9th. IT & T Conference

School of Computer Sciences

2009-10-01

Comparison of Feature Classification Algorithm for Activity Recognition Based on Accelerometer and Heart Rate Data

Dominic Maguire
Waterford Institute of Technology, 200027043@mail.wit.ie

Richard Frisby
Waterford Institute of Technology, rfrisby@wit.ie

Recommended Citation

Maguire, D., & Frisby, R. (2009). Comparison of feature classification algorithm for activity recognition based on accelerometer and heart rate data. *9th IT & T Conference*, Technological University Dublin, Dublin, Ireland, 22nd.-23rd. October. doi:10.21427/1mxx-w714

Comparison of Feature Classification Algorithms for Activity Recognition Based on Accelerometer and Heart Rate Data

Dominic Maguire [1], Richard Frisby [2]

[1] Waterford Institute of Technology
20027043@mail.wit.ie

[2] Waterford Institute of Technology
rfrisby@wit.ie

Abstract

This paper describes a project to compare two feature classification algorithms used in activity recognition in relation to accelerometer and heart rate data. Data was collected from six male and female subjects using a single tri-axial accelerometer and heart monitor attached to each subject's dominant thigh. Subjects carried out eight activities and the data was labelled semi-automatically. Features (mean, standard deviation, energy, correlation and mean heart rate) were extracted from the data using a window of 256 (3.4 seconds) and an overlap of 50%. Two classifers, *k*-NN and J48, were evaluated for activity recognition with 10-fold validation with *k*-NN ($k = 1$) achieving a better overall score of 90.07%.

Keywords: Activity recognition, feature classification

1 Introduction

Activity recognition fits into the bigger domain of context awareness by making devices aware of the activity or activities of the user [1]. The ability to recognise human activities is a key factor if computing systems are to interact seamlessly with the user's environment [2]. Context awareness is leading to the 'reinvention' of some domains such as healthcare [3] with studies examining a diverse range of applications such as hospital worker activity estimation [4], chronic disease management [5] and remote patient monitoring [6].

In context aware computing, data can be collected from a diverse range of sensors such as audio sensors, image sensors and accelerometers. Accelerometers facilitate the real-time recording of acceleration data along the x-, y- or z-axis. Due to their ever-diminishing size and embeddable nature, accelerometers can be unobtrusively worn by users. It has been noted that accelerometers have successfully crossed over to the mainstream via devices such as Apple's iPhone and Nintendo's Wii [7].

Much recent research has applied classification algorithms to accelerometer data in order to increase activity recognition accuracy [8] with some commentators stating that activity recognition is primarily a classification problem [1], [9]. Two classifiers, *k*-NN and J48/C4.5 (J48 is the Weka Toolkit [10] Java implementation of C4.5), were evaluated in this study. The Weka Toolkit is a collection of state-of-the-art machine learning algorithms and data pre-processing tools developed at the University of Waikato in New Zealand. Lombriser et al [11] identify *k*-NN and J48/C4.5 as being "the classifiers with the least complexities but rendering acceptable performance". The next section is an examination of related work which is followed by a description of the research methodology. The paper finishes with an analysis of the results and outlines some pointers for future work.

2 Related Work

Bao and Intille [12] noted that most previous studies examining activity recogition from accelerometer data were not suitable for real-world situations and were conducted either in laboratory conditions or used limited datasets. They assessed the performance of algorithms in identifying twenty activities under semi-naturalistic, simulated real-world conditions using five biaxial accelerometers. Decision table, instance-based learning, decision tree (C4.5) and naive Bayes classifiers were used with C4.5 providing the best performance recognsing everyday activities with an overall accuracy of 84%. The above study also identified the optimal single accelerometer position, for the set of activities they chose, as being on the thigh and that accuracy increased by 25% by using more than one accelerometer. Furthermore, it was shown that acceleration data could be augmented with heart rate data to determine the intensity of physical activities.

Pirttikangas et al [9] undertook a study using coin-sized sensor devices attached to four parts of the body: right thigh and wrist, left wrist and a necklace. 17 daily activties were examined using triaxial accelerometer and heart rate data. Two classifiers were used (multilayer perceptrons and kNN classifiers) with kNN achieving a 90.61% aggregate recognition rate for 4-fold cross-validation. Interestingly, heart rate data was collected but not used in the activity recognition process.

Ravi *et al* [1] collected data for eight activities using a single triaxial accelerometer worn near the pelvic region. In their introduction they outline research questions which are also relevant to this author:

- Which are the best classifiers for recognising activities; is combining classifiers a good idea?
- Which among the selected features/attributes are less important than others?
- Which activities are harder to recognise?

In this study, the performance of base-level classifiers and meta-level classifiers was compared. They found that combining classifiers using Plurality Voting provided the best overall results. Plurality Voting choses the class that has been predicted by a majority of the base-level classifiers as the final predicted class. Of the base-level classifiers, the decision tree C4.5 performed the best. In one setting, data was collected for a single subject over different days and mixed together and cross-validated. Accuracy was shown to be 97.29%. They concluded that activities can be accurately recognised using a single triaxial accelerometer.

Lombriser et al [11] demonstrated that online activity recognition algorithms could be run on their *SensorButton* miniaturised wireless sensor platform. Their main challenge was in selecting algorithms which would achieve acceptable recognition performance with limited computation resources. They examined seven office worker activities such as drinking water and using a mouse via accelerometer and light sensor data. They used k-NN and J48/C4.5 classifiers with both providing 98% accuracy during offline evaluation. Accuracy dropped during online implementation due to floating point bit accuracy on the 16-bit microcontroller. However, the k-NN classifier did slightly better (91%) than the J48/C4.5 (86%) for online recognition. The authors state that these classifiers were used due to their low complexity and acceptable rendering performance.

3 Data Collection

3.1 Hardware

The hardware used for this project included an Alive Technologies Heart Monitor and Accelerometer. This provides triaxial accelerometry at 75 Hz with a dynamic range of +/-2.7 *g* using an 8-bit resolution. Data was stored, using the proprietary `.ats` format, to a Secure Digital (SD) card on the device. Electrodes from Medick Healthcare were used to facilitate the capture of heart data. Data processing was undertaken on a 2.66 GHz Apple iMac with 4 GB of RAM.

Bao and Intille [12] identified the dominant thigh as being the optimal single accelerometer position and this was emulated in this study. The accelerometer was attached to a loose-fitting tourniquet using tape. The relatively short length of the ECG leads meant that the choice of electrode positions for monitoring heart rate was limited to the supra-xiphisternal level.

A mobile phone stopwatch was used to synchronise the accelerometer timestamp for accurate label generation.

3.2 Label Generation

Data was collected for the set of eight activities examined by Ravi *et al* [1] in a similar single accelerometer study: standing, brushing teeth, climbing up stairs, climbing down stairs, walking, running, vacuuming and situps. Each subject performed the activities in the same order as stated above. Each activity was performed for one minute except for climbing and descending stairs (in a regular two-storey house) which were carried out twice. Label generation was semi-automatic, i.e. the author recorded start and stop times for each subject and labelled each period with the specific, observed activity. Ten seconds were removed from the beginning and end of most activities to ensure the data actually corresponded to the activity being recorded. This was reduced to two seconds in the case of climbing and descending stairs. Six subjects, four males and two females, aged between 43 and 45 took part with all but one left-handed female subject positioning the accelerometer on the right thigh. All subjects enjoyed average fitness levels and were recruited by convenience sampling.

Data was stored automatically on the accelerometer SD card as an `.ats` file. This was converted to the European Data Format (EDF) using the `AtsConvert` program that accompanied the accelerometer. EDF is a format designed for biosignal exchange. The resulting `.edf` file was viewable using the accompanying `EDFview` program. This was helpful in visualising the data and validating the recorded activity start and stop times.

Data was extracted from the `.edf` file using the `edf2ascii` executable which is downloadable from `http://www.teuniz.net/edf2ascii/`. The author wrote a Java class using the Weka API (`LabelGenerator.java`) that combines the raw data and the recorded times/activities to output labelled raw data in the Weka Toolkit `.arff` (Attribute-Relation File) format. This format was chosen to simplify the subsequent feature extraction stage which also avails of the same API. The above feature extraction process was repeated for each subject with their specific, recorded times being used to label the data.

Two subject recordings had to be repeated due to absent heart rate data. It is thought that this was due to the adhesive jelly on the Medick Healthcare electrodes being dry. Once replaced, the readings were fine.

3.3 Feature Extraction

The following features were extracted from each of the three accelerometer axes in the raw data. These were found to be useful in previous studies [12], [1], [9]:

- Mean
- Standard Deviation
- Energy
- Correlation

The mean value represents the DC component of the signal over the window time frame while the standard deviation allows for the discrimination of similar accelerometer values for differing activities. Energy is a measure of the intensity of movement and is calculated by taking the sum of the squared discrete FFT magnitudes and dividing by the window length. Correlation enables the differentiation of activities that involve transition, i.e. between walking, running and stair climbing. Mean heart rate was also calculated giving a total of thirteen features.

The author wrote a Java class (`FeatureExtractor.java`) to output the features as attributes to an `.arff` file. This class avails of a range of tools to extract the features including the Apache Commons Math Library (Energy), the Weka API (Correlation) and Schildt and Holmes' advanced Java primer [13] (Standard Deviation). The `.arff` format consists of two sections: a header and a data section. The header contains the name of the relation (ActivityRecognition), a list of the attributes (extracted features) and their types. The data is then arranged as a list of instances.

Previous studies have noted the effect of window size and overlap on the performance of a classifier [12], [11]. The `FeatureExtractor` class takes these parameters as arguments which allowed for easy comparison of a range of values. The raw data was sampled at a rate of 75 Hz. A window size of 256, therefore, equates to 3.4 seconds of activity. Using a window size of 256 and overlap of 50% reduced 110587 labelled readings to 768 usable instances.

4 Results

4.1 Classifier training and testing

Feature-extracted data from five subjects were combined in one `.arff` file and trained with 10-fold cross validation using the Weka toolkit. *k*-fold validation uses *k*-1 folds for training and the remaining one for testing. J48 and *k*-NN classifiers were subsequently evaluated for activity recognition with the latter providing a better accuracy score. A range of window sizes and overlaps were tested during the feature extraction stage with a window size of 256 and an overlap of 50% giving the best results. These are shown in Table 1.

Classifier	**Accuracy**
k-NN	88.04%
J48	80.23%

Table 1: Classifier Evaluation with 10-fold Validation

Classifiers were evaluated using the sixth subject's data set as an untrained test set and this showed a slight decrease in accuracy for the *k*-NN classifier with a significant decrease in accuracy for the J48 classifier. The results are shown in Table 2.

Classifier	Accuracy
k-NN	84.83%
J48	60.67%

Table 2: Classifier Evaluation with Untrained Test Set

Adding the sixth data set to the other five showed a slight increase in accuracy for both classifiers using 10-fold validation. The results are shown in Table 3.

Classifier	Accuracy
k-NN	90.07%
J48	83.95%

Table 3: Classifier Evaluation on Six Subject Dataset

4.2 Effects of removing features on classifier accuracy

The effects of removing specific features on classifier accuracy were evaluated and the results are shown in Table 4.

Removed Feature	*k*-NN	J48
Mean Heart Rate	-5.28%	-0.63%
Mean Acceleration	-3.16%	-2.43%
Standard Deviation	-2.42%	-2.64%
Correlation	+1.17%	-2.22%
Energy	-0.52%	+0.63%

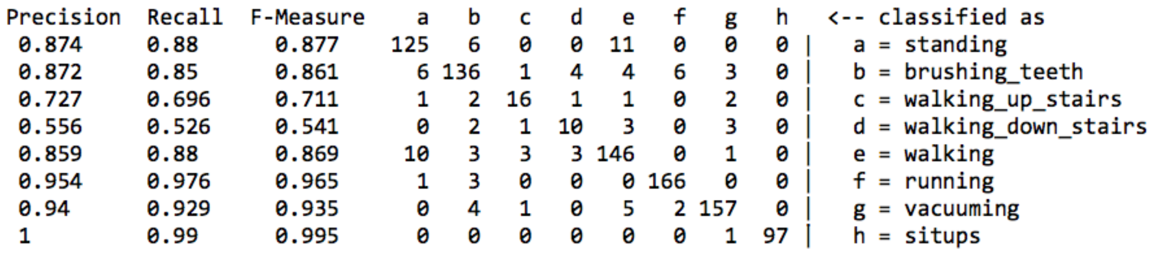
Table 4: Effect of removing specific features on classifier accuracy.

The attribute evaluator `CfsSubsetEval` was run on the data using the `BestFirst` search method. `CfsSubsetEval` considers each feature's specific predictive ability to produce a subset of features that will provide a similar degree of classifier accuracy. This process resulted in a reduction of thirteen features to seven (`meanHeartRate`, `meanZ`, `stdX`, `stdY`, `stdZ`, `energyY` and `energyZ`). The subsequent classifier evaluation using 10-fold validation produced accuracies of 80.78% for J48 and 89.33% for *k*-NN.

4.3 Specific activity recognition

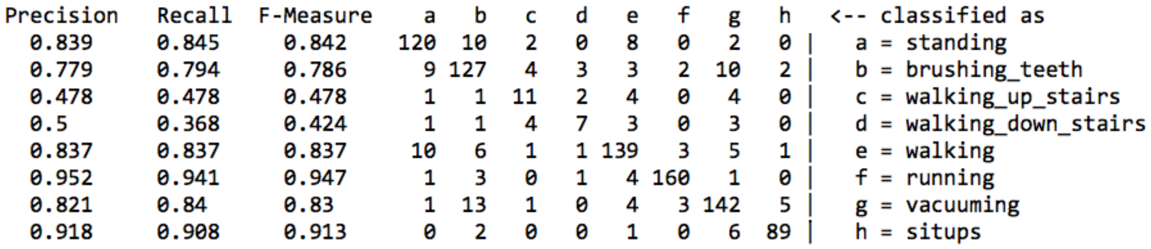
The Weka-generated confusion matrix for the *k*-NN classifier is shown in Fig 1. Precision, recall and F-measure values are also included. Precision reflects the number of correctly identified activities among those classified as the activity. Recall denotes the proportion of instances classified as activity x, among all instances that actually are activity x, and is related to the true positive value. The F-measure (2*Precision*Recall/(Precision+Recall)) combines precision and recall values in a hybrid measure of a test's accuracy [14]. The activities easiest to recognise are situps, running and vacuuming with precision values of 1.0, 0.954 and 0.94 respectively. Walking up and down stairs are shown to be difficult to recognise with the latter only having a precision value of 0.556.

The J48 classifier found situps and running the easiest to recognise with climbing and descending stairs again proving difficult to differentiate. The precision values for the latter two activities were just 0.478



```
Precision   Recall   F-Measure    a   b   c   d   e   f   g   h   <-- classified as
 0.874      0.88       0.877    125   6   0   0  11   0   0   0 |   a = standing
 0.872      0.85       0.861      6 136   1   4   4   6   3   0 |   b = brushing_teeth
 0.727      0.696      0.711      1   2  16   1   1   0   2   0 |   c = walking_up_stairs
 0.556      0.526      0.541      0   2   1  10   3   0   3   0 |   d = walking_down_stairs
 0.859      0.88       0.869     10   3   3   3 146   0   1   0 |   e = walking
 0.954      0.976      0.965      1   3   0   0   0 166   0   0 |   f = running
 0.94       0.929      0.935      0   4   1   0   5   2 157   0 |   g = vacuuming
 1          0.99       0.995      0   0   0   0   0   0   1  97 |   h = situps
```

Figure 1: Confusion matrix for the *k*-NN classifier.



```
Precision   Recall   F-Measure    a   b   c   d   e   f   g   h   <-- classified as
  0.839     0.845     0.842     120  10   2   0   8   0   2   0 |   a = standing
  0.779     0.794     0.786       9 127   4   3   3   2  10   2 |   b = brushing_teeth
  0.478     0.478     0.478       1   1  11   2   4   0   4   0 |   c = walking_up_stairs
  0.5       0.368     0.424       1   1   4   7   3   0   3   0 |   d = walking_down_stairs
  0.837     0.837     0.837      10   6   1   1 139   3   5   1 |   e = walking
  0.952     0.941     0.947       1   3   0   1   4 160   1   0 |   f = running
  0.821     0.84      0.83        1  13   1   0   4   3 142   5 |   g = vacuuming
  0.918     0.908     0.913       0   2   0   0   1   0   6  89 |   h = situps
```

Figure 2: Confusion matrix for the J48 classifier.

and 0.5 respectively. The confusion matrix for the J48 classifier is shown in Fig 2.

5 Analysis and discussion

Accuracy rates of 90.07% and 83.95% were achieved by *k*-NN ($k = 1$) and J48 classifiers respectively using a window of 256 and an overlap of 50%. This compares favourably with other studies for similar activities [9], [11] although there are also some studies in which decision tree classifiers perform better than *k*-NN [12]. Increasing the dataset size from five subjects to six increased the accuracy by 2.03% for *k*-NN and 3.72% for J48. It is envisaged, therefore, that a very large sample would produce high accuracy rates ($> 97\%$) similar to those achieved by [1] and [11].

Our final results were subject dependent, i.e. 10-fold validation was carried out on an amalgamation of the six subjects' data. The classifiers performed less well using subject independent test data and decreases in accuracy of 5.24% (*k*-NN) and 23.28% (J48) were noted. The latter substantial decrease may be due to the dataset being too small as this is known to cause instability in decision tree models [15]. Other studies have shown similar behaviour [1]. *k*-NN did show an accuracy of 84.83% for subject independent data, however, and this suggests that *k*-NN could be used for recognising certain activities without having the time and computational overhead of pre-training the classifier for a particular subject. Bao and Intille [12] indicate that such pre-trained classifiers could be used for real-time activity recognition on a range of emerging mobile devices. The MobHealth Java API [16] allows Bluetooth access to the Alive Technologies Accelerometer data stream. We used it to write software to check, in real time, that the heart rate was being recorded. It is feasible that this could be extended to include real-time classification using the Weka Tollkit API.

The complexity of building a decision tree is denoted by $O(mn \log n)$ where n is the number of instances and m is the number of features [17] and most similar studies examine the effect of removing features on accuracy. Pirttikangas et al [9] found the most important feature to be the mean acceleration and did not include mean heart rate or correlation in a subset of best features. Our study, on the other hand, showed mean heart rate to be the most significant feature for the *k*-NN classifier. Like Ravi *et al* [1], we found that energy was the least significant feature. Reducing thirteen features to seven resulted in a reduction in accuracy of only 0.74% for *k*-NN and 3.17% for J48.

The most difficult activities to identify were climbing and descending stairs. This is probably due to the fact that the time duration for these activities per subject (approximately 20 seconds) was less than the minute spent on the other six. Interestingly, Ravi *et al* [1], whose range of activities we chose for our study, also found these two activities hard to tell apart. The other six activities, including brushing teeth, returned a precision score of more than 87%. This would indicate that some upper body gestures can be identified by an accelerometer worn on the dominant thigh. The easiest activities to recognise in our study involved posture changes and included situps, running and vacuuming.

Tapia et al [18] studied the usefulness of heart rate data in differentiating the intensity of activities and found that adding heart rate data only increased subject-dependent recognition accuracy by 1.2%. Our study showed that heart rate data could improve k-NN accuracy by 5.28%. This may have been due to the fact that the subjects were of a similar age and fitness and that each subject carried out the activities in the same order. However, further study of heart rate data as a useful feature for activity recognition would be beneficial.

6 Conclusions and future work

We evaluated two classifers, k-NN and J48, for activity recognition using accelerometer and heart rate data gathered from six subjects and found that k-NN (k = 1) achieved a better overall accuracy score of 90.07%. We were able to reduce thirteen features to seven with minimal impact on classifier accuracy. Eight activities were observed with two (climbing and descending stairs) proving difficult to recognise. We successfully showed that it is possible to correctly identify 6 common activities using combined heart and accelerometer data from a sensor worn on the dominant thigh. Activities involving posture changes such as running, situps and vacuuming proved easiest to identify altough the high precision score for identifying brushing teeth (0.874) with k-NN shows that it is possible to accurately recognise predominantly upper body gestures using a thigh-worn accelerometer. A future study could examine differentiating similar activities such as brushing teeth and electric shaving.

Ravi *et al* [1] evaluated a range of base- and meta-level classifiers and found that combining classifiers using plurality voting provided the best accuracy. In preliminary tests, we found that combining classifiers improved on the accuracy of J48 by 3.72% but was 2.03% less accurate than using k-NN alone. Further investigation of this, including examining the computational overhead associated with it, could form part of any future study.

As mentioned above, it would be beneficial to further investigate the effectiveness of heart rate data as a useful feature in activity recognition, preferably with a larger subject sample. Furthermore, we would like to assess classifier performance in real-time using software incorporating the MobHealth Java API which was developed for use with the Alive Technologies Accelerometer.

References

[1] Nishkam Ravi, Nikhil Dadekar, Preetham Mysore, and Michael L. Littman. Activity recognition from accelerometer data. *American Association for Artificial Intelligence*, 2005.

[2] Venet Osmani, Sasitharan Balasubramaniam, and Dmitri Botvich. Human activity recognition in pervasive health-care: Supporting efficient remote collaboration. *J. Netw. Comput. Appl.*, 31(4):628–655, 2008.

[3] Nathalie Bricon-Souf and Conrad R. Newman. Context awareness in healthcare: A review. *International Journal of Medical Informatics*, 76:2–12, 2007.

[4] Jesus Favela, Monica Tentori, Luis A. Castro, Victor M. Gonzalez, Elisa B. Moran, and Ana I. Martínez-García. Activity recognition for context-aware hospital applications: issues and opportunities for the deployment of pervasive networks. *Mob. Netw. Appl.*, 12(2-3):155–171, 2007.

[5] J. Boyle, M. Karunanithi, T. Wark, W. Chan, and C. Colavitti. Quantifying functional mobility progress for chronic disease management. In *Engineering in Medicine and Biology Society, 2006. EMBS '06. 28th Annual International Conference of the IEEE*, pages 5916–5919, 2006.

[6] Erich P. Stuntebeck, II John S. Davis, Gregory D. Abowd, and Marion Blount. Healthsense: classification of health-related sensor data through user-assisted machine learning. In *HotMobile '08: Proceedings of the 9th workshop on Mobile computing systems and applications*, pages 1–5, New York, NY, USA, 2008. ACM.

[7] Yen-Ping Chen, Jhun-Ying Yang, Shun-Nan Liou, Gwo-Yun Lee, and Jeen-Shing Wang. Online classifier construction algorithm for human activity detection. *Applied Mathematics and Computation*, 205(2):849–860, 2008.

[8] Emmanuel Munguia Tapia. *Using Machine Learning for Real-time Activity Recognition and Estimation of Energy Expenditure*. PhD thesis, Massachusetts Institute of Technology, 2008.

[9] Pirttikangas S., Fujinami K., and Nakajima T. Feature selection and activity recognition from wearable sensors. In *International Symposium on Ubiquitous Computing Systems (UCS2006)*, 2006. International Symposium on Ubiquitous Computing Systems (UCS2006), Seoul, Korea, Oct. 11 - 13, 2006, pp. 516-527.

[10] Stephen R. Garner. Weka: The waikato environment for knowledge analysis. In *In Proc. of the New Zealand Computer Science Research Students Conference*, pages 57–64, 1995.

[11] Clemens Lombriser, Nagendra B. Bharatula, Daniel Roggen, and Gerhard Tröster. On-body activity recognition in a dynamic sensor network. In *BodyNets '07: Proceedings of the ICST 2nd international conference on Body area networks*, pages 1–6, ICST, Brussels, Belgium, Belgium, 2007. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[12] Ling Bao and Stephen S. Intille. Activity recognition from user-annotated acceleration data. In *Pervasive 2004*, pages 1–17. Springer, 2004.

[13] Herbert Schildt and James Holmes. *The Art of Java*. McGraw-Hill Osborne Media, 2003.

[14] Remco R. Bouckaert, Eibe Frank, Mark Hall, Richard Kirkby, Peter Reutemann, Alex Seewald, and David Scuse. *WEKA Manual for Version 3-6-0*. The University of Waikato, 2008.

[15] WekaDocs. Classification Trees, 2009. http://wekadocs.com/node/2 Accessed Online: 20/08/2009.

[16] MobHealth. MobHealth Java API for Accelerometer/Heart Rate Data Capture. 2008. http://sourceforge.net/projects/mobhealth.

[17] Ian H. Witten and Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques, Second Edition (Morgan Kaufmann Series in Data Management Systems)*. Morgan Kaufmann, June 2005.

[18] E.M. Tapia, S.S. Intille, W. Haskell, K. Larson, J. Wright, A. King, and R. Friedman. Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart rate monitor. pages 37–40, Oct. 2007.