A Comprehensive Comparison of Human Activity Recognition using Inertial Sensors

Hosein Nourani

A Thesis

in

The Department

of

Computer Science and Software Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of

Master of Applied Science (Computer Science) at

Concordia University

Montréal, Québec, Canada

December 2020

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: **Hosein Nourani**

Entitled: **A Comprehensive Comparison of Human Activity Recognition using Inertial Sensors**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science (Computer Science)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

_____ Chair
Dr. Marta Kersten

_____ Examiner
Dr. Marta Kersten

_____ Examiner
Dr. Essam Mansour

_____ Supervisor
Dr. Emad Shihab

Approved by _____
Lata Narayanan, Chair
Department of Computer Science and Software Engineering

_____ 2020

_____
Mourad Debbabi, Dean
Faculty of Engineering and Computer Science

Abstract

A Comprehensive Comparison of Human Activity Recognition using Inertial Sensors

Hosein Nourani

Wearables are becoming increasingly popular. Their built-in sensors (e.g., GPS, accelerometer, gyroscope, light) can provide useful data for Human Activity Recognition (HAR). During the past few years, many HAR models have been introduced with different accuracies and performance. These models have been applied in different areas such as health, fitness tracking, entertainment, or advertisement.

Given that these HAR models run on wearables, which are resource-constrained, factors like inadequate preprocessing can negatively impact the overall HAR performance. While high accuracy is essential in some applications, the device's battery life is highly critical to the end-user.

Prior studies contain a plethora of activity recognition models and pre-processing techniques that show a very high recognition performance of these models. These results are mostly reported under a specified study setup different from others, making a fair comparison among them nearly impossible. Nevertheless, to date, very few studies have conducted a side-by-side performance analysis in HAR.

Therefore, in this dissertation, we investigate some of the most used HAR techniques to understand their impact when developing an end-to-end HAR model to recognize gym exercises (e.g., "treadmill, "bicep-curl," "Russian-twist"). This study allows us to examine the accuracy performance yielding from 5 state-of-the-art featuresets in HAR models. Additionally, we focus on feature selection methods and experiment on data reduction to understand trade-offs between accuracy levels and data size.

We find that histogram bins are a valid alternative featureset in HAR, with a significant positive impact on classification performance and classifier learning rate. Moreover, our finding shows that the data reduction techniques in the feature selection phase can decrease the data size by 93% (from 119 features to 8 features) with minimal impact on model performance, resulting in a large computation saving for the model.

Acknowledgments

First of all, I would like to thank my supervisor, Dr. Emad Shihab, for his dedicated support, guidance, and trust. His motivations and overall insights in this field have made this an inspiring experience for me. I have learned a great deal from you. Words could never be enough to express my gratitude. You played a formative role in my development as a researcher and as a person.

I would also like to thank Diego Costa for his thoughtful comments and recommendations on this dissertation.

My completion of this thesis could not have been accomplished without unparalleled encouragement and supports from Dr. Leila (Elham) Montazeri, who was the greatest motivation for me to continue my path toward success.

Furthermore, I would like to thank the Data-driven Analysis of Software (DAS) research team, Mohamed Elshafei, Ahmad Abdellatif, Rabe Abdalkareem, Suhaib Mujahid, Abbas Javan, who have been a great source of help.

To conclude, I cannot forget to thank my family and friends for all the unconditional support in this very intense academic year.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Research Problem and Scope

Human Activity Recognition (HAR) using on-body sensing is one of the most prevalent assistive technologies to support older people's daily life Wang, Cang, and Yu (2019), fall risk assessment Sow, Turaga, and Schmidt (2013), physical fitness monitoring Morris, Saponas, Guillory, and Kelner (2014), or medical diagnosis González et al. (2015), to name a few. Using wearables (i.e., smartphones, smartwatches) is becoming increasingly pervasive. Wearables are small in size, relatively cheap and ubiquitously used, reveal numerous new potentials for HAR systems in research and industry. As such, in about a decade, extensive researches have been undertaken in this regard and, in result, several outstanding high-performance HAR models have been proposed in the literature.

Recognizing human activity leads to learning profound high-level knowledge about individuals' activity patterns, which contributes to developing a wide range of user-centric applications such as health, monitoring elder people Dix, Dix, Finlay, Abowd, and Beale (2003), or fitness tracking Morris et al. (2014). There are potentials in giving contextualized recommendations to the user, such as suggesting TV shows, places to visit, physical activity to perform or even improve advertisements by correlating different data results. A HAR system based on wearables allows consistent sensing without Spatio-temporal limitation since it does not depend on any pre-installed equipment in the

environment. Also, these systems do not need shared data on a server or cloud that may threaten the user's privacy as a result, sensor-based HAR have become more popular and widely used. Therefore, in this thesis, our primary focus is on HAR approaches using wearables.

However, sensor-based HAR approaches have to cope with fundamental challenges. Wearables are small in size and inherently resource-constrained; That is, one cannot scale-up their computational power. Particularly, to design a system operating wearables, crucial challenges such as the cost of computation, the minimum power required or the storage needed to perform analysis, need to be addressed. One remarkable solution is the use of pre-processing approaches aiming to optimize the size and quality of the input data. While, during the last few years, extensive studies have targeted **data pre-processing task** in HAR, there are still open topics in this phase that require more investigations.

1.2 Motivation

A typical HAR process starts with recording a movement and convert it to a stream of data using one or multiple inertial sensors. Next, the recorded stream splits into segments with shorter lengths (multiple seconds). Through the so-called *feature extraction* phase, multiple techniques are applied to each segment to extract the features. The extracted features are filtered during the *feature selection* step and fed to a classifier for recognition tasks.

To optimize data in terms of quality and quantity, previous works have mostly focused on two aspects: i) the feature extraction phase, which includes signal processing methods on sensory data to determine features that better describe a movement, and ii) the feature selection phase, which attempts to find the smallest set of features by omitting less informative or redundant features from all extracted features. Therefore, using these approaches, previous studies have provided a better quality input data for HAR models, which resulted in better recognition performance.

Several sets of features have been introduced in previous studies, and authors have shown improvements in recognition accuracy when using those features. However, a side-by-side comparison that measures the improvement, cost, and performance of featuresets, which is necessary to be used

as a benchmark for future works, has not yet been explored. Therefore, the first half of this thesis addresses this question: Based on a systematic empirical investigation of multiple state-of-the-art featuresets, we explain the difference in the performance of HAR models using each set of features. We report and discuss the impact of each featureset per activity and per classifier model. We also investigate the subject-dependency evaluation of each trained model.

Selecting the most optimal set of features and reducing the size of data in HAR have been investigated in several studies (Bishop (2006); Mujahid, Sierra, Abdalkareem, Shihab, and Shang (2017); Nguyen, Fernandez, Nguyen, and Bagheri (2017).) It has been shown that by reducing the size of data, the cost of execution and required storage decrease; However, feeding a model with less amount of data also may cause a decrease in the model's performance. The second part of this thesis aims to examine the impact of feature reduction on wearable-based HAR. Specifically, we investigate the correlation between the number of features and the recognition performance and generality both in the cross-subject evaluation and cross-trial evaluation.

1.3 Thesis Contributions

The major contributions of the thesis are as follows:

- A side-by-side comparison of how state-of-the-art features affect HAR system performance.
- A detailed investigation of the impact of data reduction on HAR system performance and generality.
- A large dataset of gym exercise activities recorded under real-life conditions publicly available for future researches on HAR and fitness tracking analysis.
- A publicly available repository of scripts contains approaches to extract 1300 most frequently used features in HAR, to help the research community to acquire a broader range of informative characteristics of sensor data.

1.4 Thesis Overview

This thesis consists of five different chapters, briefly explained below. Chapter 2 provides a background overview needed to understand different aspects of HAR, namely inertial sensors and data preprocessing, activity recognition techniques and the tools used in this thesis. This chapter concludes with a summary of state-of-the-arts studies regarding sensor-based approaches in HAR. Chapter 3 presents a comprehensive investigation of hand-crafted features, including an empirical feature analysis through an end-to-end HAR system on gym exercises. Chapter 4 shows our HAR studies and experiments regarding feature selections and data reduction on a dataset of gait activities. Similarly to chapter 3, it includes a description, results and conclusions for each experiment. Chapter 5 presents our conclusions about our studies and suggestions for further research in this area. Besides, this dissertation includes Appendix A is presenting the catalogue of our dataset for gym exercises.

Chapter 2

Human Activity Recognition using Inertial Sensors

This chapter aims to summarize the different topics related to our study. First, we focus on a commonly used approach in HAR and its phases, and then we provide an overview of similar studies to the current dissertation.

2.1 Human Activity Recognition Approach

Wearable HAR systems share a typical workflow Banaee, Ahmed, and Loutfi (2013); Janidarmian, Roshan Fekr, Radecka, and Zilic (2017); Shoaib, Bosch, Incel, Scholten, and Havinga (2014). See this approach in Figure 2.1. The procedure starts with capturing an activity into a time-series signal using inertial on-body sensors (i.e., accelerometer, gyroscope). These recorded samples are stored as a stream of data into a dataset. Next, the stream splits into successive segments, while each segment is labelled with the activity performed within that segment. To be used in classification, each segment is summarized into a feature vector called feature extraction phase, an engineering process to design a set of dimensions representing data-points in a more distinguishable way - *hand-crafted features* plays a crucial step in the HAR process. In recent years, feature extraction approaches and designing a set of hand-crafted features have been extensively investigated; However, several spots remain open, subject to research. In the remainder of this section, we summarize the different

phases of HAR approaches.

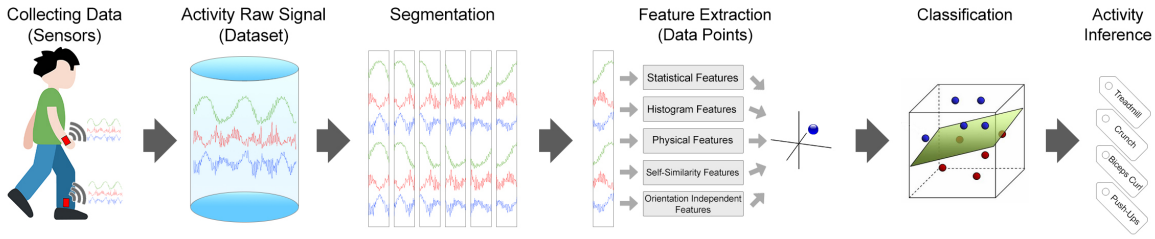


Figure 2.1: Typical Work-flow of Human Activity Recognition approaches

Motion Sensors

In the following, we describe the most commonly used inertial sensors and their functionalities in HAR:

- **The accelerometer:** responsible for measuring acceleration, typically in 3-axis (x, y, z). As any human motion in the space inherently changes its speed over time like running and walking, accelerometer is the most popular sensor for motion detection in space.
- **The gyroscope:** responsible for measuring the angular velocity in 3-axis. (x, y, z). Its main goal is to detect the human body's orientation and rotation in the space.
- **The magnetometer:** responsible for measuring magnetic fields. Generally, its goal is to find the direction toward the North. In HAR, it has a complementary role to the gyroscope in detecting the subject orientation. One of the downsides of this sensor is its sensitivity to iron, which generates significant noise in real-life applications.

There are other sensors (e.g., proximity or GPS) that have been used in HAR. However, they are not widely presented in motion detection.

Data Acquisition

In HAR using wearables to capture human activity in time series data, users can wear the sensors. Nowadays, smartphones and smartwatches are equipped with motion sensors (i.e., accelerometer, gyroscope.) Wearing them (e.g., smartwatches) or keeping them in hand or pocket

(e.g., smartphone) while their sensors are recording is a common example of HAR data acquisition. The recorded data is transmitted to a host computer and eventually is stored as a dataset for further analysis.

Challenges in acquiring data using wearable motions sensors include:

- Choosing the sensor based on the type of activity movements, i.e., a gyroscope, is more suitable in sensing tilting movements than a magnetometer.
- Choosing a body part to attach the sensor. For example, for a classifier to recognize the walking activity, sensors fixed to the feet provide more clear data than those fixed to wrists.
- Optimizing the number of sensors. While fewer sensors might negatively impact the recognition performance, more sensors on the user's body increase the processing cost and difficulties in real-life usages.
- Choosing a suitable sampling rate to capture the activity with enough resolution. The faster an activity is, the higher the sampling rate is required. However, a high sampling rate requires a more expensive sensor and bigger processing power.

These aspects of data acquisition phase have been investigated in previous studies. Table 2.1 shows the sensor setup and the target activities in related works.

Feature Extraction

Any distinctive characteristic of activity may spark an intuition in designing a HAR feature. For example: in Figure 2.2 (a), two states of *Triceps Kickback* exercise are shown. Each repetition of this exercise requires the body posture to change between these two states.[1] A triaxial accelerometer is fixed to the wrist to capture the movements towards three axes (shown in Figure 2.2 (b)). The blue and green points are indicating States A and B, respectively, through several repetitions. The following basic intuitions might be considered in feature extraction:

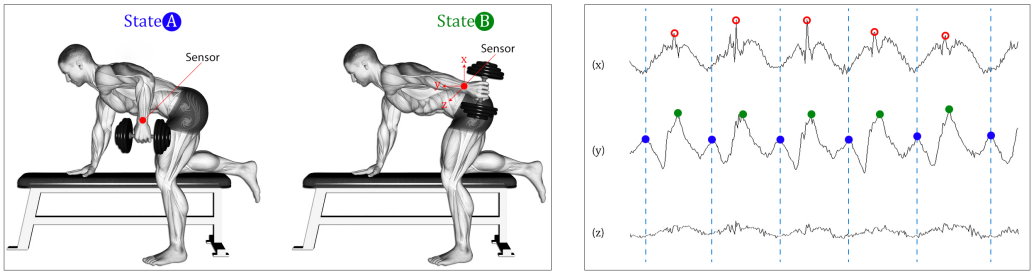
- when the wrist position is in the highest point, the rep is done, only one peak in the x-axis in each rep may lead to **using the *maximum* as a feature**

[1] A complete version of Triceps Kickback instruction includes more considerations about holding the dumbbell, inhale and exhale

Table 2.1: Data acquisition setup in previous HAR researches

Research	Sensor	Position	Frequency	Activity
Shoaib et al. (2014)	Accelerometer, Gyroscope, Magnetometer	Pocket, feet, wrist, waist, arm	50	daily routine
Sousa et al. (2017)	Accelerometer, Gyroscope	walking, sitting, standing, lying down, going up and down stairs	50	daily routine
Morris et al. (2014)	Accelerometer, Gyroscope	Curl, jumping jack, triceps extension, dumbbell row	50	Gym exercises
Yazdansepas et al. (2016)	Accelerometer	Treadmill, Seated, walking(High heels/sneakers), folding/stacking laundry, brushing teeth/ hair	100	workout & daily routine
Rosati, Balestra, and Knaflitz (2018)	Accelerometer, Gyroscope	resting, upright standing, level walking, ascending and descending stairs, uphill and downhill walking	80	daily routine
Shoaib et al. (2014)	Accelerometer, Linear Accelerator, Gyroscope, Magnetometer	walking, jogging, biking,standing, sitting,typing, writing, talking, eating, drinking coffee and smoking	50	Working & leisure

- each rep contains two minimums and two maximums in the y-axis, which leads to **counting number of peaks as a feature**
- the wrist does not move significantly toward z-axis, which leads to **less considering this axis in feature extraction**
- The exercise is composed of movements in x and y hyperplane. Therefore, **a transformation from x and y page** to a vector space that represents the movements more explicitly can be considered as a feature.



(a) (b)

Figure 2.2: (a) Two states of "dumbbell triceps dips" while a sensor attached to the wrist. (b) A triaxial accelerometer recorded the signals during five iterations of dumbbell triceps dips. Red and green bullets stand for State A and B, respectively. The red circles indicate the maximum peaks in the x signal. In the feature extraction process, an input like the y-axis signal might be considered as an intuitive feature such as mean or median to recognize each state of this activity.

Similar approaches - based on the understanding of the movements have been used to design hand-crafted features in HAR in the literature Khokhlov, Reznik, Cappos, and Bhaskar (2018); Rosati et al. (2018); Shoaib et al. (2014). From this point of view, feature sets can split into five categories: 1) statistical features, 2) histogram features, 3) self-similar features, 4) physical features, 5) orientation independent features.

Statistical features. are the most popular features in HAR. The idea is to extract statistical information from the signal using mathematical formulations. This method has been intensively investigated in HAR, and are proved to be effective in a wide range of experiments Khokhlov et al. (2018); Rosati et al. (2018); Shoaib et al. (2014). Shoaib and Bosch Shoaib et al. (2014) setup an experiment with ten subjects to recognize seven activities from daily life. They extracted two groups

of statistical features, including six time-domain features (mean, standard deviation, median, zero crossings, root means square and variance) and two frequency-domain features (Fast-Fourier Transformation (FFT) coefficients and spectral energy). They found that using both groups improves recognition performance. Khokhlov et al. (2018) pivoted a study on five daily life activities and seven subjects and four classifiers. they showed that using only the accelerometer sensor and statistical features, an increasing number of features (from 4 to 7) increases the accuracy between 3% to 20% (97.6% with KNN classifier) while adding a gyroscope sensor might increase the recognition performance by about 0.1%.

Histogram Features. are developed based on the probability distribution function of the signal of activity during a period (window size) Zardoshti-Kermani, Wheeler, Badie, and Hashemi (1995). In HAR, because each activity contains a set of small movements (as small as one sample) with specific acceleration and rotation, histogram bins indicate the difference between activities by showing the different distributions of those small movements. Xi et al., in Xi, Tang, Miran, and Luo (2017), used EMG signals in fall detection and gait analysis. They compared 15 individual features, including statistical features and histogram features on five classifiers and showed the fuzzy neural network classifier using histogram features provides the highest sensitivity and specificity with 98.70% and 98.59%, respectively. Sarbishei et al. Sarbishei (2019) used histogram features and Forward Neural Network (FNN) in a low-power real-time HAR system. They showed that while histogram bins are significantly low cost in terms of required processing time and memory usage, they are sensitive against the resolution/granularity of bins (count and width of bins). To investigate these unique characteristics of histogram features, we include them in comparison, in this study.

Self-Similar Features. Considering that 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 Morris et al. (2014). 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.

Physical Features. One intuitive idea to design a set of features from sensory data is to consider the principles of human movements. In 2011, Zhang et al. M. Zhang and Sawchuk (2011) introduced a set of features based on physical parameters of human motion. To have a robust 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, these types of features are derived based on the physical interpretations of human motion, called physical features (i.e., the correlation between the gravity and heading direction). Compared to other featuresets, these features account for a fusion of multiple sensor inputs rather than just one inputs sensor.

Orientation Independent Features. In contrast to physical features, which depend on the position and orientation of sensors, Yurtman et al. Yurtman and Barshan (2017) and Siirtola et al. Siirtola and Röning (2012) proposed features that do not rely on the variation of sensor orientation. In fact, in their model, they introduced Orientation-invariant transformations (OITs) that are inspired by the idea of *single value decomposition* Moon and Stirling (2000). 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.

Feature Selection.

Employing feature selection is very popular as it has been proved that it could significantly improve performance. Several studies have shown that a careful selection of features improves the model's performance both in accuracy and computational costs Nourani, Shihab, and Sarbishe (2019); Rosati et al. (2018); Wang et al. (2019). For example, in Nourani et al. (2019), we setup a study/experiment on time-domain features and used an ensemble feature selection (a combination of filtering method and wrapper method) and showed an identical performance while using only 10% of features. Similarly, in Yazdansepas et al. (2016), authors showed the same performance using 20% of the initial dataset.

The ain approaches in feature selection are: 1) the filter Method M. Zhang and Sawchuk (2011), 2)

the wrapper method Rosati et al. (2018), and 3)the embedded method Nourani et al. (2019). Often, researchers try different selection approaches on a given set of extracted features and choose the set providing the highest performance. Therefore, the performance achieved by using a featureset tightly bounds up both extraction and selection phases. For the same reason, in this study, we investigate the performance of featuresets as the ultimate products of feature engineering in HAR.

Classification.

In this study, we are dealing with a supervised Machine Learning (ML) task. This category of ML algorithms requires ground truth data (or labels) for learning a function that maps an input to an output. In HAR, a supervised ML infers a function (classifier) from labelled training data consisting of recorded sensory data as input and the human activities as output. A wide range of machine learning methods has been applied for the recognition of human activities. We conduct our experiments using machine learning methods including Naive Bayes Rish et al. (2001), Decision tree (DT) Friedl and Brodley (1997), K-Nearest Neighbors (KNN) , Support Vector Machines (SVM) , Feed-Forward Neural Network (FNN) , and ensemble of classifiers . The main objective of implementing different classification techniques is to review, compare and evaluate their performance considering the most heterogeneous dataset on gym exercises publicly available.

Naive Bayes. classifiers Rish et al. (2001) is one of the most known classifier models being studied since the 1950s. In this approach, the primary assumption is the independence between input features. The conditional likelihood function of each activity can be expressed as the product of simple probability density functions. Naive Bayes is one of the most popular classifiers in HAR as it is one of the lightest classifiers in recognizing human activities. Furthermore, to learn a Naive Bayes classifier, one needs a small amount of data to output results.

Decision Trees. Friedl and Brodley (1997) build a hierarchical tree using a divide-and-conquer strategy, consisting of splitting the input data into several smaller areas labelled by activity names. These decision trees are represented with decision and leaf nodes mapped to attributes and values,

respectively. 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 as they require less data preparation and computation resource for recognition Baldominos, Cervantes, Saez, and Isasi (2019); Mortazavi et al. (2014); M. Shoaib, Bosch, Incel, Scholten, and Havinga (2016). Baldominos et al. (2019) reported the best recognition performance for decision tree among a set of classifiers including Naive Bayes, KNN, FNN, and Logistic Regression.

k-Nearest Neighbors. k-Nearest Neighbors (k-NN) Duda, Hart, and Stork (2012) is a supervised classification technique that can be seen as a direct classification method because it just requires the whole dataset for recognition-no learning process in advance. K-NN algorithm uses the principle of similarity (distance) for classification. To classify a new observation, K-NN measures the distance between classes in the training set and the new observation. The distance measurement uses a similarity function (i.e., Euclidean distance.) A majority vote technique in k nearest neighbors is employed to assign this new observation to the most common class.
Attal et al. Attal et al. (2015) applied the k-NN classification to recognize twelve activities carried out by six subjects. They compared four supervised classification techniques, namely, k-Nearest Neighbor (k-NN), Support Vector Machines (SVM), Gaussian Mixture Models (GMM), and Random Forest (RF). Their approach has shown that the k-NN classifier provides the best performance (94.53% F1) compared to other classifiers ($<$ 90% F1.) Other studies based on k-NN for human activity recognition have also shown high accuracy and satisfactory segmentation results Shakya, Zhang, and Zhou (2018); Shakya et al. (2018); Wang et al. (2019,?). It is worth mentioning that, using this classifier, the computational cost of classification increases as the size of dataset grows Trabelsi, Mohammed, Chamroukhi, Oukhellou, and Amirat (2013). To address this issue, in Kose, Incel, and Ersoy (2012), the authors showed the positive impacts of data reduction techniques on k-NN performance. They also proved that increasing k improves the performance of the HAR model.

Support Vector Machine. A multi-class Support Vector Machine Suykens and Vandewalle (1999) (SVM) has been employed extensively in previous studies in HAR to discriminate among the activities Morris et al. (2014); Rosati et al. (2018); S. Zhang, Rowlands, Murray, Hurst, et al.

(2012). Assuming each data point is a co-ordinate (support vector) of feature space, Support Vector Machine (SVM) centers on the construction of a hyperplane in a high or infinite-dimensional space. SVMs work well when the number of dimensions is greater than the number of instances. Morris et al. Morris et al. (2014) used SVM to recognize 13 gym exercises in an end-to-end human activity recognition system. Their results showed, using leave-one-out cross-validation analysis, the SVM accuracy is 96% (on average).

Ensemble Learning. Ensemble learning Dietterich et al. (2002) is characterized by a combination of multiple classifiers in order to maximize accuracy. Specifically, first, multiple classifiers are employed to recognize one versus rest. Next, using a majority voting scheme, the best answer is taken as the final classification output. Some of those techniques include bagging, boosting or stacking. While ensemble methods improve accuracy significantly Nourani et al. (2019), their computational cost is relatively higher due to deriving multiple classifiers internally.

Feedforward Neural Network. A Feedforward Neural Network (FNN) is an artificial neural network with a multilayer wherein the connections between layers do not form a cycle. The FNN minimizes the error function between the estimated and the desired network outputs, representing the activity classes in the HAR context. The network's input (first layer) represents the features extracted from the sensor signal. The feed-forward architecture in this classifier is based on non-linear activations for internal layers (hidden layers).

Several studies show that FNN is efficient in non-linear classification problems, including human activity recognition. The FNN has been applied in several studies for human activity recognition, such as Baldominos et al. (2019); Z. Chen, Zhang, Cao, and Guo (2018); Zhu and Sheng (2009). In De Leonardis et al. (2018), authors used FNN to recognize eight different activities, including sitting, standing, lying down in supine position, level walking, ascending and descending stairs, uphill and downhill walking. They showed that FNN recognizes activities with 90.7% accuracy in 5-fold cross-validation. In another study Baldominos et al. (2019), using multi-layer perceptron, achieved 94.44% of classification accuracy in recognizing 13 activities including physical activities

(walking, jogging, biking, going upstairs and going downstairs), common postures (standing, sitting), working activities (typing, writing), and leisure activities (talking, eating, drinking coffee and smoking).

2.2 Related works

In this study, we target two aspects of the pre-processing phase in HAR: i) feature extraction and ii) feature selection. In this section, we review recent works about data reduction in features selection; and investigate recent studies relevant to a different type of features introduced in HAR.

2.2.1 Hand-Crafted Features In Human Activity Recognition

This section presents some studies related to feature investigation in HAR. All the works presented in this section perform similar experiments to recognize human activities using wearables. Also, they repeated the same experiment on multiple featuresets in their comparison.

In Sousa et al. (2017), the authors proposed a feature classification based on the domain of features, including i) time-domain features, ii) frequency-domain features, iii) discrete-domain features. They provide an extensive comparative study between dependent and independent-orientation features extracted from smartphones inertial sensors. They showed how using Time domain features makes a featureset dependent on the orientation of the movement. Besides, they proved that a featureset of time-domain features using accelerometer information provides the best characteristics to recognize physical activities utilizing data analysis of the smartphone inertial sensors.
Grouping and comparing features based on their inherent characteristics are useful and provide insights into similarities and differences of features. However, in HAR, featuresets are mostly composed of a combination of multiple feature domains. In this thesis, we provide featuresets designed based on an understanding of inertial movements of activities. Therefore, knowing the performance of each featureset in this study goes a long way in deciding a featureset for a future study.

In Shoaib et al. (2014) the authors evaluated the activity recognition performance with four motion sensors using nine classifiers on five body positions with four featuresets. There were three featuresets based on time-domain features and one featureset on the frequency domain. They showed

that the performance of featuresets varies based on the types of activity being recognized, the sensor position and the classification method. In featureset comparison, they concluded the time domain features always provide more informative input for classifiers than those of frequency domain. They also showed features extracted from the accelerometer performs slightly better than those from the gyroscope.

Inspired by this study, we attach multiple sensors on two body positions (wrist and ankle) and acquire data from both the accelerometer and gyroscope. However, in feature comparison, we provide five developed features with more contrast in characteristics, thereby, more applicable results for future real-life experiments. Also, the dependency on the subject is an impactful aspect of features Jordao, Nazare Jr, Sena, and Schwartz (2018), which is neglected in the literature studies. In this thesis, we evaluate the performance of featuresets using K-fold, Leave-One-Subject-Out cross-validation and Leave-One-Trial-Out cross-validation to investigate the impact of subject dependency on hand-crafted features profoundly.

In a most recent related work, Rosati et al. (2018) have targeted the performance of a real-time HAR model based on physical features versus its performance on statistical features. To build the statistical featureset, they applied 37 feature functions, including 20 time-domain functions, three frequency-domain functions, and 14 time-frequency-domain functions. To build the physical featureset, they performed a pre-processing phase, in which they highlighted positive and negative peaks, calculated single and double integration of the acceleration (antero-posterior direction and medio-lateral direction). Then, they applied 56 feature functions on these processed signals. They used a Genetic Algorithm (GA) in the feature selection phase and four classifiers, including KNN, FNN, SVM, and DT, in classification. Their results showed that the highest performance was achieved by the SVM model (97.1% and 96.7% of accuracy for using statistical features and physical features, respectively.). Although both featuresets provide a recognition performance above 96%, physical features are easier to be interpreted as their biomechanical meaning can be linked to the inertial movements of the activity, which results in more understandable features, especial in more complex activities.

In our experiments, we do not apply any feature-selection method to features compared to their study. This is essential as it maintains the original information of each featureset, which keeps the

study setup fair for side-by-side comparison. However, they applied a feature selection process on their extracted features, which may alter the final comparison results. Besides, they calculated single and double integration of acceleration signal to produce velocity and movement of the subject; however, in M. Zhang and Sawchuk (2011), authors mentioned removing gravity from the linear acceleration signal is a crucial step to build these physical features. We build on this approach and remove gravity from the linear acceleration in our physical featureset. Rosati et al. (2018) targeted seven simple activities (i.e., resting, standing, walking), which are not challenging enough to highlight the performance of featuresets thoroughly. We focused on gym exercises that carried on at a higher speed rate with more complex movements involved to show the differences more distinguishable.

2.2.2 Data Reduction in Feature Selection

Comprehensive reviews about the subject of feature reduction are available in the literature. In Janidarmian et al. (2017), the authors evaluated the performance of 293 classifiers using principal component analysis (PCA) as their feature reduction method. They applied PCA on data of 14 public datasets of accelerometer data. Using PCA, not only did they reduce the size of data, but they also normalized the data recorded from different studies. They extracted features independent of x/y/z axes and, consequently, independent from sensor orientation. Similarly, the approach in Yong, Sudirman, Mahmood, and Chew (2013) uses principal component analysis (PCA) to feed the classifiers a smaller size of the input. They found that ensemble methods of KNN provides the best recognition accuracy at the lower size of data, and Decision Tree (DT) provides the worst. They also showed that, on average, the best and worst positions for attaching sensors are the right thigh and left lower arm, respectively. Shoaib et al. Shoaib et al. (2014) experimented with ten subjects and five sensor positions to show the impact of sensor positions on activity recognition. Their results also confirmed that the right pocket (upper thigh) and wrist are respectively the best and worst positions. Comparing different featuresets, they also concluded that selecting the best sensor (accelerometer vs gyroscope) to achieve the best performance depends on body position, activity type, and classifier.

The authors in Erdaş, Atasoy, Açıcı, and Oğul (2016) extracted three feature-sets, including time-domain, frequency domain, and wavelet-domain statistics. They employed an ensemble selection on five feature selection methods and showed that their best results were achieved using time-domain features. M. Zhang and Sawchuk (2011) extracted some self-designed features called physical features and showed that these features have more contributions rather than time-domain features to the recognition system. Introducing a multi-layer classifier, they also show that different featuresets are appropriate for different activities.

More approaches on feature selection methods such as recursive feature selection Nguyen et al. (2017), correlation-based feature selection (CFS) Maurer, Smailagic, Siewiorek, and Deisher (2006), Independent Component Analysis (ICA) Mantyjarvi, Himberg, and Seppanen (2001), and Local Discriminant Analysis (LDA) Ghasemzadeh, Loseu, Guenterberg, and Jafari (2009), targeting HAR, also exist in the literature. Maurer et al. (2006) introduced CFS approaches that use the intercorrelations and feature's predictive performance to limit the search area for a good subset of features. In ICA Mantyjarvi et al. (2001), the main idea is to find a linear transformation that minimizes the statistical dependence between features.

As mentioned above, existing works mostly employ different forms of feature selection to find the best performance of their model. In this thesis, we investigate feature selection attributes independently and concerning the whole model. For the feature selection method introduced in this thesis, the most relevant previous work is from Ienco et al. Ienco and Meo (2008), who similarly divides the process into two stages and uses a hierarchical clustering followed by a wrapper method. They show that their method on various datasets outperforms filter and wrapper methods. Furthermore, using the dendrogram of features provided by hierarchical clustering gives a semantic view of feature space. However, they do not explain how much their method can reduce the size of the feature set and how it affects the generality of the models. In this thesis, we address these aspects as well as we will have a more in-depth view of the advantages of data reduction on the HAR pipeline.

2.3 Summary

In this chapter, we introduced the HAR pipeline as a multi-phase approach and highlighted standard techniques that have been used in each phase. Considering data pre-processing as a crucial step in the HAR process, we reviewed best practices in feature extraction and feature selection techniques used in the literature. As the first fold in this study, we conduct a detailed investigation through feature extraction and hand-crafted features, including our experiments, dataset, and a side-by-side comparison of the state-off-the-art featuresets, in the next chapter.

Chapter 3

Comparative Investigation on HAR Hand-Crafted Features

3.1 Introduction

In HAR, the pre-processing phase plays a crucial role in acquiring the generalizability and performance of the model Schilit, Adams, Want, et al. (1994); M. Shoaib et al. (2016); Soro, Brunner, Tanner, and Wattenhofer (2019). Specifically, the main task in this phase is to build the feature vector out of the input raw data. If the features extracted in this phase are less dependent on the subject and temporal characteristics of an activity (i.e., speed or orientation), the trained model by these features will be more robust and perform more accurately in recognition. To this aim, there are several studies targeted at providing features for HAR in the literature. Previous works mainly focused on introducing new features or providing a selection of features for the recognition model. While results show that these featuresets are performing significantly well under their study setup, to the best of our knowledge, there is no study that conducted a side-by-side comparison between featuresets to show their performance as compared to each other through the same setup. In this study, we reproduce five state-of-the-art featuresets, including 1300 features and investigate their performances through an end-to-end HAR system.

Additionally, we investigate the performance of featuresets when they are fed to different classification models and also validate different evaluation methods for each model.

In the following sections first, in Section 3.2, we describe the experiment setup including our dataset and the method to evaluate featuresets. Then, we report our results through three research questions, in section 3.3. A brief summary of this chapter is in Section 3.4.

3.2 Methods and Dataset

This section starts with the process of how we build our dataset for HAR. Then, we describe the details about the methodological framework for comparing the performance of featuresets.

3.2.1 Participants and Activities

We contacted 25 individuals to participate in between 1 and 5 sessions in this study. Subjects were selected randomly from gym members of Concordia University and had read and signed an informed *participation consent form* that was approved by Concordia University ethics office. Participants were asked to fill a form, including their personal information, their background in practicing gym exercises, and the frequency of repeating each exercise per month. The data of eight participants were excluded. First, because their activities were uncommon (i.e., too professional or too personalized (data outlier)), so we could not find multiple subjects to repeat those exercises. The second reason was due the technical errors such as a device failure or the software bugs that cause the data to become unreadable. As we wanted to conduct cross-trial validation in evaluating the models, we also excluded the data of 2 participants who had less than three reps of activities (except for treadmill). In total, 15 participants (4 female), ages 21-35, carried out 53 sessions including at least three trials of each 8 exercises (Figure 3.1). Participants varied in level of expertise in gym exercises (from 1 month to 6 consecutive years of experience).

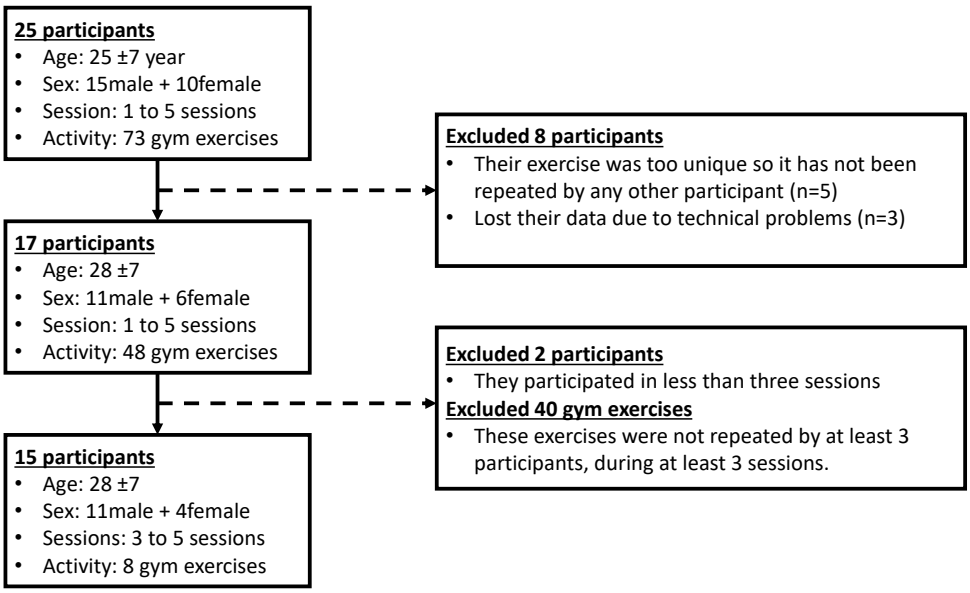


Figure 3.1: Recruitment process flowchart

Regarding the activities, we recorded the data of 73 gym exercises. However, 25 exercises, including body-weight-training and some special movements for warm-up, have been excluded as they were not common among participants. Besides, any exercise is repeated in less than three sessions by one subject that has been removed (40 exercises). It is essential to have at least two sessions (either on the same day or on multiple days) of an exercise to apply the cross-trial evaluation method. Therefore, finally, we picked 8 activities shown in Table 3.1 that meet all the constraints required in this study.

We ensured different body parts (upper-body, lower-body) get involved in the exercises as it brings the need for attaching multiple sensors to the subject's body. As expected, the exercises are from beginner to intermediate level as they are more common between participants. That is, collecting data from more advanced exercises may require recruiting participants at a certain level of expertise.

Table 3.1: Statistics of the dataset divided by type of exercise along with the experiments that involve them in. Column *Sessions* shows the total number of sessions that an exercise appeared in. Column *Subjects* shows how many subjects performed an exercise.

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	9	67	8520	Lower	A8

3.2.2 Sensors

To record the data, we employed a System-on-Chip (SoC) called Neblina (Figure 3.2 (a)). **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 a Bluetooth port, it can transmit the result to a host (e.g., cellphone or desktop computer). Neblina is equipped with all requirements for a real-time HAR system. As compared to a smartphone, Neblina is much smaller (Figure 3.2) that lets us attach it to a different part of the subject's body without making any interrupt in his/her actions de Faria and Vieira (2018) and from a technical point of view, it is designed specifically for motion tracking purposes. That is, not only does it provide access directly to its resources like sensors or memory without OS interference, it shares an interface for the user to configure the performance (i.e., sample rate, synchronization) of sensors, which is essential from the research standpoint.

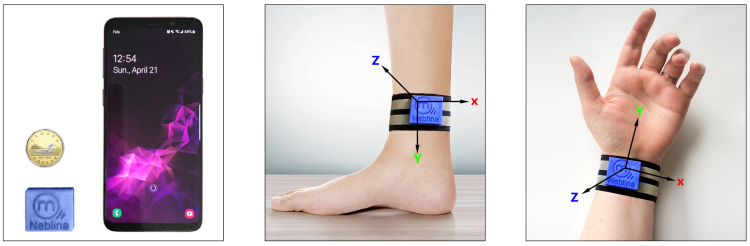


Figure 3.2: Neblina setup. (a) Compares dimensions of Neblina with a 1 dollar coin and a cellphone (Samsung Galaxy s9). (b) How Neblina located on foot using a strap. (c) How Neblina located on wrist using a strap.

Although the device is equipped with all three accelerometers, gyroscope, and magnetometer, we only store the accelerometer and gyroscope's input stream. We omit the magnetometer signal as it is highly affected by iron equipment commonly known in gyms. The frequency rate is fixed on 50Hz as the fastest gym activity is not as fast as 25 reps per second Mazo (1975).

Depending on which body parts are involved in a task (exercise), a HAR system may need to have one or more sensors to recognize a movement Wang et al. (2019). While using more sensors provide more comprehensive data for the recognition model, it limits the usability of the system and causes discomfort to users wanting to wear them de Faria and Vieira (2018); RajKumar, Vulpi, Bethi, Raghavan, and Kapila (2020). We attach two Neblina modules, one to the right wrist and one to the right ankle of participants. Two sensors set to cover the motions on upper-body and lower-body activities. The way that we fixed the modules and the orientation of the module is shown in Figure 3.2 (b) and (c).

3.2.3 Data Collection Procedures

Prior to the workout sessions, biometric data was recorded for all the participants, including body height, weight, and body fat percentage. All participants completed questionnaires to assess their background in each exercise in terms of duration and frequency (per week) of practicing. Based on these questionnaires, we aim to have a measure of participant experience level. Participants were informed about the procedure of recording movements using Neblina. They were informed about the sensor positions (on their wrist and ankle), about the supervisor's presence during the recording, and receiving instruction for in case an error occurs during the workout. This instruction aims to

avoid any irregular interruption during the workout (i.e., if the sensor moves). Two synced sensors were attached to the subject's body during the whole session. A supervisor with a stop-watch clock was recording the start/end of repetitions and the name of each exercise. After finishing each session, we transfer the data from Neblina to a host computer, where we manually adjusted the start/end moments of each exercise by visualizing the recorded signals.

In total, the process of labelling includes the following steps: (1) **participant**: before the session, each subject was asked to list the exercises that she/he is about to practice, including name, number of sets and reps, and the weights if applicable. We used this information as an initial draft for labelling. (2) **supervisor**: during the session, a supervisor manually records the type of exercise, the moment of start and stop of sets, and the number of sets. (3) **visual signal**: after finishing the session, in order to have our desired accuracy in labelling, we visually trace the signals of the accelerometer and gyroscope to refine the period assigned to each set. Manually adjusting labels, we could fix any error missed in previous steps.

Aiming to build a real-life gym exercise dataset, we focused on the following aspects:

(1) **Realistic exercise plan**: We did not limit subjects to a certain set of activities. So, they were allowed to do their own exercises at their preferred way. Although this can let subjects to perform an activity in a non-identical way, it replicates real-world condition in our data collection process. In Morris et al. (2014), the authors showed that by changing the environment from a 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 conditions is the performance of the HAR model is more close to real-world experiments.

(2) **Realistic null-class activities**: In exercise recognition, each segment is labelled with one of the target activities that the classifier tries to predict. If a segment contains an activity other than target activities, it is labelled as a null-class activity. Authors in Soro et al. (2019), asked participants to perform specific actions to reproduce the null-class activities. One of the most challenging parts of activity recognition is finding the beginning and the end of an exercise. Recording null-class activities separately and attaching it between activity periods

cannot correctly replicate the real-life transitions. Therefore, the model performance might be affected by this limitation. However, in this study, the unknown period or null-class activities are not artificially performed; instead, subjects were free to do whatever they usually do in the gym while the sensors were continuously recording their motions.

(3) **Effects of fatigue**: On some activities like running on a treadmill, the pattern of activity is significantly affected by the level of fatigue Lee, Youm, Noh, and Park (2020). In gym exercises, the longer a workout session is, the more tired subject becomes. In order to collect a more generalized dataset, in this study, each session contains 1 and 2 hours non-stop workout. Therefore, activities are recorded at different levels of tiredness of subjects.

(4) **Impact of subject experience**: Gym exercises are performed iteratively over weeks or months. By repeating an activity, subjects become more comfortable doing it, thereby getting more consistent. Keeping the consistency in performing an activity makes the activity more recognizable for a HAR model Morris et al. (2014). That is, the recognition accuracy of a HAR model might vary based on the average level of expertise of the study's subjects. The more experienced subject participate, the higher recognition accuracy is expected. However, this is not the case in real-life applications with a wide range of users (beginners to professionals). Nevertheless, it has not been addressed in most public HAR datasets Anguita, Ghio, Oneto, Parra, and Reyes-Ortiz (2013); Shoaib et al. (2014). Therefore, before each session, we asked participants about their background on doing each exercise. Aiming to observe if there is any correlation exists between the recognition performance and the level of the subject's experience.

3.2.4 Feature Extraction

In this study, we targeted five state-of-the-art featuresets as our case studies. Each featureset is designed based on a certain aspect of human activities (i.e., self-similarity or orientation dependency) and has been proved to be effective in a separate study in the past. We selected those featuresets that provide enough information to reproducibility (i.e., the definition of the feature along with preprocessing operation required to build them). To make an outlook on each featureset purpose, we

provide a description-intuition for every single feature function. Table 3.2 3.3 shows these functions along with their description/intuition.

Shared Preprocessing

The data received from sensors are transformed into a table where the columns represent signal axes and rows represent the samples. As the sample rate in this study is 50Hz, there are 50 rows per second recorded in the database. There were two shared preprocessing operations among featuresets that we applied on all input columns: i) removed rows with missing cells, ii) normalized data per column using min-max normalization and scaled them between 0 and 1. While these operations are performed before the segmentation phase, any further preprocessing required by featuresets, carried out after segmentations. It is worth mentioning that further preprocessing operations apply to each segment rather than the whole data column.

Set_A: Statistical Features (ST_Set)

The statistical features have been intensively investigated in previous studies and proved useful for activity recognition Khokhlov et al. (2018); Rosati et al. (2018); Shoaib et al. (2014). These features are computed from each sensor axis (i.e., x/y/z of the accelerometer) by applying a statistical function such as mean or variance. As a part of the processing operation to build statistical features, we calculate the *cumulative sum* for all 12 input signal columns. Therefore, we have 24 data columns (12 raw data and 12 cumulative sums) to apply 11 feature functions (S1-S11 shown in Table 3.2) to them. In total, we created a set of (24x11=) 264 features in this featureset.

Set_B: Histogram bins Features (HB_Set)

The second set of features are histogram bins which differentiate activities based on their intensity subsequences. For example, histogram bins have been proved to achieve consistently high accuracy to differentiate activities such as running and jogging, as body movements in running are more intense than that in jogging Oreifej and Liu (2013); Sarbishei (2019). Although these features are basically statistical, they have been independently studied in previous works Sarbishei (2019) and showed that they could be replaced with statistical features. Thus, as part of our comparative

study, we also consider them a separate set in our experiments. The histogram features' processing phase calculates the signal's magnitude using equation 1 for the accelerometer and gyroscope. As there are sensors on two positions (wrist and ankle), it includes 4 processed data columns to the first 12 raw data columns. Therefore, we used 16 data columns as input to build the histogram features.

$$Magnitude = \sqrt[2]{x^2 + y^2 + z^2} \tag{1}$$

To construct features, first, we binned the range of values (between 0 and 1) into 20 consecutive buckets. So, a bucket accounts for 5% of the value range and contains the total number of intervals fell into that bin. Histogram bins are indicated with the code HB in Table 3.2. In total, we produce 320 features in this featureset.

Set_C: Self-Similar Features (SS_Set)

This featureset is designed based on the quality of repetitive movements of an activity (i.e., the rising wrist in each interval of bicep curl). To extract these features, there are four processing operations performed before extracting features. We transformed x/y/z axes of each sensor signal to 4 processed data columns described as follows: 1) the magnitude of x/y/z axes using equation 1, 2) the first principal component of x/y/z axes of each sensor, 3) the first principal component of x and z axes, 4) the scaled normalized of the y-axis. Compared with the original study's setup Morris et al. (2014), in our study, the y-axis of the sensor is aligned with the user's arm while it is x in the original one. This happened due to the difference in the sensor orientation between the two studies. Built for processed signals per each sensor, we have 16 input columns to extract self-similar features from them. Therefore, there are 20 functions shown in table 3.2 that we applied to 16 input columns to build 320 features in self-similar featureset.

Set_D: Physical Features (PH_Set)

Interpreting physical motions' trajectory, such as trajectory velocities or trajectory speeds of human activities, develops the intuition behind physical features. Two mandatory prior knowledge in extracting physical features are the position and the orientation of sensors placed on the subject's

body M. Zhang and Sawchuk (2011). In our study setup, the two sensors (the right wrist and the right ankle) are placed so that their y-axis is toward gravity direction, and the x-axis is toward the subject's heading. The feature extraction, in this group, plays two different roles. First, it mathematically prepares the required signals for the extraction phase; next, it carries out the sensor fusion - combining input signals from multiple sensors. The main operations required to build this featureset composed of:

- the removing the gravity from the accelerometer signal (used in Ph1 and Ph2). We used gyroscope data as described in Waldron (n.d.) to remove the gravity from the acceleration signal.
- the Euclidean norm of three axes of each sensor (used in Ph1, Ph12, Ph13)
- the covariance matrix of acceleration data along x, y, and z-axis in each segment. (used in Ph4)
- the cumulative sum of accelerometer signal on the x-axis (used in PH6)
- the Fast Fourier Transformation (FFT) from each input signal. (used in Ph9, Ph10)

There are 13 feature functions as described in Table 3.3 (codes Ph1-Ph11). We build 46 features by applying them to the aforementioned processed signals.

Set_E: Orientation Independent Features (OI_Set)

The main idea in this featureset is providing flexibility in attaching the sensor to the body. As these features do not depend on the sensor orientation, the sensor can be loosely attached to the body, improving usability. Besides, they also make the model insensitive to sensor rotation during human movements. To build this featureset, first, we remove the direction from input data. It is achieved by projecting every data point from its original x/y/z space to another three-dimensional space but at the farthest distance between data points Yurtman and Barshan (2017). In this new space, axes' direction is defined by the 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 Janidarmian et

al. (2017). In Table 3.3, these type of features are indicated by ”OI” prefix. Therefore, we extracted 30 features to build OI_Set.

Table 3.2: Statical Functions along with the definitions and abbreviations (Statistical features, Self-Similar features, and Histogram bins features)

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 samples and the mean	MAD
S6	Median Absolute Deviation	The average distance between samples and the median	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 autocorrelation peaks	The bigger number means non-periodic activity while smaller number refers to periodic activity	NAcP
SS2	Prominent autocorrelation 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	less height refers to more fluctuations within the stream (after zero-crossing)	FAcP
SS6	Power bins (10 bins)	A 10 bins distribution of amplitudes of frequencies from 0.2-25Hz	Pbin(10)
SS7	Integrated RMS	The root-mean-square amplitude of the signal after cumulative summation	IRMS

Table 3.3: Statical Functions along with the definitions and abbreviations (Physical features and Orientation Invariant features)

Code	Function	Description/Intuition	abbreviation
Ph1, Ph2	Movement Intensity	Mean and Variance of the Euclidean norm of acceleration vector	MI
Ph3	Normalized Signal Magnitude Area	The acceleration magnitude summed over three axes within each window normalized by the window length	SMA
Ph4	Eigenvalues (Dominant Directions)	The eigenvectors of the covariance matrix of the acceleration data correspond to the dominant directions along which intensive human motion occurs.	
Ph5	Correlation (Gravity and Heading)	It shows how much the movement is aligned towards gravity direction.	CAGH
Ph6	Averaged Velocity (Heading Direction)	The Euclidean norm of the averaged velocities along y and z axes over the window.	AVH
Ph7	Averaged Velocity (Gravity Direction)	averaging the instantaneous velocity along the gravity direction at each time t over the window	AVG
Ph8	Averaged Rotation Angles (Gravity Direction)	The cumulative rotation angles around gravity direction	ARATG
Ph9	Dominant Frequency	The frequency corresponding to the maximum of FFT component magnitudes of the signal	DF
Ph10	Energy	The sum of the squared discrete FFT component magnitudes of the signal from each sensor axis	ENERGY
Ph12	Averaged Acceleration Energy	The mean value of the energy over three acceleration axes	AAE
Ph13	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)

3.2.5 Activity Recognition

In this study, to automatically recognize human activities, we trained four state-of-the-art classifiers on every featureset. The classifiers are used on various HAR studies Baldominos et al. (2019); Morris et al. (2014); Rosati et al. (2018): Support Vector Machine, Decision Tree, K-Nearest Neighbour, and Feed-forward Neural Network. The methods and their parameters setting are described in Table 3.4. We did not optimize the hyper-parameters of classifiers as improving the classification performance is out of the scope of this study; so, we keep the default values preset in their respective R package libraries.

Table 3.4: 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 Nair and Hinton (2010). 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

3.2.6 Evaluation

To evaluate the performance of the model, we need to split data into training and testing sets. In 2018, Jordao et al. (2018) conducted an extensive set of experiments to indicate the vulnerable points in HAR evaluation methods. They showed that the traditional evaluation process (k-Fold) is susceptible to bias which was mainly because of the method of splitting data. Two other alternatives for k-Folds are Leave-One-Subject-Out Jordao et al. (2018), and Leave-One-Trial-Out Cross-Validation Sena, Santos, and Schwartz (2018). As evaluation is part of our side-by-side comparison process, we employ all three validation methods in this study. The process of evaluation for

each method is explained as follows:

k-Fold Cross-Validation

The most popular approach to evaluate the performance of the HAR model is k-Fold Cross-Validation. K refers to the number of folds that the given dataset splits into. We choose k = 10 in this study. Therefore, after the feature extraction phase, we divide the whole dataset into ten approximately equal size subsets. To ensure all activities are present in both the training set and test set with equal class distribution- stratified, we divide each activity's data points separately into folds. It guarantees that the training set always contains all activities. In this work, we used k-Fold Cross-Validation to evaluate the performance of all the models. During each turn, we train and evaluate the model performance on a different fold. The average performance achieved after all ten turns accounts for the final performance of the model. Although k-Fold is very popular in HAR studies, two issues make this evaluation method less effective. First, since the data is shuffled before splitting folds, there is always a chance that the same subject's data appears in multiple folds; thus, the result of k-Fold is not subject-independent. Second, shared-data between the train set and the test set violates the basic assumption-independent folds in k-Fold. It occurs due to generating data points using the sliding window with an overlap between windows. This way, there is always a shared part (the overlap part) of data between every consecutive data-points. Although the dataset splits into the train and test, the shared part inside each data point does not. Two following validation methods address these issues.

Leave-One-Subject-Out Cross-Validation

Leave-One-Subject-Out (LOSO) Cross-Validation split the data per subject. In each turn of evaluation, the data from one subject is used as the test set, while the remaining subjects' data are used for training. The LOSO Cross-Validation technique reflects a more realistic scenario, as the model is evaluated on data from totally new subjects, which is the case in real-life HAR applications. In HAR studies, the performance achieved in this way is called subject-independent Chung, Lim, Noh, Kim, and Jeong (2019); Jordao et al. (2018). It is important to note that using LOSO may present a high variance in recognition performance from one subject to another. It

is because sometimes one activity can be carried out deliberately in different ways by different subjects. Therefore, using this method, we require a bigger dataset to generalize these kinds of variations.

Leave-One-Trial-Out Cross-Validation

In Leave-One-Trial-Out (LOTO) Cross-Validation, trials (reps) are matrices to split the dataset. A trial may contain the data of multiple subjects or only one subject. To use LOTO in HAR, we first indicate each subject's repetition with a label. For example, five repetitions of doing an exercise by a subject are labelled as repetitions 1 to 5. Then, to split the dataset, in each turn, starting from rep 1, we pick all trials with this label (e.g. rep1) for the test set and leave other reps for the train set. The main advantage of using this method compared to LOSO is that it does not necessarily need many subjects. However, in this technique, each subject should have several sessions. Although LOTO does not provide subject-independent evaluation for the model, it ensures there is no shared data between the train set and the test set, which is not the case for the k-Fold Cross-Validation, as mentioned in Section 3.2.6.

Performance Measurements. Most commonly used measures to asses the performance of a HAR model in prior works are: accuracy Brownlee (2018); Mehrang et al. (2017); M. Zhang and Sawchuk (2011) and F-measure (F1) Nourani et al. (2019); Rosati et al. (2018). These measurement units determined as follows:

- **Accuracy** measures how often the classifier is correct. Specifically, it is equal to (TN + TP) / (TP + TN + FP + FN).
- **Precision** measures when the classifier detects an activity, how often it is correct. Specifically, it is equal to TP / (TP + FP).
- **Recall** measures when a user is doing a certain activity, how often the classifier can detect it correctly. Specifically, it is equal to TP / (TP + FN). This term is also known as *Sensitivity* or *True Positive Rate*.
- **F-Score (F1)** 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 the user was doing that activity.
- **True Negative (TN):** Where we predict a non-activity period, and the user was not doing a particular activity.
- **False Positive (FP):** Where we predict a particular activity for a segment of data, however, the user is doing another specific activity or generally doing something else (out of activity given list).
- **False Negative (FN):** Where we predict either a not-activity period or a specific activity, while it is not the activity that the user is doing that.

F-measure relies on both the precision and recall. So, as compared to accuracy, it is less affected by imbalanced dataset (in terms of frequency of different activities). In this study, we used both accuracy and F1 to the performance of models.

3.3 Results

Our study aims to perform a systematic examination of the HAR pipeline (Figure 2.1) 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 the classification phase to find the model with the highest performance (RQ2). Finally, in RQ3, we target the impact of different evaluation methods on our model's performance.

3.3.1 RQ1: Which featureset provides the best performance in HAR?

As motivated earlier, choosing an appropriate featureset significantly impacts the model's 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 F1 metric for each experiment.

Table 3.5 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. 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%. **Providing the highest recognition performance along with the light computational cost of building it Fushing and Roy (2018) makes *histogram bins* an ideal candidate featureset for wearables since they are limited in resources.** On the other hand, *Orientation Independent* features achieved the lowest performance (77.44% on average). Although it reached a relatively lower performance among featuresets, it allows for flexibility in how sensors are placed on a subject.

Table 3.5: F1 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%

3.3.2 RQ2: Which classifier performs better on gym exercise recognition?

As we saw in RQ1, different classifiers perform differently, even on the 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 empirical comparison between the models' performance.

We use four popular classifiers in this experiment, including SVM, KNN, FNN, and DT. It is important to mention that we leave the models on default configuration since the optimizing hyperparameters is not a goal of this study. The default configuration of models is mentioned in the 3.2.5 section. From RQ1, we found *histogram bins* as the most informative featureset. So, in this experiment, we train all models using this featureset. The k-Fold Cross-Validation with ten folds is used for measuring classification results;

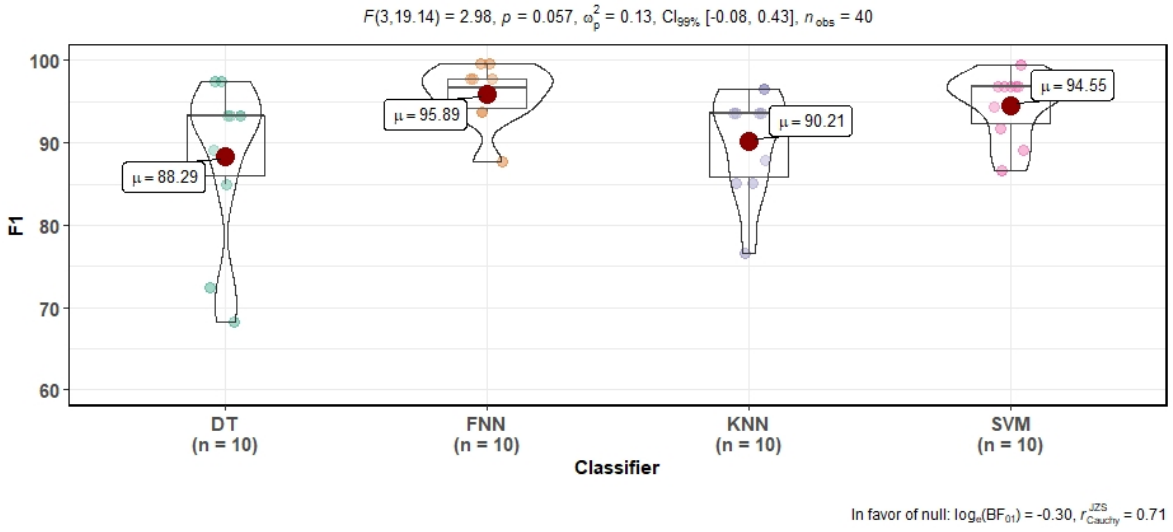


Figure 3.3: Comparison between the performance of classifiers

Figure 3.3 visualizes the distribution of a ten-round performance evaluation for classifiers in a violplot diagram. FNN and SVM show similar distribution and performance range between 85% and 99% of F1 (with the mean 95.89% and 94.55%, respectively) over ten trials. On the other hand, DT ($\mu = 88.29\%$) and KNN ($\mu = 90.21\%$) show more scattered results over trials and a bigger range, respectively 67%-98% and 77%-97% of F1. We use the two-sample Wilcoxon test to examine any significant difference between the model's performance samples.

Table 3.6: Effect size (r) of pairwise model comparisons using Wilcoxon rank sum test

	FNN	**DT**	**KNN**
DT	0.71	-	-
KNN	0.71	0.92	-
SVM	1.0.	0.83	1.0

From the test results (Figure 3.6), a large effect size is detected in all pair model comparisons (effect size r $\geq 71\%$, $p < 0.057$). Therefore, the difference between the classifiers' performance is statistically significant. **That is, FNN and SVM with $95\% \pm 1\%$ deliver the highest performances, whereas KNN and DT with 90.21% and 88.29 of F1, respectively, provide the lowest performances.**

3.3.3 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 Wang et al. (2019). However, in an empirical study, Jordao et al. Jordao et al. (2018) 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 Liu, Gao, John, Staudenmayer, and Freedson (2011) and Leave-One-Trial-Out (LOTO) Cross validation Jordao et al. (2018); Sena et al. (2018).

In k-Fold, splitting the dataset (K) is decided by researcher based on the size of dataset as well as the type of classification problemJordao et al. (2018). Table 3.1 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 subjects and trials. In fact, an activity should appears at least in two session performed by the same subject, to be eligible for LOTO Cross-Validation. We excluded the data of 2 subjects because they did not participate in at least 2 trials for all 8 exercises. In this experiment,

for LOSO we used data from 6 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 3.4 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 difference 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.

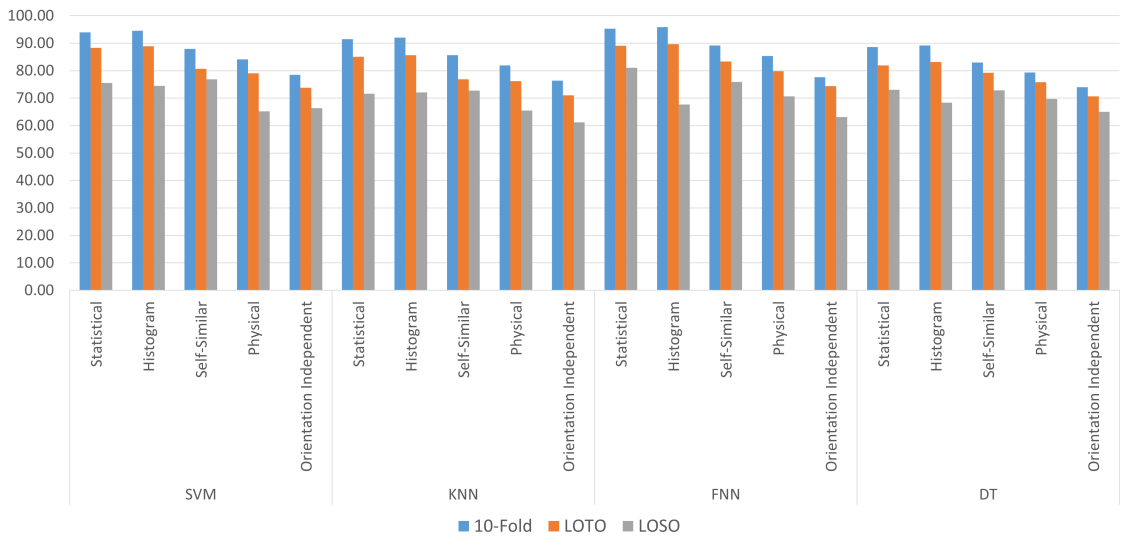


Figure 3.4: Comparison between evaluation methods (10-Fold, LOTO, LOSO)

3.4 Conclusions

Human activity recognition is an important research topic in pattern recognition and pervasive computing. Choosing the right featureset for a HAR model affects the performance significantly. The goal of this chapter was to investigate the state-of-the-art hand-crafted featuresets in HAR.

From our empirical studies, several conclusions can be made. First, by analyzing five feature-sets and using them in the four most popular classifiers in RQ1, we saw that statistical-features and histogram-features are bringing the most informative features for classifiers, with orientation-independent features providing the lowest informative features.

In RQ2, we found that FNN and SVM deliver a superior performance rather than other classifiers apart from which featureset they are using. This confirms the results of the previous studies in classifier comparison Baldominos et al. (2019); Janidarmian et al. (2017). In addition, from the experiment, Decision Tree provides the worst recognition performance from the evaluated methods.

In RQ3, we compared the leave-one-trial-out Cross-Validation with two conventional evaluation methods (k-Fold and LOSO). Results showed that LOTO and LOSO provide a more realistic result than k-Fold as they are subject independent. However, the overall performance of models using these two evaluation methods results in less accuracy than that using k-Fold, at the same dataset size. Besides, using LOTO, we could address an issue left unsolved using LOSO. In LOTO, shared trials of the same exercise between multiple subjects suppress the negative impact of the inconsistent patterns, which has been mentioned in many previous works Jordao et al. (2018); Lee et al. (2020); M. Shoaib et al. (2016).

In this chapter, we have successfully demonstrated that applying histogram bins to FNN with enough data, and using LOTO as the evaluation method can significantly help develop a HAR model for real-life scenarios. Nevertheless, with a limited amount of resources in a miniature size wearable device, an alternative featureset should be small enough to become feasible for a HAR model. Feature selection's main task in the HAR pipeline is set to play this role. Hence, in the next chapter, we perform an empirical study to investigate the feature selection phase, particularly related to the data reduction aspect. We will also design a HAR model built on the ensemble model, which is used to recognize different types of walking activities in our experiments.

Chapter 4

The Impact of Data Reduction on Wearable-Based Human Activity Recognition

4.1 Introduction

Pervasive sensors that can be conveniently worn and ubiquitously used aim at a wide range of potential applications, including various individual's health monitoring, rehabilitation, and intelligent assistance Zhao et al. (2010). These sensors have become increasingly small, more accurate, and more popular (e.g., Smartphones and wearables) Sprager and Juric (2015), leading to extensive research that improves algorithms inferring meaningful knowledge from sensor data. In recent years, many studies in Human Activity Recognition (HAR) using wearables have been carried out that provide promising performance L. Chen, Hoey, Nugent, Cook, and Yu (2012) Banaee et al. (2013); Janidarmian et al. (2017) Shoaib et al. (2014).

The process of HAR, in the well-known form, is already explained in Chapter 1 including 1) *Data collection*, 2) *Segmentation*, 3) *Feature extraction*, 4) *Classification*. First, the data is acquired by motion sensors in the form of data streams. Next, these streams are segmented using the time windows technique (e.g., a window with a length of 5 seconds shifting every 200ms). In the third phase,

the data has already been collected and segmented, so the useful features are extracted using feature extraction schemes described in Chapter 3. At this step, a feature may be considered as *relevant* if the classifier improves performance using it; and, conversely, *redundant* if it does not improve the performance of the classifier Zhao et al. (2010). Finding a set of features containing the Minimum Redundant and Maximum Relevant features (mRMR) has to be done before sending these features to the classifier. Previous studies mostly called this phase as *feature selection* phase.

In the previous chapter, we studied a wide range of features in HAR. Statistical features, self-similarity features, histogram bins are some examples of them. To build a HAR model, a straightforward solution is to extract and give all these features to the classifier, whether they are relevant or not. However, collecting and calculating features comes at a computational cost, particularly in the case of HAR, which is typically done on resource-constrained wearables. Moreover, using more features than needed could lead to many unwanted side effects, including lower model performance, overfitting and higher cost and execution time Bishop (2006); Mujahid et al. (2017); Nguyen et al. (2017). Therefore, we need a procedure to refine and intelligently select the best features.

The goal of this chapter is to examine the impact of feature reduction on wearable-based HAR. Specifically, we aim at examining the trade-off between feature reduction and model performance (RQ1), model generalizability (RQ2) and different classifiers (RQ3). We perform our experiments using step (walking, ascending/descending stairs) data collected using the Neblina system-on-module chip. Generally, our data contains more than 2,000 steps from two different subjects. We extracted a total of 119 different features from the Neblina, which were used to examine the impact of feature reduction on HAR.

Our findings showed that feature reduction could reduce the number of features by close to 90%, while only having an impact of 1-2% in model performance. Feature reduction can impact the performance of the general models (i.e., that are cross-subject); however, which subject a model is trained on does matter. Feature reduction does not have a considerable impact on most examined classifiers.

The rest of the chapter is organized as follows. The state of the art data reduction methods for HAR are presented in Section 4.2. Section 4.3 sets up our case study, providing details about the dataset, feature extraction & selection and classifiers used. Section 4.4 presents our results.

Section 4.5 discusses the relation between features and sensors. Section 4.6 concludes the thesis.

4.2 Related Work

Comprehensive reviews about the subject of feature reduction are available in the literature. In Banaee et al. (2013), the accuracy of 293 classifiers are evaluated using 14 datasets involving accelerometer sensor data. Similarly, the approach in Yong et al. (2013) uses principal component analysis (PCA) to feed the classifiers. Since the dataset contains recording data under different setups, using PCA, they extract those features that are independent from x/y/z axes and consequently independent from sensor orientation. Then, it lets them to treat identically with different datasets. The authors found the ensemble methods of KNN provide the best recognition rate and Decision Tree (DT) provides the worst. They also showed, on average, the best and worst positions for attaching sensor are right thigh and left lower arm, respectively. Similarly, Shoaib et al. (2014) performed an experiment with 10 subjects and 5 sensor positions to show impact of sensor positions on activity recognition. Their results are also confirmed that right pocket (upper thigh) and wrist are respectively the best and worst positions. It is worth mentioning that in these studies the main criteria to evaluate the position of the sensor is the amount of useful information that a sensor in that position provides for the model; however, there is another criteria for choosing sensor position called *usability* that defines how it is easy for a user to wear a sensor. From this point of view, positions such as wrist or thigh are very popular as users can wear sensors (i.e., smartwatch or smartphone in the pocket) Comparing different featuresets, they also concluded that selecting the best sensor (accelerometer vs gyroscope) to achieve best performance depends on body position, activity type, and classifier.

The authors in Erdaş et al. (2016) extracted three feature-sets including time-domain, frequency domain, and wavelet-domain statistics. They employed an ensemble selection on five feature selection methods and showed that the best results are achieved using time domain features. Zhang et al. M. Zhang and Sawchuk (2011) extracted some self-designed features called physical features

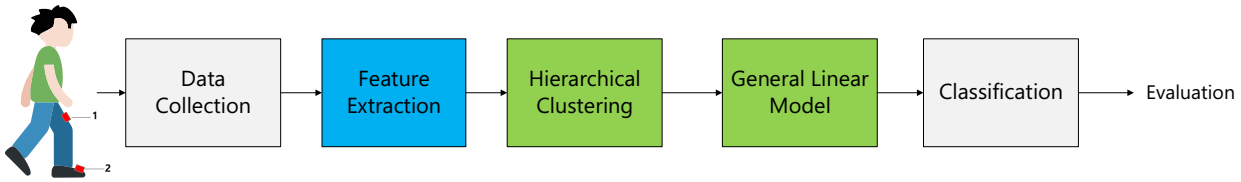


Figure 4.1: Main approach for Human Activity Recognition. Two red squares show sensor positions on the thigh and foot of the subject. The green blocks show the feature selection phase.

and showed that these features have more contributions rather time-domain features to the recognition system. Introducing a multi-layer classifier, they also show that different feature-sets are appropriate for different activities.

More approaches on feature selection methods such as recursive feature selection Nguyen et al. (2017), correlation based feature selection (CFS) Maurer et al. (2006), Independent Component Analysis (ICA) Mantyjarvi et al. (2001), and Local Discriminant Analysis (LDA) Ghasemzadeh et al. (2009), targeting HAR, also exist in the literature.

As mentioned above, existing works mostly employ different feature selection forms to find the best performance of their model. In this work, we investigate feature selection attributes solely and their impacts on the whole model. For the feature selection method introduced in this work, the most relevant previous work was conducted by Ienco et al. Ienco and Meo (2008), who similarly divides the process into two stages and uses a hierarchical clustering followed by a wrapper method. They show that their method (is not for HAR) on various datasets outperforms filter and wrapper methods. Furthermore, using the dendrogram of features provided by hierarchical clustering gives a semantic view of feature space. However, they do not explain how much their method can reduce the size of the feature set and its effects on the models' generality. This work addresses these aspects and has a deeper view of data reduction advantages in the HAR pipeline.

4.3 Study Setup

Our main approach follows a typical HAR classification solution, as shown in Figure 4.1. The approach is composed of five main phases described in the following:

4.3.1 Data Collection

In order to examine the effectiveness of our feature selection method on data reduction, we first need to collect data of certain activities. Specifically, we selected walking in flat steps, ascending up stairs and descending down stairs Kwapisz, Weiss, and Moore (2011) as our target activities. We repeat the experiments over two subjects and two sensor positions.

Sensors. We leveraged Motsai's Neblina system-on-module (SoM) solution. Neblina is a customizable module that is equipped with a tri-axial gyroscope, accelerometer, and magnetometer in conjunction with a 32-bit processor and 2X256KB of flash memory *ProMotion - Motsai Documentation* (2020). The data come from *Neblina* composes of the following features:

- Acceleration data (x/y/z).
- Gyroscope data (x/y/z).
- Magnetometer data (x/y/z).
- Force data, i.e., the acceleration vector minus gravity (x/y/z).
- Euler Angle data (yaw/roll/pitch).

We also calculate cosine for the angle of roll and pitch and call them *roll2*, *pitch2*. One column called *Step type*, which contains the labels corresponding to each step type. The collected data has been recorded on Neblina and is later downloaded to a Windows machine. The sampling rate was set to 50Hz Zhao et al. (2010). Table 4.1 gives the total steps in each trial.

Experiments. We collected data from two male participants ages 25 and 30. Similar to prior studies M. Zhang and Sawchuk (2011), we attached the sensor to the thigh Aminian and Najafi (2004) and foot of each participant and performed the trials at various indoor and outdoor locations without supervision. The sensor was strapped by an elastic belt on the front of the right thigh. Two sensor positions are shown in Figure 4.1.

4.3.2 Feature Extraction and Selection

To divide the stream into segments corresponding to each step, we leveraged a pedometer Jayalath, Abhayasinghe, and Murray (2013). Since the time-domain features have already been shown

Table 4.1: Total number of steps based on subjects, activity types, and sensor positions.

Sensor Position	**Thigh**			**Foot**		
Activity Type	Up	Down	Flat	Up	Down	Flat
Subject A	249	228	420	206	219	386
Subject B	265	250	770	222	242	504

to be quite effective for HAR as opposed to frequency-domain and wavelet features Shoaib et al. (2014), we have chosen the following time-domain features for our analysis, namely mean, median, variance, standard deviation, root mean square, mean absolute deviation and median absolute deviation. Then, we build 119 features built by applying seven feature functions to 17 input signals for each step.

Feature Selection. Many different techniques can be applied to select features. These methods generally are divided into three major categories, including a) filter methods, b) wrapper methods, and c) embedded methods Saeys, Inza, and Larrañaga (2007). In this work, we employ an embedded method that is a heuristic approach orientated toward the notion of minimizing redundancy and maximizing relevancy (mRMR). Explicitly, our method benefits from two inexpensive processing blocks (green blocks in Figure 4.1) to find an optimum set of features. Firstly, it filters highly correlated features out. Next, it ranks them, using General Linear Model (GLM), based on how much features are statistically significant and takes the top ones. These blocks are explained as follow:

Hierarchical Clustering block. This block is set to find features that have the minimum redundancy between them. More redundant features in a featureset not only do they increase the processing cost, but they also decrease the performance due to the coarse-of-dimensionality for the smaller size of the dataset. Therefore, this block aims to discover a set of low correlated features. To achieve that, we use the hierarchical clustering (HC) Murtagh and Contreras (2017) method, and measure the Spearman correlation coefficient through all features. The outcome of hierarchical clustering is a hierarchy of features in the form of a so-called dendrogram. It constructs the dendrogram by dividing features based on the correlation between features, which results in nested groups of features. Based on the level of correlation required, one can choose where to define the cut-off line. Choosing a line closer to the top makes fewer groups at a lower level of correlation and vice versa. In this work, as in Park (2013), we put the cut-off line on 0.7, and it returns 15 clusters. It means

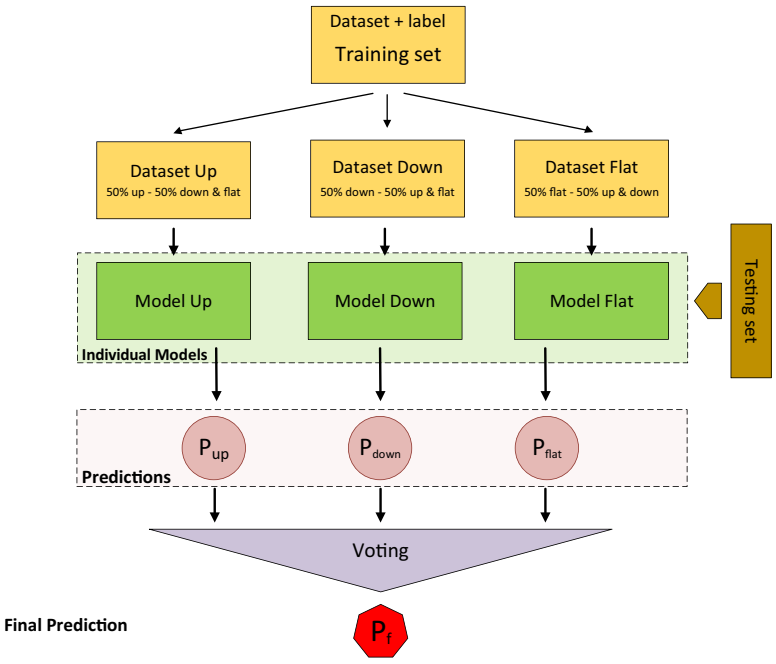


Figure 4.2: Ensemble Strategy. Training m models to detect M activities. In voting block, the best prediction gets selected.

that the correlation among clusters is between -0.3 and +0.3. In the second step, we choose the representative feature from each cluster. To aim this, we employ *Goodman and Kruskal* Davis (1967) algorithm on features of each cluster. Goodman and Kruskal is a measure that explains how much a feature can predict the other. The most accurate predictor is considered as the representative of that cluster. Then, we take these representatives to make the final featureset. Using this method, we end up with 91 features (out of 118).

General Linear Model block The main goal of this block is to measure the features in terms of their contribution in predicting response. To that aim, we train a linear model feeding features received from the prior block. Using p-value (< 0.05), we take the statistically significant features in the trained model. Taking features with a p-value less than the significance level, we will have more certain candidates to be fed with the classifier at the following phase.

4.3.3 Classification Model

In this section, we explain the structure of our classifier. More classifiers are also taken into consideration in RQ3. Janidarmian et al. (2017) and Oza and Tumer (2008) applied an ensemble

of classifiers on HAR and showed that it outperforms other conventional classifiers in dealing with more difficult problems.

Following the state-of-the-art, we used an ensemble of GLMs in this work. Intuitively, instead of training each classifier for all classes, we assign each class (step type) to one classifier - One-vs-Rest strategy. Therefore, to predict 3 step types, we train three models (individual models in Figure 4.2). Then, using a voting classifier, which is an ensemble of classifiers method, we combine their results into one final decision. In this work, we choose the class with the highest score through the voting block. For example, for certain input data, if individual models predict as following: $P_{up} = 0.8, P_{down} = 0.5, P_{flat} = 0.2$, the voting block infers *Up* as the final prediction. During training, each individual model has been provided with an exclusive dataset biased toward a certain step type.

4.3.4 Performance Evaluation

Adopting a 10-folds cross-validation strategy, we divide our data into ten folds; Nine folds to train the model and one left to test it, on each round. One step is labelled as *unknown* if the results of at least two individual-model are equal (e.g., $P_{up} = P_{down}$). Since our test set contains no unknown labelled data, any unknown step prediction is considered as false negative (FN). As there is no non-step label in the actual observation, the true negative (TN) rate is always zero. The true positive (TP) rate is composed of all correct step type predictions. In contrast, the false positive (FP) rate is all incorrect step type predictions. Using the predicted value for each step, we can calculate precision ($\frac{TP}{Tp+FP}$ and recall ($\frac{TP}{TP+FN}$). In this study, we use Accuracy, F-measure, and Mis-Classification to evaluate the performance of models.

Accuracy: Measures the rate of correctly classified step and non-steps types over all steps. It is calculated as $\frac{TP+TN}{TP+FP+TN+FN}$

F-measure (F1): Presents the harmonic mean between precision and recall. It is calculated as $2 * \frac{precision*recall}{precision+recall}$

Mis-Classification (MC): Measures the rate of incorrectly classified steps and non-steps over all of steps.

4.4 Case Study Results

In this section we present the results of our experiments that answer our research questions.

4.4.1 RQ1- How much does our feature reduction impact performance?

As motivated earlier, most wearable devices that are used for HAR are resource-constrained. Hence, reducing the number of features (and consequently, data to be collected) improves the power consumption, latency, and memory usage for these wearable devices. Simultaneously, the general belief is that reducing the number of features fed to a classifier also negatively impacts accuracy Erdaş et al. (2016). Therefore, the focus of this question is - what is the tradeoff between the amount of data we can reduce and the performance impact.

Similar to prior work, we use accuracy, F1-measure, and the misclassification rate Deng (1998) to measure the impact of performance and use the number of features used in the model to measure the data savings. To compare the two setups, we conduct one experiment using all of the features available to us and then repeat the same experiment using our reduced set of features. The differences in performance and data savings between the two experiments are then reported.

Table 4.2 shows the results of our experiments for the two subjects, A and B (results of the full model are shown in parenthesis). In each Table, the first line is the ensemble model's result (considering all steps) followed by results of individual models, i.e., step-up, step-down and walking on the flat surface. From the Table 4.2, **we see that the number of features is reduced by 92% $\pm$1% (from 119 to 8 features), while the classification accuracy is decreased by 1 - 2% for both subjects**. Alternatively, the flat walking model works better after the feature reduction for both subjects. The step-up model has the lowest accuracy (97%, which is still quite high) among the models examined. Comparing the results of two subjects at the same level of performance, the total number of features for subject A is 30% lower than subject B (8 vs. 12 features), which indicates that the reduction may be subject-specific. Either way, for both subjects, though, the reduction in features is significant.

Table 4.2: Impact of data reduction on performance of model. The numbers in parenthesis are results of base-line model (using 119 features). The model "All Steps" means that it can classify all three step types. in the case of all-steps model, while the number of features decreases from 119 to 8, the accuracy changes only 1% (from 99% to 98%)

(Subject A)

Model	**No. Features**	**Accuracy**	**F1**	**MC**
All Steps	(119) **8**	(0.99) **0.98**	(1.00) **0.99**	(0.01) **0.02**
Step Up	(119) 8	(0.97) 0.95	(0.95) 0.93	(0.03) 0.03
Step Down	(119)**7**	(0.99) 0.99	(0.99) 0.99	(0.0) 0.01
Flat walking	(119) 9	(1.00) 1.00	(1.0) 1.00	(0.0) 0.01

(Subject B)

Model	**No. Features**	**Accuracy**	**F1**	**MC**
All Steps	(119) **12**	(0.99) **0.98**	(1.00) **0.99**	(0.01) **0.02**
Step Up	(119) **11**	(0.97) 0.99	(0.95) 0.99	(0.03) 0.03
Step Down	(119) 13	(0.99) 0.99	(0.99) 0.99	(0.00) 0.01
Flat walking	(119) 12	(1.00) 1.00	(1.00) 0.99	(0.00) 0.01

4.4.2 RQ2- How does the feature reduction impact the generalizability of the model?

As we have seen from the results of RQ1, different individuals do not perform the same HAR activity in the same way Janidarmian et al. (2017). The pattern of doing the activity depends on many factors, including the physical body of the subject, his/her level of fatigue, experiences, and so on. Consequently, a model trained on one subject may not be applicable to another subject Morris et al. (2014); Shoaib et al. (2014). In our case, we are interested in examining the impact of the feature reduction on the generality of the model.

Cross-subject validation uses the data from one subject to train the model, then tests the model on data from another (independent subject). In this thesis, as our target is to examine the impacts of feature reduction, hence, similar to the case of RQ1, we repeat the cross-subject validation twice, once with all features that are available to us and once with the reduced set of features.

Table 4.3 shows results for both experiments (results of the full model are shown in parenthesis). In the top Table, we train on data from subject A and test on subject B's data and vice versa for the Table on the bottom. First, we see that the feature reduction does decrease performance, however, its

performance is comparable. Second, we notice that although the model trained on subject B's data has 3 more features (less data reduction) than the model trained using subject A's data (15 against 12), it does not provide a higher accuracy (81% vs. 80%). **Overall, we conclude that although feature reduction does impact the overall performance when evaluated across subjects, the impact is not significant. That said, again, which subject you train and test on does impact the results.**

Table 4.3: Cross-subject validation results on two subjects. A vs B means testing model of subject A on data of subject B.

Subject A v.s. Subject B

Model	No. Features	Accuracy	F1	MC
All Steps	(119)**12**	(0.93)**0.80**	(0.97)**0.89**	(0.07)**0.20**
Step Up	(119)15	(0.93)0.86	(0.90)0.81	(0.03)0.09
Step Down	(119)**10**	(0.97)0.87	(0.96)0.82	(0.03)0.17
Flat walking	(119)12	(0.97)0.86	(0.96)0.75	(0.15)0.35

Subject B v.s. Subject A

Model	No. Features	Accuracy	F1	MC
All Steps	(119)**15**	(0.96)**0.81**	(0.98)**0.89**	(0.04)**0.19**
Step Up	(119)13	(0.95)0.86	(0.92)0.79	(0.05)0.24
Step Down	(119)**12**	(0.97)0.88	(0.96)0.83	(0.02)0.08
Flat walking	(119)20	(0.99)0.84	(0.98)0.73	(0.05)0.26

4.4.3 RQ3- How does feature reduction impact different classifiers?

In most related work, the authors evaluate their feature selection method using different classifiers to identify the best model. However, different classifiers are affected by feature reduction differently. Prior work examined various different classifiers showed that they deal with feature dimensionality differently Nabian (2017). However, their setting was slightly different since they used PCA, which may reduce dimensionality, however it is not guaranteed to reduce the number of needed features since one PC may be a combination of many features.

Therefore, in this RQ, we investigate the impact of feature reduction on 6 of the most common classifiers used in HAR. Again, we build a model using all of the features available to use

Table 4.4: Impact of data reduction on six classifiers including SVM, GLM, NN, KNN, Random Forest, and Boosted Tree. The result of base-line model is written in parenthesise behind the number.

Model	N. features	Accuracy	F1	MC
GLM	(119)12	(0.99)**0.98**	(1.00)0.99	(0.01)0.02
SVM	(119)12	(0.98)0.97	(0.99)0.98	(0.02)0.03
NN	(119)12	(0.99)**0.98**	(0.99)0.99	(0.01)0.02
KNN	(119)12	(0.98)0.96	(0.99)0.98	(0.02)0.04
Random Forest	(119)12	(0.99)**0.99**	(1.00)**0.99**	(0.01)**0.01**
Boosted Tree	(119)12	(0.98)0.96	(0.99)0.98	(0.02)0.04

and compare that with a model built using the reduced feature set. We merge the data from both, subject A and B to perform this analysis. We mostly used the default parameter settings for the various models, except for the Neural Network model, in which we used a configuration that was recommended in earlier work O'Shea, Corgan, and Clancy (2016). The NN model used a 5-layer network utilizing two drop-out layers and three dense fully connected layers. Layers use rectified linear (ReLU) activation functions except for a Softmax activation on the one-hot output layer.

Table 4.4 shows the results of our experiment. As we can see from the Table, the models perform very well, with and without feature reduction. In terms of F1-measure, GLM, NN and RF slightly outperform the SVM, KNN and BT models. That said, all models do not seem to be impacted much by the feature reduction. In general, the Random Forest model seems to perform the best overall, and for that model, the feature reduction only impacts the F1-measure by 1%. **Overall, we see that most models are quite robust to the feature reduction.**

4.5 Discussion and Future Work

Limitations. One of our contributions is introducing a feature selection method that showed a significant result in reducing data size. However, this method may not provide the best results as we did not compare its result with any other conventional feature selection methods. For this reason, more validation of feature selection method is important future work. In addition, as showed in RQ2, the result might be affected by certain subject. A wider range of activities beside more number of

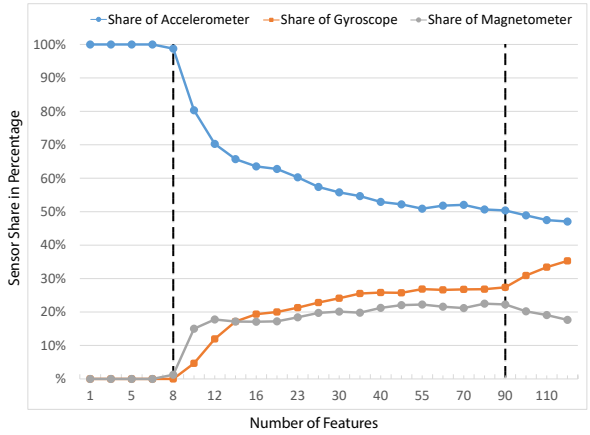


Figure 4.3: contribution of three sensors over different sizes of featureset

subjects are required to decrease the impact of subjects. So, using big enough datasets like Anguita et al. (2013) will be considered in our future works.

Features vs. Sensors. As we have shown, feature reduction is a viable way to help save the resources of wearables used in HAR. However, there is a key distinction between the features and sensors used to derive these features. Although reducing the number of features helps save computation resources, a real gain can be obtained if we could reduce the number of active sensors in a wearable. This is possible if features extracted from a sensor are completely omitted in the data reduction phase. Therefore, we run an experiment to determine which sensors provided the most contributing features. The experiment was performed on data from both subjects and included all steps in our dataset.

Figure 4.3 shows the share of each sensor vs. the total number of features for each of the three sensors on the Neblina, namely accelerometer, gyroscope and magnetometer. We observe from the figure that the accelerometer contributes the highest percentage of features, generally making up close to 50% of the features at any given point. On the other hand, the gyroscope and magnetometer, have similar contributions, which does not exceed 40%.

These results indicate that for HAR, we have the potential to not only reduce features, but perhaps do some sensor optimizations to maximize savings of wearable devices. Such optimizations are beyond the scope of this thesis, however, we plan to develop such methods and examine the effectiveness in the future.

4.6 Conclusion

In this thesis, we have investigated the impact of feature reduction on the performance of HAR. We collected step data using the Neblina system-on-module solution from two subjects and have answered three research questions related to the impact of feature reduction in terms of performance, generalizability and varying classifiers. Our findings indicate that feature reduction can have a significant reduction in using resources while achieving comparable results to a full model. Our main findings are:

- Feature reduction can reduce the number of features by close to 90%, while only having an impact of 1-2% in model performance.
- Feature reduction can impact the performance of the general models (i.e., that are cross-subject), however, which subject a model is trained on does matter.
- Feature reduction does not have a major impact on most classifiers examined.

Our analysis also have showed that the accelerometer contributes most of the features used in HAR models. In the future, we will be introducing methods that can optimize sensor operation in order to maximize the resource savings of wearables.

Chapter 5

Summary, Contribution and Future Work

This chapter concludes the thesis including a summary of results presented throughout this thesis, as well as some discussions regarding possible directions for future work.

5.1 Summary

This thesis focuses on the challenges of HAR using wearables. First, we conducted an end-to-end activity recognition pipeline to understanding the main issues in each phase of developing HAR systems. Then, we investigated different alternative featuresets, classification models and evaluation methods under an equal study set up to measure each phase's impact on the final model performance. It reveals the advantage of alternatives in each pipeline phase that have mostly been downplayed in the previous studies, especially in feature selection and feature extraction. Therefore, we evaluate the impacts of these two phases under a resource constraint condition, as it is part of the challenge in developing HAR systems in real-life. We propose a technique to reduce the input data size and show how the HAR model performs consistently under these highly limited resources conditions.

The following is a summary of the thesis chapters.

Chapter 3 presents a detailed experience of a human activity recognition system while it is

focused on a side-by-side comparison between different featuresets and classification method alternatives. In this chapter, we study the state-of-the-art featuresets and the most popular classifiers in HAR. We conduct a data collection and labelling procedure to collect 71 gym exercises carried out by 25 subjects and prepare them for quantitative analysis on HAR models. We extract 1300 hand-crafted features and design 20 classification models (combinations of five featuresets and four classifications) and evaluate them using three evaluation approaches (k-fold, cross-trial, and cross-subject). We found that: 1) Among featuresets, models using histogram bins or statistical features are providing the highest recognition performance by far, as compared to orientation-independent features and physical-features; 2) Models using FNN and SVM classifiers give the most accurate activity recognition among others, with Decision Tree (DT) recognizing the lowest accuracy level. 3) evaluating HAR models using K-fold cross-validation always yields a higher model performance than other evaluation methods. That being said, this result is always subject-dependent. On the other hand, Leave-One-Trial-Out and Leave-One-Subject-Out cross-validation methods show the model performance relatively lower, respectively. However, these methods leverage subject-independent validation, which is advantageous as it is the case for most real-life applications. We also contrast the performance of the most precise model (FNN using histogram-bins) per each activity and find out that the same family exercises (Crunch twist and Russian Twist) are the most recognition challenging for the model. In addition, we compared the convergence rate of FNN over the first 100 epochs for each featureset and show how fast the model with Histogram features can reach the maximum accuracy (at the 5th epoch.)

Chapter 4 presents the data reduction approaches in the feature selection phase in HAR processes. In this chapter, we study a tradeoff between the model's accuracy and the size of input data in the context of HAR. Performing under limited processing resources is naturally a HAR model's challenge in wearables. We empirically demonstrate the prevalence of data size and the model's accuracy rate for a HAR system. We record the data of three types of walking (up-stair, down-stair, flat walking), present a feature selection approach to reduce the data size and an ensemble model to recognize those activities. Our findings show that: 1) Only 7% (8 features) of 119 extracted features are enough for the model to deliver 99% of its original performance (while using all features). This rate varies between by maximum of 1%. 2) The data reduction insignificantly reduces the model's

generality while highly determined by the subject's data used in the training dataset. 3) The data reduction makes a relatively equal impact on different classifiers. This range starts with less than 1% decrease in the Random Forest, NN, and SVM; it ends to less than 2% for Boosted Tree, GLM, and KNN. Furthermore, we show that the more we reduce the dataset size, the more contributions the accelerometer features make to the featureset. That is, the accelerometer supplies the essential features for HAR models than two other sensors.

5.2 Contributions

The major contributions of this thesis are as follows:

- A side-by-side comparison of how state-of-the-art features affect HAR system performance.
- A detailed investigation of the impact of data reduction on HAR system performance and generality.
- A large dataset of gym exercise activities is recorded under real-life conditions and publicly available for future researches on HAR and fitness tracking analysis.
- A publicly available repository of scripts contains approaches to extract 1300 most frequently used HAR features to help the research community acquire a broader range of informative sensor data characteristics.

5.3 Future Work

We believe that our thesis makes a positive contribution towards understanding the challenges of designing a traditional HAR model. However, there are still many open challenges that need to be tackled to improve performance. We now highlight some avenues for future work.

5.3.1 Considering other factors related to feature extraction

Throughout our study, we were mainly focused on understanding of a movement and relevant feature functions, i.e., physical featureset and histogram featureset, as the key factors to extract

features. We know that other factors are also involved in designing an informative featureset. For example, a model's performance changes over a different length of overlapping (sliding window). In the future, we intend to perform more in-depth studies regarding factors like overlap, noise reduction, and sampling frequency to explore other ways of measuring the performance of a featureset in the feature extraction phase.

5.3.2 Providing real-time (online) HAR systems

In our investigation in this thesis, we perform the training and classification jobs offline. In the sense that the recognition process starts after the data is all recorded. In real-life applications, these systems called post-workout feedback systems (offline HAR.) However, there is another category called real-time (online HAR) in which the recognition occurs as the data is being recorded. Models in this category have to simultaneously deal with multiple tasks, including recording and segmenting data, extracting features, and recognizing the activity, with the minimum latency to provide a real-time experience for the user. Besides the accuracy challenge and limited storage and processing power, online HAR systems should address orientation variation, false peak detection, and latency tradeoffs. Toward this end, conducting a study that measures these aspects of the feature extraction phase can positively impact online HAR development.

5.3.3 Extending data reduction in sensor fusion phase

In chapter 4, we showed that the contribution of different types of sensors varies as we decrease the size of data (the number of features.) In our ensemble model, we noticed that this contribution is different per individual model/activity. Although omitting features decreases the processing cost, the sensors are still running and using power even if their data is not going to be used by the model. Therefore, future work will investigate the benefits of dynamic-sensor-selection to utilize energy efficiently while achieving the desired activity recognition accuracy.

5.3.4 Extending to our gym exercise dataset

One of the practical contributions of the current thesis is sharing the large dataset of gym exercises that we collected for our experiments. Although we collected a large dataset, we could not

use a significant portion of it in the interest of applying LOTO and LOSO in the evaluation phase. Therefore, we did not have enough samples per-subject and per-trial for each activity to study more advanced HAR qualifications, such as the impact of the subject's expertise level or fatigue, for that matter. So, one absolute future work could focus on collecting more data labelled appropriately and exploring these features. Meanwhile, collecting a bigger dataset will also provide more generalized results for future experiments.

Appendix A

Gym Exercise Dataset

This appendix's key element involves visually analyzing our gym dataset to glean valuable insights and understand underlying relationships and patterns that were essential during our analysis.

Figure A.1 shows the activity distribution of our dataset. The more popular exercises are at larger values (i.e., (1) Treadmill, (2) Dumbbell Bench Press, (3) Lat Pull Down.)

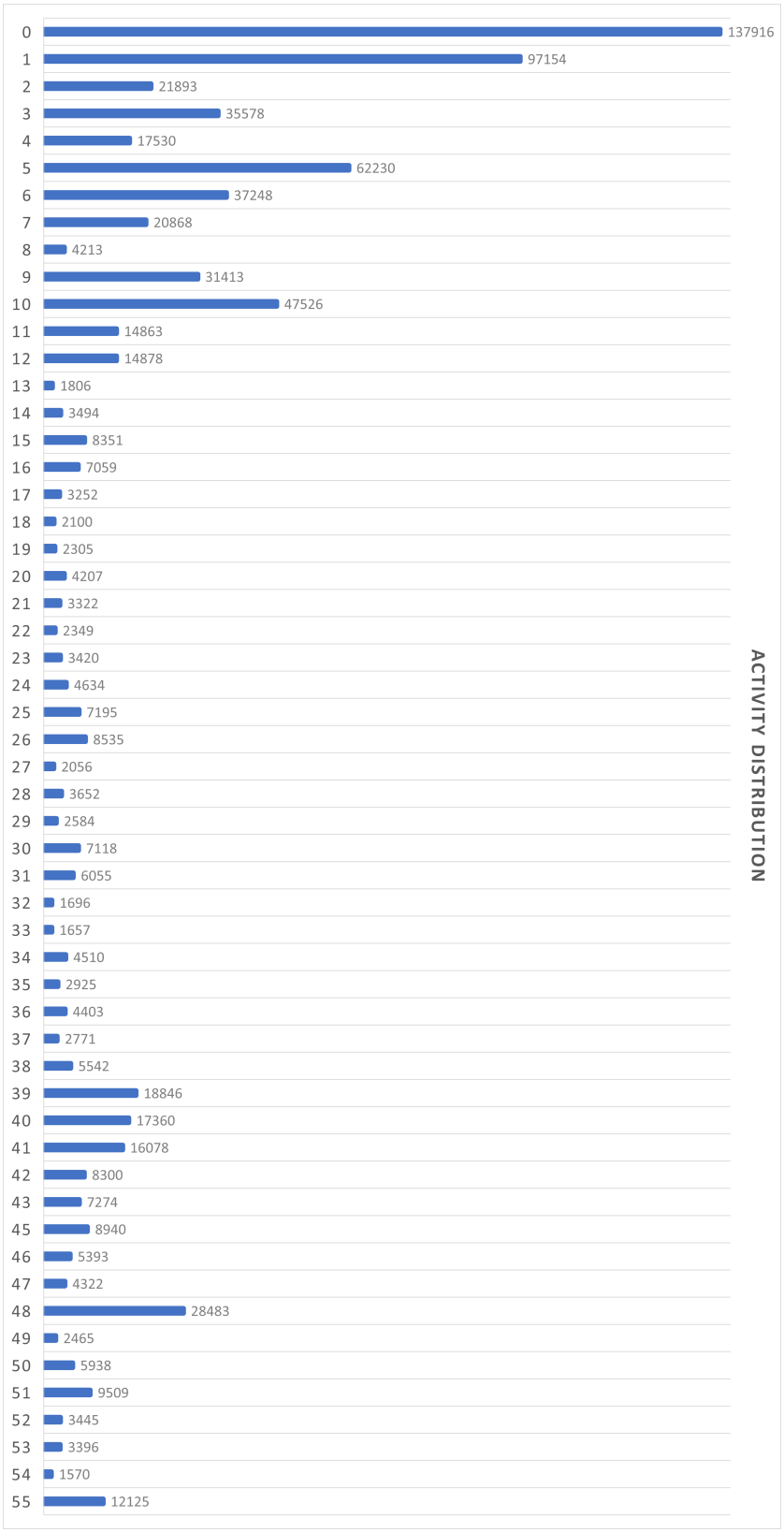


Figure A.1: The exercise distribution (total samples) in gym dataset. The exercise 0 is null activity (including any non-exercise activity a person normally does in the gym such as walking, drinking water, talking.)

There are three evaluation methods used in this thesis: 1) Leave-One-Trial-Out (LOTO), 2) Leave-One-Subject-Out (LOSO), 3) K-Fold Cross-Validation. Whereas K-Fold Cross-Validation does not need any specific consideration to be respected regarding the distribution of activities and subjects, two other evaluation methods highly enforce the dataset's measures. Regarding the LOSO, as it evaluates the model's recognition performance across different subjects, it essentially needs each exercise to be carried out by more than one subject. Figure A.2 gives the number of participants for each exercise.

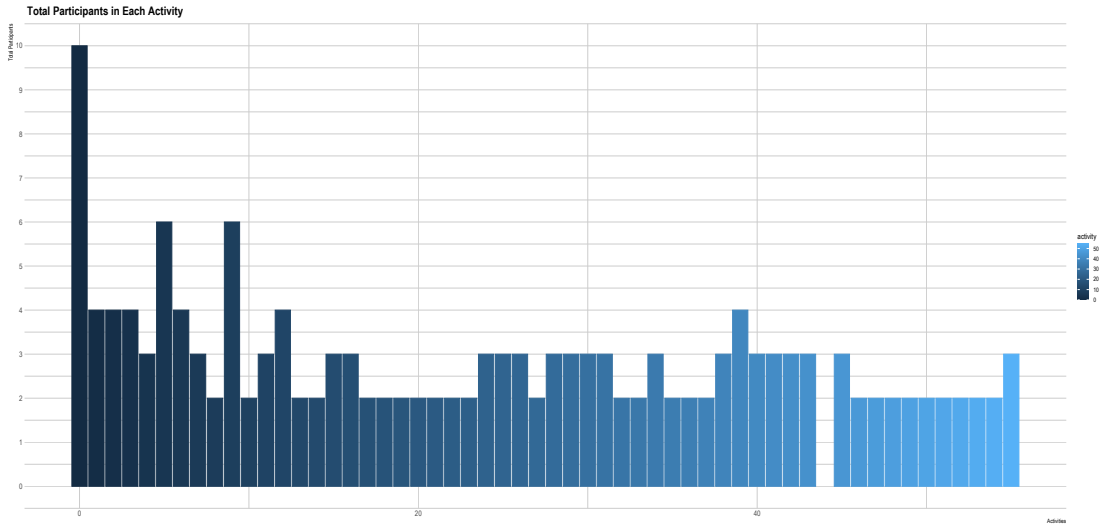


Figure A.2: Total subjects participating in each exercise.

In the LOTO Cross-Validation method, the minimum trial of each exercise matters. Figure **??** shows the number of trials for each exercises.

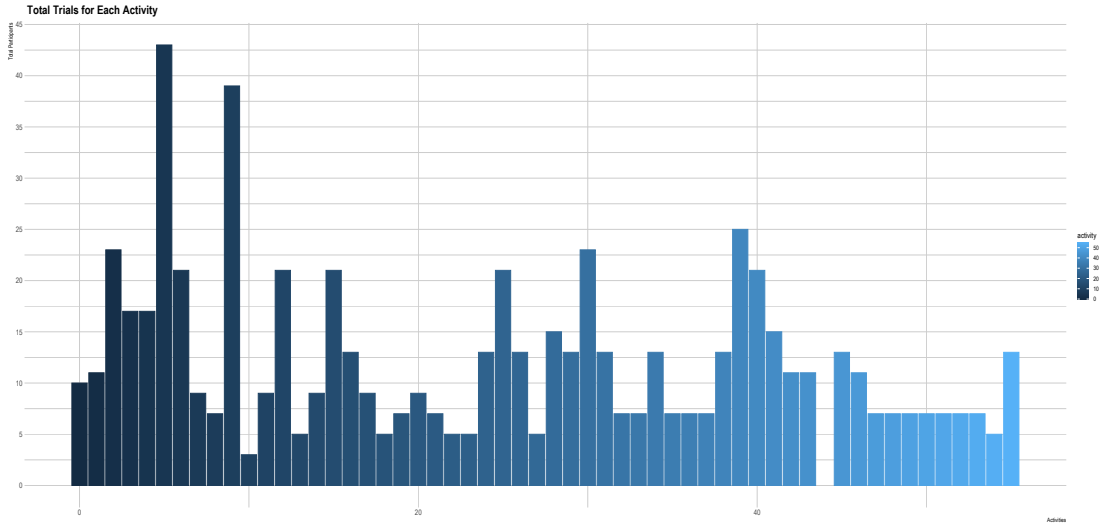


Figure A.3: Total trials carried out for each exercise. There is data from 45 exercises recorded in gym dataset

Part of our experiments was to compare the reported models' performance using these three evaluation methods. Aiming to have a fair comparison, we need identical data for all three evaluation strategies. Therefore, we need exercises that can satisfy the constraints: 1) the minimum-trials and, 2) the minimum-subjects. Figure A.4 shows total trials of each subject for every 45 exercises. From this Figure, we chose those exercises with at least three subjects who performed at least three trials. Figure **??** shows total samples recorded instead of the number of trials for each subject.

Figure A.4: Total trials per subject per exercise. There are 9 subjects and 45 exercises recorded in gym dataset

A.1 Activities

The following is a list of all considered activities in the current study.

- 1 Treadmill
- 2 ab crunch machine
- 3 Lying leg curl
- 4 Triceps Pushdown Rope
- 5 Dumbbell Bench Press Incline Dumbbell Press
- 6 barbbell bicep curl
- 7 standing calf raise
- 8 Crunch
- 9 Lat pull down
- 10 cycling
- 11 seated calf raise Calf Press Leg Press
- 12 overhead dumbbell press
- 13 Machine Shoulder (military) Press
- 14 overhead barbell press - behind the neck
- 15 Dumbbell Lateral Raise
- 16 Dumbbell Front Raise
- 17 Dumbbell Reverse Fly On Incline Bench
- 18 Barbell Upright Row
- 19 Austrailian Pull up (inverted row)

- 20 Standing Biceps Cable Curl
- 21 Lying Barbell Curl On Incline Bench
- 22 concentration dumbbell curl
- 23 Hammer Curl
- 24 Behind The Neck Lat Pull down
- 25 Seated Cable Row
- 26 pullovers machine
- 27 Horizontal bar
- 28 H Machine Row
- 29 T Machine Row - Seated machine row
- 30 Triceps Overhead Ext
- 31 Lying Close-Grip Barbell Triceps Press To Chin - Lying Triceps Press
- 32 Parallel Bar Dip
- 33 Bench Dips (Triceps Dips) (dumbbell kickback)
- 34 Lying dumbbell triceps
- 35 Incline Dumbbell Bench
- 37 Barbell Decline Bench Press
- 36 Low Cable Cross over
- 38 Cable High Cross Over
- 39 Push-up
- 40 Reverse Crunch - Flat Bench Lying Leg Raise
- 41 Russian Twist
- 42 Cable One Arm Lateral-L
- 43 Cable One Arm Lateral-R
- 44 Standing Biceps Curl
- 45 Shrug dumbbell
- 46 Pectoral Fly
- 47 band chest pulls
- 48 ab machine bend
- 49 barbell plate press
- 50 side bent pulls (kettlebell/ dumbblell)
- 51 chin-ups
- 52 Cable Crunch
- 53 Knee Hip Raise On Parallel Bars
- 54 leg extension
- 55 dumbbell fly (bench)
- 56 barbell bent-over row

References

Aminian, K., & Najafi, B. (2004). Capturing human motion using body-fixed sensors: outdoor measurement and clinical applications. *Computer Animation and Virtual Worlds*, *15*(2), 79–94. Retrieved from http://dx.doi.org/10.1002/cav.2 doi: 10.1002/cav.2

Anguita, D., Ghio, A., Oneto, L., Parra, X., & Reyes-Ortiz, J. L. (2013). A public domain dataset for human activity recognition using smartphones. In *Esann.*

Attal, F., Mohammed, S., Dedabrishvili, M., Chamroukhi, F., Oukhellou, L., & Amirat, Y. (2015). Physical human activity recognition using wearable sensors. *Sensors*, *15*(12), 31314–31338.

Baldominos, A., Cervantes, A., Saez, Y., & Isasi, P. (2019). A comparison of machine learning and deep learning techniques for activity recognition using mobile devices. *Sensors*, *19*(3), 521.

Banaee, H., Ahmed, M. U., & Loutfi, A. (2013). Data mining for wearable sensors in health monitoring systems: A review of recent trends and challenges. *Sensors*, *13*(12), 17472–17500. Retrieved from http://www.mdpi.com/1424-8220/13/12/17472 doi: 10.3390/s131217472

Bishop, C. M. (2006). Pattern recognition and machine learning (information science and statistics) springer-verlag new york. *Inc. Secaucus, NJ, USA*.

Brownlee, J. (2018). A gentle introduction to k-fold cross-validation. *Accessed October*, *7*, 2018.

Chen, L., Hoey, J., Nugent, C. D., Cook, D. J., & Yu, Z. (2012). Sensor-based activity recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *42*(6), 790–808.

Chen, Z., Zhang, L., Cao, Z., & Guo, J. (2018). Distilling the knowledge from handcrafted features

for human activity recognition. *IEEE Transactions on Industrial Informatics*, *14*(10), 4334–4342.

Chung, S., Lim, J., Noh, K. J., Kim, G., & Jeong, H. (2019). Sensor data acquisition and multimodal sensor fusion for human activity recognition using deep learning. *Sensors*, *19*(7), 1716.

Davis, J. A. (1967). A partial coefficient for goodman and kruskal's gamma. *Journal of the American Statistical Association*, *62*(317), 189–193.

de Faria, I. L., & Vieira, V. (2018). A comparative study on fitness activity recognition. In *Proceedings of the 24th brazilian symposium on multimedia and the web* (pp. 327–330).

De Leonardis, G., Rosati, S., Balestra, G., Agostini, V., Panero, E., Gastaldi, L., & Knaflitz, M. (2018). Human activity recognition by wearable sensors: Comparison of different classifiers for real-time applications. In *2018 ieee international symposium on medical measurements and applications (memea)* (pp. 1–6).

Deng, K. (1998). *Omega: On-line memory-based general purpose system classifier* (Unpublished doctoral dissertation). Carnegie Mellon University.

Dietterich, T. G., et al. (2002). Ensemble learning. *The handbook of brain theory and neural networks*, *2*, 110–125.

Dix, A., Dix, A. J., Finlay, J., Abowd, G. D., & Beale, R. (2003). *Human-computer interaction*. Pearson Education.

Duda, R. O., Hart, P. E., & Stork, D. G. (2012). *Pattern classification*. John Wiley & Sons.

Erdaş, Ç. B., Atasoy, I., Açıcı, K., & Oğul, H. (2016). Integrating features for accelerometer-based activity recognition. *Procedia Computer Science*, *98*, 522–527.

Friedl, M. A., & Brodley, C. E. (1997). Decision tree classification of land cover from remotely sensed data. *Remote sensing of environment*, *61*(3), 399–409.

Fushing, H., & Roy, T. (2018). Complexity of possibly gapped histogram and analysis of histogram. *Royal Society open science*, *5*(2), 171026.

Ghasemzadeh, H., Loseu, V., Guenterberg, E., & Jafari, R. (2009). Sport training using body sensor networks: A statistical approach to measure wrist rotation for golf swing. In *Proceedings of the fourth international conference on body area networks* (p. 2).

González, S., Sedano, J., Villar, J. R., Corchado, E., Herrero, Á., & Baruque, B. (2015). Features

and models for human activity recognition. *Neurocomputing*, *167*, 52–60.

Ienco, D., & Meo, R. (2008). Exploration and reduction of the feature space by hierarchical clustering. In *Proceedings of the 2008 siam international conference on data mining* (pp. 577–587).

Janidarmian, M., Roshan Fekr, A., Radecka, K., & Zilic, Z. (2017). A comprehensive analysis on wearable acceleration sensors in human activity recognition. *Sensors*, *17*(3), 529.

Jayalath, S., Abhayasinghe, N., & Murray, I. (2013). A gyroscope based accurate pedometer algorithm. In *International conference on indoor positioning and indoor navigation* (Vol. 28, p. 31st).

Jordao, A., Nazare Jr, A. C., Sena, J., & Schwartz, W. R. (2018). Human activity recognition based on wearable sensor data: A standardization of the state-of-the-art. *arXiv preprint arXiv:1806.05226*.

Khokhlov, I., Reznik, L., Cappos, J., & Bhaskar, R. (2018). Design of activity recognition systems with wearable sensors. In *2018 ieee sensors applications symposium (sas)* (pp. 1–6).

Kose, M., Incel, O. D., & Ersoy, C. (2012). Online human activity recognition on smart phones. In *Workshop on mobile sensing: From smartphones and wearables to big data* (Vol. 16, pp. 11–15).

Kwapisz, J. R., Weiss, G. M., & Moore, S. A. (2011). Activity recognition using cell phone accelerometers. *ACM SigKDD Explorations Newsletter*, *12*(2), 74–82.

Lee, M., Youm, C., Noh, B., & Park, H. (2020). Gait characteristics based on shoe-type inertial measurement units in healthy young adults during treadmill walking. *Sensors*, *20*(7), 2095.

Liu, S., Gao, R. X., John, D., Staudenmayer, J. W., & Freedson, P. S. (2011). Multisensor data fusion for physical activity assessment. *IEEE Transactions on Biomedical Engineering*, *59*(3), 687–696.

Mantyjarvi, J., Himberg, J., & Seppanen, T. (2001). Recognizing human motion with multiple acceleration sensors. In *Systems, man, and cybernetics, 2001 ieee international conference on* (Vol. 2, pp. 747–752).

Maurer, U., Smailagic, A., Siewiorek, D. P., & Deisher, M. (2006). Activity recognition and monitoring using multiple sensors on different body positions. In *Wearable and implantable*

body sensor networks, 2006. bsn 2006. international workshop on (pp. 4–pp).

Mazo, J. E. (1975). Faster-than-nyquist signaling. *The Bell System Technical Journal*, *54*(8), 1451–1462.

Mehrang, S., Pietila, J., Tolonen, J., Helander, E., Jimison, H., Pavel, M., & Korhonen, I. (2017). Human activity recognition using a single optical heart rate monitoring wristband equipped with triaxial accelerometer. In *Embec & nbc 2017* (pp. 587–590). Springer.

Moon, T. K., & Stirling, W. C. (2000). *Mathematical methods and algorithms for signal processing* (Vol. 1). Prentice hall Upper Saddle River, NJ.

Morris, D., Saponas, T. S., Guillory, A., & Kelner, I. (2014). Recofit: using a wearable sensor to find, recognize, and count repetitive exercises. In *Proceedings of the sigchi conference on human factors in computing systems* (pp. 3225–3234).

Mortazavi, B. J., Pourhomayoun, M., Alsheikh, G., Alshurafa, N., Lee, S. I., & Sarrafzadeh, M. (2014). Determining the single best axis for exercise repetition recognition and counting on smartwatches. In *2014 11th international conference on wearable and implantable body sensor networks* (pp. 33–38).

Mujahid, S., Sierra, G., Abdalkareem, R., Shihab, E., & Shang, W. (2017). Examining user complaints of wearable apps: a case study on android wear. In *Proceedings of the 4th international conference on mobile software engineering and systems* (pp. 96–99).

Murtagh, F., & Contreras, P. (2017). Algorithms for hierarchical clustering: an overview, ii. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *7*(6), e1219–n/a. Retrieved from http://dx.doi.org/10.1002/widm.1219 (e1219) doi: 10.1002/widm.1219

Nabian, M. (2017). A comparative study on machine learning classification models for activity recognition. *Journal of Information Technology & Software Engineering*, *7*(04), 4–8.

Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (icml-10)* (pp. 807–814).

Nguyen, T. T., Fernandez, D., Nguyen, Q. T., & Bagheri, E. (2017). Location-aware human activity

recognition. In *International conference on advanced data mining and applications* (pp. 821–835).

Nourani, H., Shihab, E., & Sarbishe, O. (2019). The impact of data reduction on wearable-based human activity recognition. In *Proceedings of the 15th workshop on context modeling and recognition* (pp. 89–94). IEEE.

Oreifej, O., & Liu, Z. (2013). Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 716–723).

O'Shea, T. J., Corgan, J., & Clancy, T. C. (2016). Convolutional radio modulation recognition networks. In C. Jayne & L. Iliadis (Eds.), *Engineering applications of neural networks: 17th international conference, eann 2016, aberdeen, uk, september 2-5, 2016, proceedings* (pp. 213–226). Cham: Springer International Publishing. Retrieved from https://doi.org/10.1007/978-3-319-44188-7_16 doi: 10.1007/978-3-319-44188-7_16

Oza, N. C., & Tumer, K. (2008). Classifier ensembles: Select real-world applications. *Information Fusion*, *9*(1), 4–20.

Park, C. H. (2013). A feature selection method using hierarchical clustering. In *Mining intelligence and knowledge exploration* (pp. 1–6). Springer.

Promotion - motsai documentation. (2020). http://docs.motsai.com/Neblina/Neblina_Development_Kit/ProMotion/index. ((Accessed on 11/13/2017))

RajKumar, A., Vulpi, F., Bethi, S. R., Raghavan, P., & Kapila, V. (2020). *Usability study of wearable inertial sensors for exergames (wise) for movements assessment and exercise.* mHealth.

Rish, I., et al. (2001). An empirical study of the naive bayes classifier. In *Ijcai 2001 workshop on empirical methods in artificial intelligence* (Vol. 3, pp. 41–46).

Rosati, S., Balestra, G., & Knaflitz, M. (2018). Comparison of different sets of features for human activity recognition by wearable sensors. *Sensors*, *18*(12), 4189.

Saeys, Y., Inza, I., & Larrañaga, P. (2007). A review of feature selection techniques in bioinformatics. *bioinformatics*, *23*(19), 2507–2517.

Sarbishei, O. (2019). A platform and methodology enabling real-time motion pattern recognition on low-power smart devices. *2019 IEEE World Forum on Internet of Things*, 257–260.

Schilit, B. N., Adams, N., Want, R., et al. (1994). *Context-aware computing applications*. Xerox Corporation, Palo Alto Research Center.

Sena, J., Santos, J. B., & Schwartz, W. R. (2018). Multiscale dcnn ensemble applied to human activity recognition based on wearable sensors. In *2018 26th european signal processing conference (eusipco)* (pp. 1202–1206).

Shakya, S. R., Zhang, C., & Zhou, Z. (2018). Comparative study of machine learning and deep learning architecture for human activity recognition using accelerometer data. *Int. J. Mach. Learn. Comput*, *8*, 577–582.

Shoaib, Bosch, S., Incel, O. D., Scholten, H., & Havinga, P. J. M. (2014). Fusion of smartphone motion sensors for physical activity recognition. *Sensors*, *14*(6), 10146–10176. Retrieved from http://www.mdpi.com/1424-8220/14/6/10146 doi: 10.3390/s140610146

Shoaib, M., Bosch, S., Incel, O., Scholten, H., & Havinga, P. (2016). Complex human activity recognition using smartphone and wrist-worn motion sensors. *Sensors*, *16*(4), 426.

Siirtola, P., & Röning, J. (2012). Recognizing human activities user-independently on smartphones based on accelerometer data. *IJIMAI*, *1*(5), 38–45.

Soro, A., Brunner, G., Tanner, S., & Wattenhofer, R. (2019). Recognition and repetition counting for complex physical exercises with deep learning. *Sensors*, *19*(3), 714.

Sousa, W., Souto, E., Rodrigres, J., Sadarc, P., Jalali, R., & El-Khatib, K. (2017). A comparative analysis of the impact of features on human activity recognition with smartphone sensors. In *Proceedings of the 23rd brazillian symposium on multimedia and the web* (pp. 397–404).

Sow, D., Turaga, D. S., & Schmidt, M. (2013). Mining of sensor data in healthcare: A survey. In *Managing and mining sensor data* (pp. 459–504). Springer.

Sprager, S., & Juric, M. B. (2015). Inertial sensor-based gait recognition: A review. *Sensors*, *15*(9), 22089–22127.

Suykens, J. A., & Vandewalle, J. (1999). Least squares support vector machine classifiers. *Neural processing letters*, *9*(3), 293–300.

Trabelsi, D., Mohammed, S., Chamroukhi, F., Oukhellou, L., & Amirat, Y. (2013). An unsupervised approach for automatic activity recognition based on hidden markov model regression. *IEEE Transactions on automation science and engineering*, *10*(3), 829–835.

Waldron, R. (n.d.). *Generic sensor api.* Retrieved from https://www.w3.org/TR/accelerometer/

Wang, Y., Cang, S., & Yu, H. (2019). A survey on wearable sensor modality centred human activity recognition in health care. *Expert Systems with Applications*.

Xi, X., Tang, M., Miran, S. M., & Luo, Z. (2017). Evaluation of feature extraction and recognition for activity monitoring and fall detection based on wearable semg sensors. *Sensors*, *17*(6), 1229.

Yazdansepas, D., Niazi, A. H., Gay, J. L., Maier, F. W., Ramaswamy, L., Rasheed, K., & Buman, M. P. (2016). A multi-featured approach for wearable sensor-based human activity recognition. In *2016 ieee international conference on healthcare informatics (ichi)* (pp. 423–431).

Yong, C. Y., Sudirman, R., Mahmood, N. H., & Chew, K. M. (2013). Human hand movement analysis using principle component analysis classifier. In *Applied mechanics and materials* (Vol. 284, pp. 3126–3130).

Yurtman, A., & Barshan, B. (2017). Activity recognition invariant to sensor orientation with wearable motion sensors. *Sensors*, *17*(8), 1838.

Zardoshti-Kermani, M., Wheeler, B. C., Badie, K., & Hashemi, R. M. (1995). Emg feature evaluation for movement control of upper extremity prostheses. *IEEE Transactions on Rehabilitation Engineering*, *3*(4), 324–333.

Zhang, M., & Sawchuk, A. A. (2011). A feature selection-based framework for human activity recognition using wearable multimodal sensors. In *Proceedings of the 6th international conference on body area networks* (pp. 92–98).

Zhang, S., Rowlands, A. V., Murray, P., Hurst, T. L., et al. (2012). *Physical activity classification using the genea wrist-worn accelerometer* (Unpublished doctoral dissertation). Lippincott Williams and Wilkins.

Zhao, Z., Morstatter, F., Sharma, S., Alelyani, S., Anand, A., & Liu, H. (2010). Advancing feature selection research. *ASU feature selection repository*, 1–28.

Zhu, C., & Sheng, W. (2009). Human daily activity recognition in robot-assisted living using multi-sensor fusion. In *2009 ieee international conference on robotics and automation* (pp. 2154–2159).