

# Human Action Recognition Using Deep Learning Methods on Limited Sensory Data

Nilay Tufek, Murat Yalcin, Mucahit Altintas, Fatma Kalaoglu, Yi Li, Senem Kursun Bahadir\*

**Abstract**—In recent years, due to the widespread usage of various sensors action recognition is becoming more popular in many fields such as person surveillance, human-robot interaction etc. In this study, we aimed to develop an action recognition system by using only limited accelerometer and gyroscope data. Several deep learning methods like Convolutional Neural Network(CNN), Long-Short Term Memory (LSTM) with classical machine learning algorithms and their combinations were implemented and a performance analysis was carried out. Data balancing and data augmentation methods were applied and accuracy rates were increased noticeably. We achieved new state-of-the-art result on the UCI HAR dataset by 97.4% accuracy rate with using 3 layer LSTM model. Also, we implemented same model on collected dataset (ETEXWELD) and 99.0% accuracy rate was obtained which means a solid contribution. Moreover, the performance analysis is not only based on accuracy results, but also includes precision, recall and f1-score metrics. Additionally, a real-time application was developed by using 3 layer LSTM network for evaluating how the best model classifies activities robustly.

**Index Terms**—Activity recognition, deep learning, CNN, LSTM, data augmentation, data balancing

## I. INTRODUCTION

**H**UMAN action recognition is a still challenging task due to the fact that actions can be carried out by different persons in various places. Moreover, length of time interval can vary for same and different types of actions. Since actions consist of sequential data, getting spatio-temporal information from the data becomes a challenge [1].

Wearable devices have started to change our lives, such as smartphones have changed our habits and behavior within ten years. Wearable devices appeared to be seen in many areas e.g. health care, fitness tracking, entertainment and they are expected to spread rapidly in the near future. For instance; [2] proposes to measure sweat rate by using wearable sensors. [3] uses wearable sensors to monitor kinematic change while doing sportive activities that require physical demands.

[4] suggests to follow up betterments of children under rehabilitation and assess them activities that accelerate the treatment process. [5] monitor health of user by wearable sensors. It reviews wearable sensor usage in the mining industry. To detect hazardous gas, to monitor brain activities of equipment drivers, to monitor fatigue, wearable devices are already in use. [6] commits a system to detect obstacles for people who has lost sense of sight by wearable sensors. [7] emphasizes the future usage of wearable devices in tourism.

In recent years, deep learning techniques outperformed hand-crafted feature extraction methods [8] and numerous deep architectures were developed particularly for wearable devices. They have great capability to extract meaningful and high-level features from data [9].

Owing to accelerometer and gyroscope sensors are wearable, human actions can be recognized via these sensors [10]. These sensors are able to be integrated in textile products and they benefit from improvements on both textile and electronics' technology [11]. Technology is used for accelerometer based systems for different purposes [12].

This study focus on activity recognition based on state-of-the-art deep learning methods using wearable sensors. Products are designed and implemented including accelerometer, gyroscope and wireless radio frequency module called ZigBee. Moreover, they are embedded with temperature, heart rate and humidity sensors.

In this work, data was collected by using these sensors at 5 Hz frequency while subjects were performing 7 different activities. The dataset was generated based on experimental study implemented under European Union project (ETEXWELD H2020: RISE. 644268) so that the dataset is called as ETEXWELD dataset. In the literature, several works have been carried out using accelerometer and gyroscope for activity recognition and authorization. UCI HAR is the first large dataset including daily activities of subjects which are measured via inertial sensors in mobile phone [13]. It includes 6 different activities including “walking”, “sitting”, “standing”, “lying”, “up-stairs” and “down-stairs”. These are similar that we used in this project, but in this study we have also added “falling” activity. Therefore, UCI HAR dataset is used in this study as a benchmark dataset to compare ETEXWELD dataset results.

In the literature, there are numerous strategies for performance analysis on sensor based activity recognition. These strategies are occurred by supervised, unsupervised and semi-supervised techniques. However, this is a classification problem and we used supervised learning methods. The UCI HAR dataset which is used as benchmark dataset is also used in several studies [14] [15]. In these works, mostly classical machine learning algorithms such as Support Vector Machines (SVM) [16], k Nearest Neighbour, Linear Discriminant Analysis (LDA) [17], Multilayer Perceptron(MLP) [18] and deep learning methods are used on this dataset.

[19] [20] recognizes daily activities by using wearable sensors and additionally environmental sound with DNN (Deep Neural Network). [21] uses S transform and Gaussian windows to extract features of the activities but its success related with

\* Corresponding author at: ITU, Mechanical Engineering Faculty 34437 Istanbul, Turkey. Tel.: +90 212 2931300x2499.  
E-mail address: kursuns@itu.edu.tr (Senem Kursun Bahadir).

length of the activity which is very dataset specific. Many studies uses kNN algorithm to classify activities on. But it is not possible to implement to kNN to real-time applications due to computational cost causes delay in recognition. [22] demonstrates that using the combination of accelerometer and gyroscope sensors together to detect human activities properly by using feature selectors and kNN algorithm. While classical machine learning algorithms present around 85% accuracy, deep learning algorithms including CNN and LSTM [23] give better result over 90s% [24]. While CNN gets 95% accuracy, [25] uses histogram of gradient (HG) and the Fourier descriptor (FD) to extract features, kNN and SVM is deployed for recognition. This kind of hand-crafted feature extraction techniques requires special feature engineering for every dataset. They achieved the latest state-of-the-art result by 97.1% using these methods on UCI HAR dataset. But, by using deep learning method we have found new state-of-the-art results on UCI HAR this dataset. Moreover, we have implemented several methods in order to compare the results of the deep learning methods including Dynamic Time Warping (DTW) [26] and kNN, one dimensional CNN (1D CNN [27]), two-dimensional CNN 2D CNN [28]), 2 layer LSTM [29], 3 layer LSTM [30], bidirectional LSTM [31], 1D CNN and LSTM on UCI HAR dataset in the paper. Additionally, we have implemented all these methods on ETEXWELD dataset to see comparison of result of the methods for sensor based activity recognition problems over UCI HAR and ETEXWELD datasets.

In this study, we have worked on how to improve accuracy, precision, recall and f1-score metrics [32] by using data augmentation and data balancing methods. After using data set balancing, we have observed much higher accuracy and precision results. Then, data augmentation is used after balancing as well and it also improved the results. All these interventions are applied only on training set. During the implementation of all methods, 10 fold cross validation was used to check how to model optimize on training dataset.

There are several contributions to literature including original deep learning architectures, improvement algorithms, a new dataset, comparison of several methods in different metrics and improving the state of the art results on public dataset in the content of this study. First of all, our results beat the state of the art results on UCI HAR public dataset in the literature. In order to achieve it, we have developed new deep learning networks with a novel data balancing algorithm. Above all, new dataset is designed and created for this project specifically, but it would be used in several projects in the future. One of the most challenging part was to create new dataset using integrated sensors and to work on a real sampled data acquired from those embedded sensors. It is not possible to collect thousands samples data from different subjects. Therefore, data augmentation approach within the study was also designed to improve generalization ability of the models. It is one of the novel step as well, because of improvement the results and being unique. For the recognition, scratch deep learning and machine learning algorithms which are used on both datasets were deployed. Finally, the comparisons of several learning methods for each dataset are shown from the aspects

of accuracy, precision, recall, time and space consumption.

## II. DESIGN ARCHITECTURE OF SENSORS EMBEDDED TO A TEXTILE STRUCTURE

End-to-end system design including the human, sensors, monitoring, and communication modules integrated to textile products is shown in Fig. 1. Several setups occurred by electronic modules and flexible covering band for various parts of the body are prepared as seen in Fig. 2. However, in this study, only chest module is used, because [33] shows that chest is the best position to place just a single wearable device to distinguish human activities. Therefore, just a single wearable device placed to the chest with its covering band and all activities were performed by this setup. Wearable sensor device can be seen on the chest clearly in Fig. 3.

All data from each sensor is tracked via a desktop application as shown in Fig. 4. Some of the activities such as walking, falling, upstairs and jogging also could be seen in Fig. 5.

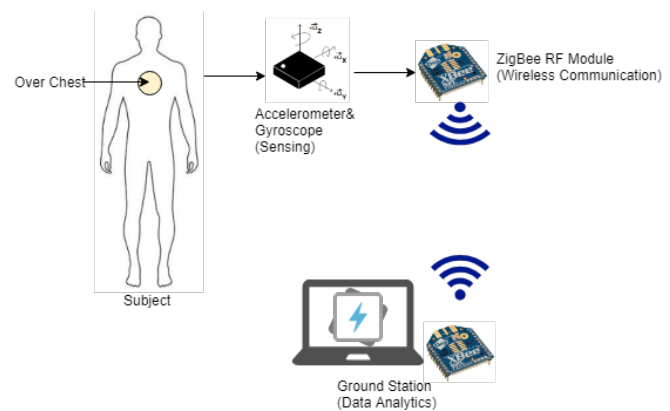


Figure 1: End to end system activity recognition system overview

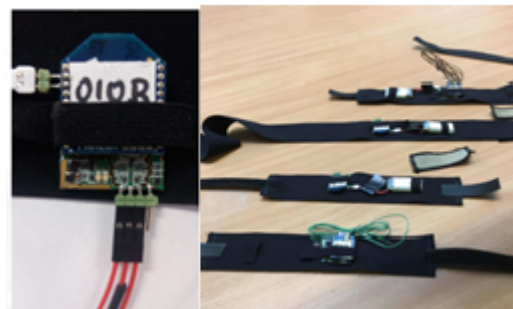


Figure 2: (a) single electronic device including all sensors; (b) couple of prototypes integrated on flexible band

## III. DATASETS

In the scope of this study, there are two similar datasets including UCI HAR [13] benchmark dataset and ETEXWELD dataset. Both are created for activity recognition via accelerometer and gyroscope sensors. These datasets include data for the actions of “walking”, “sitting”, “lying”, “standing”, “up-stairs” and “down-stairs”. Despite these datasets

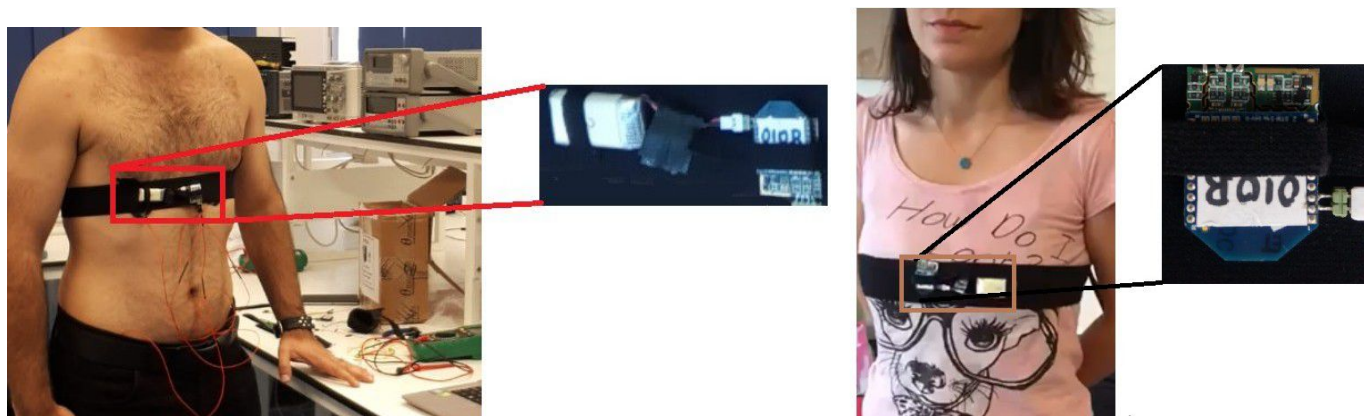


Figure 3: Wearable sensors on human physical body. (a) An early test with the experiment environment. (b) A snapshot from the data collection phase.

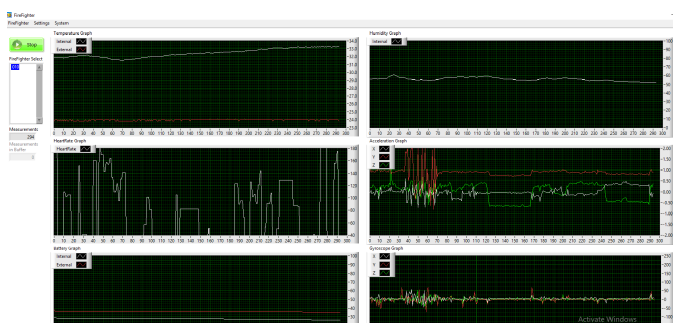


Figure 4: A view of the sensor tracker application on desktop



Figure 5: Different actions including walking, falling, upstairs and jogging

have similarities, there are also differences including sampling rate, number of actions and hardware platforms.

Firstly, while UCI HAR dataset has 50 Hz sampling rate, ETEXWELD dataset has 5 Hz sampling rate, which is highly challenging to detect activities. Requirement of a low sampling rate comes from the power consumption constraint of the system. Therefore, sampling rate and number of the samples are less in ETEXWELD dataset.

Secondly, while UCI HAR dataset has 6 different actions, ETEXWELD dataset has 7 different activities with extra added “falling” and “jogging” actions and discarded “sitting” action. Because these activities are more related to our user stories.

Finally, although first dataset is collected from the accelerometer and gyroscope sensors of a mobile device,

ETEXWELD dataset is required since the new hardware platform is a custom hardware board with different accelerometer and gyroscope sensors. Table 1 shows all physical activities in the scope of the project and the number of collected data samples. As seen on the Table 1, we can consider this as a small size dataset and some activities has much more samples compare to the others.

In order to make dataset balanced, data augmentation algorithms have been implemented and a comparison between these different type of datasets has been made.

Table I: Collected data per classes for ETextWeld dataset with the number of samples

Action	Number of Sample
Walking	10000
Jogging	9000
Lying	8000
Standing	8000
Falling	1000
Up-stairs	5000
Down-stairs	5000

#### IV. THE DESIGN OF THE AUTOMATION SYSTEM

The automated system is managed by data collection, pre-processing with data augmentation, machine learning algorithms and real time application.

##### A. Data Acquisition

The data is collected with an automation system that uses ZigBee module on both clothing architecture and ground station. This module works with 2.4 GHz frequency and transmission distance is between 10 and 20 m for indoor environment, up to 200 m for outdoor environment. For instance, Fig. 6 and Fig. 7 shows the collected data based on 3-axis accelerometer and 3-axis gyroscope, respectively. 7 different activities were carried out by 10 different subjects during data acquisition. Volunteers consist of 5 male and 5 female aged between 22 and 40 years. All experiments were conducted with the consent of the persons. After data acquisition, this collected data should be pre-processed in order to apply deep learning algorithms.

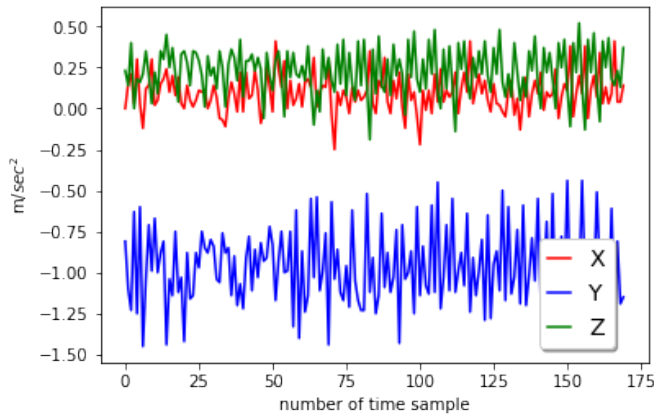


Figure 6: A capture of accelerometer data during walking action

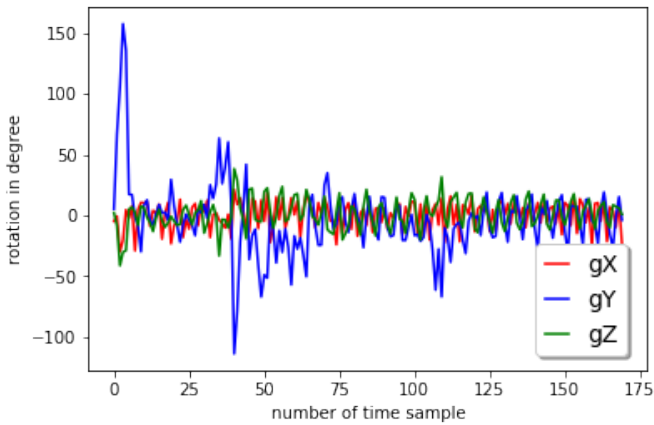


Figure 7: A capture of gyroscope data during walking action

### B. Data Preparation

Data preparation is done by preprocessing, data balancing and data augmentation. Pre-processing includes parsing the data into the format of input shape for each model with 15% to get samples throughout activity. Also, it separates the dataset into train and test data as shown in Fig. 8. Train set consist of 3 males and 3 females subjects and it corresponds 75% of the whole dataset. Other remaining 25% part which consist of 2 males and 2 females subjects which is completely different from train dataset was left as test set to determine model performance.

The distribution of activities in native train and test subsets are shown in Fig. 9.

As it is seen in Fig. 8, the native data set has unbalanced distribution for both train and test data sets. Because of the unbalanced number of collected samples for different actions, we have faced with high accuracy but low precision. In order to improve precision, data balancing algorithms were used. Sufficiency is based on random stop. The only thing we considered before stop the balancing that any proportion of number of any action should be less than 2.

After balancing, the number of samples are increased seen in Fig. 10.

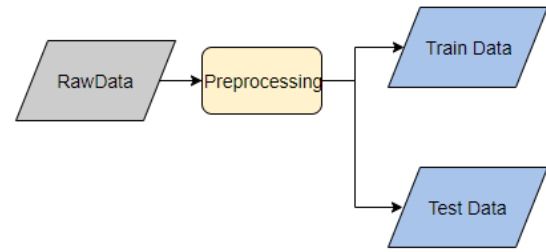


Figure 8: Preparing native dataset by filtering raw data and separating into native train and test datasets.

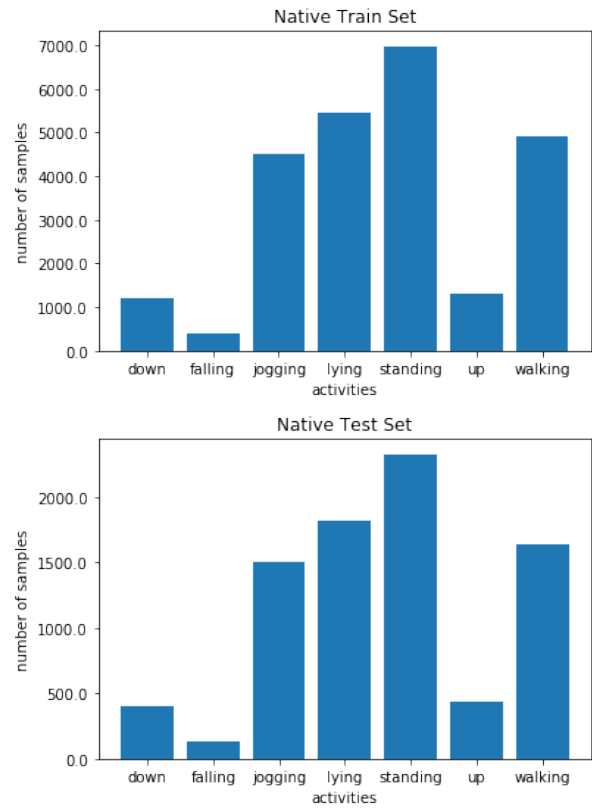


Figure 9: Native train and test dataset which is generated from collected dataset

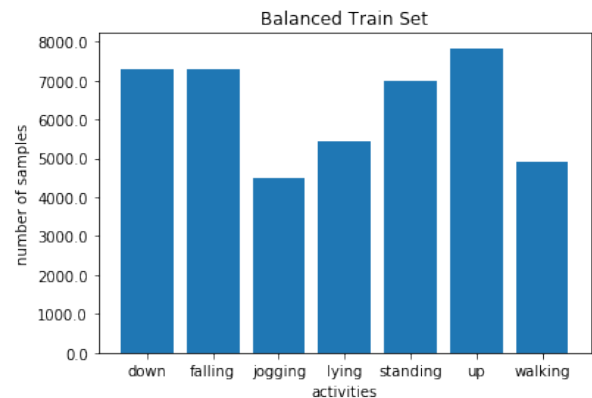


Figure 10: Sufficiently balanced train dataset



### Algorithm 1 Data Augmentation Algorithm

```

1: procedure AUGMENTDATA( $Data, n, m$ )
2:    $NewData \leftarrow data$ 
3:    $randomness \leftarrow Random(1, n)$ 
4:    $NewData \leftarrow RandomShift(r, data)$ 
5:    $randomness \leftarrow Random(1, m)$ 
6:    $NewData \leftarrow AddRandomNoise(rndmns, NewData)$ 

```

### Algorithm 2 Balancing Dataset Algorithm

```

1: procedure BALANCE( $Data, n, m$ )
2:    $randomness \leftarrow Random(1, n)$ 
3:   for  $action \in Dataset.actions$  do
4:     if  $action.len * 2 > maxAction$  then
5:        $Continue$ 
6:     if  $action.len * 2 < maxAction$  then
7:        $realSample \leftarrow getDtFrmThtAction(action)$ 
8:       for  $i \leq randomness$  do
9:          $newSample \leftarrow AugmntDt(realSample, n, m)$ 

```

The core code of balancing algorithm is also used in data augmentation. The data augmentation step was necessary due to the lack of sampled data for each action. The idea behind this augmentation algorithm is after taking any real data for any action, we would like to create similar data to improve our dataset. In order to do that, we add noise to raw data and shift it randomly left and right. While determining the noise and shifting amount, the correlation matrix was taken into account. For each action correlation matrices have been created in order to keep the similarity level of each action type in collected dataset. The correlation matrix for balanced and native train dataset can be seen in Fig. 11. The higher level pseudo-code is shown in Algorithm 1 and Algorithm 2.

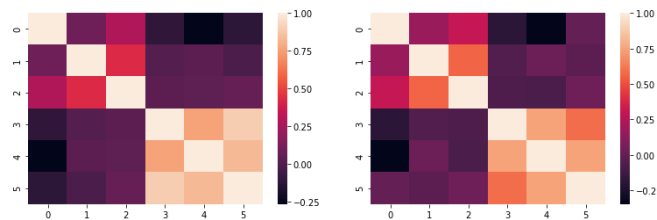


Figure 11: (a) Cross correlation matrix of each sensor signal in balanced train dataset; (b) in native train dataset

After balancing and augmentation of the dataset, the similar values in correlation matrices were observed. This was our benchmark during the determining randomness border on balancing and augmentation. The distribution of augmented dataset is shown in Fig. 12.

### C. Creation of Deep Learning Models

The research problem in this study is handled as a supervised classification process. All of the evaluated supervised deep learning [34] methods in this study are trained and tested as seen in Fig. 13 and Fig. 14.

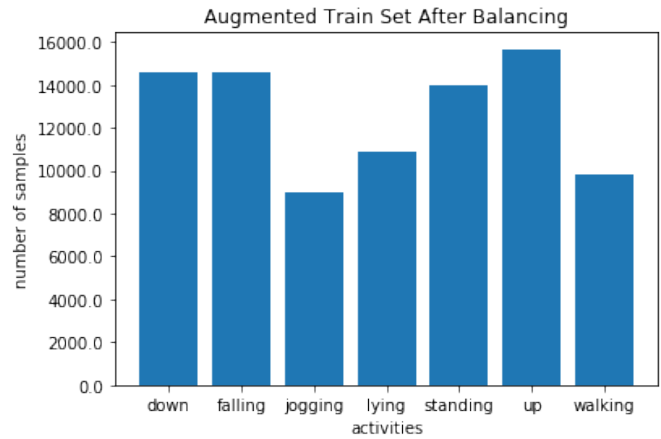


Figure 12: Augmented dataset over balanced dataset

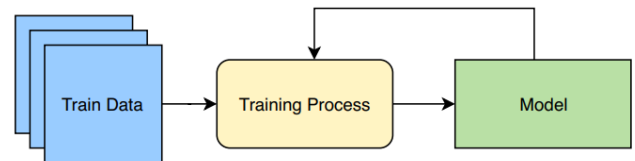


Figure 13: General training process for these machine learning approaches

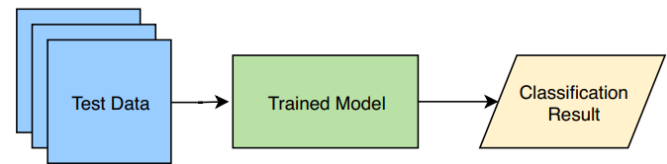


Figure 14: General testing process of any trained model

These methods are 1D CNN, 2D CNN, 2 Layer LSTM [30], 3 Layer LSTM, Bi-directional LSTM, 1D CNN+LSTM. Each network is designed and implemented with a unique architecture and is trained from scratch. In addition to these state-of-the-art deep learning methods, DTW+kNN is also implemented with the parameter  $k$  equals to 5. Moreover, all methods were applied on both UCI HAR dataset and newly created dataset.

### D. CNN Models

Convolutional neural network (CNN) is applied through signal data to extract features automatically and to classify [35] [36]. CNNs have capabilities to consider both dependencies in both time and signal dimensions. Weight sharing in CNN improves the recognition performance while decreasing the number of trainable weights. We used two different type of CNN models. Algorithm 3 shows general algorithm for CNN models.

**1D CNN Model:** Input for 1D CNN from scratch model is  $1 \times 90$  vectors. Consecutive samples of accelerometer and gyroscope data is ordered by 1D vector. For feature extraction task, architecture consists of 2 convolution layers with  $2 \times 1$

### Algorithm 3 CNN Models

```

1: procedure TRAINING(Activity, Parameters)
2:   for all activities do
3:     PreProcessing
4:     ConstructActivities
5:     while  $k \leq TotalActivities$  do
6:        $Activity(k) \leftarrow Convolutions$ 
7:        $Activity(k) \leftarrow MaxPooling$ 
8:        $Activity(k) \leftarrow Classifier$ 
9:        $Loss \leftarrow TrueActivity - Activity(k)$ 
10:  return Error

```

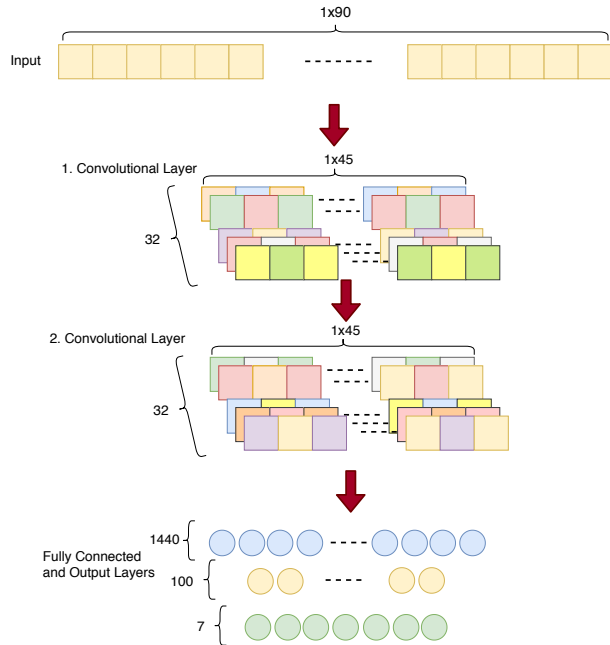


Figure 15: 1D CNN model for activity classification. Input consist of consecutive samples.

kernels, stride 2 and 1x1 kernels and stride 1, respectively. 32 different filters is used for all layers. After this step 1x1440 flattened feature vector is obtained. For classification, this features is fed to a fully connected layers which consist of 100 neurons with RELU activation function. To determine activity class, an output layer is deployed which consist of 7 neurons with softmax activation function. Learning rate is determined 0.01 and optimization method is chosen Stochastic Gradient Decent with categorical cross-entropy loss function. After 250 epochs 10 fold-cross-validation presented the best performance. Fig. 15 shows 1D CNN model.

**2D CNN Model:** Before being fed to 2D CNN, the data were converted into image format. When we consider the image as a matrix, 3 coordinates of accelerometer value comes to first 3 column and gyroscope value comes to next 3 column, respectively. Row by row, every sample is arranged horizontally throughout until the activity ends. Every activity is carried out within 3 seconds. Data input is 15x6 activity map for both model. In 1D model, convolution operator works on just one time sample instead of other in 2D model. Notice

that the kernel height is responsible for extracting temporal patterns while the kernel width extracts the correlation between neighboring axes.

Scratch model consists of 2 convolution layers, 1 max-pooling layer, 1 fully connected layer and 2 dropout layers. Learning rate was chosen 0.0001 and Adam [37] optimization method is used for learning weights. Data was fed into groups of 24 batches and the highest performance was achieved with 10 fold cross-validation as a result of 300 iterations.

To learn relations between features, 1000 neurons were used in hidden layer for classification. There is no pre-trained network for training with less effort. So, we developed completely from scratch network. Fig. 16 shows 2D CNN network model.

### E. LSTM Models

There are plenty studies that take advantage of interpreting capability of sequential data for activity recognition by using wearables [38] [39]. When implementing LSTM models, most commonly used architecture is described in [9]. A LSTM memory block consist of three gates: an input gate ( $i$ ), a forget gate ( $f$ ), and an output gate ( $o$ ), which overwrite, keep, or retrieve the memory from the memory cell ( $c$ ) at the time  $t$ .

Firstly, input gate ( $i_t$ ) and forget gates ( $f_t$ ) are computed by following equations:

$$i_t = \text{sigm}(W_i \cdot x_t + H_i \cdot h_{t-1} + C_i \cdot c_{t-1} + b_i) \quad (1)$$

$$f_t = \text{sigm}(W_f \cdot x_t + H_f \cdot h_{t-1} + C_f \cdot c_{t-1} + b_f) \quad (2)$$

Afterwards, by using an amount of the previous contents ( $c_{t-1}$ ), the current memory cell ( $c_t$ ) is updated for forgetting and the new memory ( $\tilde{c}$ ) for including. They can be calculated by following equations:

$$\tilde{c}_t = \text{tanh}(W_{\tilde{c}} \cdot x_t + H_{\tilde{c}} \cdot h_{t-1} + b_{\tilde{c}}), \quad (3)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t, \quad (4)$$

Finally, the final activation at the current position ( $h_t$ ) is calculated with the output gate ( $o_t$ ) which regulates the amount of information to output.

$$o_t = \text{sigm}(W_o \cdot x_t + H_o \cdot h_{t-1} + C_o \cdot c_t + b_o) \quad (5)$$

$$h_t = o_t \odot \text{tanh}(c_t) \quad (6)$$

$x, i, f, \tilde{c}, c, o, h \in \mathbb{R}^T$ , where  $T$  is the length of the input.  $x_t$  is the input activation at the current time ( $t$ ), and  $h_{t-1}$  is the output activations from the past time ( $t-1$ ).  $W, H,$  and  $C$  are weight matrices for input to gates, recurrent connections, and cell to gates.

In the study of Zebin at al. [40] only 6 different activities were classified with 2 layered LSTM Network. In our work, beside that two different layered LSTM Network, Bidirectional LSTM Network [41] is also used. Standard LSTM network have restrictions as the future input information cannot be reached from the current state. On the contrary, bidirectional LSTM network do not require its input data as the same

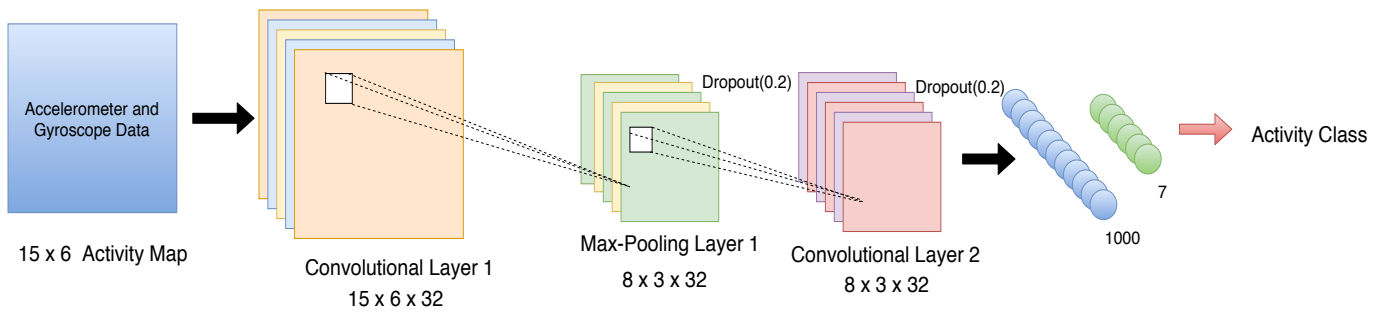


Figure 16: 2D CNN model for classification task. Input consist of 3 seconds length activities ordered by rows.

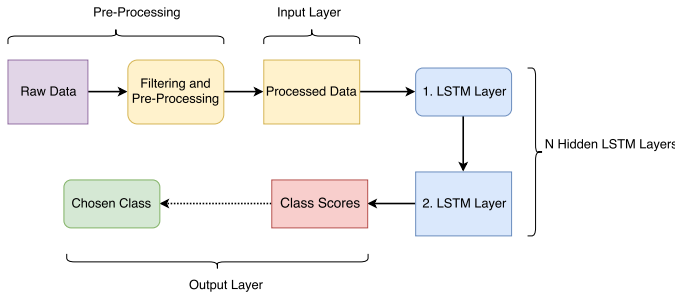


Figure 17: Overall scheme of LSTM models with different number of hidden layers

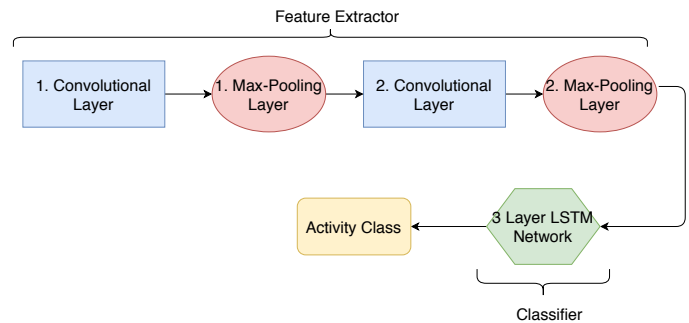


Figure 18: Hybrid model that uses CNN as a feature extractor and LSTM for classification.

dimension. Moreover, their future input information can be reached from the current state. The main idea of Bidirectional LSTM is to connect two hidden layers of opposite directions to the same output. By this structure, the output layer can access information from past and future states and interprets better.

After pre-processing step, data should be converted to fixed form for feeding LSTM network model. Data constructed with appropriate dimensions are fed to the successive LSTM blocks with vectors formed by 32 batches. Overall flow chart of the LSTM model is seen in Fig. 16 with different number of hidden layers. To understand sequential relationship between spatio-temporal data, window size and overlap rate hyperparameters are very important and were selected as 32 and 50%, respectively. 2 and 3 layers LSTM models have been deployed for two different experiments. Every layer in the network consist of 32 neurons. In experiments, 10-fold cross-validation, 0.0025 learning rate, 0.0015 loss account coefficient and 15 epochs are the common hyper-parameters for both LSTM models. Binary cross-entropy optimization has been determined as the optimization method. The dropout layer is applied with 20% probability to prevent overfitting. Bidirectional LSTM network model consists of 2 hidden layer that includes 100 and 32 hidden layer, separately.

Bidirectional LSTM network model consists of 2 hidden layer that includes of 100 and 32 hidden layer, separately. Fig. 17 shows general LSTM model with different number of hidden layers.

### F. Hybrid Models

**CNN + LSTM Model:** In this model, CNN and LSTM models took part for the recognition task as a hybrid model. At first, 1D CNN was used in different time steps as a feature extractor. After implementing 2 convolutional layer that each of them have 64 filter maps and 2 max pooling layer with stride 2, 1536 features were obtained. For classification task 3 LSTM layer that every hidden layer consist of 128 neurons was deployed. To get sequential relationship between samples and time intervals, data was fed to network as 1x124 vectors. Adam [37] optimization method was used and categorical cross-entropy selected as a loss function during training. Fig. 18 illustrate overall scheme of the model.

**KNN + DTW Model:** [42] used KNN algorithm for UCI dataset. Although KNN is classical machine learning algorithm for classification, it is very simple and effective. It is very important to determine optimal k value which has a great role in terms of classification accuracy. Algorithm was run with different k values and was chosen as 4 for best performance. DTW [43] used to calculate similarity between two temporal sequences and combined with KNN algorithm for classification task. It finds optimum mapping by using minimum distance between arbitrary length signal samples rather than just customary calculating Euclidean distance as a similarity measure criteria in KNN algorithm. Equation 8 shows how minimum distance calculation using Euclidean distance formula in Equation 7. Here, Euclidean distance was used with neighbourhood of  $k = 4$  nearest training samples with the initial condition  $D(1, 1) = |t(1) - r(1)|$ .

$$D_{Euclidean}(x, y) = \sqrt{\sum_{i=1}^k (x_i - y_i)^2} \quad (7)$$

$$D(i, j) = |t(i) - r(j)| + \min \begin{cases} D(i-1, j) \\ D(i-1, j-1) \\ D(i, j-1) \end{cases} \quad (8)$$

### G. Real-time Application

In the work carried out, one real-time desktop application which uses the deep learning method as an analyzer was implemented to track fire-fighters activity. Interface of the application can be seen on Fig. 19.



Figure 19: Real time desktop application captures walking and running activities which determined by the machine learning algorithms.

## V. EXPERIMENTAL TEST RESULTS

All results were obtained using the same hardware circumstance that involves Intel i7 2.6 GHz CPU, 8 GB RAM and without GPU. All software is implemented using Python programming language and Tensorflow library. After preparing the data, all deep learning methods on datasets were tested. In order to test it, several considerations and comparison criteria were evaluated. These criteria are based on popular evaluation metrics including precision, recall, f1-score and accuracy. All

these metrics are based on the true and false results and their real values. They are called true negative (TN), true positive(TF), false negative (FN) and false positive(FP). TP is an outcome where the model correctly predicts the positive class while TN is an outcome where the model correctly predicts the negative class. FP, a false positive, is an outcome where the model incorrectly predicts the positive class and a false negative is an outcome where the model incorrectly predicts the negative class. These outcomes help to calculate metrics as shown on Table 2.

Table II: Evaluation metrics for classification which are used in this work.

Metric	Calculation
Accuracy	$(TP+TN)/(TP+FP+FN+TN)$
Precision	$TP / (TP + FP)$
Recall	$TP/(TP+FN)$
F1 score	$2*TP/(2*TP+FP+FN)$

Training time and number of updating parameters are also measured for time and space consumption in algorithm comparisons. Training time do not effect accuracy rate. But if there is big amount of data, training procedure cost much more computation time. All of the deep learning networks that designed were tested on two different datasets including UCI HAR and ETEXWELD dataset. On UCI HAR dataset, two different approach was deployed. Firstly, experiments were carried out by using just accelerometer and gyroscope data. Then, another experiments, accelerometer, gyroscope and total data consist of accelerometer and gyroscope data together were used. Moreover, ETEXWELD dataset is used in three different training set forms which are native, balanced and augmented. The most important metric in this study is accuracy. Comparative results which based on accuracy are obtained on the UCI HAR and ETEXWELD dataset. For UCI HAR dataset when only accelerometer and gyroscope is used, accuracy results, time and space consumption points of view are shown on the Table 3. It is empirically observed the most accurate model is 3 Layer LSTM with 93%.

Table III: Test accuracy, time and space consumptions during training on UCI HAR dataset (accelerometer and gyroscope only) for each machine learning model

Models	UCI HAR Test Accuracy (%)	Training Time (minutes)	Updated Parameters
DTW + kNN	65	185	-
1D CNN	76	90	23,460
2D CNN	85	132	7,314,270
2 Layer LSTM	91	15	202, 146
3 Layer LSTM	93	20	362,342
Bi-directional LSTM	77	121	362,342
1D CNN+ LSTM	76	30	1,208,106

There is also one more parameter comes with UCI HAR dataset called "total". After adding this input into training model, the results are summarized in Table 4. As it is seen, the accuracy was improved and new state-of-the-art result was obtained with 93.7% accuracy rate.



Table IV: Test accuracy, time and space consumption during training on UCI HAR dataset for each machine learning model

Models	UCI HAR Test Accuracy (%)	Training Time (minutes)	Updated Parameters
DTW + kNN	69.1	250	-
1D CNN	76.3	125	23,460
2D CNN	85.2	200	7,314,270
2 Layer LSTM	93.7	22	202, 146
3 Layer LSTM	97.4	20	362,342
Bi-directional LSTM	90.3	140	362,342
1D CNN + LSTM	88.5	45	1,208,106

The same approach was also implemented to ETEXWELD train dataset. Results of native test accuracies are given in Table 5.

Table V: *Native* ETEXWELD data test result for accuracy, time and space

Models	ETEXWELD Test Accuracy (%)	Training Time (minutes)	Updated Parameters
DTW + kNN	72	<1	-
1D CNN	77	1	9,255
2D CNN	81	8	840,295
2 Layer LSTM	95	1.5	12,124
3 Layer LSTM	96	2	21,863
Bi-directional LSTM	86	6	21,863
1D CNN + LSTM	78	10	123,843

As it is seen in Table 5 from accuracy point of view, the best result was obtained with 3 Layer LSTM. However, accuracy may not be enough to determine whether the model is good or not. Therefore, precision, recall, f1-score and support metrics for 3-Layer LSTM have been also checked in Table 6.

Table VI: Precision, recall and F1 score metrics for each activity in 3 layer LSTM model which is occurred by *native* ETEXWELD dataset

Metrics	falling	standing up-stairs	down-stairs	walking	lying	jogging
Precision	0	0.88	0.50	0.45	0.77	0.98
Recall	0	0.81	0.44	0.55	0.84	0.99
F1-Score	0	0.85	0.47	0.50	0.81	0.98

While accuracy is relatively high, the precision, recall and f1-score are very low for several activities. When we examine the results, it is observed that, lower precision results are also small sampled activities. The reason of this low precision is that of unbalancing dataset that is trained. Confusion matrix of test results by using 3 layer LSTM model which trained with native dataset can be seen on Fig. 20.

In order to solve this problem, the data augmentation and dataset balancing have been done. After the balancing process, balanced train set for all machine learning methods is again used. As it is seen in the Table 7, accuracy is highly improved for all machine learning models and again, the best performance was obtained with 3 layer LSTM as before.

After this observation, other parameters in balanced dataset training model were re-checked. As given in Table 8, there is an important improvement.

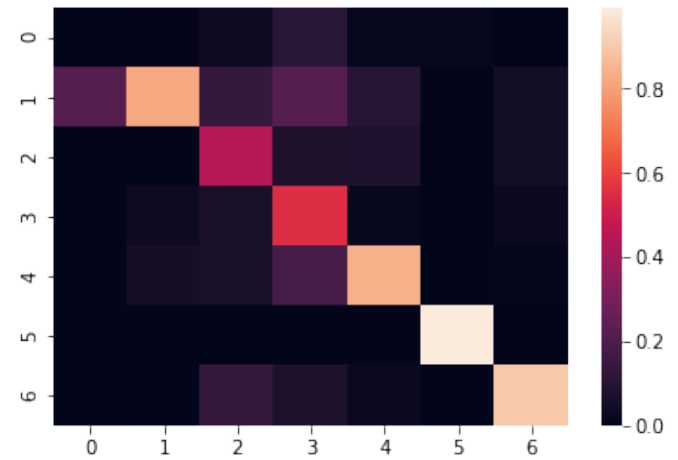


Figure 20: Normalized confusion matrix shows test results in 3 Layer LSTM model which is trained with native dataset.

Table VII: *Balanced* ETEXWELD data test result for accuracy, time and space

Models	Accuracy (%)	Training Time (minutes)	Updated Parameters
DTW + kNN	93	20	-
1D CNN	94	70	9,255
2D CNN	96	130	840,295
2 Layer LSTM	97	130	12,124
3 Layer LSTM	98	150	21,863
Bi-directional LSTM	92	270	21,863
1D CNN + LSTM	89	300	123,843

Table VIII: Precision, recall and F1 score metrics for each activity in 3 layer LSTM model which is occurred by *balanced* ETEXWELD dataset

Metrics	falling	standing up-stairs	down-stairs	walking	lying	jogging
Precision	0.90	0.90	0.67	0.84	1.00	0.99
Recall	1.00	0.97	0.97	1.00	0.81	0.99
F1-Score	0.95	0.93	0.78	0.91	0.89	0.99

It could be also clearly seen in confusion matrix how much it is improved with balanced data in Fig. 21.

After this big improvement on not only accuracy, but also precision, recall and f1-score, the dataset and model became more trustful.

Training process results on balanced data is also followed on epochs for 3 layer LSTM model. Accuracy and loss chances could be seen for both train and test set as well in Fig. 22.

In addition to this balancing, all models have been tested on the augmented dataset to see how data augmentation is effecting classification accuracy and the accuracy results on all models is seen in Table 9 and also confusion matrix can be seen in Fig. 23.

The improvement on the accuracy still exists, although it is very small. The training time is increased because of increasing iteration number. Other metrics also examined with augmented data in Table 10.

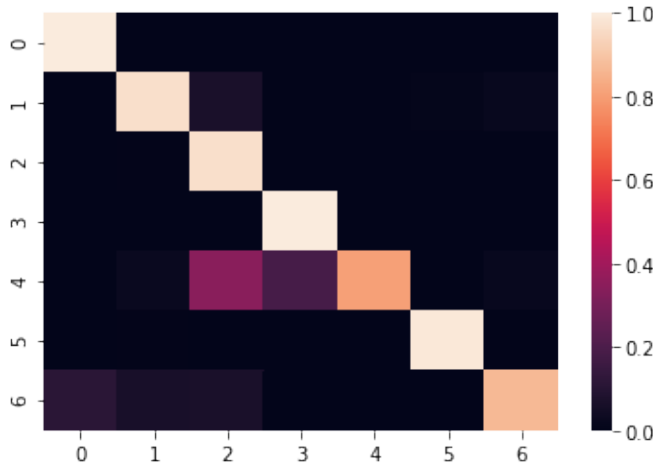


Figure 21: Normalized confusion matrix shows test results in 3 Layer LSTM model which is trained with balanced dataset.

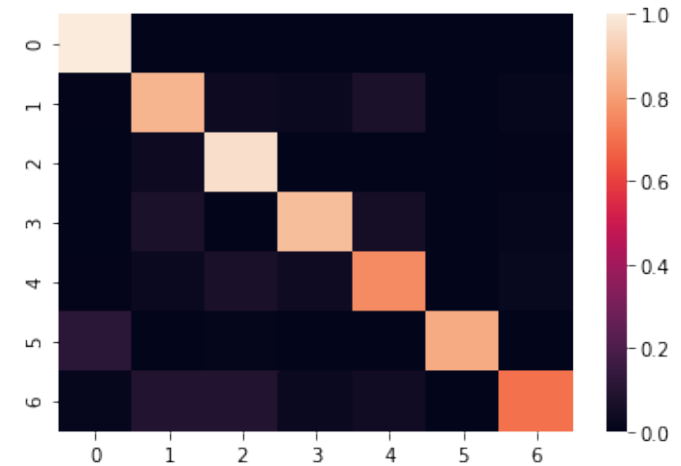


Figure 23: Normalized confusion matrix shows test results in 3 Layer LSTM model which is trained with augmented dataset after balancing.

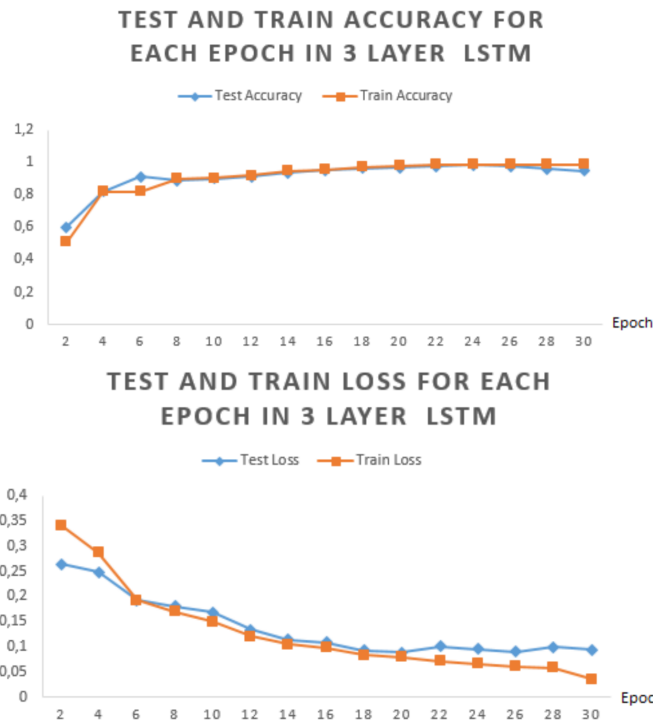


Figure 22: Train and test accuracy and loss change during each epoch for 3 layer LSTM model on balanced dataset

Table IX: *Augmented* ETEXWELD data test result for accuracy, time and space

Models	Accuracy (%)	Training Time (minutes)	Updated Parameters
DTW + kNN	94	60	-
1D CNN	94	140	9,255
2D CNN	96	200	840,295
2 Layer LSTM	97	190	12,124
3 Layer LSTM	99	330	21,863
Bi-directional LSTM	93	370	21,863
1D CNN + LSTM	90	380	123,843

Table X: Precision, recall and F1 score metrics for each activity in 3 layer LSTM model which is occurred by *augmented* ETEXWELD dataset after balancing

Metrics	falling	stand- ing	up- stairs	down- stairs	walk- ing	ly- ing	jog- ging
Precision	0.90	0.80	0.84	0.90	0.81	1.00	0.88
Recall	1.00	0.86	0.97	0.88	0.76	0.84	0.60
F1 Score	0.94	0.83	0.89	0.90	0.79	0.91	0.70

## VI. CONCLUSION

In this study, a unique dataset based on real experiments acquired from wearable devices was used for human activity recognition. We have achieved state-of-the-art results with two different data set with using deep learning techniques including from scratch trained and designed CNN and LSTM networks. It is proven that 3-Layer LSTM model is the best solution for sensor-based activity recognition problems in real time applications particularly for wearable devices. Additionally, kNN may not be an efficient option for classification of activities for big datasets like UCI HAR because of lower accuracy rate and computational cost. Therefore, we contribute to the literature while showing the comparison of different deep learning results according to accuracy, time and space complexity. Moreover, we have shown that the data augmentation on small size datasets and balancing for the unbalanced dataset are critical in order to obtain higher scores. They do not only improve test accuracy but also improve other metrics such as precision, recall and f1-score dramatically. In order to achieve that, we have design and implement our original algorithms for time series sensory data specifically as one of the novel steps. This study can be implemented to detect activities of firefighters to determine their health condition and performances. In future studies, it is planning to use multiple sensors and recognize more complicated activities.

## ACKNOWLEDGMENT

This project has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No. 644268

## REFERENCES

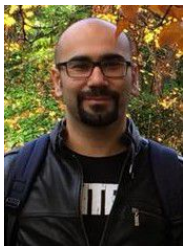
- [1] M. D. Rodriguez, J. Ahmed, and M. Shah, "Action mach a spatio-temporal maximum average correlation height filter for action recognition," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, June 2008, pp. 1–8.
- [2] P. Salvo, F. Di Francesco, D. Costanzo, C. Ferrari, M. G. Trivella, and D. De Rossi, "A wearable sensor for measuring sweat rate," *IEEE Sensors Journal*, vol. 10, no. 10, pp. 1557–1558, Oct 2010.
- [3] C. Strohrmann, H. Harms, C. Kappeler-Setz, and G. Troster, "Monitoring kinematic changes with fatigue in running using body-worn sensors," *IEEE Transactions on Information Technology in Biomedicine*, vol. 16, no. 5, pp. 983–990, Sep. 2012.
- [4] C. Strohrmann, R. E. Labrüyère, C. N. Gerber, H. J. A. van Hedel, B. Arnrich, and G. Tröster, "Monitoring motor capacity changes of children during rehabilitation using body-worn sensors," in *Journal of NeuroEngineering and Rehabilitation*, 2012.
- [5] H. Jin, Y. Shibli Abu-Raya, and H. Haick, "Advanced materials for health monitoring with skin-based wearable devices," *Advanced healthcare materials*, vol. 6, 03 2017.
- [6] S. Kursun Bahadır, V. Koncar, and F. Kalaoglu, "Wearable obstacle detection system fully integrated to textile structures for visually impaired people," *Sensors and Actuators A Physical*, vol. 179, 06 2012.
- [7] F. Ortiz Rincon, E. Tommasini, M. Rainoldi, and R. Dr. Egger, *The Future of Wearable Devices On-Site: A Scenario Technique Approach*, 01 2017, pp. 285–299.
- [8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, ser. NIPS'12. USA: Curran Associates Inc., 2012, pp. 1097–1105. [Online]. Available: <http://dl.acm.org/citation.cfm?id=2999134.2999257>
- [9] A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*, 01 2012, vol. 385.
- [10] M. Zhang and A. A. Sawchuk, "Usc-had: a daily activity dataset for ubiquitous activity recognition using wearable sensors," 09 2012, pp. 1036–1043.
- [11] S. K. Bahadır, S. Thomassey, V. Koncar, and F. Kalaoglu, "An algorithm based on neuro-fuzzy controller implemented in a smart clothing system for obstacle avoidance," *Int. J. Comput. Intell. Syst.*, vol. 6, pp. 503–517, 2013.
- [12] S. K. Bahadır, "Decision of sensor location and best classification method for entrap and muscle disease detection in healthcare smart clothing based on acceleration measurements," *Transactions of the Institute of Measurement and Control*, vol. 37, no. 8, pp. 999–1008, 2015. [Online]. Available: <https://doi.org/10.1177/0142331214552513>
- [13] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "A public domain dataset for human activity recognition using smartphones," in *ESANN*, 2013.
- [14] J.-L. Reyes-Ortiz, L. Oneto, A. Ghio, A. Samà Monsonís, D. Anguita, and X. Parra, "Human activity recognition on smartphones with awareness of basic activities and postural transitions," vol. 8681, pp. 177–184, 09 2014.
- [15] J. Zhu, R. San-Segundo, and J. M. Pardo, "Feature extraction for robust physical activity recognition," *Human-centric Computing and Information Sciences*, vol. 7, no. 1, p. 16, Jun 2017. [Online]. Available: <https://doi.org/10.1186/s13673-017-0097-2>
- [16] A. Iosifidis, A. Tefas, N. Nikolaidis, and I. Pitas, "Multi-view human movement recognition based on fuzzy distances and linear discriminant analysis," *Computer Vision and Image Understanding*, vol. 116, pp. 347–360, 2012.
- [17] J. Baek, G. Lee, W. Park, and B.-J. Yun, "Accelerometer signal processing for user activity detection," in *KES*, 2004.
- [18] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE Communications Surveys Tutorials*, vol. 15, no. 3, pp. 1192–1209, Third 2013.
- [19] T. Hayashi, M. Nishida, N. Kitaoka, and K. Takeda, "Daily activity recognition based on dnn using environmental sound and acceleration signals," in *2015 23rd European Signal Processing Conference (EUSIPCO)*, Aug 2015, pp. 2306–2310.
- [20] S. C. Mukhopadhyay, "Wearable sensors for human activity monitoring: A review," *IEEE Sensors Journal*, vol. 15, no. 3, pp. 1321–1330, March 2015.
- [21] W. Lu, F. Fan, J. Chu, P. Jing, and S. Yuting, "Wearable computing for internet of things: A discriminant approach for human activity recognition," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2749–2759, April 2019.
- [22] A. Wang, G. Chen, J. Yang, S. Zhao, and C. Chang, "A comparative study on human activity recognition using inertial sensors in a smartphone," *IEEE Sensors Journal*, vol. 16, no. 11, pp. 4566–4578, June 2016.
- [23] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, pp. 1735–80, 12 1997.
- [24] C. Ann Ronao and S.-B. Cho, "Human activity recognition with smartphone sensors using deep learning neural networks," *Expert Systems with Applications*, vol. 59, 04 2016.
- [25] A. Jain and V. Kanhangad, "Human activity classification in smartphones using accelerometer and gyroscope sensors," *IEEE Sensors Journal*, vol. 18, no. 3, pp. 1169–1177, Feb 2018.
- [26] C. J. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, Jun 1998. [Online]. Available: <https://doi.org/10.1023/A:1009715923555>
- [27] S. Ha, J. Yun, and S. Choi, "Multi-modal convolutional neural networks for activity recognition," in *2015 IEEE International Conference on Systems, Man, and Cybernetics*, Oct 2015, pp. 3017–3022.
- [28] X. Wang, L. Gao, J. Song, and H. Shen, "Beyond frame-level cnn: Saliency-aware 3-d cnn with lstm for video action recognition," *IEEE Signal Processing Letters*, vol. 24, no. 4, pp. 510–514, April 2017.
- [29] F. J. O. Morales and D. Roggen, "Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition," in *Sensors*, 2016.
- [30] F. A. Gers, J. Schmidhuber, and F. A. Cummins, "Learning to forget: Continual prediction with lstm," *Neural Computation*, vol. 12, pp. 2451–2471, 2000.
- [31] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional lstm networks," in *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, vol. 4, July 2005, pp. 2047–2052 vol. 4.
- [32] M. Sokolova, N. Japkowicz, and S. Szpakowicz, "Beyond accuracy, f-score and roc: A family of discriminant measures for performance evaluation," in *Australian Conference on Artificial Intelligence*, 2006.
- [33] L. Atallah, B. Lo, R. King, and G. Yang, "Sensor positioning for activity recognition using wearable accelerometers," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 5, no. 4, pp. 320–329, Aug 2011.
- [34] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–44, 05 2015.
- [35] N. Y. Hammerla, S. Halloran, and T. Ploetz, "Deep, convolutional, and recurrent models for human activity recognition using wearables," *CoRR*, vol. abs/1604.08880, 2016. [Online]. Available: <http://arxiv.org/abs/1604.08880>
- [36] W. Jiang and Z. Yin, "Human activity recognition using wearable sensors by deep convolutional neural networks," in *ACM Multimedia*, 2015.
- [37] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: <http://arxiv.org/abs/1412.6980>
- [38] Y. Guan and T. Ploetz, "Ensembles of deep LSTM learners for activity recognition using wearables," *CoRR*, vol. abs/1703.09370, 2017. [Online]. Available: <http://arxiv.org/abs/1703.09370>
- [39] M. Inoue, S. Inoue, and T. Nishida, "Deep recurrent neural network for mobile human activity recognition with high throughput," *CoRR*, vol. abs/1611.03607, 2016. [Online]. Available: <http://arxiv.org/abs/1611.03607>
- [40] T. Zebin, N. Peek, A. Casson, and M. Sperrin, "Human activity recognition from inertial sensor time-series using batch normalized deep lstm recurrent networks," vol. 2018, 07 2018.
- [41] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, Nov 1997.
- [42] G. Guo, H. Wang, D. A. Bell, Y. Bi, and K. Greer, "Knn model-based approach in classification," in *CoopIS/DOA/ODBASE*, 2003.
- [43] D. J. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series," in *KDD Workshop*, 1994.
- [44] O. S. Eyobu and D. S. Han, "Feature representation and data augmentation for human activity classification based on wearable imu sensor data using a deep lstm neural network," in *Sensors*, 2018.

- [45] E. Fullerton, B. Heller, and M. Munoz-Organero, "Recognizing human activity in free-living using multiple body-worn accelerometers," *IEEE Sensors Journal*, vol. 17, no. 16, pp. 5290–5297, Aug 2017.



**Nilay Tufek** studied Electronics Engineering and Computer Engineering (Double Major) at Istanbul Technical University (ITU). Also, she had been in Royal Technical University (KTH) in Sweden as an exchange student for six months. After that, she has worked in industry as software developer, team leader and researcher for seven years. She spent around five months in The University of Manchester for academic research on Horizon 2020 EU ETEXWELD Project in 2018. In parallel, she got MSc degree in Computer Engineering at ITU in

2019. Nowadays, she is working in a company as a senior artificial intelligence researcher. Moreover, she has been pursuing PhD in Computer Engineering at Bogazici University since 2019. She has 5 accepted conference papers and on going 5 patent applications.

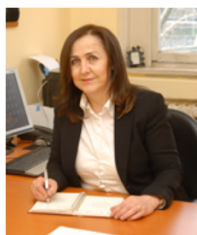


**Murat Yalcin** received the B.Sc. degree in Electronic Engineering from Uludag University in 2013 and the M.Sc. degree in Telecommunications Engineering from Istanbul Technical University in 2019. He worked at Visual Intelligence Laboratory as a research engineer for 2 years. Then, he worked in EU H2020 Project (ETEXWELD) as a researcher at The University of Manchester for 4 months. Currently, he is working as a R&D computer vision engineer at A2 Labs. Also, he is a Computer Engineering PhD student at Istanbul Technical University. His

research interests are computer vision, machine learning/deep learning and sensor-based systems. He has 3 conference paper publications.



**Mucahit Altintas** is a researcher on machine learning and pattern recognition at Istanbul Technical University, Computer Science Department. He has been in Manchester University as a visitor researcher via an EU H2020 Project. He received his B.Sc. degree from Dumlupınar University and his M.Sc. degree from Istanbul Technical University in computer science. He is currently PhD candidate at Istanbul Technical University.



**Fatma Kalaoglu** received Ph.D degree in 1991 at Istanbul Technical University. She is currently Professor in the Faculty of Textile Technologies and Design Faculty at Istanbul Technical University. Her research interests are smart textile structures, electronic textiles, clothing Technologies, clothing comfort and virtual prototyping. She is the author of more than 200 scientific papers including, SCI indexed papers, conference proceedings, book chapters, a book and a patent. She conducted 1 FP 7 project, 4 International and 6 national

projects. She is currently the coordinator of H2020 EU ETEXWELD Project.



**Dr. Yi Li** is a Life Fellow of Royal Society of Art, Commerce and Manufacturing and International Biographical Association and Fellow of the Textile Institute, and adjunct professors of a number of universities in China and a member of several professional bodies. He is the Chairman of Textile Bioengineering and Informatics Society and the Editor-in-Chief of "Journal of Fiber Bioengineering and Informatics". By securing over £10 million research funding and obtaining support from government funding bodies and industry, he established the

Textile Bioengineering Framework to conduct systematic research in textile thermal bioengineering, biomechanical engineering, sensory bioengineering and biomedical engineering, with significant research outputs in biomaterials, nano scale drug delivery systems, nano fiber based scaffolds and textile devices for tissue engineering and stents, graphene and 2D material fibers and fabrics, smart e-textile materials and components, intelligent wearables, textile material functional testing and characterization, digital apparel and clothing functional design, cloud computational technology for fashion bid data business models, textile ecological and carbon footprint and industry sustainability and strategic technology roadmap development, which has become an international platform of design and engineering textiles to promote industry-university cooperation and accelerate technology transfer. On-going projects include EU H2020 "Welding e-textile for interactive clothing" and "Fashion Bid Data Business Models". He has supervised over 50 PhD students and more than 130 research personnel. He has over 500 scientific publications, including 422 SCI/CPCI papers and peer reviewed conference papers with 10583 citations and h-index of 55 and i10-index 208 as being recognized as an international leading expert in smart functional textiles. He is selected on the list of "Highly Cited Researcher" in 2013 by THOMSON REUTERS. He owns more than 80 patents, including 46 patents granted in China, USA and Australia and 27 IP properties transferred to industry. With over 110 invited keynote/plenary lectures in conferences, he has more than 54 awards on outstanding research papers, patent inventions and technology transfer. The inventions from his team have been successfully commercialized.



**Senem Kursun Bahadir** received the B.Sc. degree in textile engineering and the M.Sc. degree in industrial engineering, and the Ph.D. degrees in automatic control/computer engineering from the University of Lille, France, and in textile engineering from Istanbul Technical University, Turkey, in 2011. From 2015 to 2016, she was the Vice Head of Textile Engineering Department and later became the Vice Dean at the Faculty of Textile Technologies and Design, Istanbul Technical University. She is currently an Associate Professor at Mechanical Engineering

Faculty. Her areas of research activities include electronic textiles, smart clothing system designs and simulation techniques in clothing area. She has authored and co-authored over 60 scientific publications in international and national peer-reviewed journals and conferences. She has been involved in many projects (European or national funding) as a Researcher and Coordinator in the field of textile technology: 1) design of textile based sensors (temperature, ultrasonic, and so on); 2) actuator implementations to textile structures (pressure and vibration motors); 3) energy harvesting and thermal heating e-textiles; and 4) design and development of interactive clothing. She is currently the ITU Coordinator of the EU H2020 Project (ETEXWELD), where the aim is the development of e-textile structures for interactive clothing. She was also awarded two international awards and one national award for textile innovation in the field of smart textiles and wearable technologies.