Article

Comparative Study on Hand-Crafted Features for Human Activity Recognition Using Sensory Data

Hosein Nourani [1,†,‡] **and Emad Shihab** [1,‡]

[1] Dept. of Computer Science and Software Engineering; h.nourani@hotmail.com

[2] Dept. of Computer Science and Software Engineering; e.shihab@concordia.com

* Correspondence: e-mail@e-mail.com; Tel.: (optional; include country code; if there are multiple corresponding authors, add author initials) +xx-xxxx-xxx-xxxx (F.L.)

† Current address: Affiliation 3

‡ These authors contributed equally to this work.

Version December 12, 2019 submitted to Journal Not Specified

Abstract: Human Activity Recognition (HAR) using sensory data refers to an emerging area of interest for healthcare, military, and security applications. Several researches are conducted to capture a certain activity from an stream of data. Basically, the most popular methods extract some attributes (features) of a signal and apply a pattern recognition model on them. There are several studies that they have proposed different set of features that show the performance is significantly improved. However, since each result have been achieved under its own setup, comparing the impact of different featuresets can not be made in a distinct form. Therefore, in this work, we split a HAR setup into its three main characteristics including dataset (types of activities), classifiers, and evaluation methods; Then, we assess the impact of using different featuresets in each characteristic of setup, separately. Toward this end, we address three challenges: (1) choosing featureset, (2) choosing classifier, and (3) choosing evaluation method. We present cross-validation results on 20 different models using 5 featuresets and 4 classifiers. For experiments, We create a dataset of 8 complex gym exercises from 13 subjects over several sessions. Results showed that models on histogram-bin features deliver the best performance (on average 87.80% of F1) relatively better than general statistical features. Among classifiers, the average classification performance of Forward Neural Network (FNN) model is reported the highest performance (95.89% of F1) using histogram bins in k-fold cross-validation. FNN in Leave-One-Trial-Out cross-validation and Leave-One-Subject-Out cross-validation achieved 89.66% and 81.59% respectively. This study provides significant experimental results on building a HAR model under realistic conditions.

Keywords: Feature Extraction; Featureset; Wearable; Motion Sensor; Neural Network, Histogram, Human Activity Recognition

1. Introduction

With the rise of life expectancy and aging of population, the development of new technologies that focuses on elderly healthcare has become a challenge [1]. Fall risk assessment of elderly patients [2], physical fitness monitoring[3], medical diagnosis [4] are to name a few. Human Activity Recognition (HAR) using Wearables is one of the most promising assistive technologies to support older people's daily life [5]. Simultaneously, using Wearables are increasingly pervasive. Wearables are small in size, relatively cheap and ubiquitously used, which has enabled enormous potential in human-centred applications. Therefore, implementation a system that uses the device resources aiming users in healthcare and improving the activity performance in the form of recognizing a certain activity or counting it [6] is vital.

Wearable sensor-based HAR systems basically share a similar approach, as shown in Figure 1. The procedure starts with recording an activity by single or multiple sensors[7]. Motion-based sensors are the most popular in HAR, which are capable to measure a movement by different metrics (e.g., acceleration, angular velocity, shake, magnetic field) [8]. When a sensor is attached to the human body, it can capture the motion of that part of the body as shown in Figure 1 step 1. In literature[3,5,7], authors have used different positions like chest, wrist, pocket, and foot to attach the sensor.

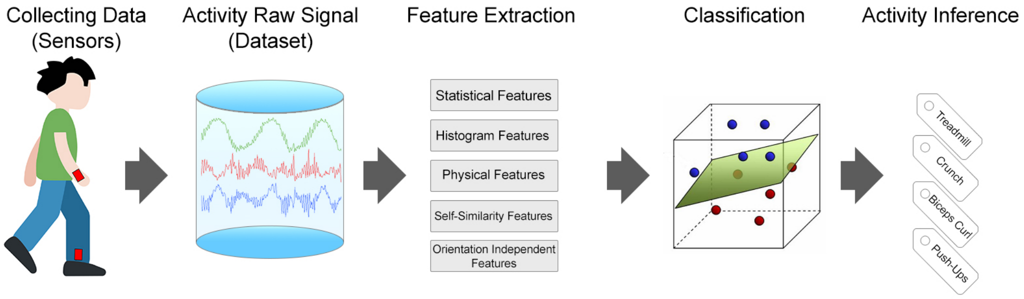


Figure 1. Typical Work-flow on Human Activity Recognition

The data collected by sensors as *time series raw signal* will be segmented into smaller pieces (Figure 1 step 2) to become easier on classification phase. In step 3, useful representative features (e.g., statistical features) for distinguishing activities are extracted from each segment - *feature extraction*[5,9]. The data extracted from step 3 will be considered as input for *classification* in step 4. The classification output is the activity name in step 5. At this step, different *evaluation* method can asses the performance of the model.
Basically, extracting features is based on a comprehensive understanding of physically how an activity has been done. Among different type of features, statistical features has been received the most attention in HAR studies. However, in recent years, other type of features, like frequency domain features and hybrid features have been combined with statistical features with the aim of improving activity recognition performance[3,5]. Although there are several studies in literature that show these featuresets outcome remarkable model performance, to the best of our knowledge, there is no empirical study investigating the performance of each set of features. Specifically, there is not a comparison study, considering different classifiers and evaluation methods under same dataset and experimental setup. One simple solution is to extract more features, however, more number of features causes more energy consumption, which can be problematic for an energy-limited device, like a smartphone[10]. Hence, such a detailed analysis can help in deciding when to best use each set of features. Therefore, there is a need to study the impact of these featureset in detail. In particular, we focus on this research question: "How and when are various featuresets, which are all state-of-the-art, best used for better recognition performance (RQ1)".

Some researchers have already investigated the impact of various features in activity recognition[3,9,11]. For example, in [3], the authors use statistical features in combination with auto-correlation features using SVM classifier and report a robust HAR system on gym exercises with around 99% of accuracy on their own dataset. On the other hand, in [12], the authors claim that the addition of physical features, feature based on physical interpretation, to statistical features while employing a multi-level classification, improves the performance to 90% of accuracy. Although the first study has reached a better performance than second study, it does not necessarily mean that the auto-correlation features are 9% more informative than physical features. These two papers are showing different results probably due to their different experimental setups. In order to compare the impact of each featureset, it is important to examine them under same experimental setup. However, these previous studies have investigated featuresets on different dataset and classification method. In order to have general answer to RQ1, we need to know the role of other factors like classification method on HAR model. In other words, we are aiming to answer: "How different do classifiers perform on different featuresets (RQ2)". We chose four classifiers in order to cover the most commonly used classification methods in the previous studies. These classifiers are: Support-Vector Machine (SVM), Feed-Forward Network (FNN), Decision Tree (DT), and K-Nearest Neighbor (KNN).

In 2018, Jordao et al. [13] revealed a basic issue regarding the process of extracting features. They showed that the traditional process of generating data points is vulnerable to bias leading skewed results. It occurs because the part of the sample's content can appear in training and testing, simultaneously. They have demonstrated that by applying non-biased ways of generating features from raw data, recognition performance has been significantly affected. Therefore, in this study we also investigate impact of three protocol of generating features: A: Extracting features over whole dataset (traditional method), B: Extracting features over sessions of recording data, and C: Extracting features over data of each subject separated. Method A is selected because it was mostly-used in previous studies []. There is an evaluation method corresponding with each way of generating features including K-fold, Leave-One-Trial-Out (LOTO), and Leave-One-Subject-Out (LOSO), respectively. In particular, we want to answer this research question: "How do different protocols impact on recognition performance? (RQ3)".

We believe that our effort will assist the readership and this will save time for future studies by not repeating the same experiments. This study can be used as a basis for making design decisions about when and what to choose these set of features for better activity recognition. The main contributions and highlights of this paper are as follows:

- To the best of our knowledge, we are the first to do such an extensive analysis of the role of state-of-the-art featuresets in activity recognition, over different classifiers and evaluation methods. We extract features from two sensors equipped with accelerometer and gyroscope on two body positions (wrist and foot). We have used four classification models in our experiments, which are all used in the state-of-the-art.
- We also investigate the recognition performance when the features are orientation-independent comparing with when they are orientation-dependent. Moreover, we target a wide range of features from low complexity which are suitable for running on smartphones (e.g., histogram) to high complexity which are useful for online HAR systems[3] for our evaluation scenarios.
- We introduce leave-one-set-out cross-validation which is similar to LOTO cross-validation. In addition, we show how much robust each state-of-the-art featureset is against different evaluation methods.
- We recognize eight gym exercises, commonly used in the state-of-the-art. Moreover, we make our data set and our labeling and extracting features application publicly available for future research in this domain [14].

The rest of the paper is organized as follows. We describe related work in Section 2. The data and the study setup including the approach and the dataset are explained in Section 3. The featuresets are described in Section 4 and our evaluation approach in Section 5. We discuss the performance evaluation in Section 6. Finally, we describe our conclusions and future work in Section 7.

2. Data Collection

Basically, a HAR system using Inertial Measurement Units (IMUs) is composed of two basic components: (1) a data acquisition unit responsible for capturing human movements, (2) a processing unit responsible for recognizing the certain activities among movements of the subject [9]. In the first component, the human movements is captured by different motion sensors such as accelerometer, gyroscope, and so on. These sensors can be located in an off-the-shelf device like a smart-phone for general HAR applications or specifically are accompanied by a storage and a processor, formed a System on Chip (SoC) for a certain purpose. The captured movements as raw data transmits to the

second component for processing operation.
Within the processing unit, there is a pattern recognition (HAR) model that classifies the input signals into certain classes of activities. This HAR model typically consists of three phases. First, there is a pre-processing operation that extracts informative features from raw signal. Second, a classifier is trained over extracted features. Third, an evaluation method to ensure that the classifier provides the required performance [15]. In other words, a HAR model should address these three aspects to be able to recognize an activity.

In literature, in order to gain more information from the sensory data, the authors came up with hundreds of hand-crafted features among time domains, frequency domain or a combination of them. While each feature represents the signal in a certain point of view, it does not mean that this feature is informative enough for a model to recognize an activity based on that. One decisive factor to build a new feature is respecting the type of activity. Wang et. al. in [5] categorized human activities based on velocity and complexity (number of phases) of an activity into three main groups: (1) The **basic activities** which happen in comparatively longer duration e.g., walking and running. (2) The complex activities that are in the form of a sequence of several phases. Each phase might be a complex or basic activity e.g., coffee time, smoking. (3) Transition activities which having a certain but temporal pattern happening between two different postures or two basic activities. e.g., stand-to-sit, push-ups and so on. From this point of view, previous works introduced different feature sets that each one target a given type of activity. Consequently, targeting more than one type of activity brings more challenges for researcher to create a suitable set of features. Since exercises in gym composed of an orchestration of different type activities (basic, transition, complex), presenting a model to effectively recognize these activities will be more challenging.

Basically, given a stream of sensory data, features are in three main categories.[5] 1) Time-domain features, 2) Frequency-domain features, 3) Hybrid features - any combination of statistical functions on signal in frequency-domain and/or time-domain.

2.1. Activities

This section describes the activities and how we chose them. As mentioned earlier, we have chosen gym exercise activities to collect data from. We captured data from 55 gym exercises. Exercises involve different parts of body including upper-body, lower-body, or entire-body. We only chose beginner to intermediate level exercises since it raises the chance of finding participants, thereby, collecting more data of each exercise for this and future works. In total, we recorded the data from 15 subjects over 25 sessions. In fact, we asked some subjects to participate multiple times in data collection. By repeating same exercise by same subject through different sessions, we reduce the impact of temporal factors i.e., tiredness on our dataset. Addition to this, in order to have more realistic scenarios, we did not limit participants to certain set of exercises, instead we left them to follow up their own exercise plan.
Subjects. We asked 15 members of a gym (4 female), ages 21-35, to participate in this study. Participants varied in level of expertise in gym exercises (from 1 month to 6 consecutive years of experience).

Comparing with literature [3,7,19,20], this way of collecting data bring following advantages:

1. **Realistic workload**: There was no instruction of how doing exercises for subjects. Although this can let subjects to perform an activity in non-identical way, it is considered as an advantage for our study since it replicates the real-world condition. In [3], the authors showed that by changing the environment from space-constrained laboratory to a real gym the segmentation performance for recognizing gym exercises has dropped by 50%. Therefore, another advantage of keeping the experiment under real-world condition is the performance of the HAR model is more reliable.
2. **Realistic null-class activities**: The unknown period or null-class activities are not artificially performed, since subjects were free to do whatever they normally do in gym.

3. **Effects of fatigue**: Since sessions are relatively long (between 1 to 2 hours), subjects mostly get tired. Thus, we can observe the impact of fatigue on performing an activity. Therefore, more generalized dataset.
4. **Impact of subject experience**: Gym exercises are performed in a iterative manner over weeks or months. By repeating an activity, subjects become more comfortable on that, consequently, they will do that exercise more consistently. Since we asked the length of practicing each exercises from participant, we can observe if there is any correlation exist between them.

2.2. Sensors

In this section, We describe how we choose sensors and how we deploy our system to collect the data. Depending on how many parts of the body that are involved in a task (exercise), a HAR system may need to have one or more sensors on different positions of the human body with the aim of capturing diverse-source information [5]. On the other hand, using more sensors limits the user's movement [21].

In a recently related work, Soro et al. [19] have targeted 10 CrossFit activities and showed that using one sensor on wrist is sufficient to reach an accuracy of 99.96%. One reason that may underlies using one sensor is that, in that case, almost all activities have moving hand involved in. However, in this study, we cover a wider range of activities which is not necessarily get performed by hand. For example, *reverse crunch* needs subject to fix hands on the ground and then draw legs in towards the torso. In this case, it is almost impossible to identify the exercise by only one sensor on wrist. Therefore, in this study, to cover activities both on lower-body and upper-body, two sensors were attached to right wrist and right foot (Figure 2 (b) and (c)). These positions (wrist and foot) are also used in previous works [17,19,22].

To record the data, we choose well-known accelerometer and gyroscope sensors. These two sensors are available on almost all Wearables such smartphones or smartwatches. In this study, as it showed in Figure 2 (a), we employed a Neblina chip, instead. The main advantage of using Neblina is its small size and ability to tightly stick to the body preventing unexpected moves during performing an activity.

Neblina is a miniature-sized box containing three tri-axial motion sensors (accelerometer, gyroscope, magnetometer) along with a processor, a flash memory, battery, and a bluetooth port. Using blue-tooth port, it can transmit the result to a host (e.g., cellphone or desktop computer). In fact, Neblina is equipped with all requirements for a real-time HAR system. Comparing with a smartphone, Neblina is much smaller (Figure 2) that lets us to attach it to different part of the subject's body without making any interrupt in his/her actions[21]. Having access directly to different resources like sensors or memory without OS interferences is another advantage of using Neblina that let us improve the efficiency of the model.

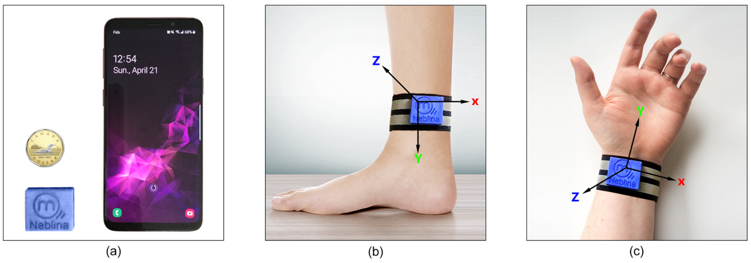


Figure 2. Neblina setup. (a) Compares dimensions of Neblina with a 1 dollar coin and a cellphone (Sumsung Galaxy s9). (b) How Neblina located on foot using a strap. (c) How Neblina located on wrist using a strap.

Although the device provides the magnetometer signals, we limited our process on using the accelerometer and gyroscope signals only. It is because the magnetometer signal can be affected by getting close to iron equipments in the gym. The frequency rate is fixed on 50Hz. It is worth to mention that the frequency rate more than 50Hz is not necessary because according to the Nyquist theorem [23], this rate is enough to record a repetitive activity with 25 cycle per second which is so much faster than the iterations of normal workouts in the gym (one iteration per 1-5 seconds).

2.3. Dataset

In this section, we explain the process of preparing our dataset for experiments.

Labeling. In order to train a HAR model in a supervised machine learning process, we need labeled data points. Quality of labeled data points at this level has a direct impact on model performance [31]. In this study, to label the data we employed a process including three phases: (1) **By participant**: before beginning of each session, each subject was asked to fill a form about list of activities, number of sets, and the weights if applicable. (2) **By supervisor**: during the session, a supervisor manually records type of exercise, the moment of start and stop, and number of repetitions. (3) **By visual signal**: after finishing the session, in order to have our desired accuracy in labeling, we visually trace the signals of accelerometer and gyroscope to refine the regains assigned to each exercise. Using this way, any error missing in one step, will be caught in the next step.

Adapting. In order to make the dataset suitable for the experiments of this study, we trimmed the dataset. As we will explain later in *Method* section, addition to k-fold cross-validation, we have two other evaluation methods including *cross-session validation* and *cross-subject validation*. These two additional evaluation methods bring some restrictions on the dataset. While *cross-session validation* requires some activities to be repeated in several sessions, *cross-subject validation* requires them to be repeated by several subjects. In addition, in RQ3, we want to compare results of different evaluation methods; Therefore, we have to use equal dataset for all evaluation methods. This is to say that for each activity to be suitable for this study, it should be repeated at least in certain number of sessions and also performed at least by certain number of subjects. Table 1 shows the statistics of the dataset after applying the restrictions. Column *Sessions* shows the total number of sessions that an exercise appeared in. Column *Subjects* shows how many subjects performed an exercise.

Table 1. Statistics of the dataset divided by type of exercise along with the experiments that involve them in.

Exercise	Subjects	Sessions	Reps	Data Point	Body Involved	Code
Lat Pull Down	6	22	218	14700	upper	A1
Bench Press	6	26	273	23230	upper	A2
Biceps curl	4	13	115	16095	upper	A3
Push-ups	5	16	181	9200	upper	A4
Treadmill	4	5	+1200	68780	Entire	A5
Ab crunch machine	4	12	108	10580	Entire	A6
Crunch Twist	3	12	98	8760	lower	A7
Russian Twist	3	8	67	8520	lower	A8

3. Method

In this section, we explain the procedure of HAR as it is provided in Figure 1.

3.1. Feature Extraction

In this study, we targeted five state-of-the-art sets of hand-crafted features. Table 2 shows 31 functions along with a description/intuition about each one. Functions are grouped into sets (first column). Each set of functions is uniquely representative of a certain aspect of the signal.

Preprocessing. We used two tri-axial sensors (accelerometer and gyroscope) on two body positions (wrist and foot); So, To reproduce features, we normally apply these 31 functions on 12 raw signals coming from sensors. However, in some cases, an extra preprocessing operation is also required which will be explained in related section. Addition to this, to ensure that the data from different resources (sensors) are at the same scales, we performed a scale normalization in advance of any other computations.

Table 2. Statical Functions along with the definitions and abbreviations

Code	Function	Description/Intuition	abbreviation
S1	Minimum	The value of the least sample	MIN
S2	Maximum	The value of the greatest sample	MAX
S3, SS8	Mean	The average of all samples	MEA
S4	Median	The middle value of samples	MEA
S5	Mean Absolute Deviation	The average distance between each sample and the mean of the stream	MAD
S6	Median Absolute Deviation	The average distance between each sample and the median of the stream	MAA
S7	Inner Quartile Range	The amount of spread in the middle part %50 of the stream	IQR
S8	Mean Crossing Rate	The rate of passing the mean along the stream	MCR
S9, SS9	Standard Deviation	how far the samples are from the mean	SD
S10, SS10	Variance	the average degree of distance between samples and mean	VAR
S11, SS11	Root Mean Square	The square root of the arithmetic mean of the squares of samples	RMS
HB	Histogram Bin	a 20 bins distribution of data	Hbin (1-20)
SS1	Number of auto-correlation peaks	The bigger number means non-periodic activity while smaller number refers to periodic activity	NAcP
SS2	Prominent auto-correlation peaks	NAcP with an extra condition that the peaks should be greater than neighbours with at least a certain distance	NAcPP
SS3	Weak autocorrelation peaks	NAcP with an extra condition that the distance between the peaks and neighbours should be less than a certain distance	NAcWP
SS4	Maximum autocorrelation value	Value of the greatest peak (except for the initial peak at zero lag)	MAXAc
SS5	Height of the first autocorrelation peak (after zero-crossing)	less height refers to more fluctuations within the stream	FAcP
SS6	Power bins (10 bins)	A 10 bins distribution of amplitudes of frequencies from 0.2-25Hz	Pbin(1-10)
SS7	Integrated RMS	The root-mean-square amplitude of the signal after cumulative summation	IRMS
Ph1	Movement Intensity	the Euclidean norm of the total acceleration vector after removing the static gravitational acceleration	MI
Ph2	Normalized Signal Magnitude Area	the acceleration magnitude summed over three axes within each window normalized by the window length	SMA
Ph3	Eigenvalues of Dominant Directions	The eigenvectors of the covariance matrix of the acceleration data correspond to the dominant directions along which intensive human motion occurs.	
Ph4	Correlation between Acceleration along Gravity and Heading Directions	It shows the human movement is either vertically or horizontally.	CAGH
Ph5	Averaged Velocity along Heading Direction	The Euclidean norm of the averaged velocities along y and z axes over the window.	AVH
Ph6	Averaged Velocity along Gravity Direction	averaging the instantaneous velocity along the gravity direction at each time t over the window	AVG
Ph7	Averaged Rotation Angles related to Gravity Direction	The cumulative rotation angles around gravity direction	ARATG
Ph8	Dominant Frequency	The frequency corresponding to the maximum of the squared discrete FFT component magnitudes of the signal from each sensor axis	DF
Ph9	Energy	The sum of the squared discrete FFT component magnitudes of the signal from each sensor axis	ENERGY
Ph10	Averaged Acceleration Energy	The mean value of the energy over three acceleration axes	AAE
Ph11	Averaged Rotation Energy	The mean value of the energy over three gyroscope axes.	ARE
OI1	Orientation Independent	result of applying PCA on Single Value Decomposition of x/y/x values of the stream	PCASVD(1-30)

3.1.1. Set_A: Statistical Features (ST_Set)

Statistical features have been intensively investigated in different applications and proved to be effective and useful for HAR [9]. These features are based on a comprehensive and intuitive understanding of how a given activity produces a set of discriminative features from measured sensor signals. We created a set of 264 features obtained by applying 11 statistical functions on 24 input signals, including (x/y/z axes of accelerometer and gyroscope, along with the cumulative sums of each axis). Functions are used in the statistical feature set are indicated by the code S1-S11 in Table 2.

3.1.2. Set_B: Histogram bins Features (HB_Set)

The second set of features are based on histogram representation of time series signal. Mathematically speaking, histogram representation of a signal is the probability distribution of signal over a period of time (often referred to as window size) [26]. In HAR, considering the fact that each activity contains a set of small movements (as small as one sample) with certain acceleration and rotation, histogram bins indicate the difference between activities by showing the different distributions of those small movements. Shirahama et. al. [27] used a histogram-based feature set for HAR. Comparing with statistical features, histogram bins have a significantly lower cost in terms of required processing time and memory usage [25]. However, they are sensitive against the resolution/granularity of bins (count and width of bins). Following prior work [28], in this work, we consider 21 bins for values between 0 through 1 of a signal. Histogram bins are indicated with the code HB in Table 2.

3.1.3. Set_C: Self-Similar Features (SS_Set)

Considering that an exercise activity is inherently more repetitive rather than a non-exercise activity, having a featureset that can capture the repetitive behaviour of signal is helpful. Morris et al. presented a featureset designed based on the idea of extracting repetitions forms of signal [3]. These features can be extracted by: 1) Calculating the convolution of a signal with a shifted version of itself (auto-correlation) or 2)Extracting the components of the signal in the frequency domain. We extracted a number of self-similar features from our data, as shown in Table 2, features with code SS1-SS11. SS6 composed of 10 features per se; Thus, in total there are 20 functions. There is a preprocessing operation required for this featureset. We transformed 3 input signals from each sensor (x/y/z inputs) to 4 processed signal describing as follows: 1) The magnitude of a/y/z axes, 2) The first principal component of all axes. 3) The first principal component of x and z axes. 4) The scaled normalized of y axis. It is important to mention that in our experiments, the y axis of sensor is along the user's arm. To build the featureset, we applied 20 functions on 16 processed signals. Therefore, this featureset contains 320 features.

3.1.4. Set_D: Physical Features (PH_Set)

One intuitive idea to design a set of features from sensory data is to take the principles of human movements into consideration. In 2011, Zhang et. al. [12] introduced a set of features based on physical parameters of human motion. To have a strong physical meaning of motion data (e.g., moving forward, backward), they assumed that the sensor position and direction are known during the experiment. In other words, this types of features is derived based on the physical interpretations of human motion, called physical features. Comparing with other featuresets, these features are made up of a fusion of multiple sensor inputs rather than just one inputs sensor. In our paper, this featureset contains 11 features, labeled with the code Ph1-Ph11 in Table 2. As a part of our pre-processing operation, we remove gravity from acceleration using gyroscope data by applying the method described in [29].

3.1.5. Set_E: Orientation Independent Features (OI_Set)

In contrast to physical features, which depend on the position and orientation of sensors, Yurtman et. al [24] proposed features that do not rely on variation of sensor orientation. In fact, in their model, they introduced Orientation-invariant transformations (OITs). They compared their model with the ordinary model - pre-defined sensor orientation, on five different datasets. Although their featureset did not have a significant impact on performance, it brought an extra added value to the model that lets it to be more robust against orientation. The OIT that they have introduced in their work is inspired by the idea of *single value decomposition*[30]. Therefore, to create this featureset, first, we project every data point from original x/y/z space to a new space with same number of dimensions but at the farthest distance between data points. The intuition here is that the direction of the axes are defined by value of the data points not by x, y or z direction. Next, we apply PCA on the transformed data and take the first 30 most informative features [31]. In Table 2, these type of features are indicated by "OI" prefix.

3.2. Activity Recognition

In this paper, to discriminate the classes (exercises), we implemented four state-of-the-art classifiers on every featureset. The classifiers are implemented on mobile phones in various studies [3,9,17]: Support Vector Machine , Decision Tree, K-nearest neighbor, and Feed-forward neural network.

Table 3. Classifier names along with hyper-parameters in this study

Classifier	Hyper-Parameters
SVM	kernel = polynomial. degree = 3. gamma = 1/(data dimension)
KNN	K = 64. Similarity Method = Euclidean distance
FNN	2 dense layers with total 1000 and 400 units [32]. One dropout layer (rate 60%). Optimizer = Adam. learning rate = 0.0001. decay = 1e-10. 100 epochs.
DT	minimum split = 20, min number of sample in leaves = round(minimum split/3), maximum depth = 30

3.2.1. Support Vector Machine

A multi-class Support Vector Machine (SVM) has been employed extensively in previous studies in HAR to discriminate among the activities [3,9,33]. Morris et al. [3] showed that SVM provides higher accuracy in HAR. Assuming each data point is a co-ordinate (support vector) of feature space, the Support Vector Machine (SVM) is a method to find an optimum hyperplane with respect two some support vectors in order to separates different classes.

3.2.2. Decision Tree

Decision Tree is an ensemble method that provides an explainable model in classification. It is suitable for running on mobile phones with reasonable recognition performance [17,18,34]. Baldominos et al. [17] reported the best recognition performance for decision tree among a set of classifiers including Naive Bayes, KNN, FNN, and Logistic Regression.
In order to distinct activities, decision trees look through every possible split in a range of values of a given feature and pick the best split between two adjacent points; The goal is that, after each split, branches become more "pure" (i.e., homogeneous activities) [35]. The "information gain" is a metric which assess error after each split. Thus, decision trees pick splits to maximize "Information Gain". There are two methods to assess the information: 1) Entropy and 2) Gini index.
Following [9,36], we used the Gini index error metric. There is also other hyper-parameters to control additional splits including maximum depth, minimum samples in leaves, and maximum number of leaf nodes. The values for these parameters are mentioned in Table 3.

3.2.3. K-nearest neighbor

Another widely used classifier in HAR is the KNN algorithm [5,37], which works based on the calculation of the distance between data-points. As a lazy learning algorithm, KNN does not need off-line training. In other words, during the classification phase, for a given test data point, KNN finds the first k nearest neighbors among training data-points. The majority of neighbors produces the classification output. Prior works showed that increasing k improves the performance on HAR model [38]. All the parameters for KNN in this study are mentioned in Table 3.

3.2.4. Forward Neural Network

By choosing a proper featureset, convention machine learning models (e.g., SVM, KNN) can achieve a good performance in activity recognition. However, we can improve it by using Deep Neural Network, which underlies the complementary information learned within the layers [39]. Recently, a number of studies have shown that Feed-forward Neural Networks achieve high performance in HAR[39,40]. A FNN is made up of a set of neurons, connected by weighted arcs, that process the input information:

$$y = f(\sum_i w_i . x_i) \tag{1}$$

where y is the output of the neuron, w_i are weights of the incoming connections, x_i are inputs to the neuron, and f is called transfer function and is selected according to the classification problem [41]. Neurons in a FNN are organized in layers. Number of neurons at first layer and last layer are corresponding with number of features and number of activities, respectively [17]. Between input and output layers a certain number of hidden layers can be inserted, whose dimensions are usually decided testing different configurations [17]. The configuration details of the network described in Table 3.

3.3. Evaluation

To evaluate the performance of the model, we need to split data into training and testing sets. For this purpose, in the context of HAR, there are traditional techniques such as k-fold cross-validation, leave-one-subject-out [13], as well as, the relatively less common techniques like leave-one-trial-out [42]. In this study, we used all three validation methods while we modified leave-one-trial-out to leave-one-set-out in order to make it more flexible for repetitive activities.

3.3.1. K-fold Cross validation

The most typical approach to evaluate the performance of HAR model is k-fold-cross validation. The idea is to use resampling procedure in a way that all the samples be used once during the testing period, mostly in the case of having limited dataset. The so called parameter k refers to the number of groups that the given dataset is to be split into. Each time one group becomes the test set and remaining $k - 1$ groups become training set. During k turns, we evaluate the model k times on different test set and train set. Finally, the performance is summarized by averaging the performance of all k turns. In this work, we apply k-fold cross validation on all the models. That is, the dataset breaks down equally into 10 parts (folds). During 10 iterations of evaluation, every part is considered as test-set and remaining parts as train-set.

3.3.2. Leave-One-Subject-Out Cross validation

Leave-One-Subject-Out (LOSO) validation is a special case of cross-validation, where a subject can be seen as a fold, hence, the number of subjects determine the number of folds. Furthermore, the LOSO validation technique reflects a realistic scenario, where a model is trained in an offline way, using the samples of some subjects, and is tested with samples of unseen subjects. It is important to note that using LOSO may present high variance in performance from one subject to another, since the same activity can be performed in different ways by the subjects.

3.3.3. Leave-One-Trial-Out Cross validation

The Leave-One-Trial-Out (LOTO) cross validation is similar to LOSO, however, instead of considering the subjects as folds, each trial (session) of doing the activity is considered as a fold. In fact, sessions might belong to same subjects of different subjects. To implement LOTO, on dataset of each subject which may contain several sessions, we assigned an index to each session. So, in order to split the dataset into test and train, we use session id instead of subject id in LOSO. The main advantage of using this method comparing with LOSO is that it needs a smaller number of subjects since each subject can have several sessions. In addition, similar to LOSO, this method does not suffer from having same content between train set and test set.

Measurements. Most commonly used measures to asses the performance of a HAR model in prior works are: accuracy[12,43,44] and F-measure[9,10]. We used accuracy and F1. Using F1 is essential since our dataset is imbalanced. Thus, since F1 relies on both the precision and recall, it is less affected by imbalanced dataset. Specifically, measurement units in this study determined as follows:

- **Accuracy** measures how often the classifier is correct. Specifically, it is equal to (TN + TP) / total.
- **F-Score** measures a weighted average of both Recall and Precision. Specifically, it is equal to (2 x Precision x Recall) / (Precision + Recall).

Where:

- **True Positive (FP):** These are cases in which we predict an activity, and user was doing that activity.
- **True Negative (TN):** Is where we predict a non-activity period, and user was not doing a certain activity.
- **False Positive (FP):** Is where we predict a certain activity for a segment of data, however, user is either doing another specific activity or generally doing something else (out of activity given list).
- **False Negative (FN):** Is where we predict either a not-activity period or a certain activity, but, it is not the activity that user is really doing that.

4. Results

Our study aims to perform a systematic examination on HAR pipeline (Figure 1) through three crucial steps. First, in RQ1, we compare different featuresets and indicate the one providing the best recognition performance; Next, using this featureset as input, we examine four classifiers in classification phase to find the model with highest performance (RQ2). Finally, in RQ3, we target the impact of different evaluation methods on performance of our model.

4.1. RQ1: Which featureset provides the best performance in HAR?

As motivated earlier, choosing an appropriate featureset significantly impacts on recognition performance. Many different featuresets have been presented in previous works. While they all are reporting remarkable performances on HAR, they can not be compared with each other due to different experimental setups that those results are achieved. Hence, we aim to investigate five state-off-the-art featuresets when all other factors (i.e., dataset, classifiers) are fixed. Each featureset is examined by four classifiers including FNN, KNN, SVM, and DT. To measure the performance, we used 10-fold cross validation and F-score metric for each experiment.

Table 4 shows the performance for each featureset (columns 2-6) on different classifiers (rows 2-5). We highlighted the best performance for each featureset in the Table. It can be seen that the best performing featuresets are *statistical featureset* (ST_Set) and *Histogram bins* (HB_Set), achieving approximately 95% of F1. However, the remaining featuresets have never exceeded 90% of F1. The highest recognition performance for *self-similar featureset*, *physical featureset*, and *Orientation independent featureset* are respectively 89.18%, 85.34%, and 78.47%.

It is worth mentioning here that regarding th computational cost of featuresets, *histogram bins* is placed at the lowest complexity (cost) with having only $O(n \log m)$ where m is number of bins. **Low computational cost along with providing the highest recognition performance make *histogram bins* an ideal candidate featureset for Wearables, since they are limited in resources**. On the other hand, *Orientation Independent* features achieved the minimum performance (77.44% on average). Although it reached to a relatively lower performance among featuresets, it allows for a flexibility in the way that sensors are placed on a subject.

Table 4. F-score for each classifier over different feature-sets using 10-fold cross validation

Classifier	**ST_Set**	**HB_Set**	**SS_Set**	**PH_Set**	**OI_Set**	Average
SVM	94.98%	94.55%	89.18%	84.15%	78.47%	87.82%
KNN	91.50%	90.21%	85.61%	81.93%	76,41%	85.50%
FNN	95.31%	95.89%	87.93%	85.34%	77.59%	88.29%
DT	88.64%	89.18%	82.94%	79.37%	74.02%	82.83%
Median	92.36%	92.92%	86.41%	82.70%	77.12%	86.30%
Average	92.74%	93.30%	86.77%	83.04%	77.44%	86.66%

4.2. RQ2: Which classifier performs better on gym exercise recognition?

As we saw in RQ1, different classifiers perform differently even on same featureset. Hence, one question that we aim to answer is whether certain classifiers perform better than others. Therefore, in this research question we do an empirically comparison between the performance of different classifiers. We use four popular classifiers in HAR including SVM, KNN, FNN, and DT. It is important to mention that we evaluate the model using default configurations since the optimizing the classification is not a goal of this study. The default configuration used for each model is mentioned in 3.2 section. From RQ1, we found *histogram bins* as the most informative featureset. So, we use it for training model in this experience. The K-fold cross-validation with 10 folds is used for measuring classification results; Thus, we have 10 results of F-Score for each classifier. To compare results, we used *Student t-test*. Figure 3 shows how results of each classifier are spread out over 10 rounds. FNN followed by SVM show a range between 85% through 99% of F-measure over 10 trials while DT and KNN are showing more divergence over trials respectively between (67%-98%) and (77%-97%) of performances. The mean for all classifiers are shown in Figure 3. Since the *p-value* is 0.057 we can reject the null hypothesis, saying that with 93% of confidence, the average performance showing in Figure 3 is what exactly that those classifiers perform in practice. In other words, FNN and SVM with 95%$\pm < 1$% are delivering highest performances whereas KNN and DT with 90.21% and 88.29 of F-Scores respectively are giving lower performances.

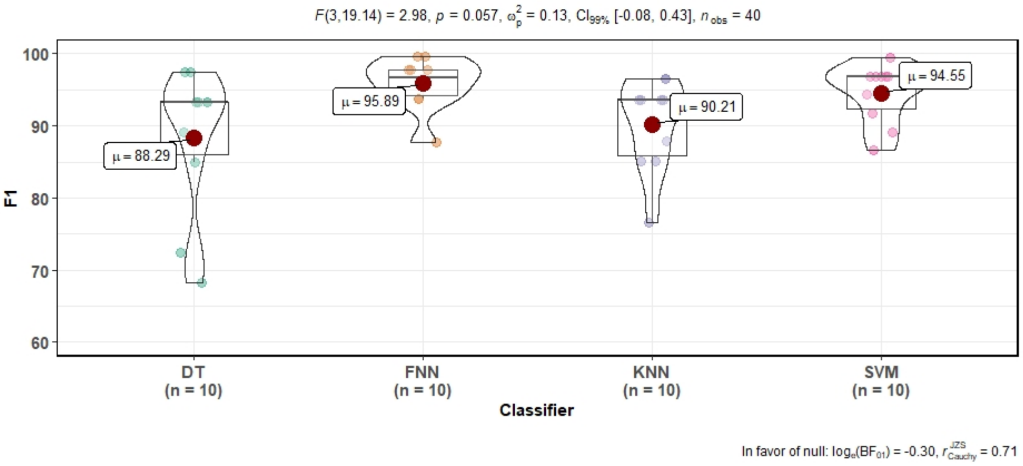


Figure 3. Comparison between the performance of classifiers

4.3. RQ3: How do different evaluation methods impact the reported HAR performance?

K-fold is one of the most popular methods to evaluate the performance of a HAR model [5]. However, in an empirical study, Jordao et al. [13] showed that the result of k-fold cross validation can be biased when using sliding windows, a technique that is commonly used in HAR. Therefore, the focus of this research question is to asses models by two state-of-the-art evaluation methods namely, Leave-One-Subject-Out (LOSO) cross validation [45] and Leave-One-Trial-Out (LOTO) Cross validation [13,42].

In K-fold, splitting the dataset (K) is decided by researcher based on the size of dataset as well as the type of classification problem[13]. Table 1 shows how the data is distributed for each activity. To split the dataset in each validation method based on number of activities and number of subjects. However, in LOSO and LOTO, it also required to respect to the distribution of activities among number of participants, and sessions of each participant. In fact, an activity should appears at least in more than one session by same subject to be eligible for LOTO cross-validation. In this experiment, for LOSO we used data from 6 subjects while compromising some activities were missing for 2 subjects. For LOTO, we have employed the data of 8 sessions while some sessions belong to same person. For those activities that appeared in more than 8 sessions, we merged their sessions to each other.

Figure 4 compares the performance of models using 10-fold cross validation (in blue), LOTO (in orange), and LOSO (in grey). As we can see from the Figure, for all featuresets and all classifiers, the evaluation technique impacts the reported performance. In fact, we see that in general, k-fold cross validation always provides better results than LOTO and LOSO. As mentioned earlier, due to the use of sliding windows in HAR, LOTO or LOSO are more realistic evaluation techniques and than k-fold cross validation. It can be seen that there is a significant distance between results of LOTO and LOSO (10%). This can be due to differently performing an exercise by different subjects in LOSO. However, in LOTO, since the model is trained by the data of at least one session of each subject it returns a better result.

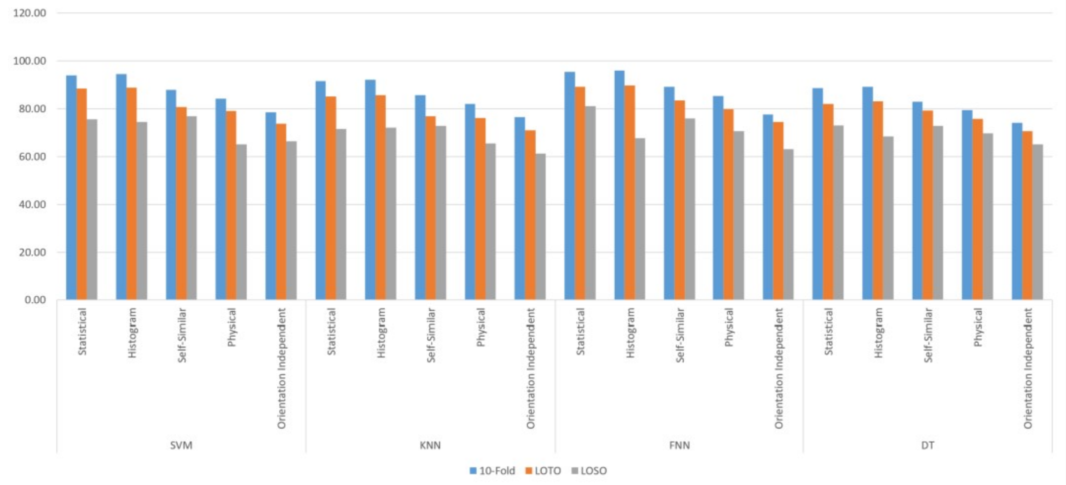
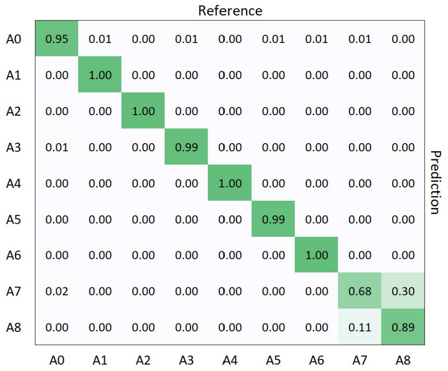


Figure 4. Comparison between evaluation methods (10-Fold, LOTO, LOSO)

5. Discussion

5.1. Performance of FNN

In this section we focus on investigating the FNN model on Histogram bins more in detail. Figure 5 shows the normalized confusion matrix for FNN model on *set_B*. One can see that activities such as Lat Pull Down and Bench Press (A1 and A2) are classified more accurately than activities such as Crunch Twist (A7) which is misclassified as Russian Twist (A8). In other words, it can be said that activities of a similar nature are more willing to get misclassified. Especially when the subject is not experienced enough on performing the exercise, recognizing the activity get harder. Addition to this the class A0 which stands for non-exercise activity is misclassified 1 percent as almost all other classes. This could be a result of variation in the distribution of data for all classes. This can also be illustrated in Figure 5 where the performance at first row is distributed across all the classes. Because of the uniform nature of data distribution among all classes, and because of a balanced nature, similar activities could be classified more accurately.


Reference
Prediction

	A0	A1	A2	A3	A4	A5	A6	A7	A8
A0	0.95	0.01	0.00	0.01	0.00	0.01	0.01	0.01	0.00
A1	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
A2	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00
A3	0.01	0.00	0.00	0.99	0.00	0.00	0.00	0.00	0.00
A4	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
A5	0.00	0.00	0.00	0.00	0.00	0.99	0.00	0.00	0.00
A6	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00
A7	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.68	0.30
A8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.11	0.89

Figure 5. Normalized confusion matrix for FNN classifier using histogram dataset(set_B). A0 in this table stands for non-exercise data points.

5.2. Learning Speed

The last but not the least aspect worthy to mentioned is various converging speed of FNN between using different featuresets. As mentioned in RQ1, training an FNN on Histogram bins after 100 epochs delivers the best performance comparing with result of training at same number of epochs on other featuresets. In this section, we investigate velocity of FNN on reaching its best performance during first 100 epochs. Figure 6 shows the performance of FNN models using different featuresets. As we expect, after 100 epochs, the models on histogram bins, statistical features, self-similarity features have been reached a higher level of F1 (all above 90%) while the had been stable almost after first 20 epochs. On the other hand, the trend for both models on physical features and orientation independent are below 90% while they have not been stable even at greater number of epochs (close to 100). This is to say that FNN can be trained faster when it is feed with histogram bins, statistical features, or self-similarity features rather two other featuresets. Interesting, for FNN on histogram features, the first time to touch the best performance is at the 5th epoch. This number for models on statistical features and self-similarity features happens after 14 epochs and 12 epochs respectively.

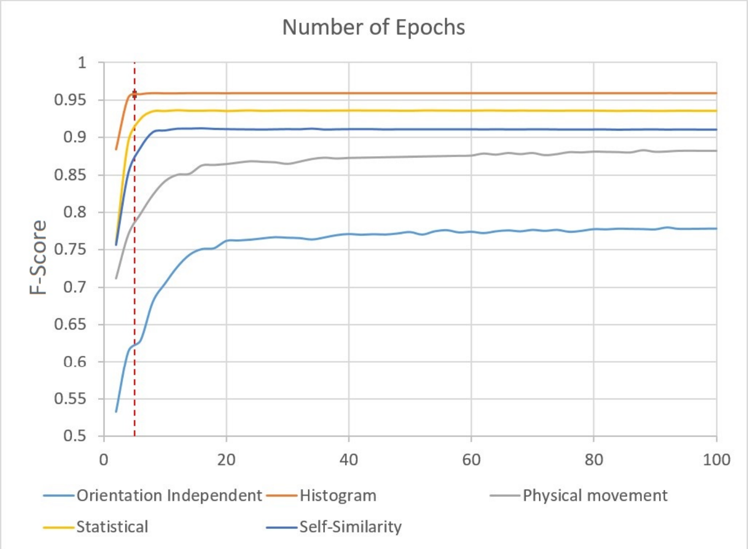


Figure 6. F-Score of FNN during first 100 epochs using 5 featuresets

6. Conclusions

Human activity recognition is an important research topic in pattern recognition and pervasive computing. In this paper, we have studied on the state-of-the-art models using hand-crafted features and traditional models. From RQ1 and RQ2, it turned out that FNN and histogram bins can deliver a superior performance rather other 19 pairs of classifiers and featuresets. It is also important to mention that the number of bins and width of each bin play important roles on extracting informative features. In RQ3, comparing leave-one-trial-out cross validation with two conventional evaluation methods (k-fold and LOSO), we saw that LOTO and LOSO provide a more realistic result rather K-Fold at the expense of declining the performance. Addition to this, we figured out, LOTO can address two issues which are not solved LOSO, including: 1) it is applicable on datasets with less number of subjects comparing with LOSO which requires more subjects. 2) it can suppress the impact of performing differently of an exercise by different subjects.

Author Contributions: For research articles with several authors, a short paragraph specifying their individual contributions must be provided. The following statements should be used “conceptualization, X.X. and Y.Y.; methodology, X.X.; software, X.X.; validation, X.X., Y.Y. and Z.Z.; formal analysis, X.X.; investigation, X.X.; resources, X.X.; data curation, X.X.; writing–original draft preparation, X.X.; writing–review and editing, X.X.; visualization, X.X.; supervision, X.X.; project administration, X.X.; funding acquisition, Y.Y.”, please turn to the CRediT taxonomy for the term explanation. Authorship must be limited to those who have contributed substantially to the work reported.

Funding: Please add: “This research received no external funding” or “This research was funded by NAME OF FUNDER grant number XXX.” and and “The APC was funded by XXX”. Check carefully that the details given are accurate and use the standard spelling of funding agency names at https://search.crossref.org/funding, any errors may affect your future funding.

Acknowledgments: In this section you can acknowledge any support given which is not covered by the author contribution or funding sections. This may include administrative and technical support, or donations in kind (e.g., materials used for experiments).

Conflicts of Interest: Declare conflicts of interest or state "The authors declare no conflict of interest." Authors must identify and declare any personal circumstances or interest that may be perceived as inappropriately influencing the representation or interpretation of reported research results. Any role of the funders in the design of the study; in the collection, analyses or interpretation of data; in the writing of the manuscript, or in the decision to publish the results must be declared in this section. If there is no role, please state "The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results".

Abbreviations

The following abbreviations are used in this manuscript:

MDPI	Multidisciplinary Digital Publishing Institute
DOAJ	Directory of open access journals
TLA	Three letter acronym
LD	linear dichroism

Appendix A

Appendix A.1

The appendix is an optional section that can contain details and data supplemental to the main text. For example, explanations of experimental details that would disrupt the flow of the main text, but nonetheless remain crucial to understanding and reproducing the research shown; figures of replicates for experiments of which representative data is shown in the main text can be added here if brief, or as Supplementary data. Mathematical proofs of results not central to the paper can be added as an appendix.

Appendix B

All appendix sections must be cited in the main text. In the appendixes, Figures, Tables, etc. should be labeled starting with 'A', e.g., Figure A1, Figure A2, etc.

References

1. Hong, Y.J.; Kim, I.J.; Ahn, S.C.; Kim, H.G. Activity recognition using wearable sensors for elder care. 2008 Second International Conference on Future Generation Communication and Networking. IEEE, 2008, Vol. 2, pp. 302–305.
2. Sow, D.; Turaga, D.S.; Schmidt, M. Mining of sensor data in healthcare: A survey. In *Managing and mining sensor data*; Springer, 2013; pp. 459–504.
3. Morris, D.; Saponas, T.S.; Guillory, A.; Kelner, I. RecoFit: using a wearable sensor to find, recognize, and count repetitive exercises. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2014, pp. 3225–3234.
4. González, S.; Sedano, J.; Villar, J.R.; Corchado, E.; Herrero, Á.; Baruque, B. Features and models for human activity recognition. *Neurocomputing* **2015**, *167*, 52–60.
5. Wang, Y.; Cang, S.; Yu, H. A survey on wearable sensor modality centred human activity recognition in health care. *Expert Systems with Applications* **2019**.
6. Schilit, B.N.; Adams, N.; Want, R.; others. *Context-aware computing applications*; Xerox Corporation, Palo Alto Research Center, 1994.
7. Shoaib, M.; Bosch, S.; Incel, O.D.; Scholten, H.; Havinga, P.J.M. Fusion of Smartphone Motion Sensors for Physical Activity Recognition. *Sensors* **2014**, *14*, 10146–10176. doi:10.3390/s140610146.
8. Hassan, M.M.; Uddin, M.Z.; Mohamed, A.; Almogren, A. A robust human activity recognition system using smartphone sensors and deep learning. *Future Generation Computer Systems* **2018**, *81*, 307–313.

9. Rosati, S.; Balestra, G.; Knaflitz, M. Comparison of Different Sets of Features for Human Activity Recognition by Wearable Sensors. *Sensors* **2018**, *18*, 4189.
10. Nourani, H.; Shihab, E.; Sarbishe, O. The Impact of Data Reduction on Wearable-Based Human Activity Recognition. Proceedings of the 15th Workshop on Context Modeling and Recognition. IEEE, 2019, CoMoRea '19, pp. 89–94.
11. Zhang, M.; Sawchuk, A.A. Human daily activity recognition with sparse representation using wearable sensors. *IEEE journal of Biomedical and Health Informatics* **2013**, *17*, 553–560.
12. Zhang, M.; Sawchuk, A.A. A feature selection-based framework for human activity recognition using wearable multimodal sensors. Proceedings of the 6th International Conference on Body Area Networks. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2011, pp. 92–98.
13. Jordao, A.; Nazare Jr, A.C.; Sena, J.; Schwartz, W.R. Human activity recognition based on wearable sensor data: A standardization of the state-of-the-art. *arXiv preprint arXiv:1806.05226* **2018**.
14. Nourani, H. Gym Exercises Dataset. https://github.com/h0111in/Gym-Exercises-dataset/, 2019. [Online; accessed 05-May-2019].
15. Kołodziej, M.; Majkowski, A.; Tarnowski, P.; Rak, R.J.; Gebert, D.; Sawicki, D. Registration and Analysis of Acceleration Data to Recognize Physical Activity. *Journal of Healthcare Engineering* **2019**, *2019*.
16. Krishnan, N.C.; Juillard, C.; Colbry, D.; Panchanathan, S. Recognition of hand movements using wearable accelerometers. *Journal of Ambient Intelligence and Smart Environments* **2009**, *1*, 143–155.
17. Baldominos, A.; Cervantes, A.; Saez, Y.; Isasi, P. A Comparison of Machine Learning and Deep Learning Techniques for Activity Recognition using Mobile Devices. *Sensors* **2019**, *19*, 521.
18. Mortazavi, B.J.; Pourhomayoun, M.; Alsheikh, G.; Alshurafa, N.; Lee, S.I.; Sarrafzadeh, M. Determining the single best axis for exercise repetition recognition and counting on smartwatches. 2014 11th International Conference on Wearable and Implantable Body Sensor Networks. IEEE, 2014, pp. 33–38.
19. Soro, A.; Brunner, G.; Tanner, S.; Wattenhofer, R. Recognition and Repetition Counting for Complex Physical Exercises with Deep Learning. *Sensors* **2019**, *19*, 714.
20. Anguita, D.; Ghio, A.; Oneto, L.; Parra, X.; Reyes-Ortiz, J.L. A public domain dataset for human activity recognition using smartphones. Esann, 2013.
21. de Faria, I.L.; Vieira, V. A Comparative Study on Fitness Activity Recognition. Proceedings of the 24th Brazilian Symposium on Multimedia and the Web. ACM, 2018, pp. 327–330.
22. Anwary, A.; Yu, H.; Vassallo, M. An automatic gait feature extraction method for identifying gait asymmetry using wearable sensors. *Sensors* **2018**, *18*, 676.
23. Mazo, J.E. Faster-than-Nyquist signaling. *The Bell System Technical Journal* **1975**, *54*, 1451–1462.
24. Yurtman, A.; Barshan, B. Activity recognition invariant to sensor orientation with wearable motion sensors. *Sensors* **2017**, *17*, 1838.
25. Sarbishei, O. A Platform and Methodology Enabling Real-Time Motion Pattern Recognition on Low-Power Smart Devices. *2019 IEEE World Forum on Internet of Things* **2019**, pp. 257–260.
26. Zardoshti-Kermani, M.; Wheeler, B.C.; Badie, K.; Hashemi, R.M. EMG feature evaluation for movement control of upper extremity prostheses. *IEEE Transactions on Rehabilitation Engineering* **1995**, *3*, 324–333.
27. Shirahama, K.; Köping, L.; Grzegorzek, M. Codebook approach for sensor-based human activity recognition. Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct. ACM, 2016, pp. 197–200.
28. Xi, X.; Tang, M.; Miran, S.M.; Luo, Z. Evaluation of feature extraction and recognition for activity monitoring and fall detection based on wearable sEMG sensors. *Sensors* **2017**, *17*, 1229.
29. Accelerometer. https://www.w3.org/TR/accelerometer/. (Accessed on 07/03/2019).
30. Moon, T.K.; Stirling, W.C. *Mathematical methods and algorithms for signal processing*; Vol. 1, Prentice hall Upper Saddle River, NJ, 2000.
31. Janidarmian, M.; Roshan Fekr, A.; Radecka, K.; Zilic, Z. A Comprehensive Analysis on Wearable Acceleration Sensors in Human Activity Recognition. *Sensors* **2017**, *17*, 529.
32. Nair, V.; Hinton, G.E. Rectified linear units improve restricted boltzmann machines. Proceedings of the 27th international conference on machine learning (ICML-10), 2010, pp. 807–814.
33. Zhang, S.; Rowlands, A.V.; Murray, P.; Hurst, T.L.; others. Physical activity classification using the GENEA wrist-worn accelerometer. PhD thesis, Lippincott Williams and Wilkins, 2012.

34. Shoaib, M.; Bosch, S.; Incel, O.; Scholten, H.; Havinga, P. Complex human activity recognition using smartphone and wrist-worn motion sensors. *Sensors* **2016**, *16*, 426.
35. Bishop, C.M. Pattern recognition and machine learning (information science and statistics) springer-verlag new york. *Inc. Secaucus, NJ, USA* **2006**.
36. Masum, A.K.M.; Barua, A.; Bahadur, E.H.; Alam, M.R.; Chowdhury, M.A.U.Z.; Alam, M.S. Human Activity Recognition Using Multiple Smartphone Sensors. 2018 International Conference on Innovations in Science, Engineering and Technology (ICISET). IEEE, 2018, pp. 468–473.
37. Shakya, S.R.; Zhang, C.; Zhou, Z. Comparative Study of Machine Learning and Deep Learning Architecture for Human Activity Recognition Using Accelerometer Data. *Int. J. Mach. Learn. Comput* **2018**, *8*, 577–582.
38. Kose, M.; Incel, O.D.; Ersoy, C. Online human activity recognition on smart phones. Workshop on Mobile Sensing: From Smartphones and Wearables to Big Data, 2012, Vol. 16, pp. 11–15.
39. Chen, Z.; Zhang, L.; Cao, Z.; Guo, J. Distilling the knowledge from handcrafted features for human activity recognition. *IEEE Transactions on Industrial Informatics* **2018**, *14*, 4334–4342.
40. Zhu, C.; Sheng, W. Human daily activity recognition in robot-assisted living using multi-sensor fusion. 2009 IEEE International Conference on Robotics and Automation. IEEE, 2009, pp. 2154–2159.
41. Zhang, L.; Zhang, B. A geometrical representation of McCulloch-Pitts neural model and its applications. *IEEE Transactions on Neural Networks* **1999**, *10*, 925–929.
42. Sena, J.; Santos, J.B.; Schwartz, W.R. Multiscale DCNN Ensemble Applied to Human Activity Recognition Based on Wearable Sensors. 2018 26th European Signal Processing Conference (EUSIPCO). IEEE, 2018, pp. 1202–1206.
43. Brownlee, J. A gentle introduction to k-fold cross-validation. *Accessed October* **2018**, *7*, 2018.
44. Mehrang, S.; Pietila, J.; Tolonen, J.; Helander, E.; Jimison, H.; Pavel, M.; Korhonen, I. Human activity recognition using a single optical heart rate monitoring wristband equipped with triaxial accelerometer. In *EMBEC & NBC 2017*; Springer, 2017; pp. 587–590.
45. Liu, S.; Gao, R.X.; John, D.; Staudenmayer, J.W.; Freedson, P.S. Multisensor data fusion for physical activity assessment. *IEEE Transactions on Biomedical Engineering* **2011**, *59*, 687–696.

Sample Availability: Samples of the compounds ...... are available from the authors.