A Method for Activity Recognition Partially Resilient on Mobile Device Orientation

Nikola Jajac, Bratislav Predic, Dragan Stojanovic
University of Nis, Faculty of Electronic Engineering
18000 Nis, Serbia
{nikola.jajac, bratislav.predic, dragan.stojanovic}@elfak.ni.ac.rs

ABSTRACT

This paper demonstrates a method for activity recognition partially resilient on mobile device orientation, by using data from a mobile phone embedded accelerometer. This method is partially resilient on mobile device orientation, in such a way that a mobile device can be rotated around only one axis for an arbitrary angle. The classifier for activity recognition is built using data from one default orientation. This method introduces a calibration phase in which the phone's orientation is determined. After that, accelerometer data is transformed into the default coordinate system and further processed. The solution is compared with the method that built a classifier using data from multiple orientations. Three classifiers were tested and a high accuracy of around 90% was achieved for all of them.

1. INTRODUCTION

The advance of low-cost and low-power sensors has led to their massive integration into modern mobile devices that have become powerful sensing platforms. By using data from sensors like accelerometers, gyroscopes, digital compasses, light sensors etc. it is possible to describe the context of a mobile device user in much more detail than by using only location data. A richer description of a user context provides better adaptation of mobile content, services and resources, enabling the user to stay focused on the task at hand.

An activity of the user represents an important aspect of a context, because it directly impacts user's ability to interact with the mobile device and applications. By employing the information about the current activity, the mobile device can adapt its interfaces, filter the content it provides, or perform a specified action, to support the user in the best possible way.

The greatest possibilities for application of activity recognition systems lay in the healthcare domain. For example, such systems can be used for elderly care support or for long-term health/fitness monitoring [1]. Current methods for tracking activities, like paying a trained observer or relying on self-reporting are time and resource consuming tasks, and are error prone. An automatic system for recognizing activities could help reduce errors that arise from previously mentioned methods. Also, such system enables users to go about their daily routines, while the data collection and processing are done in the background, and do not interfere with their current activities.

In recent years a lot of work has been done on activity recognition from accelerometer data. Since an accelerometer is a standard part of modern mobile devices (like mobile phones and tablets), they can be used in activity recognition. An advantage in using these devices is that they are already commonly used by a lot of people that would not have to wear an additional device to perform activity recognition, which greatly increases the acceptance of such a system.

This paper focuses on recognizing activities from accelerometer data processed at a mobile phone. Activity recognition is formulated as a classification problem. This paper considers the healthcare domain for activity recognition system application. For this reason it is important to recognize physical activities, such as: walking, running, walking up/down stairs etc. Examples of possible users in this domain are the elderly and persons with certain disabilities. It can be assumed that such users keep their mobile phone relatively (but not completely) fixed. In the paper it is assumed that the phone position is fixed, but that the phone orientation is only partially fixed. The orientation is partially fixed in such a way that the axis that is perpendicular to the phone's screen is parallel to the ground and the phone can be rotated freely around that axis. For experiments in this paper, the mobile phone was worn in the right front pants pocket (as one of the places where people usually carry mobile phones) and the screen of the phone was facing the user. The orientation where the bottom side of the phone is facing the ground is considered the default orientation. First, three classifiers were built using data from multiple orientations. These classifiers were tested using data from the same orientations. Then, classifiers were built again, but using data from the default orientation only. These new classifiers were tested using data from multiple orientations, transformed into the default coordinate system prior to testing, according to the method proposed in this paper. After that, classifier performances from these two tests were compared.

Using results from previously mentioned tests, we created an application for activity recognition that runs on a mobile device in real time and tested the impact that data transformation has on the performance of such an application, specifically in terms of processor load. The processor load is important because if such application is to be accepted by users, it must not significantly decrease battery life, or the performance of the device in everyday tasks.

The proposed method represents only an intermediate step in the development of a method for activity recognition with no restrictions in mobile device orientation.

The rest of the paper is structured as follows: section 2 provides an overview of related work on activity recognition. Section 3 describes the process of accelerometer data collection. In sections 4 and 5, mixed orientation data and data reorientation processing approach, for activity recognition resilient on mobile device orientation, are presented and evaluated. Section 6 presents an evaluation of how much the activity recognition application and data transformation phase participate in the processor load. Section 7 gives the conclusions about the paper and outlines plans for future work.

2. RELATED WORK

In recent years there has been a lot of research related to recognizing activities from accelerometer data. In [2] authors used data from 5 biaxial accelerometers worn simultaneously on different parts of the body. Used accelerometers could detect acceleration up to ±10G. Accelerometers were mounted onto hoarder boards and firmly attached to different body parts. Data was collected from 20 subjects performing various everyday tasks without researcher supervision. The following features were computed on sliding windows of accelerometer data: mean, energy, frequency-domain entropy and correlation. A number of classifiers were trained and tested with the calculated data, where decision trees showed the best result, recognizing activities with an accuracy of 84%.

Ravi et al. in [3] attempted to perform activity recognition using a single triaxial accelerometer worn near the pelvic region. Data was collected by 2 subjects performing 8 different activities. Similarly to [2] the features were computed using the sliding windows technique. Four features were extracted: mean, standard deviation, energy and correlation. Extracted features were used to train and test 5 base-level classifiers, and in addition to that, 5 meta-level classifiers. Authors concluded that meta-level classifiers in general outperform base-level classifiers and that plurality voting, which combines multiple base-level classifiers, shows the best results. The authors also showed that out of the used features, energy is the least significant one, and that there is no significant change in accuracy when this feature is avoided.

Kwapisz et al. in [4] tried to recognize activities by using data from a single acceleration sensor, but they used data from an acceleration sensor embedded into a standard mobile phone. These accelerometers typically detect acceleration up to ±2G along three axes. Their research methodology follows the one in [2, 3]. The authors collected data from 29 subjects, extracted 6 basic features and tested 3 classifiers, where multilayer perceptrons showed the best result, recognizing activities with an accuracy of 91.7%. The authors showed that activity recognition can be performed successfully by using acceleration data from a mobile phone.

The unifying fact for papers [2 - 4], no matter if one or more accelerometers are used, is that the position and the orientation of the accelerometer is fixed while performing all of the examined activities. This fact can be probably expected in case of specialized devices as in [2, 3]. In case of using a standard mobile phone as in [4], the method puts strains on how someone carries a mobile device, which could decrease acceptability of such a system.

Sun et al. in [5] tried to recognize activities by using acceleration data from a mobile phone, in a setting where the position and the orientation of the phone vary. They restrict their hypothesis space to 6 possible positions (6 pockets) and 4 orientations of the mobile phone. The data from all position and orientation combinations were collected. The authors added acceleration magnitude at each sample, as an additional sensor reading dimension. By using collected data, several features were calculated: mean, variance, correlation, FFT energy and frequency-domain entropy. Calculated features were used to train and test SVM (Support Vector Machine) models. Generated SVM model recognizes activities with an accuracy of 93.1% throughout all tested positions and orientations.

Thiemjarus in [6] applied a different approach. Accelerometer was mounted on a belt-clip device which was worn by test subjects in a fixed position on a body, but which could be mounted in 4 different orientations. Data was collected by 13 subjects that performed a routine comprised of 6 activities. The first step in data analysis was device orientation detection. The orientation detection was also formulated as a classification problem. The features used for orientation detection were mean along Y and Z axis. Orientation detection was performed for an activity routine performance, which contains approximately 5 seconds of data for each tested activity, while activity recognition was window-based. The second step was signal transformation using the appropriate transformation matrix, and the third step was activity recognition itself. Author achieved a subject-independent classification accuracy of 90.9%.

A possible problem with the approach in [6] is that orientation detection is done on a data set that includes information about all tested activities. The open issue is how this approach can be applied in a real world scenario, when the user does not perform a specified activity routine. A unifying fact for [5, 6] is that the hypothesis space is limited to a number of orientations. Again, the open issue is how such a system would perform when given data from an unknown orientation. In this paper we propose a method that only partially limits the device orientation, in a way that the device can be rotated only around one axis, but for an arbitrary angle. This arbitrary rotation practically creates an infinite number of possible orientations, in contrast of previous approaches, which are all limited to a certain number of orientations. To achieve this, a calibration phase which precedes activity recognition is introduced.

3. ACCELEROMETER DATA COLLECTION

As a test device, a smartphone Samsung I9001 Galaxy S Plus which runs on Android operating system version 2.3.5 is used. The accelerometer embedded in this phone detects acceleration up to ±2G. Data from the accelerometer has three attributes: acceleration along X, Y and Z axis, represented by floating point values. Sampling rate for the accelerometer was set to *SENSOR_DELAY_FASTEST* to achieve the highest possible accuracy.

An application for recording data from the accelerometer has been developed. Data was collected by several test users. The recording process is as follows: while standing still, test user selects the activity he is going to perform and starts the recording. After that the user has ten seconds to place the phone in the pocket in the desired orientation. After ten seconds a beep sound is played and for two seconds the gravity vector is extracted from accelerometer data. After two seconds another beep sound is played and an average value for the gravity vector is saved to a file. To extract the gravity vector from accelerometer data a simple low-pass filter is used. The user can then start to perform the specified activity. Another 2 seconds after the second beep, the application starts to record acceleration data to another file. After finishing with the activity the user stops the recording.

Data was collected while performing 6 different activities:

- Standing
- Walking
- Running
- Walking up stairs
- Walking down stairs
- Sitting.

For each activity data was collected for the default orientation and for 3-4 other orientations, depending on the activity. Some of the non-default orientations matched between activities and some did not. To minimize mislabeling a portion of data was removed from the beginning and the end of each recording.

4. A MIXED ORIENTATION DATA APPROACH

The first approach to the free orientation problem we test is building a classifier from data collected from all orientations, very similar to [5]. In this approach, the next step after data collection is feature extraction. The features were extracted from accelerometer data using a window size of 512 samples with 256 samples overlapping between consecutive windows. Three features were extracted from each of the three axes, giving a total of nine attributes for building a classifier. The features extracted were:

- Mean
- Standard deviation
- Correlation.

We specified these features, that are calculated using data in time domain, because we apply the activity recognition system in real-time locally on a device. For this reason, the features should be relatively simple to compute, to reduce power consumption and processor load. The selected features do not require signal representation in the frequency domain, and thus can be computed relatively fast. Also, the mean is used in the standard deviation calculation, and the mean and the standard deviation are used in correlation calculation, which further increases computation speed.

Extracted features were used to train and test 3 classifiers available in the WEKA Data Mining Toolkit [7], which are commonly used in activity recognition [2 – 4, 6]:

- C4.5 decision tree
- Naïve Bayes
- K-nearest neighbors.

We are mainly interested in the performance of the decision tree classifier, since it requires the least amount of computation in the classification phase, which is important when the system is applied in a real-time locally on a device. For the testing we used 10-fold cross validation, and the results are shown in Table 1. All of the tested classifiers showed excellent results in recognizing activities, which is consistent with previous work [5]. The results are slightly better than in [5] which can be probably contributed to a specific data set and the fact that the data was collected by a single test user.

Table 1. Classifier accuracy – Mixed orientation data approach

Classifier	Accuracy (in %)
C4.5 Decision Tree	98.8
Naïve Bayes	99.5
K-nearest neighbors	99.8

5. A DATA REORIENTATION PREPROCESSING APPROACH

The second approach to the free orientation problem is based on building a classifier from data collected in the default orientation. In the classifying phase, transformation of data collected in various orientations into the default coordinate system is performed, prior to the feature extraction and classification.

The classifiers for testing are built in the same way as in the mixed orientation data approach, but now only data collected from the default orientation is used. To get the most precise results, it is important to test the classifier with data from all available orientations, which includes also data from the default orientation. For this reason, a portion of data from the default orientation was omitted in classifier building, and was used later in classifier testing. In this way we avoid overfitting, which can happen when the same data is used for training and testing. No data transformation was done on data used for building a classifier.

As previously mentioned, in this paper we assume that the phone can be rotated only around the Z axis, and consequently, we assume that there is no change in the acceleration along the Z axis when performing some activity in the default and non-default position. This means that we do not need to transform the Z coordinate, just X and Y coordinates. To achieve that, we use a rotation matrix for rotation around the Z axis for an angle θ. The rotation matrix is given in (1). To calculate angle θ we use the information about gravity vectors.

$$R = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{1}$$

By using the gravity vectors from all of the recordings in the default orientation, we computed an average gravity vector for the default orientation (gravity vector is defined as data from the accelerometer, by three attributes: X, Y and Z). Since the phone's screen is facing the user, the Z axis is practically parallel to the ground, so we take into account only the X and Y components of the vector.

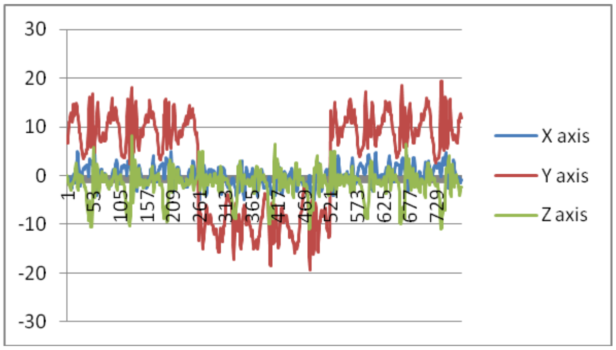

Figure 1. The accelerometer data collected while walking

In the classifying phase, we firstly calculate the difference between the angle of the average gravity vector in the default orientation and the angle of the gravity vector for the current orientation, and this difference is angle θ we need to transform accelerometer data into the default coordinate system. In the next step, each sample from the accelerometer is transformed into the default coordinate system using the rotation matrix given in (1). Figure 1 shows data from the accelerometer while walking. The vertical axis represents the acceleration in m/s^2. Samples from the acceleration sensor are represented along the horizontal axis. Samples 1-250 represent data when the phone is in the default orientation, samples 251-500 represent data when the phone is in a non-default orientation and samples 501-750 represent the same data as samples 251-500, but transformed into the default coordinate system. Transformed data is then used to extract features in the same way as in the mixed orientation data approach. For testing we used data from non-default orientations, and data from default orientation. Data from default orientation was treated the same way as data from non-default orientations, and was transformed accordingly to its gravity vector. We tested the same 3 classifiers as in the mixed orientation data approach, and the results are shown in Table 2.

Table 2. Classifier accuracy – Data reorientation preprocessing approach

Classifier	Accuracy (in %)
C4.5 Decision Tree	86.5
Naïve Bayes	91.5
K-nearest neighbors	95.0

The results obtained are lower than in the mixed orientation data approach, but are still above 90% threshold, except for the decision tree classifier. For this reason we analyze the decision tree classifier further. The confusion matrix for the decision tree classifier is shown in Figure 2.

```
 a   b   c   d   e   f   <-- classified as
89   0   0   0   0  14 |   a = RUNNING
 6  31   2   2  32   1 |   b = SITTING
 0   0  36   0   0  10 |   c = STAIRS_DOWN
 0   0   6  39   0   5 |   d = STAIRS_UP
 0   0   0   0  55   0 |   e = STANDING
 0   0   0   0   0 251 |   f = WALKING
```

Figure 2. Confusion matrix for the decision tree classifier

It can be seen that a lot of instances that represent sitting are classified as standing, which is not intuitive, because these two activities should be easy to distinguish. This is a consequence of how WEKA generates the decision tree. When we look at the generated decision tree shown in Figure 3, we can see that sitting and standing are distinguished by mean value along the Z axis (MeanZ). WEKA makes the split on value -9.804189 which is the maximum value for MeanZ for sitting. It can be expected that the maximum value for MeanZ will vary for different recordings, since it can't be expected from someone to sit in exactly the same way every time. Since the minimum value for MeanZ for standing is -1.07289 the split should be done on the value -5.4385, which is halfway between the maximum for sitting and the minimum for standing. This would generate a much more robust tree with higher accuracy. Such decision tree would correctly classify all of the sitting instances, previously misclassified as standing. Consequently, decision tree accuracy increases to 92%.

```
StandardDeviationY <= 0.574771
|   MeanZ <= -9.804189: SITTING (37.0)
|   MeanZ > -9.804189: STANDING (34.0)
StandardDeviationY > 0.574771
|   CorrelationYZ <= -0.215697: RUNNING (55.0)
|   CorrelationYZ > -0.215697
|   |   MeanZ <= -2.416916
|   |   |   CorrelationYZ <= 0.206422: STAIRS_DOWN (18.0)
|   |   |   CorrelationYZ > 0.206422: STAIRS_UP (18.0)
|   |   MeanZ > -2.416916: WALKING (115.0)
```

Figure 3. Generated decision tree

To demonstrate the benefits of this approach, the same classifier was tested with the same test data, but this time no data transformation was performed prior to classification. Based on the results shown in Table 3, it can be concluded that a classifier built using data from only one orientation, cannot classify instances from other orientations with a high success rate. With the reorientation preprocessing included, the classification accuracy results (shown in Table 2) increase significantly, which demonstrates the advantage of this approach, compared to the one assuming a fixed orientation at all times.

Table 3. Classifier accuracy – Single orientation classifiers without reorientation

Classifier	Accuracy (in %)
C4.5 Decision Tree	54.4
Naïve Bayes	48.5
K-nearest neighbors	58.8

To compare the data reorientation preprocessing approach against the mixed orientation data approach, the classifier built with the mixed orientation data approach was retested with a data set consisting of classifier training data, as well as data including orientations that were not used in classifier training. Accuracy of the mixed orientation data approach, when handling data including unknown orientations, can be analyzed in this manner. Evaluation results are shown in Table 4 and are comparable to the results achieved with the data reorientation preprocessing approach. It can be concluded that the mixed orientation data approach can also handle data from previously unknown orientations, but with a decrease in accuracy compared to the data reorientation preprocessing approach.

Table 4. Classifier accuracy - Mixed orientation data approach with unknown orientations

Classifier	Accuracy (in %)
C4.5 Decision Tree	85.0
Naïve Bayes	83.8
K-nearest neighbors	88.5

6. EVALUATION OF MOBILE CPU LOAD

Using results from the previous test we built an application for activity recognition in real-time locally on the device. The application implements the data reorientation preprocessing approach and uses a prepared decision tree as a classifier. To test how much the application and data transformation phase participate in the processor load we used DDMS (Dalvik Debug Monitor Service).

We ran DDMS for thirty seconds while the application was active on the device. The results are shown in Figure 4. The first line of the figure represents the main thread of the application in which all of the application processing is done. Different methods are represented with different colors on the timeline. This figure focuses on the period between two feature extractions. The boxes marked with the number 1 represent feature calculation. We notice three color groups, where each one, looking from the left to the right, represents calculation of mean, standard deviation and correlation. The box marked with the number 2 represents data transformation. We can see how data transformation is performed whenever a new sample is read from the accelerometer and that feature extraction is performed only when the window shifts for the specified number of samples.

The Figure 5 shows the timeline again but this time in more details. Similar to Figure 4, the box marked with the number 1 represents mean calculation (green color on the timeline), and the box marked with the number 2 represents data transformation. It can be seen that data transformation requires a small portion of processor time, compared to processor time required to calculate just one feature, but is performed more often.

The entry point of the application is the *onSensorChanged* function, which is called whenever a new sample from the accelerometer is read. This function encapsulates all of the application processing and participates in the processor load with 22.9%. Also, the function *transformData*, which encapsulates all of the data transformation, participates in the processor load with 1.3%. It can be concluded that although data transformation is performed more often than feature extraction, it doesn't increase processor load significantly.

Figure 4. Result of the DDMS tool applied on the activity recognition application

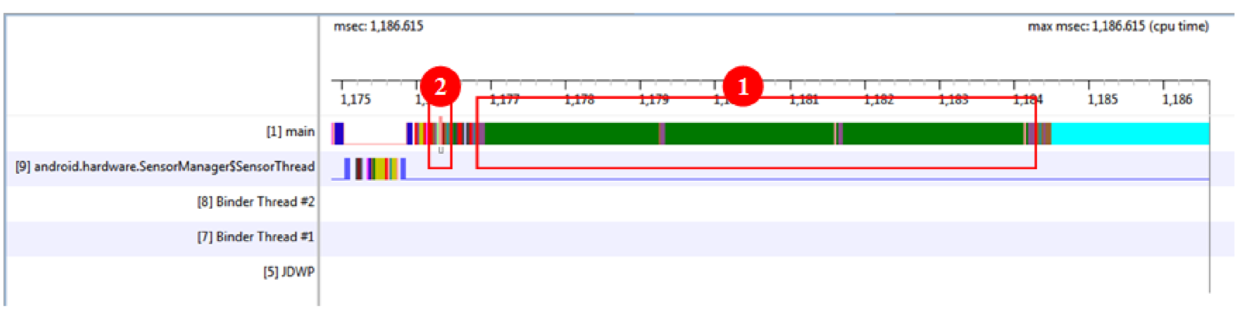

Figure 5. Result of the DDMS tool in more detail

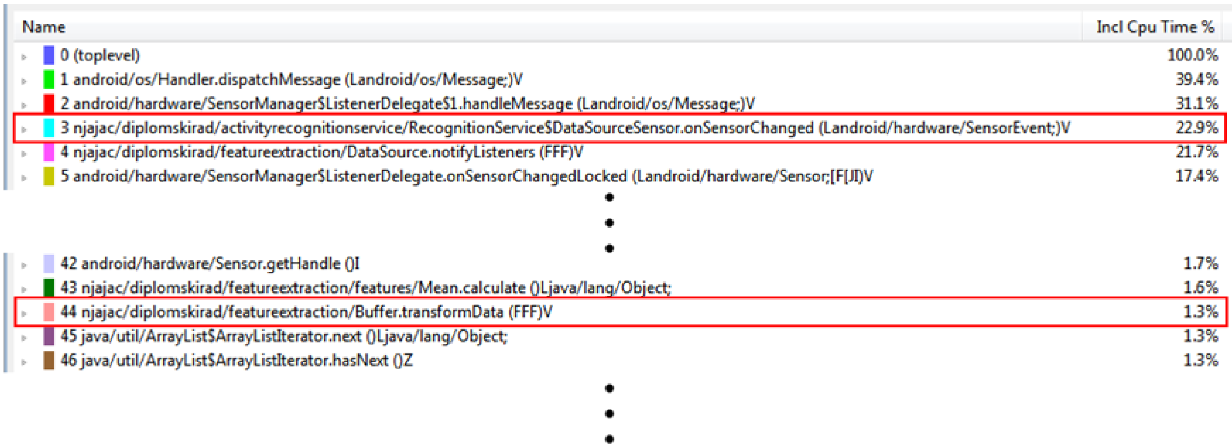

Figure 6. Participation of individual functions in processor load

7. CONCLUSION AND FUTURE WORK

In this paper a method for orientation independent activity recognition from accelerometer data is described. An accelerometer embedded in a mobile phone was used. Orientation of the phone was partially fixed in such a way, that the phone could be rotated only around one axis, but the angle of orientation was arbitrary. To determine the angle of rotation a calibration phase was introduced, in which the user has to stand still for a couple of seconds with the phone placed in the desired orientation. In this period the gravity vector is extracted and the difference between the angle of that gravity vector and the angle of the average gravity vector in the default orientation is calculated. This difference is the angle of rotation of the phone. After that the user can start to perform activities freely. The data from the accelerometer is transformed into the default coordinate system, the features are extracted and the activity recognition is performed.

This method showed slightly reduced accuracy compared to the method when a classifier is built from data collected from various orientations, when data from the predefined orientations only is considered, but the results are still above the threshold (accuracy above 90%). When data from not predefined orientations is considered as well, the proposed method demonstrates increased accuracy. The most significant advantage of this method is that it requires data collection from only one orientation, so less data is required for training. Also it makes no assumption on the orientation in the classifying phase; there are no predefined orientations, so the system will work with data from any orientation. The drawback is the existence of the calibration phase, so the process in not fully transparent. This probably makes this method unusable in some areas of application, like in elderly care for example, but in others, like in fitness monitoring, we believe that is acceptable because the calibration phase is very short and requires very little effort from the user. In return the user can place the phone in any orientation.

In this paper the phone orientation is assumed to be partially fixed, which still limits the user to a certain degree. Demonstrated method represents only an intermediate step in development of a method which would acquire a full three-dimensional orientation in the calibration phase and impose no limits in the phone's orientation.

8. ACKNOWLEDGMENTS

This paper was realized as a part of the project "Studying climate change and its influence on the environment: impacts, adaptation and mitigation" (43007) financed by the Ministry of Education, Science and Technological Development of the Republic of Serbia within the framework of integrated and interdisciplinary research for the period 2011-2014.

9. REFERENCES

[1] J. Lester, T. Choudhury, G. Booriello. A Practical Approach to Recognizing Physical Activities. *Proceedings of the 4th International Conference on Pervasive Computing*, 2006, pp. 1-16.

[2] L. Bao, S. S. Intille. Activity recognition from user-annotated acceleration data. *Proceecedings of the 2nd International Conference on Pervasive Computing*, 2004, pp. 1-17.

[3] N. Ravi, N. Dandekar, P. Mysore, M. L. Littman. Activity recognition from accelerometer data. *Proceedings of the Seventeenth Conference on Innovative Applications of Artificial Intelligence (IAAI-05)*, 2005, pp. 1541-1546.

[4] J. R. Kwapisz, G. M. Weiss, S. A. Moore. Activity Recognition using Cell Phone Accelerometers. *ACM SIGKDD Explorations Newsletter*, 2010, vol. 12, issue 2, pp. 74-82.

[5] L. Sun, D. Zhang, B. Li, B. Guo, S. Li. Activity Recognition on an Accelerometer Embedded Mobile Phone with Varying Positions and Orientations. *Proceedings of the 7th International Conference on Ubiquitous Intelligence and Computing*, 2010, pp. 548-562.

[6] S. Thiemjarus. A Device-Orientation Independent Method for Activity Recognition. *International Conference on Body Sensor Networks*, 2010, pp. 19 - 23.

[7] I. H. Witte, E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques*, 3rd ed. Morgan Kaufmann, 2011