction

Inertial sensors are useful for measuring and classifying motions because they are convenient to use during daily activities. They are also advantageous because they place few constraints on the environment and can be used over relatively long periods of time. In general, a commercial inertial sensor system used for monitoring motion in daily life includes software with its own embedded motion classification algorithm, and the system uses a sensor attached to a fixed location on the body of the subject to perform the monitoring (i.e., fall detection or daily life monitoring for older adults). The advantages of this approach include the relatively accurate classifications of a set of motions predefined in the system, and the classification results can be used as indicators to evaluate and compare the accuracy of each sensor [1]. However, the possible types of motion in daily life are highly varied and depending on a study's purpose, the motion to be detected may be diverse. Therefore, the algorithms of such software cannot be revised by the investigator to restrict the classification to a predefined set of motions. Moreover, the software is limited because of its inability to classify more varied types of motion.

Recently, a wide range of studies has been conducted based on deep-learning models, including studies on motion classification referred to as human activity recognition (HAR) using inertial sensor data. Various models have been tried by many researchers, but the convolutional neural network (CNN), recurrent neural network (RNN) including long short-term memory (LSTM), and hybrid methods are representative of motion classification using inertial sensors depending on the data (Figure 1). In the study of Li et al. (2019), a CNN model was used to develop algorithms that can classify the motions of unrelated activities performed by the driver of a car with a free hand while driving [2]. For five motions, from taking a phone call to text messaging and eating food, the reported accuracy was 87%. In the study of Rivera et al. (2017), an RNN model was used to develop algorithms that can classify hand movements related to six motions including door opening and door closing, and the reported accuracy was 80.09% in the absence of noise and 74.92% in the presence of noise [3]. Kreuzer et al. (2017) developed algorithms that can classify the steps in human gait as five motions, from the loading response to the swing phase [4]. The LSTM neural network was used, and its mean classification accuracy was shown to be 92%. As such, various attempts have been to develop a suitable neural network for effective data classification, with reported accuracies in the range of 80–90%.

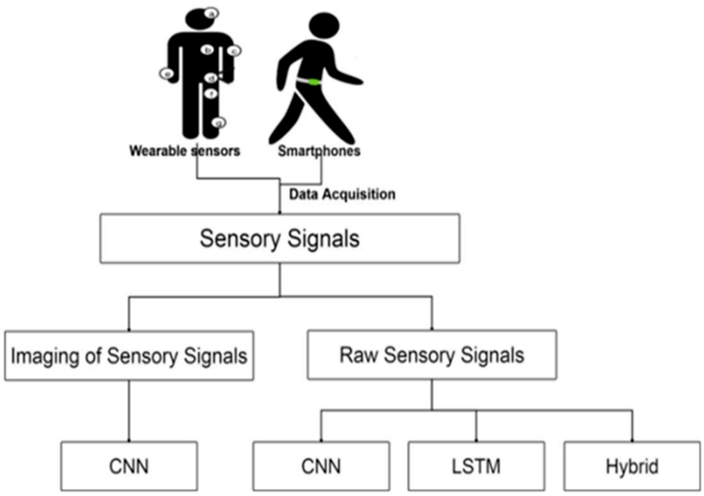

Figure 1. Categorization of proposed deep-learning models [5].

The inertial sensor data used for such motion classification is mainly utilized in the form of image-based post-processed data and time series raw data. In the case of image-based data, CNN models are mainly used, and for time series raw data, CNN or LSTM is used according to the judgment of the researcher. A hybrid model using both of these models is also used depending on the analysis data and the purpose [5]. Since the CNN model operates based on image classification, it requires high computational resources according to feature extraction through relatively numerous filters and mappings [6,7]. On the other hand, the LSTM model is highly competent for classification using raw sequences of time series signals [8]. In addition, LSTM is a model that solves the vanishing gradient problem in which the RNN does not remember the preceding value as the number of input increases, and it shows excellent performance when modeling long-term data such as time series data [9]. In the case of the hybrid model, there have been reports of improved accuracy compared to the single model (i.e., CNN or LSTM) depending on the conditions, but since the CNN layer is included, more training time and computational resources are required [10]. Therefore, in this study, since time-sequential inertial sensor data is used to minimize training time and computing resources, it is judged that the use of a single LSTM model using time series data is more suitable than a hybrid of CNN based on spatial correlation.

The development of motion classification algorithms based on deep-learning models generally requires a long training time and high-specification PC. The model type and architecture as well as the number of training iterations also exert a considerable influence on the outcome.

Factors affecting the accuracy of motion classification can be very diverse, not only by the type of model but also by data acquisition methods (number and attachment locations of sensors, sampling rate, etc.), pre-processing, etc. [11,12]. However, if we look only at the application of the same model under the same conditions, the main factors influencing the training time and accuracy of deep-learning models are the number of hidden units and epochs. An epoch is a single training iteration of the overall data, and an increase in the number of epochs corresponds to an increase in the number of training iterations, and hence an increase in the accuracy of the deep-learning model [13]. A hidden unit acts as a neuron in the hidden layer and performs data computation. An increase in the number of hidden units indicates an increase in the level of computation and hence an increase in the accuracy; however, the training time per epoch also increases [14]. Thus, in order to construct an efficient deep-learning model, it is necessary to properly control these two factors: the numbers of epochs and hidden units.

In this study, motion classification was performed using low-sampling single accelerometer data using a deep-learning model on a common PC with typical specifications. In addition, the training time and classification accuracy according to the adjustment of the number of epochs and hidden units were compared.

2. Methods

2.1. Experimental Procedures and Data Collection

The participants in this study consisted of 24 healthy undergraduates in their 20s, and among them, data were damaged in 2 of them, so data from a total of 22 people were used for training (14 males and 8 females, age 24.1 ± 1.7 years, height 171.2 ± 10.8 cm, weight 70.2 ± 15.5 kg). For data collection, ActivPAL4 (PAL Technologies Ltd., Glasgow, UK), an accelerometer sensor that can be attached to the middle of the right thigh, was used (Figure 2). The size of the sensor was 23.5 mm (wide), 43 mm (long), 5 mm (thick), and weighed 9.5 g. Additionally, it had a measurement range of −4 g (gravitational acceleration) to 4 g, and data was stored in 10 bits for each sample. Double-sided tape was used between the sensor and the skin for fixing the sensor, and a nonwoven tape wider than the sensor size was additionally covered and fixed on the attached sensor. The 3-axis acceleration data were recorded at a 20 Hz sampling frequency.

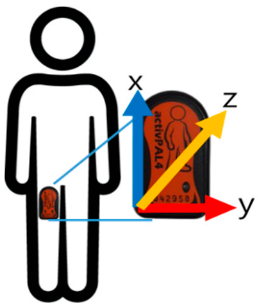

Figure 2. Sensor attachment position (middle of right thigh) and coordinate axis.

All participants were instructed to perform approximately 5 min of each of the seven randomly assigned motions consistently (lying, running, sitting on a chair, sitting on the floor, climbing stairs, chasing each other, and walking). In the case of walking, running, and climbing stairs, all subjects were required to perform each motion at a predetermined (limited) speed. Table 1 shows the description of the actions to be performed. For the data of each motion, 150 s (i.e., 3000 frames) of data in the middle were used for analysis to exclude the transition part out of the total of 5 min (300 s). In other words, 21,000 * 3 matrix data consisting of 3-axis (x, y, and z-axis) data of 7 motions with 3000 frames each were used for analysis after each motion labeling.

Table 1. Motions used for the classification and their descriptions.

Motion	Description
Lying	lying on the floor with their arms and legs outstretched in a supine position
Running	running at their own pace on a flat, curved track
Sitting on a chair	sitting on a chair without any movement
Sitting on the floor	sitting cross-legged position on the floor without any movement
Stairs	going up and down the 21 stairs in succession
Chasing each other	a game in which multiple participants run around with tails on their backs and grab each other's tails
Walking	walking at their own pace on a flat, curved track

Figure 3 shows the changes along the three-axis (x, y, z) acceleration for each measured motion. The dynamic and static motions are shown in sequence from the first motion of walking, sitting on a chair, running, sitting on the floor, climbing stairs, lying, and chasing each other.

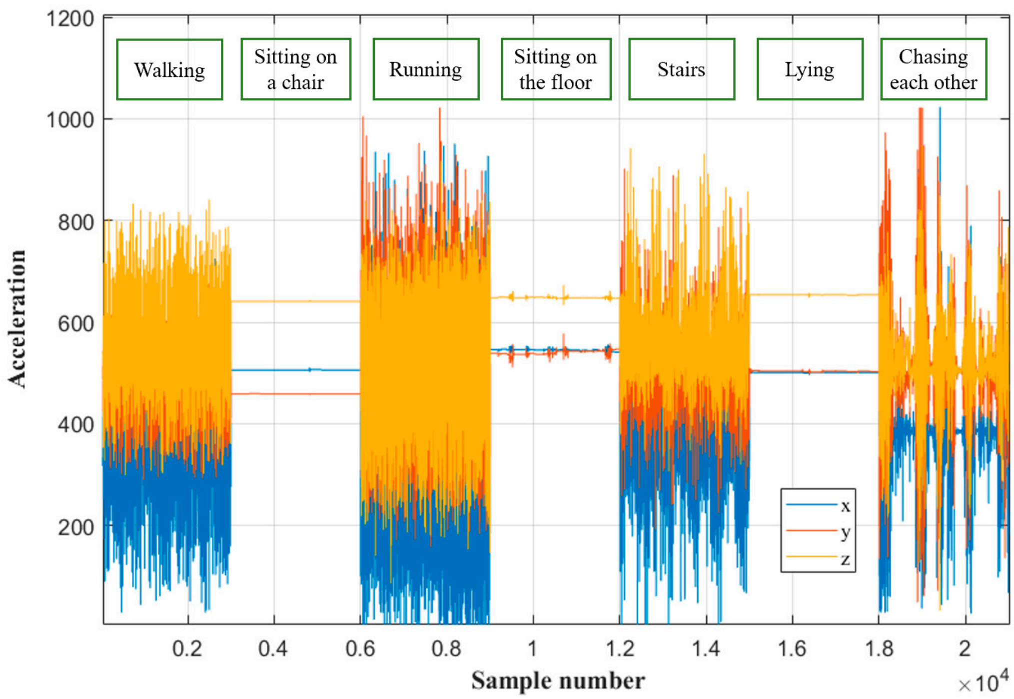

Figure 3. The example of three-axis (x, y, z) data of the acceleration sensor for each motion. In this case, the value on the vertical axis of the figure is the rescaled value of the acceleration and is expressed as a change of 170 mV per 1 g based on approximately 510 mV (0 g). For example, 680 mV = +1 g, 340 mV = −1 g, etc.

2.2. Deep-Learning Model Construction

For the classification of motion, the LSTM deep-learning model was used, and the deep-learning code provided by the MATLAB (R2021a) was applied.

The hidden layer of the deep-learning model used in this study was constructed as shown in Figure 4. The LSTM layer was divided into an encoder and a decoder for the sequence-to-sequence technique, which enables independent predictions for each time step of the sequence data. The fully connected layer performs numerous computations because all the neurons of one layer are connected to all the neurons of the next layer. This data preprocessing step aligns the data in one dimension, whereas the complete connection leads to higher accuracy (output) than that of an incomplete connection [15]. The softmax layer is where the class probability is determined. Using input with dimensions based on the number of input classes enables the probability a given data instance belongs to each dimension to be estimated. In this study, seven class inputs in total were provided for the motion, and the probability of each data instance signifying each of the motions was determined. The probability of each motion determined in the softmax layer was used to classify the data in the classification layer. Lastly, the motion of the input was determined as the output.

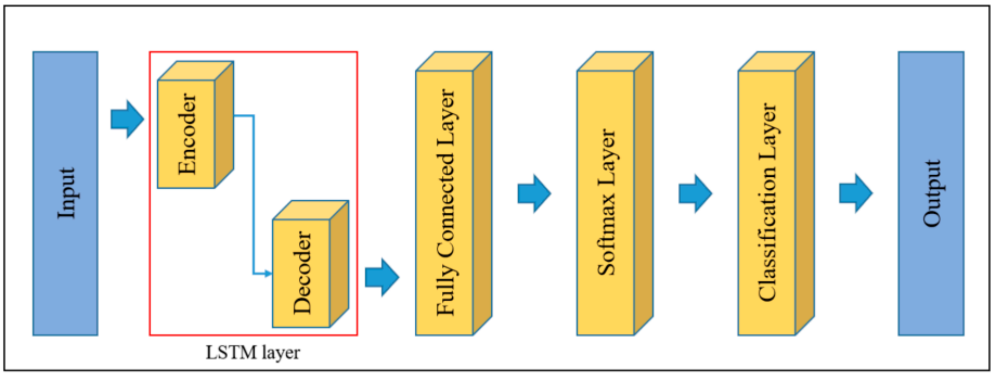

Figure 4. Model construction.

The number of hidden units (100, 150, and 200) and the number of epochs (150, 300, 500, 750, and 900), which is the number of training iterations, were varied, and the accuracy and training time for each case were determined. In the training set for model training, the data from 60% of the participants were used. In the test set for verifying the model accuracy, the data from 30% of the participants were used. In the validation set for verifying the accuracy during training, the data from 10% of the participants were used. For the model training and operation, a desktop PC with conventional performance (CPU: i5-9400f 4.1 GHz, GPU: RTX 2060, and RAM: 16 GB) was used. Simultaneous work of other applications was not performed during model training, but the internet connection and the operation of Windows were not restricted.

3. Results

The accuracy and training time for the motion classification models according to the numbers of hidden units and epochs used during training are presented in Tables 2 and 3, respectively, whereas the visual representations are given in Figures 5 and 6. The accuracy increased as the number of epochs increased from 150 to 300 regardless of the number of hidden units. Depending on the conditions, the accuracy was the highest at approximately 300 and 500 epochs and after that, it showed a general decreasing trend. A detailed look at the training model accuracy in each condition reveals that the accuracy was the highest at 300 epochs for hidden units 150 and 200 and at 500 epochs for hidden unit 100. For 150 hidden units, the accuracy after 750 epochs slightly increased although a decreasing trend was found overall. The training time of each model increased from a minimum of 28.1 min (100 hidden units and 150 epochs) to a maximum of 320 min (200 hidden units and 900 epochs), which verifies the increase in training time in relation to increases in the numbers of hidden units and epochs. However, across all conditions, the model with the highest accuracy (99.82%) had 150 hidden units and 300 epochs and took 78.15 min to train.

Table 2. Training accuracy according to numbers of epochs and hidden units.

Training Accuracy (%)		Epoch				
		150	300	500	750	900
Hidden unit	100	93.46%	98.02%	98.50%	94.86%	90.53%
	150	96.19%	99.82%	95.32%	98.58%	97.54%
	200	95.94%	98.90%	95.98%	94.74%	90.99%

Table 3. Training time according to numbers of epochs and hidden units.

Training Time (min)		Epoch				
		150	300	500	750	900
Hidden unit	100	28.10	50.37	81.01	120.06	148.37
	150	49.41	78.15	112.39	202.35	231.14
	200	66.29	99.02	183.43	276.18	320.01

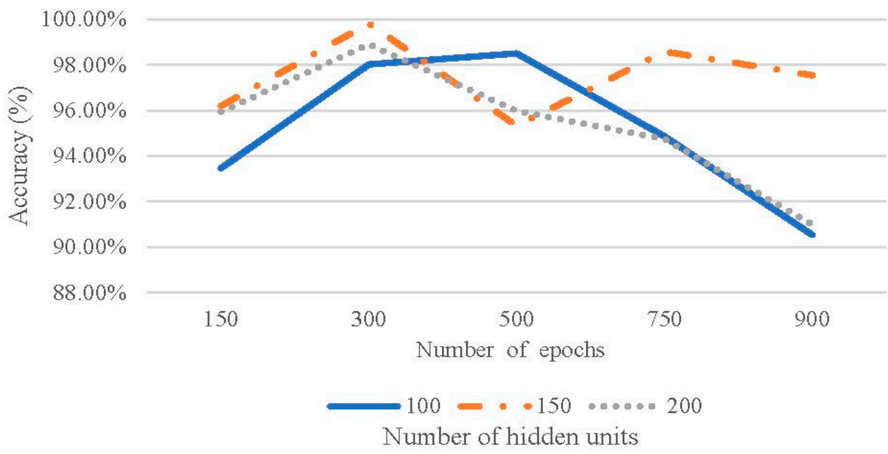

Figure 5. Accuracy results for the number of hidden units and epochs.

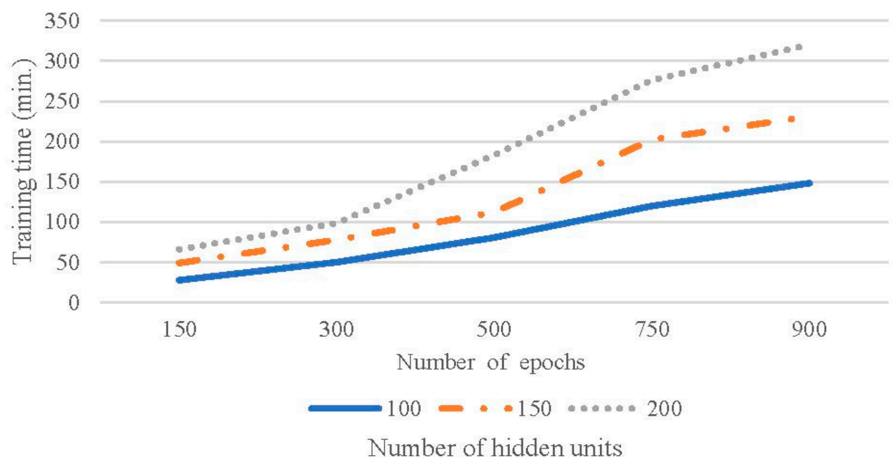

Figure 6. Training time results for the number of hidden units and epochs.

Figure 7 presents the training progression for 900 epochs and 100 hidden units. The accuracy graph shows the accuracy according to the increase in epochs, where the solid line indicates the training accuracy for the training data and the dotted line indicates the validation accuracy. The loss graph shows the error (or loss) based on the increase in epochs, where the solid line indicates the loss according to the training progression and the dotted line indicates the validation loss; here, lower values indicate better training performance, in contrast to accuracy. On the accuracy graph, increases in the number of epochs first led to increases in training accuracy and validation accuracy. Then, after a certain level, a fall in validation accuracy was observed. On the loss graph, an interval of increases in validation loss was observed, after which a gap between the training accuracy and loss appeared. The validation accuracy was the highest at approximately 400 epochs in the 900 epochs and 100 hidden units' condition.

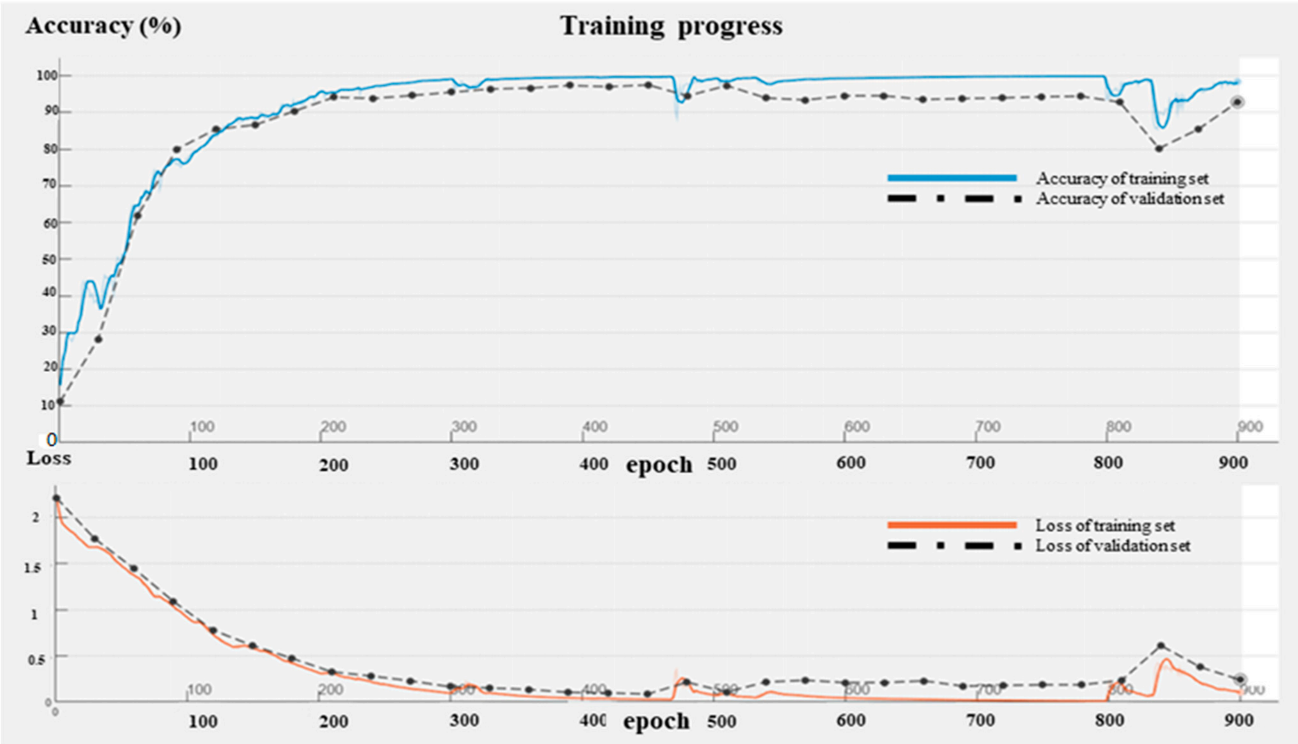

Figure 7. An example of a graph of training progression for 900 epochs and 100 hidden units.

Tables 4 and 5 present the confusion matrices for 150 hidden units and 300 epochs and for 200 hidden units and 900 epochs, respectively. The rows in each table indicate the classification result, and the columns indicate the actual class. The diagonal cells of the matrix indicate the successful classifications, whereas the non-diagonal cells indicate misclassifications. A greater number of errors were found in the training results in Table 5 than in Table 4, despite the higher numbers of epochs and hidden units.

Table 4. Confusion matrix for 150 hidden units and 300 epochs.

	Lying	**Running**	**Sitting on a Chair**	**Sitting on the Floor**	**Climbing Stairs**	**Chasing Each Other**	**Walking**
Lying	20,981	-	-	7	-	11	-
Running	-	20,948	-	17	-	7	4
Sitting on a chair	-	4	20,991	-	-	-	9
Sitting on the floor	-	2	-	20,852	50	-	-
Climbing stairs	19	10	-	124	20,950	22	-
Chasing each other	-	17	-	-	-	20,960	-
Walking	-	19	9	-	-	-	20,987
Correct	99.999%	99.998%	99.999%	99.993%	99.998%	99.998%	99.999%
Incorrect	0.001%	0.002%	0.001%	0.007%	0.002%	0.002%	0.001%

Table 5. Confusion matrix for 200 hidden units and 900 epochs.

	Lying	Running	Sitting on a Chair	Sitting on the Floor	Climbing Stairs	Chasing Each Other	Walking
Lying	16,862	-	-	-	2	4	-
Running	-	20,980	-	499	891	10	9
Sitting on a chair	-	4	20,991	-	-	-	16
Sitting on the floor	-	-	-	18,931	21	-	1
Climbing stairs	47	-	-	268	16,569	2	-
Chasing each other	4091	15	-	1302	3517	20,984	18
Walking	-	1	9	-	-	-	20,956
Correct	80.296%	99.999%	99.999%	90.148%	78.9%	99.999%	99.998%
Incorrect	19.704%	0.001%	0.001%	9.852%	21.1%	0.1%	0.002%

4. Discussion

The purpose of this pilot study was to compare the training time of the LSTM model using a common PC with typical specifications. Specifically, a simplified experimental protocol was used to determine the effect of the number of epochs and hidden units, and only a single accelerometer with a low sampling frequency (20 Hz) was used. In addition, to minimize model complexity, a basic LSTM model was used.

The results of this study showed the model had the best accuracy within the range of 300–500 epochs for the three conditions of hidden units (100, 150, and 200), followed by a gradually decreasing trend (Table 2, Figure 5). Hence, the increase in training time based on the increase in the two factors (numbers of epochs and hidden units) did not always yield a high accuracy. The results reveal that, whereas the overall model accuracy varied according to the number of hidden units, the highest accuracy was found at approximately 250–400 epochs. This result was compared with the results of previous studies with similar conditions (regarding motion classification using the LSTM algorithm). Abbaspour et al. (2020) collected the acceleration data of six motions including smartphone-based walking and running, then checked the accuracy of the deep-learning model after training and subsequent motion classification [16]. Two methods were used for the algorithm: a CNN alone and a CNN combined with LSTM (CNN–LSTM). The accuracy of the two models was compared, and the CNN model achieved a 94% accuracy at 200 epochs whereas the CNN–LSTM model achieved a 97% accuracy at 250 epochs. For the CNN–LSTM model, the accuracy was higher than the CNN model despite the need for an additional 50 epochs. In the study of Kim et al. (2020), the participants were instructed to wear an inertial measurement unit (IMU) sensor and perform 14 motions in total [17]. During each motion, 18 datasets (including nine datasets of 3-axis acceleration, 3-axis gyroscope, and 3-axis magnetic field data) were obtained for use in deep learning. Moreover, the training was performed with 64 hidden units and 400 epochs. As a result, the loss was found to have adequately converged after 350 epochs, with a 94.73% accuracy, indicating adequate classification. In these previous studies, the accuracy was high in epoch ranges similar to those of the present study despite the variation in the number of hidden units and sizes of the total datasets.

The risk of overfitting was also confirmed through training progression and confusion matrix (Figure 7, Tables 4 and 5). Although the increase in model complexity and training frequency caused by the increases in hidden units and epochs resulted in the improvement of accuracy, overfitting occurred after a certain point, reducing accuracy. Thus, a confusion matrix was constructed to examine the relationship of misclassifications to different motions. A greater number of errors were found in the training results in Table 5 in comparison with Table 4, despite the higher numbers of epochs and hidden units. This was attributed to overfitting, which is the excess learning of the training data by the model, whereby errors occur when processing new data (i.e., the validation or test dataset) rather than the previously learned data (i.e., the training dataset). In other words, as a hidden unit is used in the internal computation of deep learning, this may increase structural complexity because of the increased number of computations per epoch in deep models with multiple hidden units, which has been reported to increase the probability of overfitting [18]. The overfitting may arise in the presence of too many epochs or hidden units. Table 5 shows that the motions lying, sitting on the floor, and climbing stairs had in fact been misclassified as chasing each other. In chasing each other, the variability of motion is high because the participant increases speed to catch the opponent's tail or protect his or her own tail from the opponent during slow movements. Hence, the variation in acceleration is substantially high (Figure 3) and was irregular across all participants so that an algorithm that overfit the training data could not adequately classify data with such high variability. In order to solve this problem, it is suggested that the number of attached sensors be increased [19], or an improved LSTM model such as a lightweight deep-learning model with reduced computational power and latency [20] or LSTM with Deep Q-Network [21] be attempted.

There were several limitations of this study. First, the sensor data used in this study was different from the generally collected motion data. When performing dynamic movements (i.e., walking, running, climbing stairs), the movements of all subjects were controlled at a constant speed for each movement. This may be different from the motion data collected in a daily living environment, and this was considered to be the reason for showing a fairly high accuracy even though a simple model was used in this study. Results of the model used in this study when validated using the WISDM dataset, which is public data measured with a position (right thigh), sampling (20 Hz), and single sensor (accelerometer) similar to this study [22], showed a classification accuracy of approximately 84%. Although the sensor used was different (not attached to the body by using the sensor built into the smartphone and with a different sensor measuring range), it was revealed that this is a lower value than that showing an accuracy of up to 90% or more in previous studies applying various models (Bi-directional LSTM, CNN, Convolutional LSTM, LSTM-RNN, etc.) [23–25]. This is a pilot study focusing only on the comparison of the accuracy and training time according to the change of the hidden unit and epoch of the basic LSTM model using a single acceleration sensor for arbitrary motions. Most of the previous studies suggest only the results according to the selected hidden unit and epoch for the deep-learning model, and these reports have limitations in direct discussion with the current results. However, approaches using various sensor combinations and improved models are in progress to improve classification accuracy [11,12,26–28], and it will be necessary to compare them through various sensor conditions, improved models, and data processing in terms of training efficiency in the future.

In the case of commercial activity classification software using a traditional IMU-based hierarchical algorithm such as using threshold, the accuracy of general activity classification is very high, but classification error occurs a lot when changing postures (i.e., transition) such as sit-to-stand [29,30]. In general, the application of deep-learning models is often applied to reduce such classification error, but in this study, a fragmentary comparison of training time and accuracy was performed using data for each activity except transition. Therefore, when classifying continuous motion in real life, the classification accuracy presented in this paper is predicted to significantly decrease. In the future, it is necessary to improve the model and apply various dynamic situations, and in this process, a comparison of the accuracy of the model through various combinations of the number and types of sensor data should be performed.

Additionally, the number of subjects used for training and validation of the model is very small. The accuracy varied according to the data constituting each item, even when the proportions of training, test, and validation datasets were equal. This is presumed to be due to the small sample size (n = 22) resulting in deviations according to the condition. Thus, a larger number of participants should be recruited for further studies to reduce such deviations. In addition, because the data in this study were collected solely from healthy undergraduates in their 20s, the classification of the motions of individuals in other age groups with varying physical functions by the trained model could lead to errors. Thus, further studies regarding the development of motion classification models should use an experimental design that reflects a wider range of participant ages.

5. Conclusions

The motion classification accuracy of the LSTM-based model was high, which implies that deep-learning models can learn to classify motion based on the form of data desired by the investigator in lieu of the conventional fixed set of motions of a sensor and its software. The training complexity increased as the number of hidden units increased so that the runtime of a single epoch increased, and as the number of epochs increased, the number of iterations increased. Hence, the increase in hidden units and epochs resulted in an increase in training time, as presented in Table 2 for actual conditions. However, the increase in model complexity and training time did not ensure an increase in accuracy (Table 3). Thus, before such problems arise, training should be terminated to improve the accuracy and construction of the algorithm more efficiently with a simultaneous fall in training time. The possibility of training in epoch ranges without overfitting or reduced accuracy was verified, even in a typical PC environment. The results of this study are anticipated to contribute to the development and use of efficient motion classification models.

Author Contributions: Conceptualization, J.-S.C. and G.-R.T.; methodology, K.-Y.K. and S.-G.L.; software, K.-Y.K.; validation, J.-G.K., H.K. and K.-Y.K.; formal analysis, K.-Y.K. and S.-G.L.; investigation, K.-Y.K. and J.-S.C.; writing—original draft preparation, K.-Y.K. and J.-S.C. writing—review and editing, K.-Y.K., S.-G.L. and J.-S.C.; visualization, K.-Y.K.; project administration, J.-S.C. and G.-R.T.; funding acquisition, J.-S.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2022R1A2C1010491).

Institutional Review Board Statement: The protocol of our study was confirmed by the Ethics Committee of Konkuk University (IRB No. 7001355-201507-HR-066).

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Data is available on reasonable request.

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

References

1. Steeves, J.A.; Bowles, H.R.; Mcclain, J.J.; Dodd, K.W.; Brychta, R.J.; Wang, J.; Chen, K.Y. Ability of Thigh-Worn ActiGraph and activPAL Monitors to Classify Posture and Motion. *Med. Sci. Sports Exerc.* **2015**, *47*, 952–959. [CrossRef] [PubMed]
2. Li, L.; Xieb, Z.; Liangc, Y.; Horreyd, W.; Xu, X. Recognition of manual driving distraction through deep-learning and wearable sensing. In Proceedings of the 10th International Driving Symposium on Human Factors in Driver Assessment, Training and Vehicle Design, Santa Fe, NM, USA, 24–27 June 2019; pp. 22–28.
3. Rivera, P.; Valarezo, E.; Choi, M.T.; Kim, T.S. Recognition of Human Hand Activities Based on a Single Wrist IMU Using Recurrent Neural Networks. *Int. J. Pharma Med. Biol. Sci.* **2017**, *6*, 114–118. [CrossRef]
4. Kreuzer, D.; Munz, M. Deep Convolutional and LSTM Networks on Multi-Channel Time Series Data for Gait Phase Recognition. *Sensors* **2021**, *21*, 789. [CrossRef] [PubMed]
5. Ramanujam, E.; Perumal, T.; Padmacathi, S. Human Activity Recognition with Smartphone and Wearable Sensors Using Deep Learning Techniques: A Review. *IEEE Sens. J.* **2021**, *21*, 13029–13040. [CrossRef]
6. Um, T.T.; Babakeshizadeh, V.; Kulic, D. Exercise motion classification from large-scale wearable sensor data using convolutional neural networks. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 2385–2390.
7. Wan, S.; Qi, L.; Xu, X.; Tong, C.; Gu, Z. Deep Learning Models for Real-time Human Activity Recognition with Smartphones. *Mob. Netw. Appl.* **2020**, *25*, 743–755. [CrossRef]
8. Ronao, C.A.; Cho, S.B. Human activity recognition with smartphone sensors using deep learning neural networks. *Expert Syst. Appl.* **2016**, *59*, 235–244. [CrossRef]

9. Kumar, S.; Sharma, A.; Tsunoda, T. Brain wave classification using long short-term memory network based OPTICAL predictor. *Sci. Rep.* **2019**, *9*, 5153. [CrossRef]
10. Rajagukguk, R.A.; Ramadhan, R.A.A.; Lee, H.J. A Review on Deep Learning Models for Forecasting Time Series Data of Solar Irradiance and Photovoltaic Power. *Energies* **2020**, *13*, 6623. [CrossRef]
11. Gu, F.; Chung, M.H.; Chignell, M.; Valaee, S.; Zhou, B.; Liu, X. A survey on deep learning for human activity recognition. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–34. [CrossRef]
12. Zhang, S.; Li, Y.; Zhang, S.; Shahabi, F.; Xia, S.; Deng, Y.; Alshurafa, N. Deep learning in human activity recognition with wearable sensors: A review on advances. *Sensors* **2022**, *22*, 1476. [CrossRef]
13. Manurung, A.H.; Budiharto, W.; Prabowo, H. Lgorithm and modeling of stock prices forecasting based on long short-term memory (LSTM). *ICIC Express Lett.* **2018**, *12*, 1277–1283.
14. Pei, W.; Dibeklioğlu, H.; Tax, D.M.J.; van der Maaten, L. Multivariate Time-Series Classification Using the Hidden-Unit Logistic Model. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 920–931. [CrossRef] [PubMed]
15. Zhao, J.; Deng, F.; Cai, Y.; Chen, J. Long short-term memory-Fully connected (LSTM-FC) neural network for PM2.5 concentration prediction. *Chemosphere* **2019**, *220*, 486–492. [CrossRef] [PubMed]
16. Abbaspour, S.; Fotouh, F.; Fotouhi, H.; Vahabi, M.; Linden, M. Deep Learning-Based Motion Activity Recognition Using Smartphone Sensors. *IADIS Int. Conf. E-Health* **2020**, *20*, 128–134.
17. Kim, K.; Cho, Y.K. Effective inertial sensor quantity and locations on a body for deep learning-based worker's motion recognition. *Autom. Constr.* **2020**, *113*, 103126. [CrossRef]
18. Bouktif, S.; Fiaz, A.; Ouni, A.; Serhani, M.A. Optimal Deep Learning LSTM Model for Electric Load Forecasting using Feature Selection and Genetic Algorithm: Comparison with Machine Learning Approaches. *Energies* **2018**, *11*, 1636. [CrossRef]
19. Chung, S.; Lim, J.; Noh, K.J.; Kim, G.; Jeong, H. Sensor data acquisition and multimodal sensor fusion for human activity recognition using deep learning. *Sensors* **2019**, *19*, 1716. [CrossRef]
20. Agarwal, P.; Alam, M. A lightweight deep learning model for human activity recognition on edge devices. *Procedia Comput. Sci.* **2020**, *167*, 2364–2373. [CrossRef]
21. Zhou, X.; Liang, W.; Kevin, I.W.K.; Wang, H.; Yang, L.T.; Jin, Q. Deep-learning-enhanced human activity recognition for Internet of healthcare things. *IEEE Internet Things J.* **2020**, *7*, 6429–6438. [CrossRef]
22. Weiss, G.M.; Yoneda, K.; Hayajneh, T. Smartphone and smartwatch-based biometrics using activities of daily living. *IEEE Access* **2019**, *7*, 133190–133202. [CrossRef]
23. Oluwalade, B.; Neela, S.; Wawira, J.; Adejumo, T.; Purkayastha, S. Human Activity Recognition using Deep Learning Models on Smartphones and Smartwatches Sensor Data. In Proceedings of the HEALTHINF 2021—14th International Conference on Health Informatics, Online, 11–13 February 2021.
24. Gupta, S. Deep learning based human activity recognition (HAR) using wearable sensor data. *Int. J. Inf. Manag. Data Insights* **2021**, *1*, 100046. [CrossRef]
25. Pienaar, S.W.; Malekian, R. Human Activity Recognition using LSTM-RNN Deep Neural Network Architecture. In Proceedings of the 2019 IEEE 2nd Wireless Africa Conference (WAC), Pretoria, South Africa, 18–20 August 2019; pp. 1–5.
26. Wang, J.; Chen, Y.; Hao, S.; Peng, X.; Hu, L. Deep learning for sensor-based activity recognition: A survey. *Pattern Recognit. Lett.* **2019**, *119*, 3–11. [CrossRef]
27. Straczkiewicz, M.; James, P.; Onnela, J.P. A systematic review of smartphone-based human activity recognition methods for health research. *NPJ Digit. Med.* **2021**, *4*, 1–15. [CrossRef] [PubMed]
28. Ferrari, A.; Micucci, D.; Mobilio, M.; Napoletano, P. Trends in human activity recognition using smartphones. *J. Reliab. Intell. Environ.* **2021**, *7*, 189–213. [CrossRef]
29. Aminian, S.; Hinckson, E.A. Examining the validity of the ActivPAL monitor in measuring posture and ambulatory movement in children. *Int. J. Behav. Nutr. Phys. Act.* **2012**, *9*, 119. [CrossRef]
30. Mitchell, T.; Borner, K.; Finch, J.; Kerr, J.; Carlson, J.A. Using Activity Monitors to Measure Sit-to-Stand Transitions in Overweight/Obese Youth. *Med. Sci. Sports Exerc.* **2017**, *49*, 1592–1598. [CrossRef]