

Received August 23, 2020, accepted September 8, 2020, date of publication September 14, 2020, date of current version September 25, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3024003

Design of a Wearable Wireless Multi-Sensor Monitoring System and Application for Activity Recognition Using Deep Learning

GOKMEN ASCIOGLU¹ AND YAVUZ SENOL²

¹Department of Electrical and Electronics Engineering, Graduate School of Natural and Applied Sciences, Dokuz Eylül University, 35390 Izmir, Turkey

²Department of Electrical and Electronics Engineering, Engineering Faculty, Dokuz Eylül University, 35390 Izmir, Turkey

Corresponding author: Gokmen Ascioğlu (gokmen.ascioglu@ogr.deu.edu.tr)

This work was supported in part by Dokuz Eylül University under Grant 2020.KB.FEN.011.

ABSTRACT Human activity monitoring and recognition systems assist experts in evaluating various health problems including obesity, cardiac diseases and, sports injury detection. However, these systems have two challenging points; monitoring activities for outdoor applications and extracting relevant features using hand-crafted techniques from multi-dimensional and large datasets. To address these challenges, we have focused on new dataset generation for activity recognition, a novel design of a sensor-based wireless activity monitoring system, and its application to deep learning neural networks. The designed monitoring system consists of one master and four slave devices, and can collect and record acceleration and gyroscope information. The slave devices were attached on arm, chest, thigh, and shank areas of the human body. Activity data were collected and recorded from sixty healthy people for thirteen activity types including drink from cup and cleaning table. These activities were divided into three activity categories as basic, complex, and all, which is the combination of basic and complex activities. Obtained datasets were fed into deep learning neural networks namely convolutional neural network (CNN), long-short term memory (LSTM) neural networks, and convolutional LSTM (ConvLSTM) neural networks. The performance of each neural network for each category type was separately examined. The results show that ConvLSTM outperforms CNN and LSTM as far as activity recognition is concerned.

INDEX TERMS Activity recognition, deep learning, wireless sensor network, wearable sensors, data augmentation.

I. INTRODUCTION

Activity monitoring and recognition are popular research fields in nowadays. They have great potential to improve life quality in the field of health [1]–[4]. For example, these systems can be used to observe older adults at home as healthcare or help physically impaired people in the process of treatment as rehabilitation assistance.

Activity monitoring can be performed using non-invasive or invasive methods [5]–[8]. The non-invasive method is based on computer vision which consists of one or more cameras. The invasive method relies on the data collected on the human body using various sensors such as accelerometer, gyroscope, and electromyography (EMG). Vision-based

The associate editor coordinating the review of this manuscript and approving it for publication was Lorenzo Mucchi.

systems have some disadvantages such as high cost and limited indoor use. Moreover, these systems cause the invasion of personal privacy [9]. Sensor-based systems including smartwatches and smartphones overcome many of these disadvantages [10], [11]. These systems are inexpensive, usable in both indoor and outdoor applications, and less privacy-intrusive.

With the advances in communication technologies, wearable wireless sensor networks (WSNs) have particularly gained importance for activity monitoring. The main advantages of the wearable WSNs are small size, lightweight, and portability. These networks consist of master and small battery-powered slave devices with limited computation and radio communication capabilities. Kouris and Koutsouris designed WSN, which is called as WISE, using Bluetooth technology. The sensor nodes transmit the collected 3-axis

accelerometer, walking speed, and heartbeat information to a smartphone [12]. Tufek *et al.* created WSN to record acceleration and gyroscope information on the human body using Zigbee technology [13]. Hossain *et al.* created a network using LoraWAN technology. Sensor nodes transmit the accelerometer data to LoraWAN gateway [14]. Balli *et al.* created WSN consists of smartwatches and smartphone. Smartwatches transfer the acquired accelerometer data to a smartphone for storage and processing [15].

The studies on activity recognition are performed using various traditional and modern machine learning methods [16]–[20]. Among the most popular traditional machine learning techniques are support vector machine (SVM), k-nearest neighbor (kNN), and artificial neural network (ANN). The common aspects of these machine learning methods are the lack of feature extraction capability. These methods require explicit human intervention in the form of feature extraction for training. However, it is known that manual feature extraction technique limits generalization capability since it is based on human experience and domain of knowledge. Furthermore, this process causes time-consuming [21]. Recently, deep learning (DL) branch of modern machine learning has become a popular choice as a powerful tool for activity recognition [22]–[24]. DL extracts features automatically through hierarchical architectures and offers superior performance. It does not need specialized knowledge and expertise.

In the literature, human activities are categorized with different concepts as basic and complex. Shoaib *et al.* [25] explain that basic activities are repetitive activities such as walking, writing, and sitting. Complex activities are not repetitive and may involve various hand gestures such as smoking and ascending stairs. Bharti *et al.* [26] categorize activities as stationary (e.g., sitting and standing), simple (e.g. ascending and descending stairs), and complex (e.g. cooking and cleaning). In this study, we have combined given these definitions and, commented that basic activities are repetitive motions and not include any hand gestures, whereas complex activities are repetitive or non-repetitive motions including hand gestures. According to our definition, walking, running, and jumping for basic activities, whereas smoking, writing, and drinking coffee for complex activities can be given as example.

There are numerous studies on human activity recognition using deep learning neural networks. Murad and Pyun [27] proposed a model LSTM based on deep recurrent neural network for basic and complex activity recognition. They used separately five public benchmark datasets recorded by wearable sensors. Experimental results show that their DL models outperform traditional machine learning methods such as SVM and kNN. San-Segunda *et al.* [28] compare performance between Hidden Markov Model (HMM) and CNN to classify basic activities. With this purpose, they collect the accelerometer signals from different smartwatches and smartphones and conclude that DL has better results than HMM. Jiang and Yin [29] enable Deep CNN to

automatically learn the optimal features from activity images consisting of accelerometer and gyroscope signal sequences. Their approach is evaluated on three public datasets for activity recognition. They present that deep learning outperforms Support Vector Machine (SVM) algorithm in terms of recognition accuracy and computational cost. Chen and Xue [30] construct a convolutional neural network for basic activity recognition. They create a dataset consists of 31688 label samples from eight typical activities. They make a performance comparison between CNN and SVM classifier with different features. Experimental results show that CNN gives better results than SVM.

In this study, a wearable wireless activity monitoring system was designed to collect and record 3-axis acceleration and 3-axis gyroscope information. Then, wireless sensors were placed on the arm, chest, shank, and thigh parts of the human body. Sensor outputs were collected and recorded by master device during various types of activity, including writing on paper and hand washing. Obtained data were fed into deep learning neural networks, including CNN, LSTM and ConvLSTM. To analyze network performances, we divided activities into three categories as basic, complex, and all. Then, the network performance was separately examined for each category.

This study includes additional activities compared to other benchmark datasets with respect to complex activity types. Moreover, the proposed wearable WSN enables the extension of the hardware system and can be used in the outdoor environment.

II. METHOD

A. SYSTEM DESIGN

To gather acceleration and gyroscope datasets, we designed a wireless system consisting of two main devices named as master and slave. The master device establishes and manages a wireless network, as well as records incoming signals from the slave devices. Slave devices connect to the network created by master device via Wi-Fi and send Inertial measurement unit (IMU) sensors data to the master. Fig. 1 shows a block diagram of the system.

Raspberry Pi Zero Wireless is the chosen master device due to its small size, low-cost, and internal Wi-Fi module. It supports Linux Operating System with readily available programmable tools required to operate the master device. For the slave devices, NodeMCU Wi-Fi development module was selected. It has a sufficiently small size and is low-cost. Furthermore, NodeMCU allows the utilization of built-in microcontroller of the Wi-Fi module, dispensing with the need for external microcontroller. Since NodeMCU development module has SPI and I2C bus line, it makes the IMU sensor (MPU6050) a native sensor with I2C features.

The slave level acquisition frequency was configured as 40Hz. Slave devices are powered by 900mAh LiPo batteries with a lifetime of about 6 hours on a full charge. When the battery switch is in “off” mode, external power can be

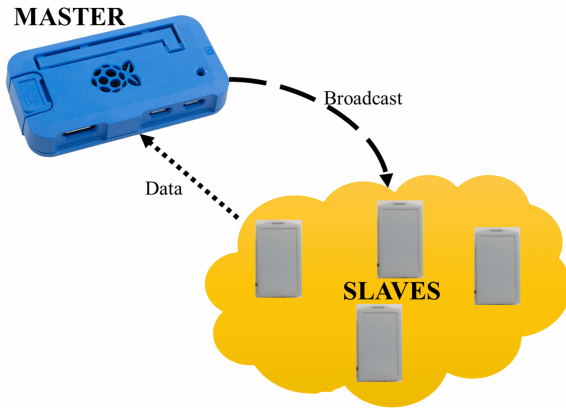


FIGURE 1. Block diagram of designed wireless monitoring system.

connected to charge on integrated circuit via the micro USB. Charging does not require batteries to be removed from its slot. Slave devices are shielded by a box to prevent hardware problems and isolate the circuit from users. The enclosure box has attached elastic rope and stopper. The rope and the stopper allow the user to comfortably mount and secure the sensor on the human body. Fig. 2 shows the hardware design of the monitoring system.

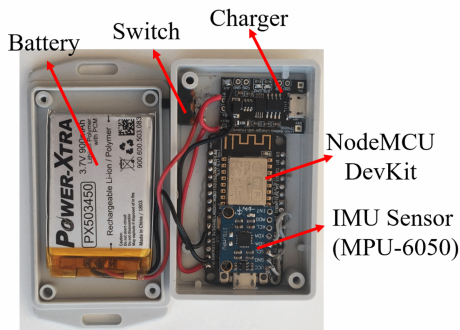


FIGURE 2. Hardware design of slave devices.

The Raspberry Pi Wi-Fi module generates a network to which slave nodes whose MAC addresses are registered can connect and are assigned static IP addresses. Master-slave communication employs User Datagram Protocol (UDP). The software on the master device (Python script) accepts interactively various management commands which can be submitted via a web interface in Fig. 3 that runs on the same wireless network. Management commands range from broadcasting option to viewing connected devices on the network.

When the battery switch set to “on” mode, the slave devices connect to the sensor network and await commands from the master device; this command is broadcast to all slaves. The master device periodically broadcasts counter to the slave devices, which prompt the slaves to collect and

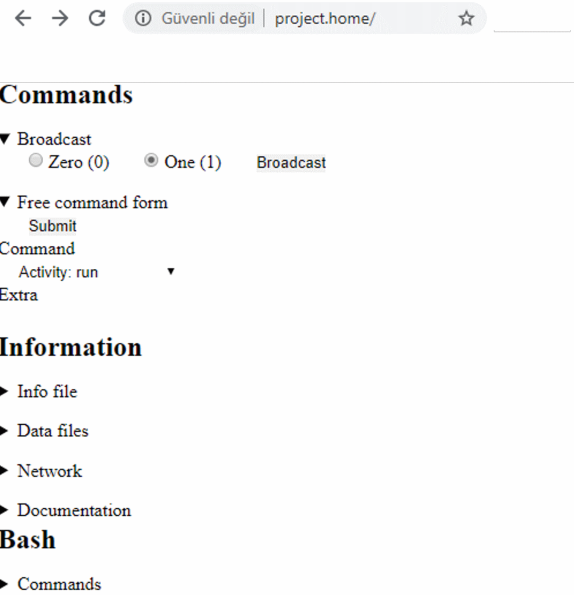


FIGURE 3. WSN management interface of designed monitoring system.

return sensor data. These broadcasts also serve to synchronize slave devices. Slave devices add data counter and checksum to each sensor data. Master device collects and after verifying the data records it in a file. Each sensor data is saved in a separate file defined by the global name specified by the user and the IP assigned to the slave devices.

Communication protocol between master and slave devices was chosen as IEEE 802.11n. The protocol has advantages such as high data rate and longer distance at indoor and outdoor when compared with 802.11 b/g.

B. DATA COLLECTION

Sensors were attached on the thigh, shank, arm and chest parts of the human body. Before the actual activity, each sensor is calibrated by gathering data while the subject is in a standing position. Only then acceleration and angular rate data sets were collected. Before sending data to the master device, raw acceleration and gyroscope data are divided by the sensitivity scale factor to get the value in terms of ‘g’ and ‘deg/sec’. The sensor placements are given in Fig. 4.

We collected data on 60 healthy users (37 males and 23 females) with age ranging from 20 to 40 (mean 25.9 ± 3.8), height 160-195 cm (mean 179 ± 3.0) and weight 50-103 kg (mean 76.8 ± 3.4). Each user was briefed on what the study is about and was asked to perform a total of thirteen different activities, which is combination of eight basic and five complex. Table 1 gives all activities and their abbreviations. Fig. 5 shows several pictures of the data collection moment for four different activity types, which is descending stairs, writing on paper, walking downhill, and running.

For each subject, 65-minutes of data were collected for thirteen different activities by recording 5 minutes of data per



FIGURE 4. Placements of sensor on the human body.

TABLE 1. Categories and activities.

	All Activities	Abbreviation
Basic Activities	Walking on flat ground	WG
	Walking uphill	WU
	Walking downhill	WD
	Ascending stairs	ASC
	Descending stairs	DES
	Running	RN
	Jumping	JP
	Sit down - Stand up	SS
Complex Activities	Cleaning table	CT
	Drink from cup	DC
	Open-close refrigerator	OCR
	Writing on paper	WP
	Hand washing	HW

activity. A capture of activity signals taken from thigh part of leg and arm during walking uphill and cleaning table is given in Fig. 6 and 7, respectively.

C. DATA PRE-PROCESSING

During activities, incoming accelerometer signals from sensors were noisy. To remove noise, moving average filter was used. The data range of accelerometer and gyroscope signals was not at the same scale. To uniform data range, min-max normalization was separately applied on each feature of the dataset. This is important to prevent dominant characteristics of large values according to fewer values during training. Then, the signals were divided into frames with non-overlapping sliding windows. The main challenge was to find the correct window size. In the literature, there are different approaches. For example; Karantonis *et al.* [31] used 1-second window size without overlapping, and Preece *et al.* [32] used 2-seconds window size with 1-second overlapping. The small window size affects adversely

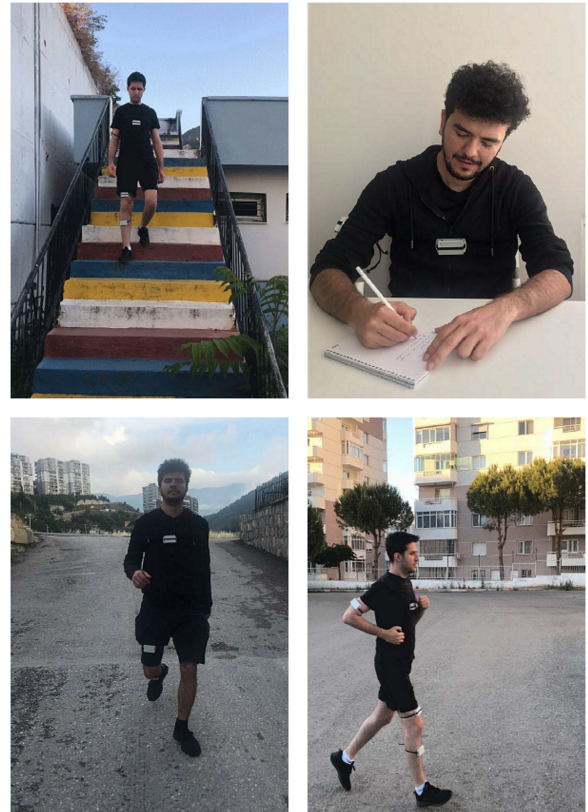


FIGURE 5. Data collection moment for descending stairs, writing on paper, walking downhill, and running.

computational cost, and wide window size may cause overfitting. In this study, we determined window size as 1.5-seconds by observing the average period duration of activities. Due to this selected window-size, we obtained total of 12000 frames for each activity and totally 156000 frames from all activities. Then, data augmentation was performed by adding noise into frames and number of frames was increased five times.

D. DEEP LEARNING APPLICATIONS

The produced frames were divided into two sections as 70% training and 30% testing. Then, the frames belong to each category fed into three different deep learning neural networks; CNN, LSTM, and ConvLSTM.

CNN is a feed-forward network with feature extraction layer [33], [34]. Overall structure has five layers, which consist of input, convolutional and pooling for feature extraction, fully-connected, softmax activation function, and classification as given in Fig. 8. In the convolutional layer, one or more filters (kernels), which slides in each direction with determined strides, are applied on input frames and produce many local features with convolution operations. To ensure nonlinearity, the RELU activation function is performed on extracted features. These are forwarded to the pooling layer for reducing the spatial dimension. The extracted features are fed into fully connected layers, which comprises one or more

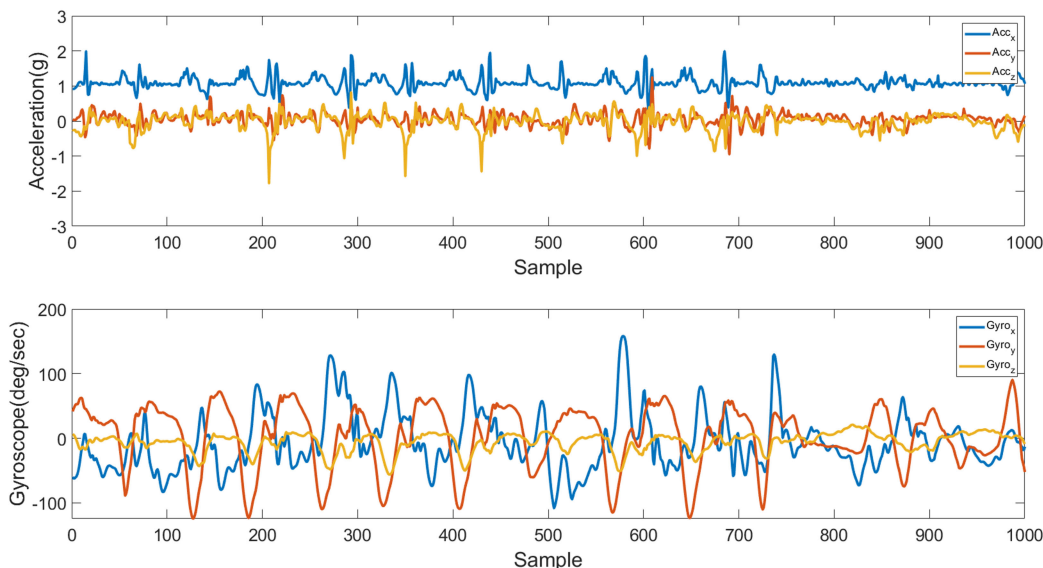


FIGURE 6. A capture of raw acceleration and gyroscope data for walking uphill from thigh part of leg.

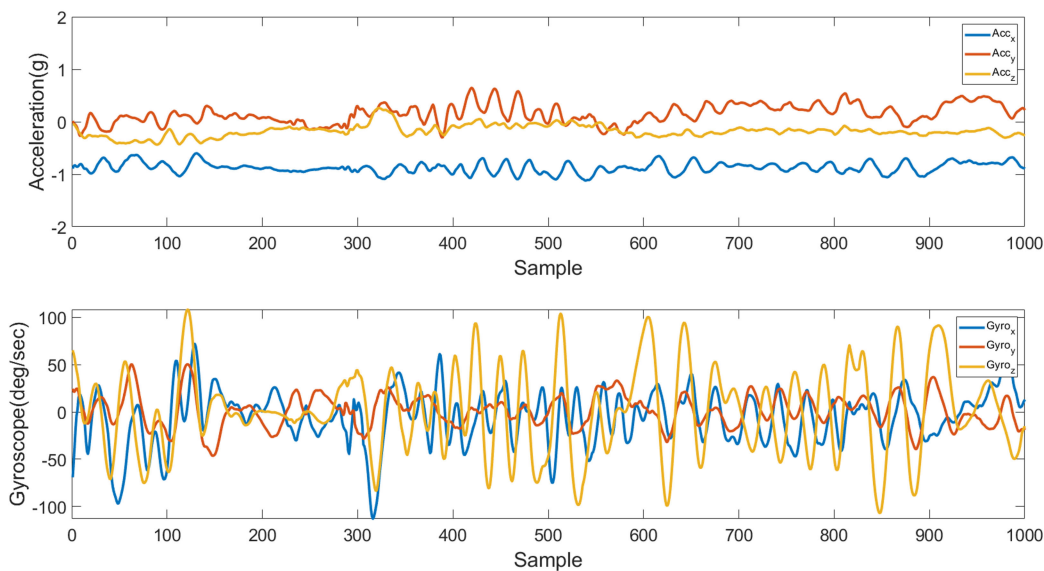


FIGURE 7. A capture of raw acceleration and gyroscope data for cleaning table from arm.

layers. The last layer of fully connected layer allows directing output into a class label using softmax activation function and classification layer.

LSTM NN is a particular type of recurrent neural networks (RNN), which overcome vanishing and exploding gradient problems [35], [36]. It has five parts; input layer, LSTM layers, fully-connected layers, one softmax activation function, and one classification layer [37]. LSTM layers have particular memory cells, which include gates such as input, forget, and output. These gates control which information should be forgotten, remembered and updated. Fully-connected layer is a feed-forward network that uses the extracted features by

LSTM layers. The softmax activation function makes probability distribution for class labels and classification layer decides the final class. The architecture of LSTM NN is given in Fig. 9.

ConvLSTM is a hybrid model, which combines the feature extraction part of CNN and LSTM NN, as seen in Fig. 10 [13]. In CNN part, features are extracted. The created feature vector is given as input for LSTM layers, which model the temporal dynamics of the extracted features. The last layer outputs of LSTM passes through fully-connected layers, softmax activation function, and classification layer for class label.

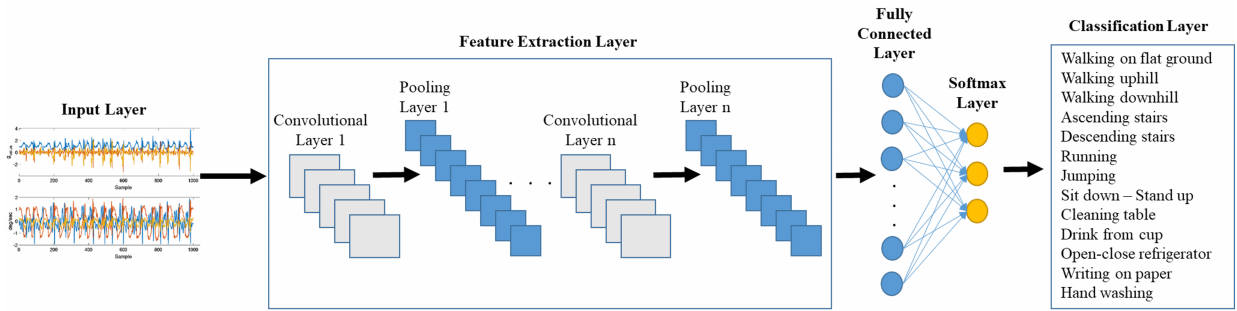


FIGURE 8. CNN architecture.

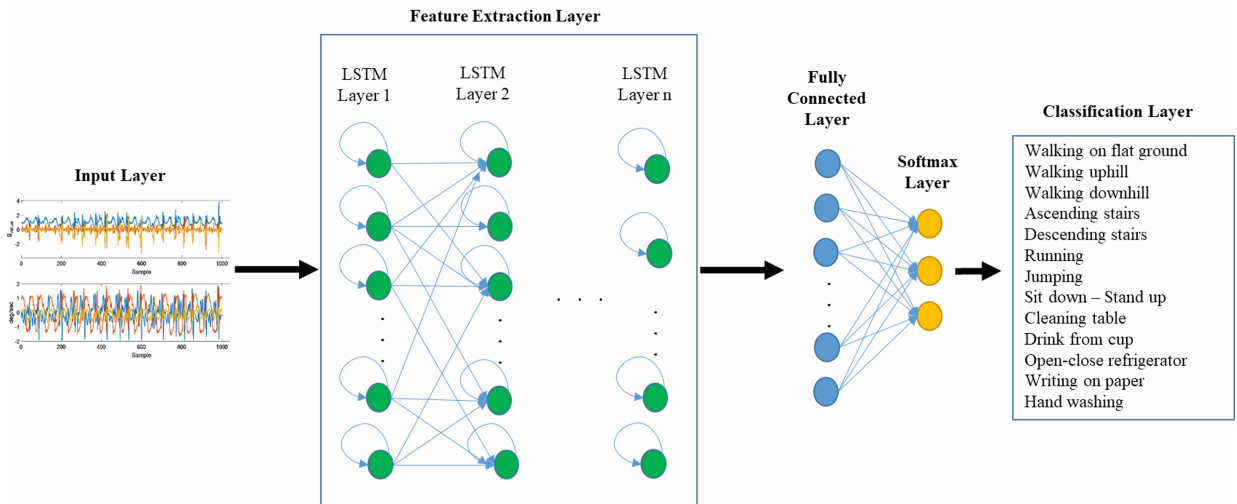


FIGURE 9. LSTM architecture.

TABLE 2. Specification of GPU in experiments.

Parameters	Value
Graphic Card Chip	Geforce RTX2070 Super
Graphic Memory Capacity	8GB DDR6
Clock Frequency	1770MHz
Cuda Kernels	2560
Memory Bandwidth	448GB/sec
Memory Speed	14Gbps

The factor that can affect neural network performance is hyperparameters. They include variables which determine the network structure (e.g., number of layers) and the variables which determine how the network is trained (e.g. learning rate, kernel size, and optimization algorithm). The values of these variables were set before training. They were determined to obtain the best performance by fine-tuning.

Training and test processes of deep learning applications were performed on graphical processor unit (GPU) to accelerate working time [38], [39]. The specifications of the chosen GPU are given in Table 2. Processes were conducted in deep learning toolbox and Experiment Manager of MATLAB 2020a.

E. EVALUATION METRICS

Classification performances of deep learning neural networks were evaluated using four metrics given as (1), (2), (3), and (4).

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall(Sensitivity) = \frac{TP}{TP + FN} \tag{3}$$

$$F_{score} = 2 \times \frac{(Recall \times Precision)}{Recall + Precision} \tag{4}$$

where TP is true positive rate, TN is true negative rate, FP is false positive rate, and FN is false negative rate.

III. RESULTS

The classification performances of the proposed models for the test frames are evaluated in this section. We divided activities into three categories as basic, complex, and all. CNN, LSTM, and ConvLSTM were separately applied to each category. To get the best performances from networks, different architectures were constructed and the values of

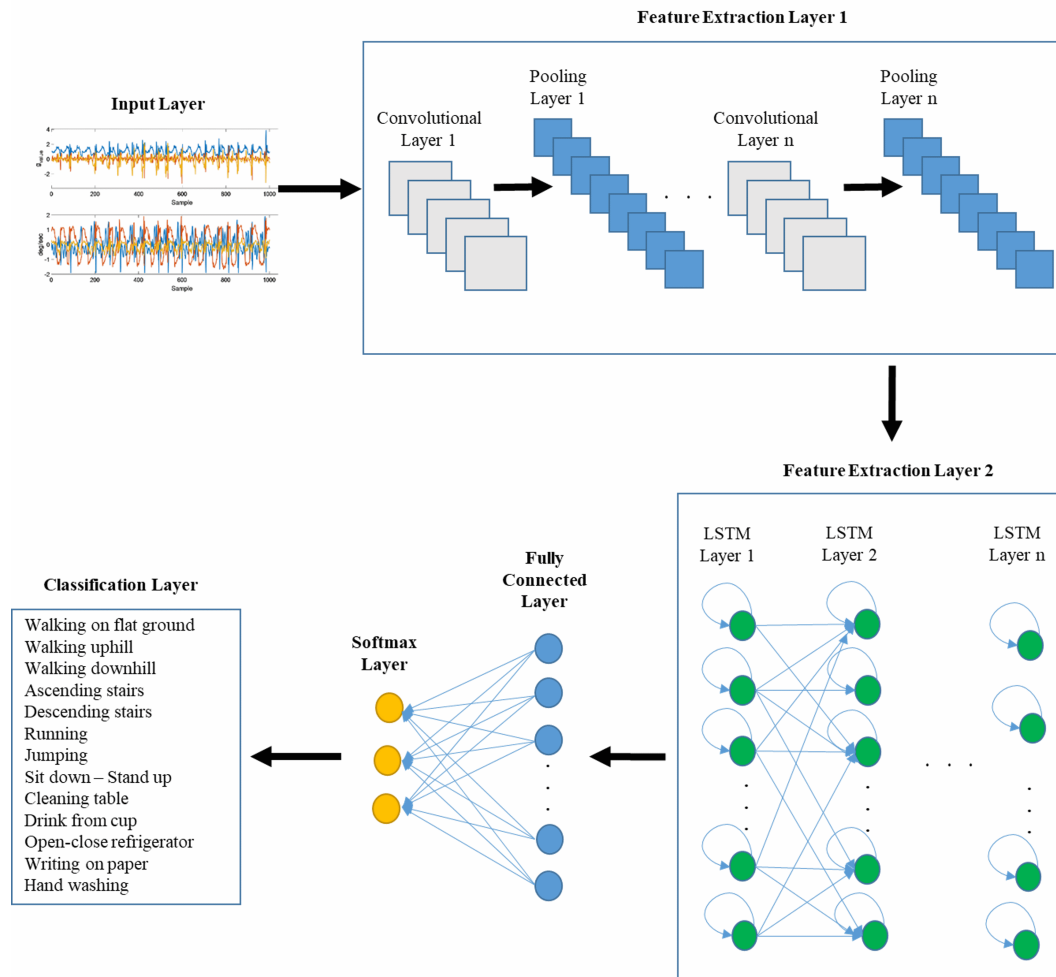


FIGURE 10. Convolutional LSTM architecture.

hyperparameters were determined by fine-tuning. For each activity, 42000 and 18000 frames were taken as training and test sets, respectively. Total frame numbers used in networks changed depending on the number of activities in each category. The results of the test frames will be discussed separately for each category in the following order.

A. CLASSIFICATION OF BASIC ACTIVITY CATEGORY

Basic activities consist of walking on flat ground, walking uphill, walking downhill, ascending stairs, descending stairs, running, jumping, and sit down - stand up. These motions are gait-based and approximately periodic signals. The motion speed causes minor changes in the duration of a period.

Firstly, CNN was applied for the classification of basic activities. The best network performance was obtained from architecture; consists of three convolutional layers, three RELU activation functions, three pooling layers, three fully-connected layers, one softmax activation function, and one classification layer. In the first convolutional layer, 96 kernels with 3×3 size were applied to the input frameset. The output of the convolutional layer went through the RELU

activation function for non-linearity. In the pooling layer, 2×2 maximum pool filters reduced the dimension of extracted frames from the convolutional layer. 128 kernels with 3×3 size and 2×2 maximum filters for second convolutional and pooling layers and, 256 kernels with 3×3 size and 2×2 maximum filters for third convolutional and pooling layers were used for feature extraction. Sliding steps of filters were adjusted as one and two in convolutional layers and pooling layers, respectively. The extracted information was taken as input for neural network with three fully-connected layers. The number of neurons in layers were determined as 128, 192, and 8, respectively. The output of the last fully-connected layer by passing through softmax activation function fed into classification layer for decision. The other important hyperparameters as well as size and numbers of filters are learning rate, weight optimization algorithm, and mini-batch size for the network performance. These parameters were determined as 0.008, adaptive moment optimization, and 64, respectively.

Secondly, LSTM was applied for classification. The neurons in layer have particular memory units to remember

information for a longer time. The most successful network performance was obtained from architecture with three LSTM layers, three fully-connected layer, one softmax activation function, and one classification layer. The neuron numbers in LSTM layers were taken as 128, 192 and 256, respectively. The output of the last LSTM layer fed into the fully-connected layer with 64, 96, and 8 neurons, respectively. Softmax activation function determined the probabilistic distribution of potential outcomes. Classification layer decided on an activity label using income information from activation function. The other hyperparameters; weight optimization algorithm, learning rate, and mini-batch size were set to adaptive moment estimation, 0.007, and 128, respectively.

Thirdly, convLSTM was applied for classification. In this neural network, convolutional and pooling layers of CNN was added to LSTM layers for feature extraction. Extracted features from CNN layers were used as input for LSTM layers. The architecture of feature extraction layers, filter sizes, and numbers used in CNN was not changed. The number of neurons was determined as 64 and 96 in LSTM layers. The outputs of last LSTM layer fed into the fully-connected layer with 8 neurons. The classification results were obtained with passing through softmax activation function of the fully-connected layer. Hyperparameters were tuned to stochastic gradient descent algorithm, 0.03, 64, and 0.8 for weight optimizer, learning rate, mini-batch size, and momentum coefficient, respectively.

The classification performance of each neural network based on eight defined activity types is given in Fig. 11, Fig. 12, and Fig. 13. ConvLSTM improves the recognition

		Target Class								Precision
		WG	WU	WD	ASC	DES	RN	JP	SS	
Predicted Class	WG	96.0%	1.2%	1.8%	1.5%	1.4%	0.7%	0.5%	0.8%	92.3%
	WU	0.8%	94.0%	0.8%	2.2%	1.9%	2.6%	1.8%	0.3%	90.0%
	WD	0.9%	0.8%	93.7%	1.7%	1.4%	1.2%	1.1%	0.2%	92.7%
	ASC	0.8%	1.9%	2.2%	92.5%	2.0%	1.8%	2.1%	0.5%	89.1%
	DES	1.2%	1.5%	0.4%	1.2%	91.9%	1.0%	1.5%	0.4%	92.7%
	RN	0.2%	0.4%	0.6%	0.3%	0.4%	92.3%	0.3%	0.3%	97.3%
	JP	0.1%	0.0%	0.5%	0.4%	0.6%	0.2%	92.5%	0.6%	97.4%
	SS	0.0%	0.2%	0.0%	0.2%	0.4%	0.2%	0.2%	96.9%	98.7%
	Recall	96.0%	94.0%	93.7%	92.5%	91.9%	92.3%	92.5%	96.9%	

FIGURE 13. Confusion matrix obtained from ConvLSTM for classification of basic activities.

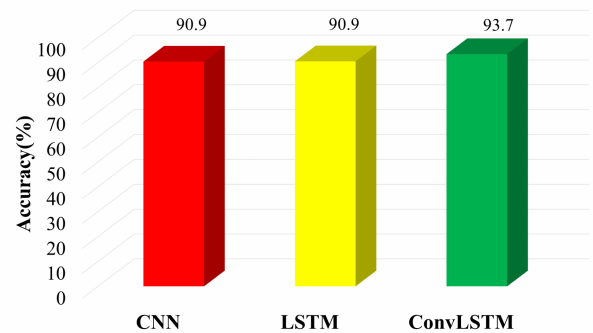


FIGURE 14. Average accuracy of deep neural networks for basic activity category.

of each activity between 0.5% and 5.5% with respect to the other two networks. In Fig. 14, the comparison of classification accuracy is given for all three networks. This shows that LSTM and CNN have almost the same average performances for basic activity recognition. However, ConvLSTM outperforms other two networks and gives 93.7% average classification accuracy.

B. CLASSIFICATION OF COMPLEX ACTIVITY CATEGORY

Complex activities consist of cleaning table, drink from cup, open-close refrigerator, hand washing, and writing on paper. These activities are based on hand gesture.

Firstly, CNN was used for classification. The network has presented the best performance with the same CNN architecture that was used for the classification of basic activities. The only difference in this CNN comes from hyperparameter settings. The number of kernels was decreased to 64, 96, and 192 for convolutional layers and, kernel sizes were fixed as 3 × 3. In pooling layers, 2 × 2 maximum pooling was used for sub-sampling. The sliding steps were adjusted as one stride for kernels and two strides for maximum pooling. The extracted features fed into the neural network with three fully-connected layers. The neuron numbers of fully-connected layers were adjusted as 64, 96, and 5, respectively. Optimization algorithm, learning rate, and mini-batch size were chosen as adaptive moment estimation, 0.01, and 64, respectively.

		Target Class								Precision
		WG	WU	WD	ASC	DES	RN	JP	SS	
Predicted Class	WG	91.0%	1.5%	2.4%	1.2%	3.0%	1.5%	0.8%	0.2%	89.5%
	WU	2.7%	90.2%	3.6%	2.5%	2.0%	1.2%	0.7%	0.3%	87.4%
	WD	2.0%	2.2%	90.4%	2.0%	1.2%	1.3%	0.3%	1.4%	89.6%
	ASC	2.2%	2.4%	1.2%	91.0%	4.7%	2.0%	1.0%	0.3%	86.8%
	DES	1.9%	3.0%	1.9%	2.8%	88.0%	1.9%	0.2%	1.7%	86.7%
	RN	0.2%	0.3%	0.2%	0.1%	0.7%	90.1%	4.8%	0.7%	92.7%
	JP	0.0%	0.4%	0.2%	0.0%	0.0%	1.9%	92.0%	0.9%	96.4%
	SS	0.0%	0.0%	0.1%	0.4%	0.4%	0.1%	0.2%	94.5%	98.7%
	Recall	91.0%	90.2%	90.4%	91.0%	88.0%	90.1%	92.0%	94.5%	

FIGURE 11. Confusion matrix obtained from CNN for classification of basic activities.

		Target Class								Precision
		WG	WU	WD	ASC	DES	RN	JP	SS	
Predicted Class	WG	90.5%	2.2%	1.4%	2.6%	0.8%	1.2%	1.4%	0.5%	89.9%
	WU	2.8%	90.1%	2.9%	1.9%	2.2%	1.0%	1.7%	0.1%	87.7%
	WD	2.4%	1.4%	89.0%	2.1%	1.8%	1.6%	1.2%	1.4%	88.2%
	ASC	1.4%	2.7%	2.7%	90.9%	3.3%	1.3%	2.8%	2.3%	84.6%
	DES	2.0%	1.9%	2.5%	1.4%	90.7%	1.4%	0.2%	2.7%	88.2%
	RN	0.3%	0.8%	0.8%	0.5%	0.8%	91.5%	1.7%	1.7%	93.2%
	JP	0.2%	0.4%	0.3%	0.6%	0.3%	1.1%	90.3%	1.8%	95.0%
	SS	0.4%	0.5%	0.4%	0.0%	0.1%	0.9%	0.7%	94.5%	96.7%
	Recall	90.5%	90.1%	89.0%	90.9%	90.7%	91.5%	90.3%	94.5%	

FIGURE 12. Confusion matrix obtained from LSTM for classification of basic activities.

Secondly, LSTM was used for classification. The most successful results were obtained from architecture consists of three LSTM layers, two fully-connected layers, softmax activation function, and classification layer. The neuron numbers in LSTM layers were determined for 128, 256, and 96. The output in LSTM layers fed into fully-connected layer with 32 and 5 neurons, respectively. The other parameters were determined as adaptive moment estimation, 0.001, and 64 for optimizer, learning rate, and mini-batch size, respectively.

Thirdly, ConvLSTM was used for classification. The extracted features in CNN fed into LSTM. The kernel sizes and number of kernels has the same hyperparameter settings used in feature extraction layer of CNN architecture. LSTM were adjusted as two layers, and the neuron numbers in LSTM layers were determined as 128 and 128. The last output of LSTM fed into fully connected layer with 5 neurons. The best results were obtained with adaptive moment estimation, 0.009 learning rate, and 128 mini-batch size.

The classification performance of neural networks was separately given for each activity type in Fig. 15, Fig. 16, and Fig. 17. ConvLSTM gives better performance between 0.6% and 6.2% than the other two networks for each activity. Fig. 18 shows average classification performance for all networks. ConvLSTM gives the best performance with an average success rate of 94.0%. CNN and LSTM have approximately same average classification performance as 90.8% and 90.5%, respectively.

		Target Class					Precision
		CT	DC	OCR	WP	HW	
Predicted Class	CT	90.3%	3.5%	2.2%	2.3%	1.0%	90.9%
	DC	1.7%	91.0%	3.4%	4.1%	1.3%	89.6%
	OCR	3.9%	1.7%	89.7%	1.9%	2.0%	90.4%
	WP	1.8%	1.6%	2.3%	89.9%	3.7%	90.5%
	HW	2.3%	2.2%	2.4%	1.8%	92.0%	91.3%
Recall		90.3%	91.0%	89.7%	89.9%	92.0%	

FIGURE 17. Confusion matrix obtained from ConvLSTM for classification of complex activities.

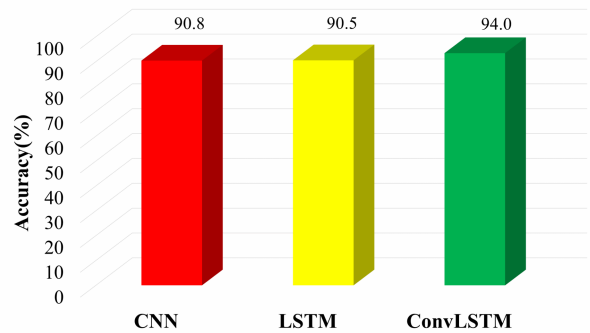


FIGURE 18. Average accuracy of deep neural networks for complex activity category.

		Target Class					Precision
		CT	DC	OCR	WP	HW	
Predicted Class	CT	90.1%	2.5%	3.1%	1.4%	2.1%	90.8%
	DC	2.1%	91.5%	2.5%	2.3%	3.0%	90.2%
	OCR	2.3%	2.2%	90.9%	2.1%	2.7%	90.7%
	WP	1.8%	1.8%	1.0%	92.0%	2.3%	93.0%
	HW	3.7%	2.0%	2.5%	2.2%	89.9%	89.6%
Recall		90.1%	91.5%	90.9%	92.0%	89.9%	

FIGURE 15. Confusion matrix obtained from CNN for classification of complex activities.

		Target Class					Precision
		CT	DC	OCR	WP	HW	
Predicted Class	CT	94.7%	1.6%	2.0%	2.1%	2.0%	92.4%
	DC	0.8%	95.0%	2.3%	2.4%	0.9%	93.6%
	OCR	2.5%	1.2%	91.5%	1.2%	0.9%	94.0%
	WP	0.7%	1.2%	1.2%	93.7%	1.1%	95.7%
	HW	1.3%	1.0%	3.0%	0.6%	95.1%	94.1%
Recall		94.7%	95.0%	91.5%	93.7%	95.1%	

FIGURE 16. Confusion matrix obtained from LSTM for classification of complex activities.

C. CLASSIFICATION OF ALL ACTIVITY CATEGORY

All activity category consists of both basic and complex activities.

Firstly, CNN was applied for classification. The best results were obtained from the architecture, which made up of three convolutional layers, three RELU activation functions, three pooling layers, four fully-connected layers, one softmax activation, and one classification layer. Kernel sizes were set to 3×3 , 4×4 , and 3×3 , and number of kernels was adjusted as 128, 192, and 256 for convolutional layers. Activation functions were applied to the outputs of all convolutional layers for non-linearity. Three maximum pooling filters with 3×3 , 2×2 , and 2×2 size were used to reduce the dimension of features and choose dominant characteristic of extracted features from convolutional layers. Sliding strides were tuned to one and two for each convolutional layer and each pooling layer, respectively. The extracted frames fed into neural network include four fully-connected layers. From the beginning to the end of the network, there are 192, 256, 128, and 13 neurons in these layers. The other hyperparameters such as learning rate, weight optimization algorithm, momentum coefficient, and mini-batch size were taken as 0.005, stochastic gradient descent momentum, 0.6, and 128, respectively.

Secondly, LSTM was applied for classification. The architecture consists of five LSTM layers, three fully connected layers, one softmax activation function, and one classification layer. The neuron numbers used in LSTM layers are 256, 256, 324, 128, and, 128, respectively. Neuron numbers of fully connected layers set to 128, 64, and 13. Learning rate,

		Target Class													Precision
		WG	WU	WD	ASC	DES	RN	JP	SS	CT	HW	DC	WP	OCR	
Predicted Class	WG	86.1%	4.0%	4.9%	2.1%	2.2%	1.9%	1.1%	1.0%	0.0%	0.0%	0.0%	0.0%	0.2%	83.2%
	WU	3.2%	80.3%	5.2%	3.9%	2.3%	1.0%	0.9%	0.6%	0.2%	0.1%	0.2%	0.0%	0.2%	81.9%
	WD	2.4%	3.6%	84.1%	4.1%	1.2%	3.1%	2.0%	1.0%	0.0%	0.0%	0.2%	0.3%	0.1%	82.4%
	ASC	3.9%	3.5%	1.4%	86.1%	6.9%	2.1%	1.7%	0.9%	0.4%	0.1%	0.0%	0.1%	0.1%	80.3%
	DES	2.8%	2.7%	2.8%	1.8%	84.7%	8.4%	0.8%	0.8%	0.0%	0.0%	0.3%	0.2%	0.3%	80.2%
	RN	1.5%	3.7%	0.5%	0.2%	1.5%	82.8%	0.4%	2.7%	0.1%	0.1%	0.3%	0.4%	0.2%	87.7%
	JP	0.0%	1.1%	0.2%	1.0%	0.8%	0.4%	92.8%	3.3%	0.5%	0.0%	0.7%	0.3%	0.5%	91.3%
	SS	0.1%	0.7%	0.4%	0.4%	0.1%	0.0%	0.1%	89.1%	0.0%	0.0%	0.2%	0.0%	0.1%	97.7%
	CT	0.0%	0.0%	0.0%	0.2%	0.0%	0.0%	0.1%	0.2%	80.2%	4.9%	5.1%	2.3%	2.5%	84.0%
	HW	0.0%	0.1%	0.1%	0.0%	0.1%	0.1%	0.0%	0.0%	3.7%	87.0%	6.9%	2.4%	4.9%	82.6%
	DC	0.0%	0.0%	0.1%	0.0%	0.2%	0.2%	0.0%	0.1%	4.1%	3.1%	80.1%	5.9%	4.2%	81.7%
	WP	0.0%	0.2%	0.2%	0.1%	0.0%	0.0%	0.1%	0.0%	2.7%	1.8%	2.6%	83.9%	6.6%	85.4%
	OCR	0.0%	0.1%	0.1%	0.1%	0.0%	0.0%	0.0%	0.3%	8.1%	2.9%	3.4%	4.2%	80.1%	80.7%
	Recall	86.1%	80.3%	84.1%	86.1%	84.7%	82.8%	92.8%	89.1%	80.2%	87.0%	80.1%	83.9%	80.1%	

FIGURE 19. Confusion matrix obtained from CNN for classification of all activities.

		Target Class													Precision
		WG	WU	WD	ASC	DES	RN	JP	SS	CT	HW	DC	WP	OCR	
Predicted Class	WG	88.2%	2.1%	4.2%	3.7%	3.2%	1.9%	1.3%	2.6%	0.3%	0.2%	0.4%	0.4%	0.3%	81.1%
	WU	1.8%	87.3%	2.2%	3.1%	3.3%	2.5%	2.1%	2.0%	0.4%	0.1%	0.2%	0.3%	0.2%	82.7%
	WD	2.4%	1.6%	86.9%	2.3%	2.1%	2.0%	2.4%	2.4%	0.1%	0.0%	0.2%	0.5%	0.4%	84.1%
	ASC	2.3%	3.1%	1.6%	84.0%	4.6%	2.9%	1.5%	2.1%	0.2%	0.1%	0.0%	0.1%	0.1%	81.9%
	DES	3.0%	2.2%	1.9%	3.2%	83.9%	1.2%	2.3%	2.7%	0.2%	0.0%	0.3%	0.7%	0.3%	82.3%
	RN	2.1%	1.3%	1.2%	2.7%	1.7%	85.3%	2.3%	3.4%	0.4%	0.2%	0.3%	0.6%	0.7%	83.5%
	JP	0.0%	1.2%	0.9%	0.3%	0.3%	1.3%	86.0%	3.3%	0.3%	0.1%	0.3%	0.3%	0.5%	91.6%
	SS	0.1%	0.7%	1.4%	0.2%	0.4%	2.1%	1.3%	80.9%	0.3%	0.1%	0.2%	0.7%	0.7%	90.8%
	CT	0.0%	0.0%	0.0%	0.1%	0.1%	0.1%	0.2%	0.4%	80.4%	2.2%	4.3%	1.3%	3.2%	87.1%
	HW	0.0%	0.1%	0.3%	0.2%	0.0%	0.2%	0.3%	0.1%	4.1%	91.1%	5.9%	3.4%	2.9%	83.9%
	DC	0.0%	0.1%	0.2%	0.1%	0.4%	0.2%	0.1%	0.0%	3.7%	3.2%	82.1%	3.6%	5.4%	82.8%
	WP	0.1%	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.1%	3.5%	0.9%	2.8%	84.0%	4.3%	87.6%
	OCR	0.0%	0.1%	0.1%	0.1%	0.0%	0.3%	0.2%	0.0%	6.1%	1.8%	3.0%	4.1%	81.0%	83.7%
	Recall	88.2%	87.3%	86.9%	84.0%	83.9%	85.3%	86.0%	80.9%	80.4%	91.1%	82.1%	84.0%	81.0%	

FIGURE 20. Confusion matrix obtained from LSTM for classification of all activities.

optimization algorithm, and mini-batch size was taken as 0.006, adaptive moment estimation, and 128, respectively.

Thirdly, ConvLSTM was applied for classification. The feature extraction layer of CNN has been fully taken over with its own entire setup. LSTM layers have 96, 192 and, 128 neurons, respectively. The neuron numbers in fully connected layers were set to 32 and 13. Weight optimization algorithm, learning rate, and mini-batch size were set to adaptive moment estimation, 0.008, and 128, respectively.

Confusion matrices in Fig. 19, Fig. 20, and Fig. 21 shows the classification performance of all the networks concerning to the defined architecture and hyperparameters. When the classification accuracies are examined, it can be seen that ConvLSTM performs better than the other networks and gives an average of 89.3% success rate considering all activities. Fig. 22 presents illustrative accuracy performance comparison for all three networks. Here, general performance of the networks for all activities is given.

D. EVALUATION

In this study, we have particularly focused on designing a monitoring system for outdoor environment and automated feature extraction technique using deep learning neural networks. Firstly, a wireless monitoring system, which consists of one master and four slave devices, was designed. Due to its low-cost, easy placement to multiple points of the human body, and lightweight, this system provides superiority over long-term use in outdoor environments compared to visual-based systems, smartwatches, and mobile phones [40], [41]. Secondly, the performance comparisons of three different deep learning neural networks for three different activity categories were done and the average accuracy of these networks was given in Table 3. When the results were evaluated, ConvLSTM has the highest accuracy for each activity category in the range of 89.3% to 94.0%. CNN and LSTM have average accuracy over 90% for classification of basic and complex activities, however their performance

		Target Class													Precision
		WG	WU	WD	ASC	DES	RN	JP	SS	CT	HW	DC	WP	OCR	
Predicted Class	WG	90.4%	1.4%	1.6%	2.3%	3.5%	1.0%	0.8%	1.4%	1.0%	0.2%	0.1%	0.3%	0.2%	86.8%
	WU	1.4%	90.2%	1.8%	2.8%	1.3%	0.9%	1.4%	2.3%	0.3%	0.1%	0.3%	0.2%	0.3%	87.3%
	WD	1.3%	0.4%	91.2%	1.4%	2.2%	1.1%	0.3%	3.5%	0.2%	0.0%	0.4%	0.2%	0.1%	89.1%
	ASC	2.4%	1.4%	1.3%	89.5%	2.1%	1.3%	2.0%	2.0%	0.8%	0.1%	0.2%	0.6%	0.2%	86.1%
	DES	2.9%	1.9%	1.1%	2.0%	86.1%	1.2%	1.9%	2.9%	0.4%	0.0%	0.3%	0.2%	0.3%	85.1%
	RN	0.3%	0.8%	1.4%	1.2%	1.4%	92.4%	1.5%	0.2%	0.4%	0.1%	0.2%	0.1%	0.5%	91.9%
	JP	0.4%	1.0%	0.3%	0.2%	2.1%	0.4%	91.0%	0.4%	0.3%	0.1%	0.2%	0.2%	0.1%	94.1%
	SS	0.1%	1.0%	0.6%	0.1%	1.0%	1.3%	0.6%	86.7%	0.2%	0.1%	0.7%	0.2%	0.2%	93.4%
	CT	0.0%	1.2%	0.2%	0.0%	0.1%	0.1%	0.0%	0.1%	88.6%	0.3%	2.7%	1.8%	3.8%	89.6%
	HW	0.2%	0.2%	0.1%	0.0%	0.0%	0.1%	0.1%	0.2%	1.5%	93.6%	3.2%	2.4%	2.7%	89.9%
	DC	0.5%	0.1%	0.2%	0.2%	0.1%	0.0%	0.2%	0.1%	2.4%	0.8%	88.1%	2.5%	3.1%	89.6%
	WP	0.1%	0.2%	0.1%	0.2%	0.1%	0.2%	0.0%	0.1%	2.0%	2.4%	1.6%	87.8%	2.1%	90.6%
	OCR	0.2%	0.2%	0.1%	0.1%	0.0%	0.0%	0.2%	0.1%	1.9%	2.2%	2.0%	3.5%	86.4%	89.2%
	Recall	90.4%	90.2%	91.2%	89.5%	86.1%	92.4%	91.0%	86.7%	88.6%	93.6%	88.1%	87.8%	86.4%	

FIGURE 21. Confusion matrix obtained from ConvLSTM for classification of all activities.

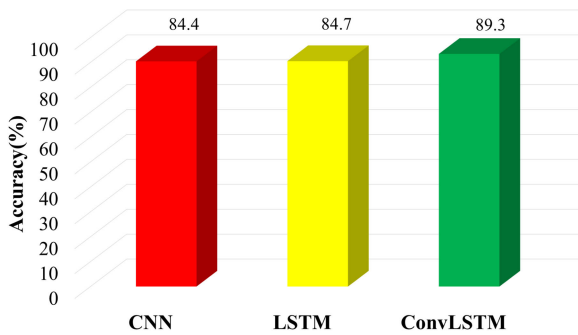


FIGURE 22. Average accuracy of deep neural networks for all activity category.

TABLE 3. Accuracy comparison of network performances for activity categories.

Activity Category	Number of Activity Types	CNN	LSTM	ConvLSTM
Basic	8	90.9%	90.9%	93.7%
Complex	5	90.8%	90.5%	94.0%
All	13	84.4%	84.7%	89.3%

decreases by approximately 6% for the classification of all activity category. Moreover, Table 4 presents F_{score} values of networks in each activity category. The results show that ConvLSTM has the highest F_{score} values for each activity category between 0.894 and 0.937.

IV. DISCUSSION

There are many studies on wearable wireless network systems for activity monitoring [13], [34], [42]. These are mostly based on Bluetooth and Zigbee technology due to low-power consumption. In this study, a Wi-Fi based wearable wireless

TABLE 4. F_{score} comparison of network performances for activity categories.

Activity Category	Number of Activity Types	CNN	LSTM	ConvLSTM
Basic	8	0.909	0.903	0.937
Complex	5	0.908	0.905	0.924
All	13	0.844	0.847	0.894

monitoring system was proposed with a higher data transfer rate and longer communication distance.

There are various public benchmark datasets in literature [43]–[47]. They focus on sensing environment, object or body motion. Each one includes different specifications with respect to activity types, number of activity, sensor variety, number of sensors, sampling rate, and sample size. A comparison of benchmark datasets and the dataset used in this study is given in Table 5.

UCI HAR, WISDM, and BILKENT datasets are based on basic activities including walking, ascending stair, descending stair, and standing. PAMAP2 dataset has also complex activities including watching television, play soccer, and drive car. OPPORTUNITY dataset has complex activities such as open-close door, drink, and clean table. In this study, there are different activities, which are not available in given benchmark datasets, containing writing on paper and hand washing.

There are also various studies that present wearable wireless sensor networks for activity monitoring and apply DL neural networks for activity recognition. These studies use Zigbee or Bluetooth technology for their WSNs. They also use smartphone as wireless sensor node or master device. Chen and Xue [30] utilized from mobile phone to collect and record acceleration data. They recorded a total of 31688 samples for eight basic activities. Ronao and Cho [22] used a

TABLE 5. Benchmark analysis for public dataset and the dataset used in this study.

Dataset	Sensing Types	Number of Sensors	Number of Activities	Sampling Rate	Number of Volunteer	Total Dataset Size
OPPORTUNITY	Environment, Object, Motion	72	17	32	12	701366
UCI HAR	Motion	1	6	50	30	748406
WISDM	Motion	1	6	20	36	1098209
PAMAP2	Motion	4	18	100	9	3850491
BILKENT	Motion	5	19	512	8	14592000
This study	Motion	4	13	40	60	9360000

mobile phone to collect data for six basic activities. The total number of dataset is approximately 700000. Tufek *et al.* [13] designed a wearable WSN using Zigbee technology to collect motion signals for seven different basic activities. Total number of sample size is 46000. When compared with collected data using custom design, our study offers a large dataset and more activity types including complex motions.

In this study, activity recognition was realized with DL neural networks using collected datasets. Three different neural networks were used including CNN, LSTM and ConvLSTM. The performance comparison of neural networks was done for three different activity category. Ordóñez and Roggen [21] made a performance comparison between CNN and ConvLSTM using Opportunity dataset. ConvLSTM improves the accuracy about 6% over results previously reported by using CNN. Xu *et al.* [48] used Opportunity and PAMAP2 dataset to recognize activities by using ConvLSTM and CNN. The results showed that ConvLSTM improves the accuracy of about 6% for Opportunity dataset and 2% for PAMAP2 dataset in respect to CNN. In this study, addition to CNN and ConvLSTM, LSTM NN was used. The performance of ConvLSTM is about 3% to 5% better than others for three different activity categories. The obtained results have shown that ConvLSTM approves previous studies.

V. CONCLUSION

This paper presents a wearable wireless multi-sensor activity monitoring system and its application in deep learning for activity recognition. The designed monitoring system has superior advantages such as low-cost, lightweight, and easy placement on various points of the human body. It is also quite convenient for outdoor applications. Using this system, total of thirteen datasets, which consist of indoor and outdoor activities, were collected from sixty healthy individuals.

The activities were categorized as basic, complex, and all. Basic activity category is based on gait, complex activity category is based on hand gesture, and all activity category is the combination of these two categories. Three deep learning neural networks were used for recognition processes. The architectures and hyperparameters of the neural networks have been optimized to allow the best performances. The results show that ConvLSTM outperforms the other two networks for each activity category. Moreover, the results also present that the basic activity category can be recognized more accurately than others.

In future work, we are planning to present a public benchmark dataset by expanding available dataset with respect to activity types and sample size.

REFERENCES

- [1] A. Palaniappan, R. Bhargavi, and V. Vaidehi, "Abnormal human activity recognition using SVM based approach," in *Proc. Int. Conf. Recent Trends Inf. Technol.*, Apr. 2012, pp. 97–102.
- [2] T. B. Supriyatna, S. M. Nasution, and R. A. Nugraheni, "Human activity recognition using support vector machine for automatic security system," *J. Phys. Conf. Ser.*, vol. 1192, no. 1, Mar. 2019, Art. no. 012017.
- [3] J. Qi, P. Yang, A. Waraich, Z. Deng, Y. Zhao, and Y. Yang, "Examining sensor-based physical activity recognition and monitoring for healthcare using Internet of Things: A systematic review," *J. Biomed. Informat.*, vol. 87, pp. 138–153, Nov. 2018.
- [4] D. De, P. Bharti, S. K. Das, and S. Chellappan, "Multimodal wearable sensing for fine-grained activity recognition in healthcare," *IEEE Internet Comput.*, vol. 19, no. 5, pp. 26–35, Sep. 2015.
- [5] Y.-D. Lee and W.-Y. Chung, "Wireless sensor network based wearable smart shirt for ubiquitous health and activity monitoring," *Sens. Actuators B, Chem.*, vol. 140, no. 2, pp. 390–395, Jul. 2009.
- [6] S. C. Mukhopadhyay, "Wearable sensors for human activity monitoring: A review," *IEEE Sensors J.*, vol. 15, no. 3, pp. 1321–1330, Mar. 2015.
- [7] L. Tao, T. Burghardt, S. Hannuna, M. Camplani, A. Paiement, D. Damen, M. Mirmehdi, and I. Craddock, "A comparative home activity monitoring study using visual and inertial sensors," in *Proc. 17th Int. Conf. E-Health New., Appl. Services (HealthCom)*, Oct. 2015, pp. 644–647.
- [8] F. Shahmohammadi, A. Hosseini, C. E. King, and M. Sarrafzadeh, "Smartwatch based activity recognition using active learning," in *Proc. IEEE/ACM Int. Conf. Connected Health Appl., Syst. Eng. Technol. (CHASE)*, Jul. 2017, pp. 321–329.
- [9] P. Lago, F. Lang, C. Roncancio, C. Jiménez-Guarín, R. Mateescu, and N. Bonnefond, "The contextact A4H real-life dataset of daily-living activities," in *Proc. Int. Interdiscipl. Conf. Modeling Using Context*. Springer, 2017, pp. 175–188.
- [10] C. Dobbins and R. Rawassizadeh, "Towards clustering of mobile and smartwatch accelerometer data for physical activity recognition," *Informatics*, vol. 5, no. 2, p. 29, 2018.
- [11] R. Akhavian and A. H. Behzadan, "Smartphone-based construction workers' activity recognition and classification," *Autom. Construct.*, vol. 71, pp. 198–209, Nov. 2016.
- [12] I. Kouris and D. Koutsouris, "Activity recognition using smartphones and wearable wireless body sensor networks," in *Proc. Int. Conf. Wireless Mobile Commun. Healthcare*. Springer, 2011, pp. 32–37.
- [13] N. Tufek, M. Yalcin, M. Altintas, F. Kalaoglu, Y. Li, and S. K. Bahadir, "Human action recognition using deep learning methods on limited sensory data," *IEEE Sensors J.*, vol. 20, no. 6, pp. 3101–3112, Mar. 2020.
- [14] T. Hossain, M. A. R. Ahad, T. Tazin, and S. Inoue, "Activity recognition by using LoRaWAN sensor," in *Proc. ACM Int. Joint Conf. Int. Symp. Pervasive Ubiquitous Comput. Wearable Comput. (UbiComp)*, 2018, pp. 58–61.
- [15] S. Balli, E. A. Sağbaş, and M. Peker, "Human activity recognition from smart watch sensor data using a hybrid of principal component analysis and random forest algorithm," *Meas. Control*, vol. 52, nos. 1–2, pp. 37–45, Jan. 2019.
- [16] Z. Chen, Q. Zhu, Y. C. Soh, and L. Zhang, "Robust human activity recognition using smartphone sensors via CT-PCA and online SVM," *IEEE Trans. Ind. Informat.*, vol. 13, no. 6, pp. 3070–3080, Dec. 2017.
- [17] S. Sani, N. Wiratunga, and S. Massie, "Learning deep features for kNN-based human activity recognition," in *Proc. CEUR Workshop*, 2017, pp. 95–103.

- [18] P. Lubina and M. Rudzki, "Artificial neural networks in accelerometer-based human activity recognition," in *Proc. 22nd Int. Conf. Mixed Design Integr. Circuits Syst. (MIXDES)*, Jun. 2015, pp. 63–68.
- [19] Y. Liu, L. Nie, L. Liu, and D. S. Rosenblum, "From action to activity: Sensor-based activity recognition," *Neurocomputing*, vol. 181, pp. 108–115, Mar. 2016.
- [20] S. Abdelhedi, A. Wali, and A. M. Alimi, "Fuzzy logic based human activity recognition in video surveillance applications," in *Proc. 2nd Int. Afro-Eur. Conf. Ind. Advancement (AECIA)*, Springer, 2016, pp. 227–235.
- [21] F. Ordóñez and D. Roggen, "Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition," *Sensors*, vol. 16, no. 1, p. 115, Jan. 2016.
- [22] C. A. Ronao and S.-B. Cho, "Human activity recognition with smartphone sensors using deep learning neural networks," *Expert Syst. Appl.*, vol. 59, pp. 235–244, Oct. 2016.
- [23] N. Y. Hammerla, S. Halloran, and T. Plötz, "Deep, convolutional, and recurrent models for human activity recognition using wearables," 2016, *arXiv:1604.08880*. [Online]. Available: <http://arxiv.org/abs/1604.08880>
- [24] V. Bianchi, M. Bassoli, G. Lombardo, P. Fornaciari, M. Mordonini, and I. De Munari, "IoT wearable sensor and deep learning: An integrated approach for personalized human activity recognition in a smart home environment," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8553–8562, Oct. 2019.
- [25] M. Shoaib, S. Bosch, O. Incel, H. Scholten, and P. Havinga, "Complex human activity recognition using smartphone and wrist-worn motion sensors," *Sensors*, vol. 16, no. 4, p. 426, Mar. 2016.
- [26] P. Bharti, D. De, S. Chellappan, and S. K. Das, "HuMAN: Complex activity recognition with multi-modal multi-positional body sensing," *IEEE Trans. Mobile Comput.*, vol. 18, no. 4, pp. 857–870, Apr. 2019.
- [27] A. Murad and J.-Y. Pyun, "Deep recurrent neural networks for human activity recognition," *Sensors*, vol. 17, no. 11, p. 2556, Nov. 2017.
- [28] R. San-Segundo, H. Blunck, J. Moreno-Pimentel, A. Stisen, and M. Gil-Martín, "Robust human activity recognition using smartwatches and smartphones," *Eng. Appl. Artif. Intell.*, vol. 72, pp. 190–202, Jun. 2018.
- [29] W. Jiang and Z. Yin, "Human activity recognition using wearable sensors by deep convolutional neural networks," in *Proc. 23rd ACM Int. Conf. Multimedia*, 2015, pp. 1307–1310.
- [30] Y. Chen and Y. Xue, "A deep learning approach to human activity recognition based on single accelerometer," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2015, pp. 1488–1492.
- [31] D. M. Karantonis, M. R. Narayanan, M. Mathie, N. H. Lovell, and B. G. Celler, "Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring," *IEEE Trans. Inf. Technol. Biomed.*, vol. 10, no. 1, pp. 156–167, Jan. 2006.
- [32] S. J. Preece, J. Y. Goulermas, L. P. J. Kenney, and D. Howard, "A comparison of feature extraction methods for the classification of dynamic activities from accelerometer data," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 3, pp. 871–879, Mar. 2009.
- [33] C. Wang, J. Shi, Q. Zhang, and S. Ying, "Histopathological image classification with bilinear convolutional neural networks," in *Proc. 39th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2017, pp. 4050–4053.
- [34] Y.-L. Hsu, H.-C. Chang, and Y.-J. Chiu, "Wearable sport activity classification based on deep convolutional neural network," *IEEE Access*, vol. 7, pp. 170199–170212, 2019.
- [35] R. Pascanu, T. Mikolov, and Y. Bengio, "Understanding the exploding gradient problem," *ArXiv*, vol. abs/1211.5063, 2012.
- [36] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.
- [37] G. Santos, P. Endo, K. Monteiro, E. Rocha, I. Silva, and T. Lynn, "Accelerometer-based human fall detection using convolutional neural networks," *Sensors*, vol. 19, no. 7, p. 1644, Apr. 2019.
- [38] E. Buber and B. Diri, "Performance analysis and CPU vs GPU comparison for deep learning," in *Proc. 6th Int. Conf. Control Eng. Inf. Technol. (CEIT)*, Oct. 2018, pp. 1–6.
- [39] S. Shi, Q. Wang, P. Xu, and X. Chu, "Benchmarking state-of-the-art deep learning software tools," in *Proc. 7th Int. Conf. Cloud Comput. Big Data (CCBD)*, Nov. 2016, pp. 99–104.
- [40] F. Shahmohammadi, A. Hosseini, C. E. King, and M. Sarrafzadeh, "Smartwatch based activity recognition using active learning," in *Proc. IEEE/ACM Int. Conf. Connected Health Appl., Syst. Eng. Technol. (CHASE)*, Jul. 2017, pp. 321–329.
- [41] J. Cheng, X. Chen, and M. Shen, "A framework for daily activity monitoring and fall detection based on surface electromyography and accelerometer signals," *IEEE J. Biomed. Health Informat.*, vol. 17, no. 1, pp. 38–45, Jan. 2013.
- [42] P. Bharti, D. De, S. Chellappan, and S. K. Das, "HuMAN: Complex activity recognition with multi-modal multi-positional body sensing," *IEEE Trans. Mobile Comput.*, vol. 18, no. 4, pp. 857–870, Apr. 2019.
- [43] D. Roggen, A. Calatroni, M. Rossi, T. Holleczeck, K. Forster, G. Troster, P. Lukowicz, D. Bannach, G. Pirkl, A. Ferscha, J. Doppler, C. Holzmann, M. Kurz, G. Holl, R. Chavarriaga, H. Saha, H. Bayati, M. Creatura, and J. D. R. Millan, "Collecting complex activity datasets in highly rich networked sensor environments," in *Proc. 7th Int. Conf. Netw. Sens. Syst. (INSS)*, Jun. 2010, pp. 233–240.
- [44] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "A public domain dataset for human activity recognition using smartphones," in *Proc. ESANN*, vol. 3, 2013, p. 3.
- [45] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, "Activity recognition using cell phone accelerometers," *ACM SIGKDD Explor. Newslett.*, vol. 12, no. 2, pp. 74–82, Mar. 2011.
- [46] A. Reiss and D. Stricker, "Introducing a new benchmarked dataset for activity monitoring," in *Proc. 16th Int. Symp. Wearable Comput.*, Jun. 2012, pp. 108–109.
- [47] B. Barshan and M. C. Yükek, "Recognizing daily and sports activities in two open source machine learning environments using body-worn sensor units," *Comput. J.*, vol. 57, no. 11, pp. 1649–1667, 2014.
- [48] C. Xu, D. Chai, J. He, X. Zhang, and S. Duan, "InnoHAR: A deep neural network for complex human activity recognition," *IEEE Access*, vol. 7, pp. 9893–9902, 2019.



GOKMEN ASCIOĞLU received the B.S. degree in electronics and communication engineering from Suleyman Demirel University, Isparta, and the M.S. degree in electrical and electronics engineering from Dokuz Eylül University, Izmir, in 2014. From 2012 to 2016, he was a Research Assistant with BMC Automotive, where he has been working as an Electrical Vehicle and Advanced Driving Assistance Systems Engineer, since 2019. His research interests include artificial intelligence, embedded systems, and signal processing.



YAVUZ SENOL received the B.S. and M.S. degrees in electrical-electronics engineering from Dokuz Eylül University, Izmir, Turkey, and the Ph.D. degree in electrical and electronic engineering from the University of Sussex, Brighton, U.K., in 1994.

He is currently an Associate Professor with the Department of Electrical-Electronics Engineering, Dokuz Eylül University. His research interests include space instrumentation, machine learning, signal processing, embedded systems, and robotics. He realized some projects for private companies and TUBITAK in related subjects.

...