A study on IMU-Based Human Activity Recognition Using Deep Learning and Traditional Machine Learning

Chengli Hou
Information Engineering School of NanChang University
Nanchang, China
e-mail: hchengli@email.ncu.edu.cn

***Abstract*—Human Activity Recognition (HAR) has been an increasingly popular range to do researches which stems from the ubiquitous computing. And lately, identifying activities during daily life has become one of more and more challenges. Subsequently, more and more methods can be used in the recognition of human activities such as Support Vector Machine (SVM), Random Forests (RF) which are the representatives of Traditional Machine Learning (TML) and also some Deep Learning (DL) methods like Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN). However, neither TML nor DL is suitable for all kinds of situations and various datasets. As a result, we would like to explore more about such consequences. In this paper, we discover a discrepancy and phenomenon that different sizes of collected HAR datasets may produce influences on the effectiveness of traditional machine learning methods as well as the deep learning architectures. We conduct experiments on two kinds of different datasets USC-HAD and WISDM with the best accuracy nearly 90% in DL and 87% in TML. Due to the consequences of the experiments we give a conclusion on the individual heterogeneity problems of the HAR datasets--when dealing with the HAR datasets of small scales, the TML structures are more suitable. However, conversely, when the datasets have large amount of datasets. Specifically, DL approaches such as CNN and LSTM are more sensible choices.**

Keywords-HAR; deep learning; machine learning; USC-HAD; WISDM

I. INTRODUCTION

Human Activity Recognition (HAR) is witnessed as a very crucial problems at present filled with significant research challenges. It has a vast majority of applications including public health , medical care, personal surveillance, as well as security, brain-computer interface (BCI) , physical training , military and so on[1]. Furthermore, it mainly focuses on accuracy , robustness and real-time capability[2].Plenty of Inertial Measurement Unit (IMU) based technical facilities are used to pick up raw sensor-based data for HAR such as gyroscopes, accelerometers which is called acceleration-based method. It requires participants to wear a wide range of devices for collecting data instead of cameras. The data collected from these wearable sensing machines can approximately be used in recognizing activities of human-being on a basis of deep learning, machine learning as well as pattern recognition. With the advancement of technologies, nowadays, many electronic equipment have their own built-in accelerators which is convenient and handy for data collecting.

Today, deep learning has developed dramatically when applying in Computer Vision (CV) and Natural Language Processing (NLP) including methods like Stochastic Gradient Descent (SGD) which consists of procedures like showing the input vector for a few examples, next computing the outputs and the errors, and then computing the average gradient for those examples. The SGD have the advantage of adjusting the weights accordingly. While Convolutional Neural Networks (CNN) are designed to process data that comes in the form of multiple arrays. In addition, Recurrent Neural Networks (RNN) are produced for tasks that involve sequential inputs, such as speech and language which have the advanced models named Long Short-Term Memory (LSTM) . [3, 4]

Additionally, there also exist some traditional machine learning methods like Support Vector Machine (SVM) which maps the data into a higher dimensional input space and constructs an optimal separating hyperplane in this space[5]. Random Forests(RF) are a combination of tree predictors such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest[6] and .etc.

However, the two types of methods are not suitable for all the situations and datasets because they both have their own special characteristics and different architectures . So we can observe a lot of differences when using DL and TML to different kinds and sizes of HAR datasets. These can possibly attribute to the sensor data collected in HAR are all time series data which may differ individually. Thus, for the reasons like these, we are supposed to explore the effects and results when putting DL methods and TML methods into use on HAR sensor data.

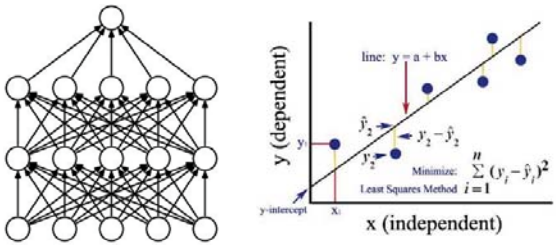


Figure 1. Different datasets with DL or TML methods.

In this paper, we first adopt two different settings in Fig.1 to impose researches on the effects of deep learning and traditional machine learning methods respectively using the raw data picked from HAR datasets. Moreover, we draw the

978-1-7281-6136-5/20/$31.00 ©2020 IEEE

conclusion that the traditional machine learning approaches are suitable for datasets which have relatively small scale. On the contrary, the deep learning methods may be suitable for the datasets that are based on a large scale of participants.

II. Related work

Many researches have done a load of work and made great efforts on different types of ways in dealing with the datasets about human activity recognition.. Some researches put forward the different activity recognition methods on a basis of distinguishing sensors such as radar, WIFI and so on which give the thoughts of how to collect sensor data in HAR. Subsequently, the researchers also make their attempts to try different approaches to deal with the human activities recognition Those attempts can probably be summed up in two sorts. Firstly, they carry out the human activity recognition using traditional machine learning methods like SVM, RF, Logistic Regression (LR) and perception. In addition, adopting deep learning approaches like CNN, LSTM to distinguish the activities from each other. The HAR datasets are also crucial with plenty of organizations and groups carry out experiments to collect sensor data from different human activities that are suitable for different situations. or to and finally various kinds of datasets concerning HAR.

A. Activity Recognition Based on Different Kinds of Sensor

The sensors utilized in the activity recognition can be of all kinds such as radar in [1]. The Radar based human gait recognition has been of great interest due to the increasingly smaller size and lower cost of software-defined and wireless radar platforms that are readily available. It has made possible applications of indoor radar which has the ability to facilitate gait recognition remotely, and in little or no light, without the potential of exposing the human body. The use of WIFI also occupy the vital position during activity recognition. The researchers in [2] present a novel real-time, device-free and privacy-preserving WiFi-enabled IoT platform for occupancy sensing, which are designed to owe an optimal tradeoff between performance and scalability. And with regard to the evaluation of the applications, the experimental results illustrate that the platform works efficiently with a high accuracy in terms of occupancy detection. Moreover, another type of sensor in Activity Recognition (AR) is perhaps visual. As in [3], it showed a novel recurrent convolutional architecture called Long-term Recurrent Convolutional Networks (LRCNs) suitable for large-scale visual learning which is end-to-end trainable, and demonstrate the value of these models on benchmark video recognition tasks, image description and retrieval problems. The video narration challenges utilizes UCF-101, COCO2014 datasets respectively and also uses the video from a CRF.

B. Traditional Machine Learning Methods

Recently, many TML methods can be used on HAR datasets for instance, random forests, SVM , KNN and so on. In [4], they adopted random forest classifier to validate the effects of their novel wearable system which was on a basis of a new set of 20 computationally efficient features. And furthermore they obtain very encouraging results with human activities recognition in a relatively high accuracy. In addition, in[5] , they developed two predictive models: a random forest classifier to predict activity type and a random forest of regression trees to estimate Metabolic Equivalents (METs) for evaluating the validity of correctly classifying types of Physical Activity (PA) behavior and predicting Energy Expenditure (EE) with the result of the hip accelerometer, obtaining an top average accuracy. And then, in [6], the AR coefficients are extracted as features for activity recognition with classification of the human activities performing with support vector machine. The average recognition results for four activities (running, still, jumping and walking) using the proposed AR-based features are seen very good and in the meanwhile, the highest classification accuracy is achieved by SVM used on the reference attributes and angles in [7].

C. Deep Learning Methods

Lately, deep neural network architectures have made significant progress in a great range of fields like pattern and image recognition, particularly the convolutional neural networks which is possibly the most efficient and effective deep models and widespread used in computer vision and etc. In [8] the researchers found that CNN works better than performance comparison classifiers SVM and an 8-layer Deep Belief Network (DBN) which can reach an significant average accuracy without any feature extraction methods. Besides, in [9], the researcher A. Ignatov found that the proposed model demonstrated state-of-the-art performance of CNN while requiring low computational cost and no manual feature engineering when testing on UCI HAR dataset.

In addition, when it comes to the LSTM, the researchers found that their approaches using LSTMs with temporal segments improves significantly over baseline and achieves comparable state-of-the-art performances. When it comes to [10], the researchers tackle the challenges through Ensembles of deep Long Short Term Memory (LSTM) networks which consist of the modified training procedures for LSTM networks and various sets of LSTM learners into classifiers. They focus on data-driven generation of diversity allowing for robust LSTM resembles which are energetic. Subsequently, in [11] ,the researchers rigorously explore deep, convolutional, and recurrent approaches across three representative datasets that contain movement data captured with wearable sensors as well as introducing a novel approach to regularization for recurrent networks. Some researchers also had the opinions that the combinations of CNN and RNN(LSTM) may have better effects on a basis of the recent success of recurrent neural networks for time series domains in [12]. They propose a generic deep framework for activity recognition based on convolutional and LSTM recurrent units. The results show that the framework outperforms competing deep non-recurrent networks on the challenge dataset .

D. Human Activity Recognition (HAR) Datasets

The **Opportunity dataset** consists of annotated recordings from on-body sensors from 4 participants instructed to carry out common kitchen activities. Data is recorded at a frequency of 30Hz from 12 locations on the body,

which is annotated with 18 mid-level gestures (e.g. Open Door / Close Door) in [13]. They deployed 72 sensors of 10 modalities in 15wireless and wired networked sensor systems in the environment , in objects, and on the body to yield over 25 hours of sensor data in [14].

The **PAMAP2 dataset** is a new dataset owing recordings from 18 activities performed by 9 participants, with accelerometer, gyroscope, magnetometer, temperature and heart rate data collected data from Inertial Measurement Units (IMU) that situated on chest, ankle and hand. The IMU last for a duration more over than 10 hours. The activities generally include household activities and a various kinds of exercise activities like playing football and so on in [15].

The **Skoda dataset** is a dataset covering the problem of recognizing activities of assembly-line workers inside a car production environment.[10]. The researchers collected data from 27 on-body sensors including seven IMUs, eight FSRs on each arm, and four Ubisense tags that work under different sampling frequencies. And at the same time, they are translated into 10 manipulative gestures such as checking the boot, opening/closing engine bonnet, boot and doors, and turning the steering wheel with the new string-matching-based method for spotting activities in continuous data streams. The top method in the benchmark has a 74 percent accuracy rate in a 560-minute-long recording in [16].

III. Main Work

We study HAR problem by means of both traditional machine learning methods such as SVM, KNN and random forests, and deep learning methods such as CNN and LSTM. We first introduce them while highlighting their characteristics and then introduce the effectiveness of the different structures and architectures used in the recognition of human activities and HAR datasets.

A. Support Vector Machine (SVM)

Support Vector Machine (SVM) is a generalized linear classifier utilizing the Hyper-margin hyperplane classifies data in a supervised learning manner. SVM can be nonlinearly classified by the kernel method, which is one of the common kernel learning methods.

Given that the question that separates the set of training data $(\boldsymbol{x_1}, y_1)$, $(\boldsymbol{x_2}, y_2)$, $\dots(\boldsymbol{x_m}, y_m)$ into two classes and $X_i \in R^N$ *or* $X_i = [x_1, x_2, x_3 \dots x_n]$ is a feature vector with the learning goal binary variable $y_i \in \{-1, +1\}$ seen as its class label representing positive class or negative class. The optimal values for w and b can be found by solving a constrained minimization problem, using Lagrange multipliers $\alpha_i (i = 1, \dots m)$ in which α_i and b are found by using an SVC learning algorithm in [17]. Those $\boldsymbol{X_i}$ with nonzero α_i are the "support vectors". For $\boldsymbol{K(x, y) = x \cdot y}$, this corresponds to constructing an optimal separating hyperplane in the input space R^N.[18] Mainly, we utilize the Gaussian Kernel in our paper which is shown in (2). The kernel functions can be vital to SVM because we notice that in the linear SVM dual problem, both the objective function and the decision function involve only the inner product between the input instance and the instance. The inner product of the objective function $x_i \cdot x_j$ can be substituted by kernel function $\mathrm{K}(x_i, x_j) = \varphi(x_i) \cdot \varphi(x_j)$ and the objective function is showed on (3) while the decision function is displayed as (4).

- $\mathrm{f}(\mathbf{x}) = \mathrm{sign}(\sum_{i=1}^{m} \alpha_i \, y_i K(\boldsymbol{x_i}, \boldsymbol{x}) + b)$ (1)
- $\mathrm{K}(X_i, X) = \exp\left(-\frac{\|X_i - X\|^2}{2\sigma^2}\right)$ (2)
- $\mathrm{W}(\alpha) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j \, y_i y_j K(x_i, x_j) - \sum_{i=1}^{N} \alpha_i$ (3)
- $\mathrm{f}(\mathrm{x}) = \mathrm{sign}\left(\sum_{i=1}^{N} \alpha_i y_j \varphi(x_i) \cdot \varphi(x) + \mathrm{b}\right) = \mathrm{sign}(\sum_{i=1}^{N} \alpha_i y_i K(x_i, x) + b)$ (4)

However, the SVM algorithm is usually more suitable for dividing the linear models which seems to not have a satisfying result dealing with HAR. It is probably typically useful for HAR datasets with specific characteristics.

B. K-Nearest Neighbor(KNN)

The KNN algorithm extracts the classification label of the most similar data (nearest neighbor) of the feature in the sample set. And finally, the category with the most occurrences in K is selected as the classification of the new data where x_i and x_j mean the two kinds of classes and p means 1,2,3⋯,n. Generally speaking, the most important step is to calculate the distance between the values. Some approaches to calculate the distances are as following (5)-(8).

- The Lp distance: $l_p(x_i, x_j) = (\sum_{i=1}^{N} \left|x_i^{(l)} - x_j^{(l)^p}\right|)^{\frac{1}{p}}$ (5)
- The Euclidean distance: $l_2(x_i, x_j) = (\sum_{i=1}^{N} \left|x_i^{(l)} - x_j^{(l)^2}\right|)^{\frac{1}{2}}$ (6)
- The Manhattan distance: $l_1(x_i, x_j) = \sum_{i=1}^{N} \left|x_i^{(l)} - x_j^{(l)}\right|$ (7)
- The L∞ distance : $l_\infty(x_i, x_j) = \max\left|x_i^{(l)} - x_j^{(l)}\right|$ (8)

In this paper, we choose the parameters with neighbors equal to 10 and impose distance as weights as well as set Multi-fold cross validation in 20 when utilizing the KNeighborsClassifier. The traditional machine learning methods KNN may have increasingly worse effects as the increase in the neighbors in HAR as a result of the complexity in time. So as the participants in the datasets become larger, the KNN may probably not have a satisfying results.

C. Random Forests(RF)

Random Forests (RF) is an algorithm that integrates multiple trees through the idea of integrated learning when its basic unit is the decision tree, and its essence belongs to a branch of machine learning - Ensemble Learning and each decision tree is a classifier. The base of the RF is information gain. It demonstrates information of the feature X is known to reduce the uncertainty of the information of the class Y, and generally the difference between entropy H(Y) and condition entropy H(Y|X) is called the mutual information as the next formula (9) shows how to calculate information gate.

- $\mathrm{H}(\mathrm{Y}, \mathrm{X}) = \mathrm{H}(\mathrm{Y}) - \mathrm{H}(\mathrm{Y}|\mathrm{X})$ (9)

We set the criterion as 'gini' and impose n_estimators on 200 as parameters when using RandomForestClassifier on a basis of sklearn. But the random forest algorithm is prone to overfitting if have some wrong parameters so when the datasets turn larger, the RF may be overfitting not lead to the willing results as we think.

D. Convolutional Neural Networks(CNN)

Convolutional neural networks have significant effects to identify the different prominent patterns of HAR's signals and collected data. Specifically, the lower layers obtain the local information aimed at capturing the basic movement in HAR while the higher layers gain the signals representing for high-level. It is noticeable that each layer might probably have a range of convolution and pooling tier as well as fully connected layers with different parameters. The Convolutional Neural Network (CNN) has established itself as a powerful technique for human activity recognition where convolution and pooling layers are applied along the temporal dimension of sensor signals [19]. And generally a sliding window strategy is very important adopted to divide the time series signal into short pieces of signals which are suitable for CNN. The overall CNN structure used in this paper is shown in Fig.2. Aiming at developing more accurate and decisive activity recognition algorithm, we adopted techniques such as pooling, dropout regularization, the Rectified Linear Unit (ReLU) activation function and soft-max classifier that are popularly used in several deep learning tasks.

The input vector of sensor data is defined as $x_k^0=[x_1, x_2, x_3, \cdots x_M]$, when M represents for the amount of sensor data in every window after completing the segmentation.

The output of the convolutional layer might be measures as following, in which g represents for the index of the convolutional layer, A illustrates the size of the convolutional kernel and w_a^i shows the weight in i^{th} feature map a^{th} index of the sliding kernel.

$$c_k^{t,i} = \alpha(b_i + \sum_{a=1}^{A} w_a^i x_{i+a-1}^{0,i}) \tag{10}$$

In addition, we use MaxPooling1D and MaxPooling2D in this paper respectively during CNN while the output of the max-pooling operations are selecting the max value among the inputs and it can be measured below with the explanation that the pooling size is ***N,*** and ***S*** represents the stride.

$$g_k^{t,i} = maximum(c_{n+k\times s}^{t,i}) \quad n \in N \tag{11}$$

When utilizing the feature vector into the fully-connected layer and with soft-max classifier to distinguish the activities, so there is a great need to flatten it as $g^t = [g_1, g_2, g_3, \cdots g_\varphi]$ where φ illustrates the units of the final pooling operation. Subsequently, finally, we apply the flattened feature vector into the dense layer with the soft-max function.

The deviation and error during the forward propagation are then applying to back propagation periods and assuming the loss function is Loss, we want to solve the gradient of the loss function versus the weight ***W*** and the bias ***b***. Firstly, we assume the output of the l^{th} convolution layer are:

$$Z^{[l]} = \sum_m \sum_n W_{m'n'}^{[l-1]} Z_{m,n}^{[l-1]} + b^{[l]} \tag{12}$$

In which the $Z_{m,n}^{[l-1]}$ represents the output of the previous pooling layer, and at the same time, $\boldsymbol{m'}, \boldsymbol{n'}$ show the size of the kernel size and in addition ***m, n*** illustrates the length and width of the inputs.

When it comes to the gradient of the loss function to the parameter ***W*** and ***b***, it uses the chain rule and given by (13)-(15):

- $\frac{\partial Loss}{\partial W_{m'n'}^{[l]}} = \sum_{i=0}^{H} \sum_{i=0}^{H} dZ_{i,j}^{[l]} Z_{i+m',j+n'}^{[l-1]}$ (13)
- $\frac{\partial Loss}{\partial b^{[l]}} = \sum_{i=0}^{H} \sum_{i=0}^{H} dZ_{i,j}^{[l]}$ (14)
- $dZ_{i,j}^{[l]} = \frac{\partial Loss}{\partial Z_{i,j}^{[l]}} = \frac{\partial Loss}{\partial Z^{[l+1]}} \frac{\partial Z^{[l+1]}}{\partial Z^{[l]}} = dZ^{[l+1]} \frac{\partial Z^{[l+1]}}{\partial Z^{[l]}}$ (15)

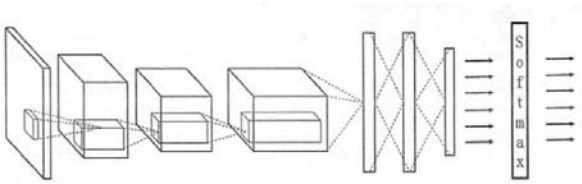


Figure 2. The CNN architecture.

E. Long Short-Term Memory(LSTM)

Long Short Term Memory Network (LSTM), which successfully solves the defects of the original cyclic neural network, has become the most popular RNN. Additionally, LSTM has been successfully applied in many fields such as speech recognition, picture description, and natural language processing. This architecture is recurrent because a few of the junctions in the LSTM network form a cycle, in which the current time ***t*** is thought to be the conditions of the previous time ***t-1*** .Subsequently, LSTM units or also called cells are designed to avoid the influence of the diminishing gradients when incorrect derivatives are being backpropagated through a plenty of layers in recurrent networks[20]. The LSTM uses two gates to control the contents of the cell state $\boldsymbol{C_t}$. First is the forget gate, which determines how much of the cell state at the previous moment is retained to the current moment. Secondly it is the input gate that illustrates to what extent the input to the network at the current time is saved to the cell state. In addition, the LSTM utilizes an output gate to control the number of the cell state is given to the current output value of the network. The forget gate is given by:

$$f_t = \sigma\big(W_f \cdot [h_{t-1}, x_t] + b_f\big) \tag{16}$$

In which w_f represents the weight matrix of the forget gate while $[h_{t-1}, x_t]$ shows joining two vectors into one longer vector. Furthermore, σ illustrates the sigmoid function. Similarly, the input gate is as following:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{17}$$

Next we calculate the cell state $\widetilde{\boldsymbol{C_t}}$ used to describe the current input, which is calculated based on the last output and current input simultaneously which is shown below:

$$\widetilde{C_t} = \tanh(W_h \cdot [h_{t-1}, x_t] + b_c) \tag{18}$$

And at present we calculate the current cell state $\boldsymbol{C_t}$ which is made up of the previous cell state $\boldsymbol{C_{t-1}}$ multiply in elements by the forget gate $\boldsymbol{f_t}$ and then utilizing the currently input cell state $\widetilde{\boldsymbol{C_t}}$ multiply in elements by the input gate $\boldsymbol{i_t}$ gathering together that is given by:

$$C_t = f_t \cdot* C_t + i_t \cdot* \widetilde{C}_t \tag{19}$$

When specifically, $\cdot*$ means multiplying by elements. And in this way, because of the control of the forget gate, the LSTM network can save the information from a long time ago and at the same time can avoid current insignificant content entering the memory. The output gate is displayed as following:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{20}$$

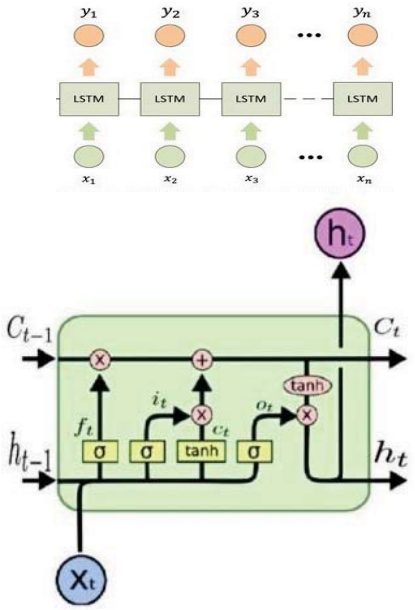


Figure 3. The LSTM architecture.

Finally, the most important part is the final output of the LSTM which is decided by both the output gate and cell state that are illustrated by:

$$h_t = O_t \cdot* \tanh(c_t) \tag{21}$$

Formulas (19)-(24) have concluded all the forward propagation of the LSTM. We also have the formula that shows the error deviation passing forward at any k time described by:

$$\delta_k^T = \prod_{j=k}^{t-1} \delta_{o,j}^T W_{oh} + \delta_{f,j}^T W_{fh} + \delta_{i,j}^T W_{ih} + \delta_{\tilde{c}_t}^T W_{ch} \tag{22}$$

We train the LSTM architecture for one to three LSTM layers and then flatten the feature vector. Then we connect it to the Fully Connected (FC) dense layer that is visualized in Fig.3. We notice a better evaluation results with the single LSTM layer.

IV. Experiments

While in the following section of two parts we will illustrate the datasets and evaluation, experiments of the traditional machine learning methods and deep learning approaches.

A. USC-HAD and WISDM Dataset

The dataset University of Southern California Human Activity Dataset (USC-HAD) is specifically produced to sum up the most basic and common human activities in daily life from a large scale and various group of human subjects. The USC-HAD dataset is composed of 14 subjects (7 male, 7 female) to participate in the data collection within twelve activities that are among the most basic and common human activities in people's daily lives. The more detailed information is shown in the following Table 1 and 4,.and the visualization of the dataset is illustrated in Fig.4 [21].

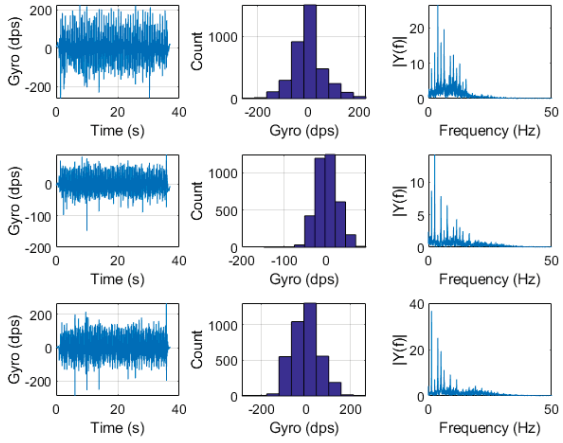
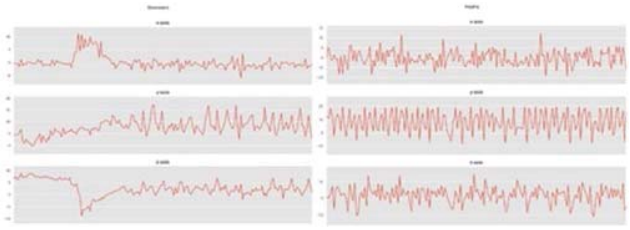


Figure 4. The visualization of the USC-HAD dataset.

The other dataset is WISDM (Wireless Sensor Dating Mining) which explores the research issues related to mining sensor data from the powerful mobile devices like mobile phones. It builds useful applications which is made up of six activities: walking, jogging, ascending stairs, descending stairs, sitting, and standing where each example is labeled with the activity that occurred while the data was being collected. In addition, the dataset record acceleration in three axes where the z-axis captures the forward movement of the leg and the y-axis captures the upward and downward motion. Fig.5 plots the accelerometer data for all three axes of six activities performed by a single user. We take the jogging and downstairs as two examples to show as the following. Specifically, for most activities the y values have the largest accelerations as a result of Earth's gravitational pull and Fig.6 represents the training examples by activity type and user [22].



(a)downstairs (b)jogging

Figure 5: Acceleration Plots for the Two Typical Activities

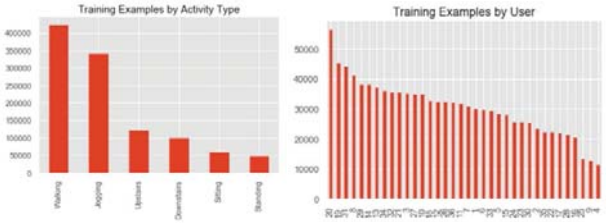


Figure 6. Training Examples by activity type and user.

TABLE I. STATISTICS OF THE PARTICIPATING HUMAN SUBJECTS

	Age	Height(cm)	Weight(kg)
range	21-49	160-185	43-80
mean	30.1	170	64.6
std	7.2	6.8	12.1

B. Description and Result of the Experiments

1) Experiments of the TML structures

In order to explore deeply and find whether the Traditional Machine Learning (TML) algorithm has the more abilities to deal with such small scale dataset in comparison with the Deep Learning (DL) algorithm we make up our attempts to train the HAR datasets WISDM and USC-HAD with KNN, SVM as well as the random forests that are all the Traditional Machine Learning Methods (TML) classifiers to contrast with the effectiveness with the DL approaches. The following Fig.7 and Fig.8 as well as Table 2 and Table 3 illustrate the more details about the experiments on the TML.

TABLE II. THE EVALUATION ON THREE TML METHODS UTILIZING WISDM DATASET

TML approaches	SVM	KNN	Random Forests
Test accuracy	77%	67.8%	**82.7%**
F1 score	68%	62%	**73%**

TABLE III. THE EVALUATION ON TRADITIONAL MACHINE LEARNING METHODS USING USC-HAD DATASET

TML approaches	SVM	KNN	Random Forests
Test accuracy	52.5%	52.1%	**67.9%**
F1 score	35%	**62%**	41%

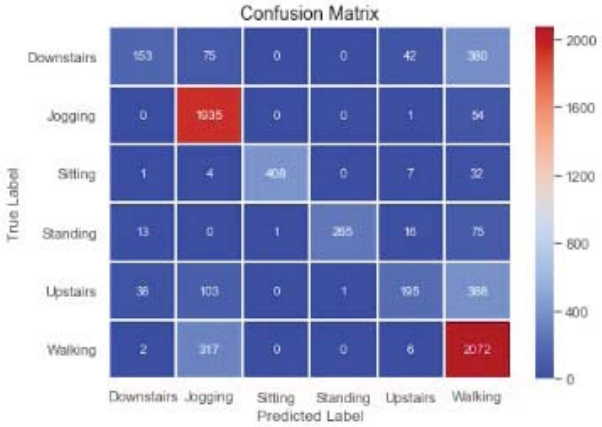
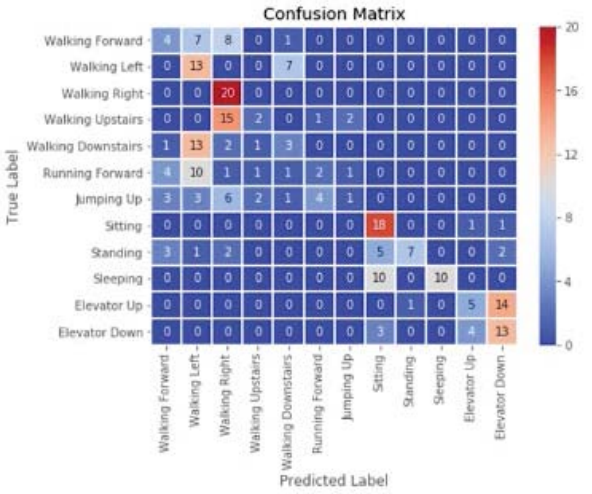
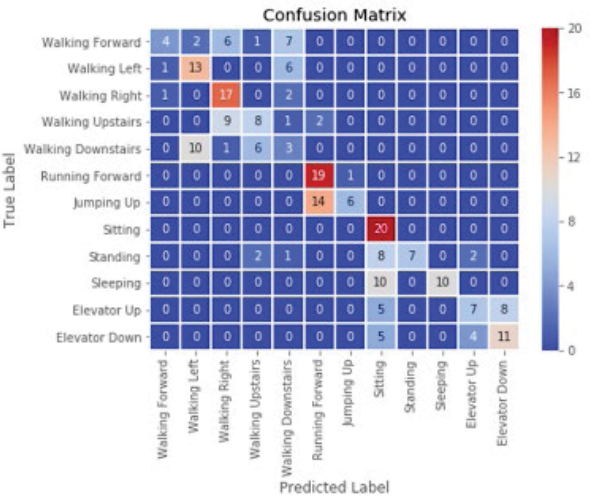


Figure 7. The confusion matrix on Random Forests(the best in WISDM).



(1)KNN



(2)Random Forests

Figure 8. The confusion matrix on KNN and RF using USC-HAD dataset.

From the aforementioned figures and tables, we have the opinions that traditional machine learning methods really have

the better effects than the deep learning algorithms when the training datasets are in a small scale and different people are used in training and test processes. Conversely, the deep learning approaches are supposed to on a basis of a relatively huge number of data in the datasets and more suitable for the division of the training and test datasets which are mainly the same or similar to each other.

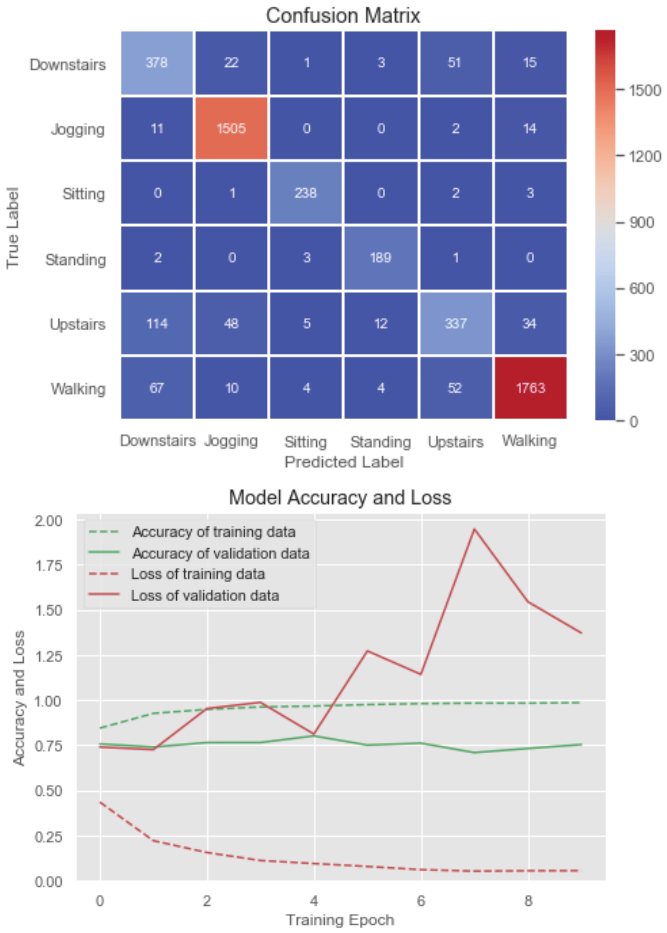
2) Experiments of the DL architectures

CNN: Firstly, with regard to the CNN we had mainly two kinds of approaches of Convolutional Neural Network (CNN), Conv1D and Conv2D respectively used in two representative HAR datasets USC-HAD and WISDM to explore the different suitable situations for applying Traditional Machine Learning(TML) and Deep Learning(DL). We have done several attempting experiments to probe into the suitable convolutional layers and finally, we establish three distinguishing sorts of Conv1D architectures with convolutional layers ranging from one to three in each structure. In terms of the convolutional kernel size as to it was elaborated in [8], there are only three options: 1, 2 or 3 in order to extract information between the three different axes so they set the convolution kernel width to 2. In fact, we try the convolution kernel with the width of 2 and it performs the best with our models .When it comes to the Conv2D modeling, in order to confirm the comparison abilities and the reliability of our experiments we decide to utilize the same conditions. They are arranged in the infrastructure corresponding to Conv2D with some different parameters. We choose the two types of prototypes are both utilized the ReLU activation function which are regularly used in plenty of experiments completed before in some paper which can always show great efforts and results. Moreover, we make use of the soft-max classifier in the last fully connected (dense) layer and the categorical cross-entropy as the loss function.

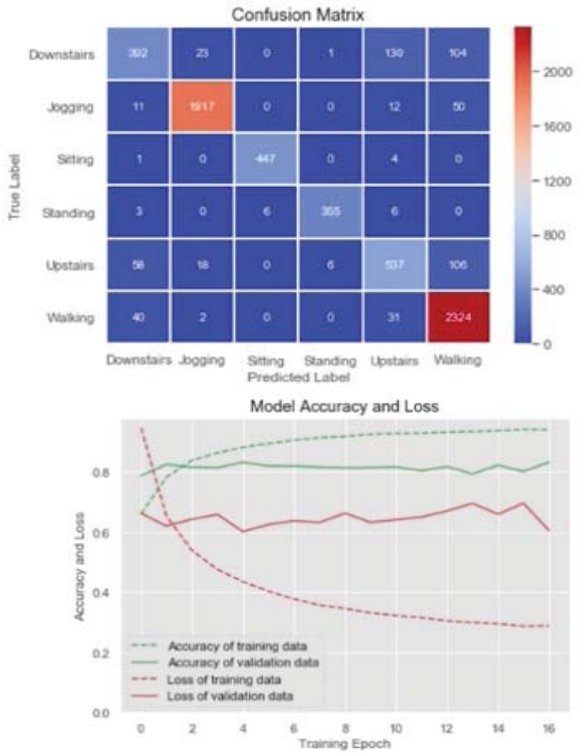
TABLE IV. ACTIVITIES AND THEIR BRIEF DESCRIPTIONS

	Activity	Description
1	Walking forward	The subject walks forward in a straight line
2	Walking left	The subject walks counter-clockwise in a full circle
3	Walking right	The subject walks clockwise in a full circle
4	Walking upstairs	The subject goes up multiple flights
5	Walking downstairs	The subject goes down multiple flights
6	Running forward	The subject runs forward in a straight line
7	Jumping	The subject stays at the same position and continuously jumps up and down
8	Sitting	The subject sits on a chair either working or resting. Fidgeting is also considered to belong to this class
9	Standing	The subject stands and talks to someone
10	Sleeping	The subject sleeps or lies down on a bed
11	Elevator up	The subject rides in an ascending elevator
12	Elevator down	The subject rides in a descending elevator

Subsequently, we tried a wide range of optimizers to optimize the established architectures of the CNN network. For attempts, we apply optimizers SGD which is the most common optimization method for calculating the mini-batch gradient every iteration and then updating the parameters. However, when using SGD, choosing the right learning rate is difficult. And it tends to converge to local optimum. Subsequently, in some cases, it may be trapped at the saddle point. Secondly, we utilize the Momentum optimizer with the characteristics that accelerate SGD in the relevant direction as well as accelerating convergence. But it might be not such suitable in our experiments. Then we make attempts to use the Adagrad optimizer which has shown the drawbacks that it still relies on manually setting a global learning rate. In addition, if the setting is too large, the regularizer will be too sensitive and the adjustment of the gradient might possibly be too large. So, finally we choose the RMSprop optimizer which is really significant for handling non-stationary targets. The confusion matrix as well as the line charts of the model accuracy and loss training of the WISDM and USC-HAD datasets can clearly be seen in Fig.9 and 10 respectively and for more details about the experiments, we show them in Table 5 and 6 severally.



(1) Conv2D structure with two convolutional layers on WISDM(the best)



(2) Conv1D structure with one convolutional layer on WISDM(the best)

Figure 9. The best confusion matrix and line charts in WISDM.

TABLE V: THE CLASSIFICATION ACCURACY ON SIX ACTIVITIES OF WISDM DATASET

Activities\Architecture	**1 to 3 Conv1D layer(s)**		
Downstairs	61.23%	**85.08%**	70.62%
Jogging	94.72%	**95.73%**	94.92%
Sitting	**99.56%**	99.34%	91.81%
Standing	**96.22%**	66.76%	87.3%
Upstairs	57.79%	65.1%	**70.48%**
Walking	97.37%	97.87%	**98.33%**
Test accuracy	**91%**	**91%**	90%
Baseline error	**11.27%**	9.30%	9.57%
F1 score	90%	**91%**	**91%**
Activities\Architecture	**1 to 3 Conv2D layer(s)**		
Downstairs	68.09%	**80.43%**	57.54%
Jogging	97.26%	**98.24%**	96.03%
Sitting	96.72%	97.54%	**99.78%**
Standing	93.33%	**96.72%**	92.7%
Upstairs	53.64%	**61.27%**	60.97%
Walking	91.58%	**92.79%**	83.94%
Test accuracy	87.2%	**90.2%**	84%
Baseline error	12.84%	9.83%	**15.96%**
F1 score	83%	**86%**	84%

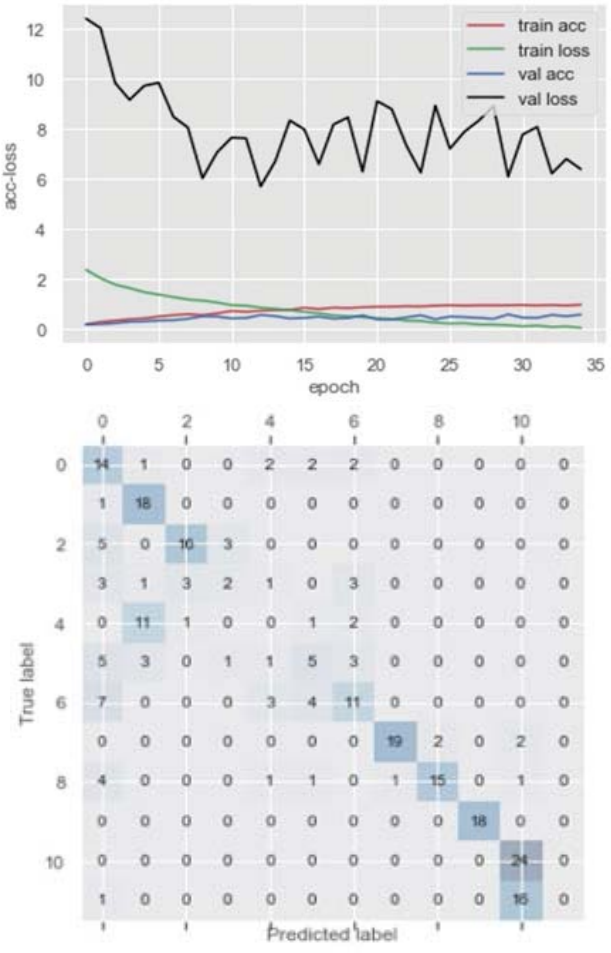
The Table 5 shows that utilizing the two convolutional layers might have the most significant results in comparison with the other two other situations. And totally speaking, CNN is good at recognizing the activities of jogging, sitting, standing and walking but may possibly cannot tell the differences from upstairs and downstairs. Mainly because such two activities are so similar to each other.

TABLE VI. THE EVALUATION ON THREE DIFFERENT STRUCTURES IN CONVOLUTIONAL LAYERS

Evaluation\Architecture	**1 to 3 Conv2D layer(s)**		
Test accuracy	**59.2%**	45.4%	36.3%
F1 score	**59%**	45%	36%

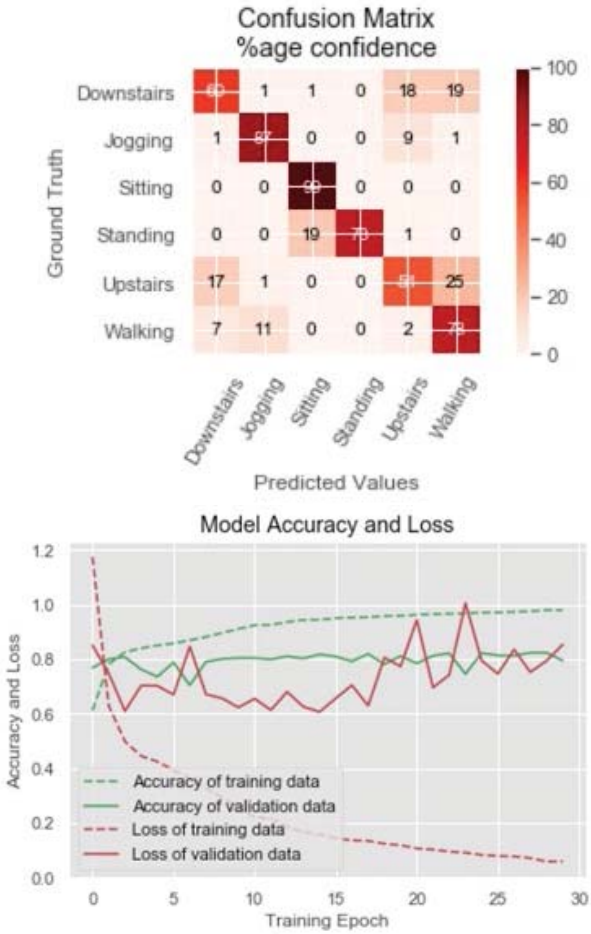
From the Table 6 we find that the deep learning methods like CNN might probably be poor at recognizing the HAR activities when the dataset is in a small scale and it include the different human beings while during the training process as well as the testing process.

LSTM: When it comes to the LSTM architecture in deep learning, we build from one to three layers of LSTM the same suitable as the CNN structure with the timeperiods setting at 80. The confusion matrix and the line charts of the model accuracy and loss on the datasets WISDM and USC-HAD are able to see in Fig.11 and Fig.12 respectively and for more details about the LSTM experiments, Table 7 and 8 shows them.



Conv2D structure with one convolutional layer on USC-HAD(the best)

Figure 10. The confusion matrix and line charts of the model accuracy and loss training on USC-HAD.



LSTM structure with two LSTM layers on WISDM(the best)
Figure 11. The confusion matrix and line charts on WISDM.

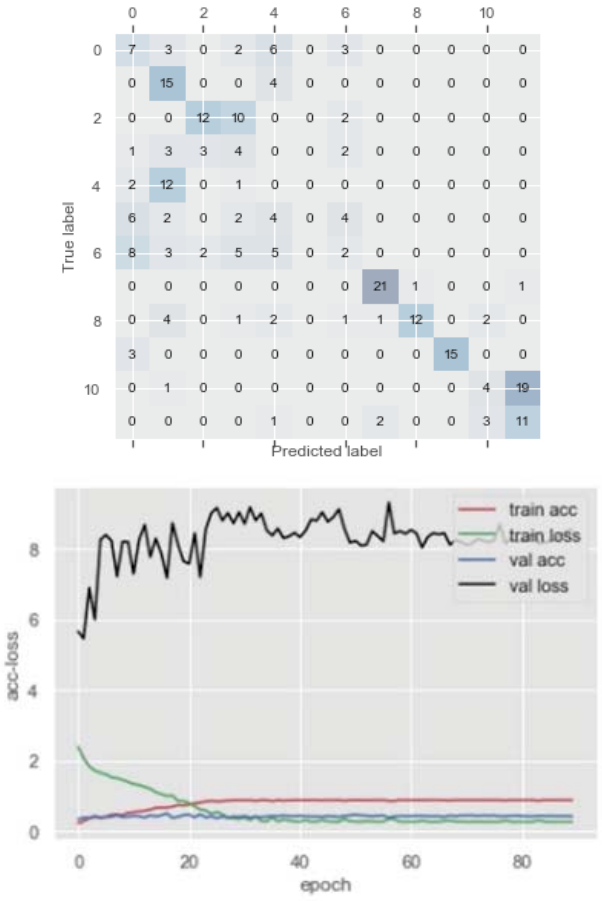
TABLE VII. THE CLASSIFICATION ACCURACY ON 3 DIFFERENT STRUCTURES ON WISDM

Activities\Architecture	1 to 3 LSTM layer(s)		
Downstairs	46.46%	**62.15%**	60.15%
Jogging	89.75%	**94.87%**	87.74%
Sitting	**99.56%**	99.34%	**99.56%**
Standing	74.56%	**95.95%**	78.11%
Upstairs	**62.76%**	60.97%	54.07%
Walking	**82.1%**	75.09%	78.43%
Test accuracy	80%	**81%**	78%
Baseline error	20.46%	18.92%	**21.81%**
F1 score	80%	**81%**	80%

From Table 7, we get to know that the two LSTM layers might be the most suitable for analyzing the HAR dataset-WISDM. In addition we discover clearly that LSTM architecture have more negative effects and efforts than the CNN structure of all the three sorts.

TABLE VIII. THE EVALUATION ON THREE DIFFERENT STRUCTURES TRAINING ON USC-HAD

Evaluation\Architecture	1 to 3 LSTM layer(s)		
Test accuracy	**42.9%**	38.3%	24.6%
F1 score	**43%**	37%	25%



LSTM architecture with one LSTM layer on USC-HAD(the best)
Figure 12. The confusion matrix and line charts on USC-HAD.

From Table 8, we discover that the one LSTM layer might be the best structure for the dataset-USC-HAD. Moreover, we find out the same phenomenon as the WISDM dataset that LSTM architecture truly makes a worse job than CNN. Specifically, it is obvious that LSTM or we can say deep learning algorithm may be not fit for the little range dataset like USC-HAD. Perhaps it may on a basis of a larger scale and will show a better consequence.

V. CONCLUSION

In this paper, we find the differences between the effects and results of deep learning approaches and traditional machine learning methods when dealing with and analyzing the various sizes of the choosing datasets. Hence we select two kinds of different HAR datasets –WISDM with a large scale range of collected sensor data while the USC-HAD has a relatively smaller amount of data. In order to explore deeply about this phenomenon therefore, we utilize the traditional deep learning methods like KNN, SVM as well as the random forests for the validation of our discoveries. Finally we jump to the conclusion towards the individual heterogeneity problems of the HAR datasets. We find that when the HAR datasets are in a small scale and the participants who are

collected sensor data from are in a small group as well and then the traditional machine learning structures are more applicable to obtain a probably satisfying testing accuracy and results. However, when the datasets have the characteristics of a large scale and specifically, deep learning methods such as CNN and LSTM are better choices.

REFERENCES

[1] M. S. Seyfioglu, A. M. Ozbayoglu, and S. Z. Gurbuz, "Deep convolutional autoencoder for radar-based classification of similar aided and unaided human activities," *IEEE Transactions on Aerospace and Electronic Systems,* vol. 54, no. 4, pp. 1709-1723, 2018.

[2] J. Yang, H. Zou, H. Jiang, and L. Xie, "Device-Free Occupant Activity Sensing Using WiFi-Enabled IoT Devices for Smart Homes," *IEEE Internet of Things Journal,* vol. 5, no. 5, pp. 3991-4002, 2018.

[3] J. Donahue *et al.*, "Long-term recurrent convolutional networks for visual recognition and description," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 2625-2634.

[4] P. Casale, O. Pujol, and P. Radeva, "Human Activity Recognition from Accelerometer Data Using a Wearable Device," in *Pattern Recognition and Image Analysis*(Lecture Notes in Computer Science, 2011, pp. 289-296.

[5] K. Ellis, J. Kerr, S. Godbole, G. Lanckriet, D. Wing, and S. Marshall, "A random forest classifier for the prediction of energy expenditure and type of physical activity from wrist and hip accelerometers," *Physiol Meas,* vol. 35, no. 11, pp. 2191-203, Nov 2014.

[6] Z.-Y. He and L.-W. Jin, "Activity recognition from acceleration data using AR model representation and SVM," in *2008 international conference on machine learning and cybernetics*, 2008, vol. 4, pp. 2245-2250: IEEE.

[7] M. Luštrek and B. J. I. Kaluža, "Fall detection and activity recognition with machine learning," vol. 33, no. 2, 2009.

[8] Y. Chen and Y. Xue, "A Deep Learning Approach to Human Activity Recognition Based on Single Accelerometer," presented at the 2015 IEEE International Conference on Systems, Man, and Cybernetics, 2015.

[9] A. Ignatov, "Real-time human activity recognition from accelerometer data using Convolutional Neural Networks," *Applied Soft Computing,* vol. 62, pp. 915-922, 2018.

[10] Y. Guan, T. J. P. o. t. A. o. I. Plötz, Mobile, Wearable, and U. Technologies, "Ensembles of deep lstm learners for activity recognition using wearables," vol. 1, no. 2, p. 11, 2017.

[11] N. Y. Hammerla, S. Halloran, and T. J. a. p. a. Plötz, "Deep, convolutional, and recurrent models for human activity recognition using wearables," 2016.

[12] F. J. Ordonez and D. Roggen, "Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition," *Sensors (Basel),* vol. 16, no. 1, Jan 18 2016.

[13] R. Chavarriaga *et al.*, "The Opportunity challenge: A benchmark database for on-body sensor-based activity recognition," vol. 34, no. 15, pp. 2033-2042, 2013.

[14] D. Roggen *et al.*, "Collecting complex activity datasets in highly rich networked sensor environments," in *2010 Seventh international conference on networked sensing systems (INSS)*, 2010, pp. 233-240: IEEE.

[15] A. Reiss and D. Stricker, "Introducing a New Benchmarked Dataset for Activity Monitoring," presented at the 2012 16th International Symposium on Wearable Computers, 2012.

[16] T. Stiefmeier, D. Roggen, G. Ogris, P. Lukowicz, and G. Tr, "Wearable Activity Tracking in Car Manufacturing," *IEEE Pervasive Computing,* vol. 7, no. 2, pp. 42-50, 2008.

[17] V. Vapnik, *The nature of statistical learning theory*. Springer science & business media, 2013.

[18] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: a local SVM approach," in *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, 2004, vol. 3, pp. 32-36: IEEE.

[19] S. Ha and S. Choi, "Convolutional neural networks for human activity recognition using multiple accelerometer and gyroscope sensors," in *2016 International Joint Conference on Neural Networks (IJCNN)*, 2016, pp. 381-388: IEEE.

[20] S. Hochreiter, Y. Bengio, P. Frasconi, and J. Schmidhuber, "Gradient flow in recurrent nets: the difficulty of learning long-term dependencies," ed: A field guide to dynamical recurrent neural networks. IEEE Press, 2001.

[21] M. Zhang and A. A. Sawchuk, "USC-HAD: a daily activity dataset for ubiquitous activity recognition using wearable sensors," in *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, 2012, pp. 1036-1043: ACM.

[22] J. R. Kwapisz, G. M. Weiss, and S. A. J. A. S. E. N. Moore, "Activity recognition using cell phone accelerometers," vol. 12, no. 2, pp. 74-82, 2011.