

Physical Activity Recognition From Smartphone Accelerometer Data for User Context Awareness Sensing

Johan Wannenburg and Reza Malekian, *Member, IEEE*

Abstract—Physical activity recognition of everyday activities such as sitting, standing, laying, walking, and jogging was performed, through the use of smartphone accelerometer data. Activity classification was done on a remote server through the use of machine learning algorithms, data was received from the smartphone wirelessly. The smartphone was placed in the subject's trouser pocket while data was gathered. A large sample set was used to train the classifiers and then a test set was used to verify the algorithm accuracies. Ten different classifier algorithm configurations were evaluated to determine which performed best overall, as well as, which algorithms performed best for specific activity classes. Based on the results obtained, very accurate predictions could be made for offline activity recognition. The kNN and kStar algorithms both obtained an overall accuracy of 99.01%.

Index Terms—Accelerometer, activity recognition, machine learning, smartphone.

I. INTRODUCTION

THE use of smartphones has become an integral part of our daily lives. These smartphones are ever improving and developing into pocket size computers capable of performing complex calculations and operations [1]. A smartphone has a collection of various sensors on-board that could be useful in the field of human activity recognition and monitoring. These devices are the perfect gateway into the measuring of user specific parameters such as physiological parameters, physical activities, as well as, a number other factors that could give more insight into our daily routines [2]. Human activity recognition is a very important field of research and could prove meaningful in applications in healthcare, smart environments, and security. However, this could prove somewhat complex [3]. Physical feature extraction is an integral part of context awareness and activity monitoring, it could be used to improve predictions and decision-making in mobile health

monitoring. Accurate discernment between normal activity and medical related emergencies could result in the possibility of minimizing human intervention in ubiquitous health monitoring through; an example of this would be fall detection [1], [4]. There are a number of different approaches to physical activity recognition, in this paper a system that utilizes a smartphone's accelerometer data along with machine learning algorithms to classify activity is proposed. The approach will make use of feature extraction and selection techniques in order to optimize the classification algorithm. The smartphone will act as a wearable data collection device while a remote server will be responsible for all processing and classification of physical activity. Various machine learning algorithm will be evaluated in order to determine which algorithm performs best overall, as well as, for unique individual activities.

The novelties explored and implemented in this paper are as follows.

- 1) The implemented approach can identify and classify both dynamic and static physical user activity.
- 2) The classification process utilizes the accelerometer data of the user's smartphone, eliminating the need for an external worn device.
- 3) The design makes use of feature selection in order to reduce the number of required features, thus reducing the model complexity while maintaining classification accuracy.
- 4) The approach is dynamically designed to allow for the training and classification of new activities such as falling, walking up stairs, and down stairs.
- 5) A comparison of how well various classifier algorithms perform is the focus of this paper. Thus identifying which classifiers perform better for specific activities.

This paper is organized as follows. Section II will describe concepts and techniques required for successful activity recognition by referring to related works. In Section III an overview of the system design will be given. Section IV explains the design implementation, here all aspects of the design will be discussed, including the exact role of the smartphone and the remote processing server. In Section V the experimental procedure and datasets will be discussed. All findings and results are evaluated and summarized in Section VI. Section VII gives some aspects to focus on in future work that may improve the developed system. Finally, this paper is concluded in Section VIII.

This work was supported in part by the Research Development Programme under Grant AOX220 and in part by the National Research Foundation under Grant 90908.

The authors are with the Department of Electrical, Electronic, and Computer Engineering, University of Pretoria, Pretoria 0002, South Africa (e-mail: johan.wannenburg@gmail.com; reza.malekian@ieee.org).

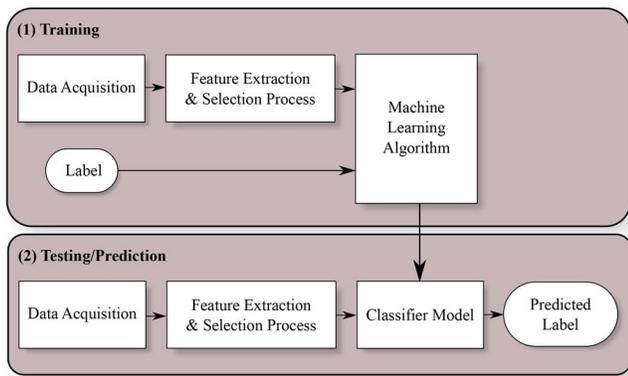


Fig. 1. Flow diagram of the proposed supervised machine learning-based classification system.

II. ACTIVITY RECOGNITION THROUGH ANALYSIS OF RELATED WORKS

Physical activity recognition can be done in a number of ways which usually consists of a battery powered wireless sensor node, measuring accelerometer data [5]. This paper focuses on obtaining accelerometer from a smartphone, other current research has shown interest in this and it has shown promising results and successful implementation [1]–[3], [5], [6]. Activity recognition is not a new concept and has been implemented for some time. One of the earliest breakthroughs in activity recognition research was performed by Bao and Intille [7], this paper is significant and highly cited. Incel *et al.* [8] have also made a significant contribution in activity recognition through the use of smartphones, through identifying and summarizing various approaches. Physical activity can be classified into either static postures, such as sitting, standing, lying; or dynamic motions, such as running, walking, stair climbing, and so forth [9]. These can be classified as primitives, other research seeks to identify and classify composite activities, namely a sequence of primitives, and this might give more insight into user activity and context. Activity recognition consists of four stages: 1) preprocessing; 2) feature extraction; 3) model training; and 4) classification [2]. When putting these basic steps into a proposed system, a generic model can be used to explain the system flow while highlighting important steps such as data acquisition, input data labels, feature extraction and selection, the selected machine learning algorithm and classification can be seen in Fig. 1.

Generally activity recognition starts by the collection of data from an accelerometer. Accelerometers are very popular sensors in the field of biomedicine and can be used for tremor analysis, assessment of physical activity, qualification of metabolic energy expenditure [9], measurement of gait parameters [10], fall detection [11], postural detection and transition [12], [13], and the measurement of activities performed in daily living [14], [15]. Most of these applications make use of computational techniques that require the implementation of machine learning algorithms, these algorithms serve to solve nonlinear multivariate problems.

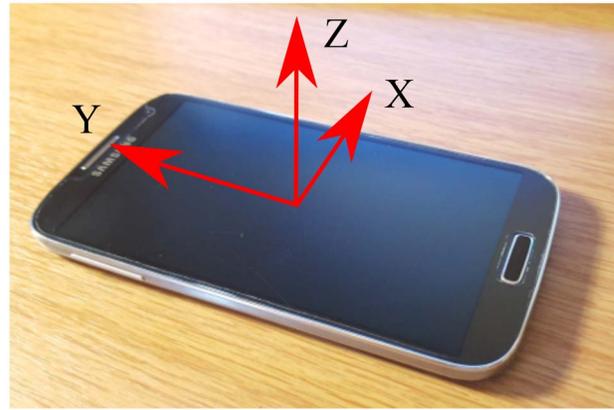


Fig. 2. Smartphone with accelerometer axial orientation.

A tri-axial accelerometer is a device capable of measuring acceleration in 3-D, namely the x , y , and z axes [3]. However, this raw acceleration data needs to be processed before it can be used for activity classification [6]. Fig. 2 shows the accelerometer axial orientation of the smartphone used in this paper.

A. Feature Extraction

Feature extraction is the most important part of activity recognition, since the selected features will determine the accuracy of activity recognition. There are various features, each having significant qualities relating to the specific activity being performed. Most features are derived or calculated based on statistical models, therefore include values such as mean, median, max, min, standard deviation, correlation coefficients, etc. [2], [5]. Features such as correlation are very useful for differentiating activities that involve translation in only one direction from those in multiple directions [16]. These features are usually classified as being either time or frequency-based features, both aspects are very important and can improve the accuracy of the model, however, the features are selected based on the application and in order to optimize the classification algorithm. The main frequency components regarding activity recognition have found to be located between 1 and 18 Hz [5]. The aforementioned study was done based on an accelerometer worn around the ankle, thus, meaning if the accelerometer were to be placed near the hip the acceleration forces would be lower than at the ankle, hence frequencies measured at the hip are far lower and reduced to below 5 Hz [5], [17]. Lower frequencies allow for lower computation cost which indirectly translate to lower power consumption [5]. Both dc and ac frequency components could provide useful features for activity recognition. The dc component is mostly related to the influences of gravity and can thus be used to determine static posture, while the ac components can be used to classify dynamic motion [3]. Feature extraction is usually performed on data obtained using a sliding window approach, with a 50% overlap; where this window refers to a set time period based on the sampling frequency and sampling rate. This window should be selected in order for one complete activity to be observed in a single window [5].

B. Feature Selection

The performance of a classification algorithm drastically depends on dimensions of the feature space or vector, therefore it is critical to minimize and reduce the effects of dimensionality [9]. One way to reduce dimensionality would be to apply feature selection algorithms, these algorithms are used to detect and discard features that make a minimal contribution to the success of the classifier [9]. Feature selection algorithms usually make use of sequential search algorithms such as branch bound searches and the Pudil algorithm. Feature selection is also a good approach to prevent biasing while training, since some features could bias the classifier model. There are two general approaches to feature selection as follows.

- 1) Filter methods, where features are scored and ordered based on a search algorithm. The ordered list of features shows which features contribute the most to the model.
- 2) Wrapper methods make use of various combinations of the features and compare results for a given classifier, this information is then used to evaluate which feature combinations result in the most accurate model. This approach is very time consuming and requires a tremendous amount of processing power.

C. Classification

Activity recognition is usually treated as a supervised machine learning problem, this is due to the fact that the desired outputs are controlled and an unsupervised model may result in a more complex model [1], [3], [6]. However, there are approaches utilizing unsupervised models such as hidden Markov models [9]. Some of the most common supervised models include decision tree classifiers such as the J48 [1] and random forest [3], support vector machines (SVMs), K -nearest neighbors, Naïve Bayes, logistic regression and multilayer perceptron such as artificial neural networks (ANNs). In order to reduce computational and model complexity a supervised approach will be used. Certain classifiers are more accurate when determining specific activities, for example multilayer perceptron models are better at predicting slightly more complex activities such as jogging and stair climbing [6]. For the simpler activities such as sitting and standing, those that detect posture a decision tree was found to achieve higher accuracies [2]. In order to ensure that the system achieves the best possible accuracy for activity classification a combination of classifiers can be used. Here the optimal classifiers for each activity is selected and combined by a fusion method. There are a number of different classifier combination approaches, the most popular includes majority/plurality voting and average of probabilities [3]. There are other slightly more complex approaches implemented and tested by Ravi *et al.* [16] such as boosting, bagging and stacking, collectively known as meta-level classifiers. Plurality voting has been proven to show good results when combining classifiers [16].

D. Challenges in Activity Recognition

Activity recognition does suffer from certain factors, this is expected since it is a complex nonlinear problem. The developed classifiers can very easily be overcomplicated and result

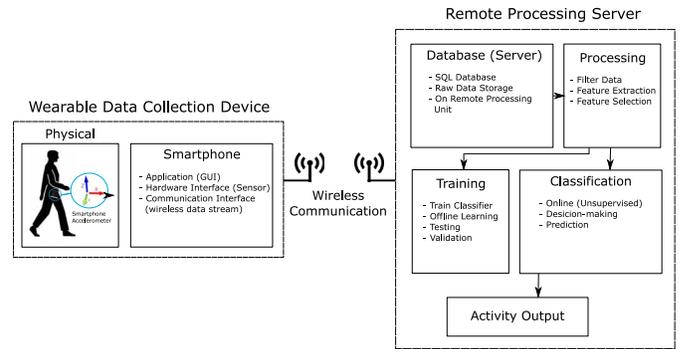


Fig. 3. System block diagram.

in long computational periods, as well as, possibly suffer from high biasing or variance. Thus it is crucial to select and minimize the features used as inputs to the classifier. Another factor that may result in an inaccurate model is the change in smartphone reference or orientation. This could happen when the smartphone moves around in the user's pocket, or if it is placed with a different orientation that the orientation the classifier was trained with. This can be eliminated by eliminating the effect of orientation by applying the concept of square summing, where all three axis data is combined [2], [5]. There are other ways that random orientation can be eliminated while still making use of data from all three axes [18].

III. SYSTEM OVERVIEW

In order to successfully develop a system capable of physical activity recognition it is important to decide on the expected outcomes and by using this to develop an approach. The expectation of this system is to accurately recognize and classify physical human activity through the use of accelerometer data obtained by means of a smartphone. The objective would thus be to successfully classify both static posture (standing, sitting, and lying), as well as, dynamic motion (walking, jogging, cycling, etc.) through the use of 3-axis accelerometer data obtained from a smartphone. Five activity classes have been identified in this paper and are as follows: 1) standing; 2) sitting; 3) laying down; 4) walking; and 5) jogging. These five activities make up the basic static and dynamic activities a person performs throughout their daily lives, thus forming a crucial part in successfully recognizing physical activity. The 3-axis accelerometer data was collected from the user's smartphone while in his/her trouser pocket. This places the smartphone relatively close to the hip where successful activity recognition can take place [6].

The raw data was then sent to a remote server where processing and classification takes place. This results in two main subsystems: 1) the wearable data collection device and 2) the remote processing unit. The acquisition of raw data takes place on the wearable data collection device which was implemented on a smartphone platform, more specifically an Android device. This raw accelerometer data was obtained from the smartphone's accelerometer sensor. It was required that the smartphone had an application running allowing it to perform data acquisition and a connection to the remote server.

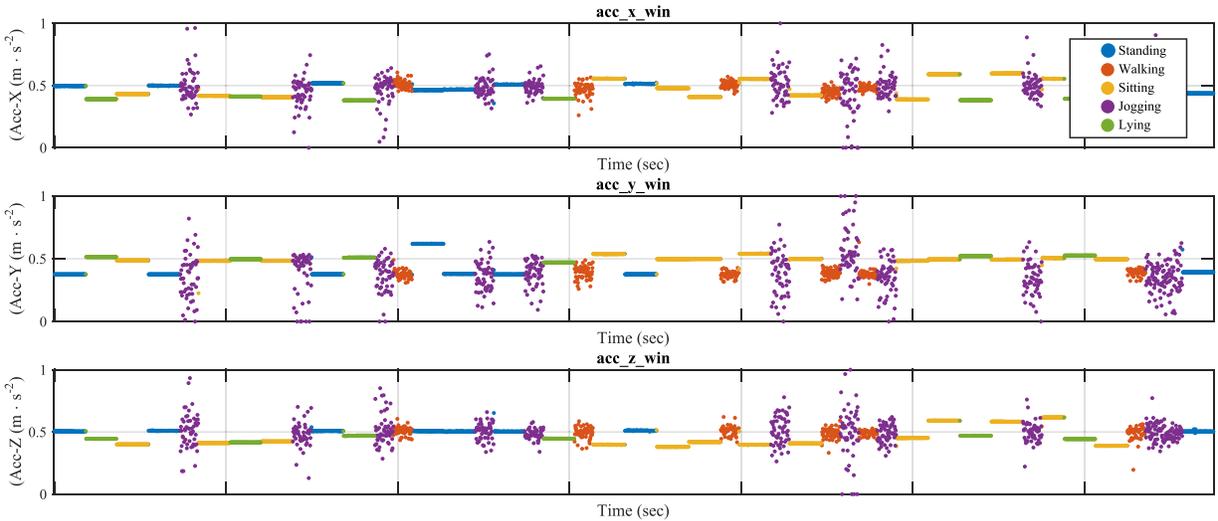


Fig. 4. Plot of the sampled x , y , and z data including activity labeling.

Once the data has been acquired by the smartphone, it was sent to the remote server for further processing and classification. It is important that this communication is wireless in order to get realistic data where the user can perform normal everyday physical activities without any discomfort or restrictions. A Wi-Fi-based connection through an access point was established between the sensing device and server. Once the data was received by the remote server, hosted on a personal computer, the data is stored, a database approach was taken for the offline data which is used to train the model. Online classification will require an approach that can operate in real time. If the data is in the correct format feature extraction and feature selection is applied to the data in order to gain meaningful information for the activity classification process. The activity classification process will be split into two sections: 1) training, where the classifier model will be trained offline and 2) classification, where activity recognition will take place on unsupervised data. An overview of the proposed system can be seen in Fig. 3.

The proposed method is tailored for individual users, meaning that the activity recognition is designed for a single user, but implementable on various different users. Such a design supports independent user activity and is thus dynamic in the sense that new activities can be trained and classified with ease. This allows for the research focus to be placed on the comparison of various classifiers and how accurately they classify a specific activity, which could lead to the utilization of a classification technique based on a combination of classifier results.

IV. DESIGN IMPLEMENTATION

A. Wearable Data Collection Device (Smartphone)

As explained in the experimental procedure the smartphone will be responsible for acquiring the accelerometer data. The smartphone made use of the MATLAB Mobile application which enabled a connection to the remote server over a network. This application also allows the control of any of the

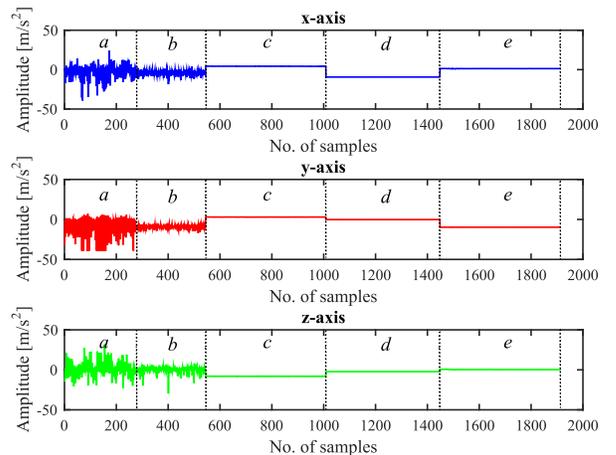


Fig. 5. Raw accelerometer data of the x , y , and z axes indicating the five possible types of activity.

smartphones sensors remotely. This approach was selected in order to allow the test subject to perform activities normally while the researcher controls the data acquisition remotely. A sampling rate of 10 Hz was selected based on the fact that important frequencies at the hip would be below 5 Hz. According to the Nyquist criterion this should be twice the maximum frequency component, resulting in a sampling frequency of 10 Hz. This means that a sample is taken every 100 ms. Fig. 4 shows a plot of the sampled data, here each activity segment can be identified.

B. Remote Processing Server

1) *Preprocessing*: Once the data acquisition has taken place each of the individually saved activity sets are appended to create one large data set that will be used for training and testing. This large data set is generated by appending the individual activities in a random order to reduce biasing. Fig. 5 displays each of the unique raw activity axial data by means of indicators a through e . Each of the analyzed activities can

TABLE I
TIME-BASED AND FREQUENCY-BASED FEATURES

Time Based Features	Frequency Based Features
Mean	Mean
Maximum	Maximum
Minimum	Minimum
Median	Median
Standard Deviation	Skewness
Signal Magnitude Area (SMA)	Kurtosis
Mean Deviation	Principal Component Analysis (PCA)
Principal Component Analysis (PCA)	
Interquartile Range (IQR)	

be identified as follows: 1) jogging; 2) walking; 3) sitting; 4) laying down; and 5) standing. It can also be observed that the raw data consists of the three mentioned axes, namely, x , y , and z .

Hereafter the data was normalized using unity-based normalization, where the data is normalized to a range between 0 and 1. Normalization allows all data to be treated the same and reduce biasing and variance. The normalized data was then split into the training and test data, which was a 70%/30% split. From here all processing operations were performed on both training and test data. Now that the raw data is formatted in such a way that it can be used in the next step, windowing. This process involves systematically passing a sliding window over the data set. Windowing is necessary since features are usually based on statistical variables, meaning that they are calculated from a set of values. The window allows for the creation of a set of values from which a unique feature can be extracted for each window. The window size is very important and according to various research should contain at least one cycle of an activity to uniquely identify it, this means that the window size can be anywhere between 1 and 7.5 s depending on the required activity recognition [2], [5]. A window size of 1 s was selected for this design, which was found to be optimal according to related work [17]. The selected window process is only used to isolate a group of samples for feature extraction therefore a rectangular window was used. The window should also be overlapping, this allows for data to be analyzed in a flowing manner rather than just set data segments; meaning that specific activities will not be mislabeled. To be more specific it was found that an overlap of 50% is optimal for feature selection [2], [3], [16], [17], [19]. Based on this knowledge a window with 50% overlap was selected, meaning that if the sampling rate was selected as 10 Hz a 1 s window consists of ten samples. Of these ten samples only half of them will be new samples in every window, this is due to the 50% overlap.

2) *Features and Feature Extraction*: Making use of the windowed data feature extraction was performed, a total of 46 features were selected. A summary of the features is given in Table I, all features were performed on the x , y , and z axes except for the calculation of the signal magnitude area.

Once all the features have been calculated (Table II) the data is formatted into a feature matrix, thus simplifying the way it is stored and ordered. This method also allows for the data to

TABLE II
SELECTED FEATURES WITH CONTRIBUTION RATINGS

No.	Gain rating	Feature	Domain
1	2.02528	Y min	time
2	1.94741	Y max	time
3	1.94121	Y PCA	frequency
4	1.93739	Y median	frequency
5	1.89951	Z mean	frequency
6	1.85121	X mean	frequency
7	1.8351	Z max	time
8	1.79218	Y median	time
9	1.77667	X max	time
10	1.75652	Z min	time
11	1.74489	Y mean	time
12	1.74489	Y max	frequency
13	1.74448	Y PCA	time
14	1.68438	X min	time
15	1.63542	Z median	time
16	1.60395	Z max	frequency
17	1.60395	Z mean	time
18	1.60233	Z PCA	time
19	1.5962	Z PCA	frequency
20	1.57716	X PCA	frequency
21	1.57298	X PCA	time
22	1.56491	X median	time
23	1.56476	X mean	time
24	1.56476	X max	frequency
25	1.45363	SMA	time
26	1.28189	X std dev	time
27	1.25357	Z std dev	time
28	1.22494	Z IQR	time
29	1.22488	X IQR	time

be exported and saved in a compatible format for the WEKA suite. All preprocessing and feature extraction was performed in MATLAB, while the feature selection, classification and verification took place in WEKA [20].

3) *Feature Selection*: The calculated features were exported to WEKA for feature selection. A filter method known as InfoGainAttributeEval was applied to the data; in which attributes were evaluated and ranked with regard to their contribution to the detection of the relevant activity classes. This method relies on measuring the information gain of a feature with respect to the class, and ranking the features based on their contribution. A ranker search method was utilized, where attributes are ranked according to individual evaluation. Using this approach, the 29 highest contributing features of the proposed 46 were selected for classification. This process assisted in reducing dimensionality, as well as, improved the time period it took for training and testing. Table I lists the priority order of the features after applying the InfoGainAttributeEval feature selection algorithm.

4) *Classification*: As mentioned the classification process was also completed using the WEKA suite. Seven main classifiers were identified as possible candidates for supervised physical activity recognition. Each of the classifiers were evaluated in order to see which classifier performs best as an overall model. It was also noted which classifier showed the best performance per activity. The seven classifiers were set up

TABLE III
TESTED MACHINE LEARNING ALGORITHMS

No.	Classifier	Parameters
1	Support vector machine	SVM algorithm
2	Multilayer Perceptron	17 hidden layer neurons
3	Naïve Bayes	-
4	Naïve Bayes	BayesNet
5	k-Nearest Neighbour	n = 1
6	k-Nearest Neighbour	n = 5
7	Naïve Bayes	Bagging method
8	Tree method	J48
9	Multilayer Perceptron	25 hidden layer neurons
10	kStar	-

with various configurations resulting in ten classifier configurations. The classifiers used in this paper are listed in Table III, each were evaluated with the same data sets, same number of features, as well as, the same windowing technique.

V. EXPERIMENTAL PROCEDURE AND DATASETS

A. Experimental Procedure

For purposes of this experiment the test procedure was kept very simple and only consisted of one activity being performed rather than a fixed set of activities. The reason for this would be that if a set sequence of activities were to be performed by each subject and then used to train the model that sequence will bias the model to a set sequence and not to a random order of performed activities, as the case would be in normal everyday activity. The activity was recorded for a set time interval, this interval was chosen as 36 s. Therefore, the test subject selects and performs a specific activity for 36 s. The 36 s is then later truncated into 30 s to ensure activity change is not interpreted as an activity since this will reduce the model's accuracy.

B. Datasets

In order to obtain a large sample set it had to be decided what constituted as a large sample set, a set that allowed for accurate classification of physical activity. The objective of this paper was to obtain roughly 5 hours' worth of accelerometer data. It was required that there be an equal distribution of each activity, and it should also be kept in mind that the sample set will be split into training (70%) and test (30%) data. Table IV shows a summary of the collected data sample set.

The final sample set consisted of data collected from ten different test subjects, made up of a total of 505 different activity recordings. The sample set was roughly 187 920 samples (at a sampling rate of 10 Hz), worth approximately 313.2 min or 5.22 h. Meaning that 3.65 h (roughly 438 possible activities) was used as training data and the other 1.57 h (roughly 188 possible activities) was used for testing.

VI. RESULTS AND DISCUSSION

Using the data set described in Section V in combination with the ten classifier configurations from Section IV reasonable results were obtained. Table V gives a summary of the

TABLE IV
SUMMARY OF DATASET

Training Data		
	No. of activities	Percentage
Jogging	3816	14.51%
Laying	4258	16.19%
Sitting	6859	26.07%
Standing	7034	26.74%
Walking	4340	16.50%
Total	26307	100%
Test Data		
	No. of activities	Percentage
Jogging	1557	13.81%
Laying	1639	14.54%
Sitting	3188	28.27%
Standing	2923	25.92%
Walking	1968	17.45%
Total	11275	100%

TABLE V
SUMMARY OF CLASSIFIER PERFORMANCE

No.	Classifier	Classification accuracy
1	SVM	94.32%
2	ANN (a=17)	98.74%
3	Naïve Bayes	91.10%
4	Naïve Bayes (BayesNet)	97.92%
5	kNN (n=1)	99.01%
6	kNN (n=5)	99.00%
7	Bagging (Naïve Bayes)	91.10%
8	Tree	98.80%
9	ANN (a=25)	98.88%
10	kStar	99.01%

TABLE VI
ACTIVITY CLASSIFICATION ACCURACY FOR EACH CLASSIFIER

	Best Classifier per Activity				
	Jogging	Laying	Sitting	Standing	Walking
1	97.62%	88.47%	96.20%	92.54%	96.19%
2	98.14%	99.02%	99.22%	99.08%	97.71%
3	94.80%	93.47%	84.35%	92.37%	95.22%
4	96.98%	99.02%	98.87%	98.53%	95.33%
5	98.91%	99.02%	99.18%	99.11%	98.63%
6	99.04%	99.15%	99.28%	99.01%	98.37%
7	94.93%	93.35%	84.38%	92.37%	95.17%
8	97.94%	99.15%	99.22%	99.32%	97.76%
9	98.14%	99.27%	99.25%	99.38%	98.53%
10	98.09%	99.02%	99.28%	99.01%	98.58%

overall model accuracy for each of the classifier configurations. From this table it is seen that kNN and kStar have the highest classification accuracy for offline activity recognition with an accuracy of 99.01%. The model that showed the lowest accuracy was the Naïve Bayes variants with an accuracy of 91.10%. Each of these algorithms show a high level of accuracy and could prove acceptable for online real-time activity recognition.

Table VI shows a summary of how well each classifier configuration performed with respect to activity. Based on

TABLE VII
CONFUSION MATRIX FOR CONFIGURATION 3 CLASSIFIER

		Predicted Class				
		Jogging	Laying	Sitting	Standing	Walking
Actual Class	Jogging	1476	0	0	0	81
	Laying	17	1532	88	0	2
	Sitting	39	457	2689	0	3
	Standing	200	0	1	2700	22
	Walking	86	0	3	5	1874

TABLE VIII
CONFUSION MATRIX FOR CONFIGURATION 9 CLASSIFIER

		Predicted Class				
		Jogging	Laying	Sitting	Standing	Walking
Actual Class	Jogging	1528	2	7	6	14
	Laying	3	1627	2	5	2
	Sitting	5	8	3164	5	6
	Standing	3	4	4	2905	7
	Walking	12	3	5	9	1939

the results there is not one classifier that achieves the highest classification accuracy for each activity. It was found that when classifying jogging configuration 6 yielded the highest accuracy with 99.04%, and configuration 3 the lowest with 94.80%. When it came to the classification of laying down configuration 9 showed the best results with an accuracy of 99.27% while configuration 1 had the worst accuracy with only 88.47%. Sitting was classified most accurately by configuration 6 with 99.28% and configuration 7 had the lowest accuracy 84.38%. As for standing, classifier 9 showed the best results with 99.38% accuracy while configurations 3 and 7 had the lowest accuracy (92.37%). Walking was something that was classified well by all algorithms and was best classifier by configuration 5 (98.63%) and configuration 7 had the lowest accuracy with 95.17% which was still deemed very high.

Tables VII and VIII show the confusion matrices for configurations 3 and 9; where 9 had the highest overall accuracy and 3 the lowest. This was used to show how well a classifier predicted an activity for each activity of the test data set. The test data set used in this experiment consisted of a total of 11 275 possible activities to be classified, as discussed in Section V.

When comparing the acquired results with that of related works, it was found that the classifiers produced results comparable to related works [6], [7], [16], [21]. Table IX shows a comparison of results between this paper and the results obtained by them. According to this summary of results the classifier accuracies obtained in this paper was comparable with that of others.

VII. IMPROVEMENTS AND FUTURE PROSPECTS

Even though the results obtained are acceptable for offline activity classification for this to be a successful means of online real-time activity classification there are a few things that need to be taken into consideration. First, the training needs to be done using data from more than ten test subjects in order to design a truly nonbiased model, perhaps open data

TABLE IX
COMPARISON OF RESULTS FOR VERIFICATION

Classifier	Classification accuracy	Related work accuracy	Reference
SVM	94,32%	98,15%	[16]
Naïve Bayes	91,10%	34,94%	[7]
		98,86%	[16]
kNN (n=1)	99,01%	98,15%	[16]
kNN (n=5)	99,00%	98,15%	[16]
Bagging (Naïve Bayes)	91,10%	98,15%	[16]
Tree	98,80%	77,31%	[7]
		85,10%	[6]
ANN (Multilayer Perceptron)	98,88%	91,70%	[6]
kStar	99,01%	98,10%	[21]

sets could be used. More activity types could also be included to create a better understanding of the user's activity pattern, this could include activities such as stairs climbing and falling. Some work also includes activities that are performed on a regular basis that are not necessary related to physical activity such as driving. With an increase in data the algorithm could also be improved so that it is user independent, resulting in a dynamic algorithm that would not need to be trained by an individual user before use, but rather perform classifications accurately in real time. As seen in this paper certain algorithms perform better for specific activities, in order to improve the overall activity recognition accuracy meta classification methods could be considered and included, this could include methods such as plurality voting [16]. It is a method that performs classification based on a combination of classifier results, this could ensure that the best classifier could be selected for each specific activity. The sliding window could also have an effect on the model's performance when implementing it in real time. When the window has a large overlap it requires a higher number of resources, this could cause delay and thus not result in a truly real-time classification implementation.

VIII. CONCLUSION

The results obtained in this paper were in line with what was expected, and offline activity classification was performed successfully. A thorough analysis of features and feature selection was done in order to optimize algorithm dimensionality. The implemented approach works well and is dynamic. It could easily be used with other accelerometer hardware or to detect other activities such as fall detection and climbing stairs. A number of classifications were analyzed to determine which classifier would yield the best results, and according to the obtained results kNN was the most accurate model. However, multiple classifiers were evaluated to determine if certain classifiers perform better for a specific given activity. This could potentially be used to implement plurality voting which could result in optimal accuracy for each activity. It can also be concluded that even though this is acceptable for offline activity classification more work will be need to be done to migrate this to an online real-time model implementation.

REFERENCES

- [1] A. O. Kansiz, M. A. Guvensan, and H. I. Turkmen, "Selection of time-domain features for fall detection based on supervised learning," in *Proc. World Congr. Eng. Comput. Sci.*, vol. 2. San Francisco, CA, USA, 2013, pp. 796–801.
- [2] P. Siirtola and J. Roning, "Ready-to-use activity recognition for smartphones," in *Proc. IEEE Symp. Comput. Intell. Data Min. (CIDM)*, Singapore, Apr. 2013, pp. 59–64.
- [3] A. Bayat, M. Pomplun, and D. A. Tran, "A study on human activity recognition using accelerometer data from smartphones," *Proc. Comput. Sci.*, vol. 34, pp. 450–457, Aug. 2014.
- [4] C.-C. Yang and Y.-L. Hsu, "A review of accelerometry-based wearable motion detectors for physical activity monitoring," *Sensors*, vol. 10, no. 8, pp. 7772–7788, Jan. 2010.
- [5] M. A. Álvarez de la Concepción, L. M. Soria Morillo, L. Gonzalez-Abril, and J. A. Ortega Ramírez, "Discrete techniques applied to low-energy mobile human activity recognition. A new approach," *Expert Syst. Appl.*, vol. 41, no. 14, pp. 6138–6146, Oct. 2014.
- [6] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, "Activity recognition using cell phone accelerometers," *ACM SIGKDD Explor. Newslett.*, vol. 12, no. 2, pp. 74–82, 2010.
- [7] L. Bao and S. S. Intille, "Activity recognition from user-annotated acceleration data," in *Proc. PERSASIVE*, Vienna, Austria, 2004, pp. 1–17.
- [8] O. D. Incel, M. Kose, and C. Ersoy, "A review and taxonomy of activity recognition on mobile phones," *BioNanoScience*, vol. 3, no. 2, pp. 145–171, Jun. 2013.
- [9] A. Mannini and A. M. Sabatini, "Machine learning methods for classifying human physical activity from on-body accelerometers," *Sensors*, vol. 10, no. 2, pp. 1154–1175, 2010.
- [10] R. E. Mayagoitia, J. C. Lötters, P. H. Veltink, and H. Hermens, "Standing balance evaluation using a triaxial accelerometer," *Gait Posture*, vol. 16, no. 1, pp. 55–59, 2002.
- [11] A. K. Bourke, J. V. O'Brien, and G. M. Lyons, "Evaluation of a threshold-based tri-axial accelerometer fall detection algorithm," *Gait Posture*, vol. 26, no. 2, pp. 194–199, 2007.
- [12] D. M. Karantonis, M. R. Narayanan, M. Mathie, N. H. Lovell, and B. G. Celler, "Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring," *IEEE Trans. Inf. Technol. Biomed.*, vol. 10, no. 1, pp. 156–167, Jan. 2006.
- [13] B. Najafi *et al.*, "Ambulatory system for human motion analysis using a kinematic sensor: Monitoring of daily physical activity in the elderly," *IEEE Trans. Biomed. Eng.*, vol. 50, no. 6, pp. 711–723, Jun. 2003.
- [14] I. C. Gyllensten and A. G. Bonomi, "Identifying types of physical activity with a single accelerometer: Evaluating laboratory-trained algorithms in daily life," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 9, pp. 2656–2663, Sep. 2011.
- [15] A. Godfrey, A. K. Bourke, G. M. O'laighin, P. van de Ven, and J. Nelson, "Activity classification using a single chest mounted tri-axial accelerometer," *Med. Eng. Phys.*, vol. 33, no. 9, pp. 1127–1135, 2011.
- [16] N. Ravi, N. Dandekar, P. Mysore, and M. L. Littman, "Activity recognition from accelerometer data," in *Proc. Nat. Conf. Artif. Intell.*, vol. 3. Pittsburgh, PA, USA, 2005, pp. 1541–1546.
- [17] D. M. Karantonis, M. R. Narayanan, M. Mathie, N. H. Lovell, and B. G. Celler, "Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring," *IEEE Trans. Inf. Technol. Biomed.*, vol. 10, no. 1, pp. 156–167, Jan. 2006.
- [18] J. Morales, D. Akopian, and S. Agaian, "Human activity recognition by smartphones regardless of device orientation," in *Proc. IS&T/SPIE Electron. Imag.*, vol. 2014. San Francisco, CA, USA, 2014, pp. 1–12.
- [19] Z. He and L. Jin, "Activity recognition from acceleration data based on discrete cosine transform and SVM," in *Proc. IEEE Int. Conf. Syst. Man Cybern. (SMC)*, San Antonio, TX, USA, Oct. 2009, pp. 5041–5044.
- [20] M. Hall *et al.*, "The WEKA data mining software: An update," *ACM SIGKDD Explor. Newslett.*, vol. 11, no. 1, pp. 10–18, 2009.
- [21] D. Y. Mahmood and M. A. Hussein, "Intrusion detection system based on K-star classifier and feature set reduction," *Int. Org. Sci. Res. J. Comput. Eng.*, vol. 15, no. 5, pp. 107–112, 2013.



Johan Wannenburg received the B.Eng. degree in electronics engineering from the University of Pretoria, Pretoria, South Africa, in 2014, where he is currently pursuing the M.Eng. degree in electronics engineering.

He is an Assistant Lecturer with the Department of Electrical, Electronic, and Computer Engineering, University of Pretoria. His current research interests include e-health, biosensors, body area networks, and finding novel ways of improving health monitoring through concepts such as context awareness.



Reza Malekian (M'12) is currently a Senior Lecturer with the Department of Electrical, Electronic, and Computer Engineering, University of Pretoria, Pretoria, South Africa. His current research interests include advanced sensor networks, Internet of Things, and mobile communications.

Dr. Malekian is also a Chartered Engineer and a Professional Member of the British Computer Society. He is an associate editor for the IEEE Internet of Things Journal.