# The application of machine learning in multi sensor data fusion for activity recognition in mobile device space

PLEASE CITE THE PUBLISHED VERSION

http://dx.doi.org/10.1117/12.2177115

PUBLISHER

VERSION

VoR (Version of Record)

REPOSITORY RECORD

Al-Marhoubi, Asmaa H.A., Sara Saravi, and Eran A. Edirisinghe. 2019. “The Application of Machine Learning in Multi Sensor Data Fusion for Activity Recognition in Mobile Device Space”. figshare. https://hdl.handle.net/2134/21854.

# The Application of Machine Learning in Multi Sensor Data Fusion for Activity Recognition in Mobile Device Space

Asmaa H. Marhoubi, Sara Saravi, Eran A. Edirisinghe
Computer Science Department, Loughborough University, United Kingdom

## ABSTRACT

The present generation of mobile handheld devices comes equipped with a large number of sensors. The key sensors include the Ambient Light Sensor, Proximity Sensor, Gyroscope, Compass and the Accelerometer. Many mobile applications are driven based on the readings obtained from either one or two of these sensors. However the presence of multiple-sensors will enable the determination of more detailed activities that are carried out by the user of a mobile device, thus enabling smarter mobile applications to be developed that responds more appropriately to user behavior and device usage. In the proposed research we use recent advances in machine learning to fuse together the data obtained from all key sensors of a mobile device. We investigate the possible use of single and ensemble classifier based approaches to identify a mobile device's behavior in the space it is present. Feature selection algorithms are used to remove non-discriminant features that often lead to poor classifier performance. As the sensor readings are noisy and include a significant proportion of missing values and outliers, we use machine learning based approaches to clean the raw data obtained from the sensors, before use. Based on selected practical case studies, we demonstrate the ability to accurately recognize device behavior based on multi-sensor data fusion.

**Keywords:** Activity Recognition, Mobile phone, Mobile Sensors, Gyroscope, Accelerometer, Multi Sensor.

## 1. INTRODUCTION

The vast majority of the current handheld mobile devices come with a large number of in-built sensors. Amongst these sensors the GPS sensor, ambient light sensor (ALS), the gyroscope, the accelerometer, the compass, the rotation sensor and the proximity sensor are considered the key sensors. Mobile device sensors are mainly categorized into three main groups, namely, motion sensors, environmental sensors and position sensors. Motion sensors are used to measure rotational and acceleration forces along three axes of the devise (x, y and z – see section 2); environmental sensors measure environmental parameters such as the level of illumination intensity, presence of other objects closer to the device etc., and position sensors measure the physical position of the handheld device such as the GPS location, orientation with respect to the North etc. Some are software-based sensors while others are hardware-based sensors. Hardware based sensors are physical parts that are positioned within the handheld device. They perform the measurements directly and gather the readings. Accelerometer and Gyroscope are examples of it. Software based sensors rely on one or more hardware based sensors to derive their readings. The rotation sensor is an example of this type.

The presence of the above sensors allows mobile applications to be built and to be operated in a more adaptive, user friendly manner, largely improving the user's quality of experience. A common and straightforward application is the use of the ALS to identify the screen surface illumination and using it to control the backlight in order to make the screen appear brighter or lighter. This not only improves user experience but also extends the devices battery life. Many commercial systems exists that best utilizes individual sensors to drive mobile applications more effectively. A detailed literature review carried out by us revealed that there have been a few attempts by the research community to combine the separate sub readings of an individual device for activity recognition. In [4] Kwapisz et. al., proposed the use of three classification approaches to fuse together the x, y and z-accelerometer readings to identify activities such as walking, jogging, sitting, standing, walking up or down the stairs etc. In [3] Hinckley et. al. integrated a number of sensors that demonstrate several scenarios of usage of a device such as changing the orientation between landscape and portrait modes, when the device is held like a phone to enable recording memos and when the user picks up the device, for it to automatically power up. However this paper did not propose any approaches to fuse together different sensor readings nor proposed any method for modelling behavior based on a single senor reading. Based on this initial exploratory work

Image Sensing Technologies: Materials, Devices, Systems, and Applications II,
edited by Nibir K. Dhar, Achyut K. Dutta, Proc. of SPIE Vol. 9481, 94810G
© 2015 SPIE · CCC code: 0277-786X/15/$18 · doi: 10.1117/12.2177115

of [3], some other researchers have explored the different individual sensors on mobile devices, where they for example demonstrated the use of the tilt sensor for tilting, information selection and browsing on a mobile device.

The above literature review has revealed the limited scope of research carried out in the mobile device domain for multi-sensor data fusion focused at more detailed activity recognition of a mobile device. We believe that the sensor data can be efficiently fused and used in predicting more detailed activities such as a user walking with a mobile phone but also holding the phone still in order to enable viewing something displayed of a screen, a person sitting and just flipping the phone and hence not bothered about viewing the content of the display. The identification of such activities will require at least the fusion of data gathered via two sensors. There may also be many different data classification approaches in machine learning that may have different accuracies of performance in their ability to correctly classify events. In particular it will be useful to investigate the use of the more recent advancements in machine learning algorithms such as ensemble classification approaches such as bagging, random forests etc., as against using the more traditional neural network (single classifier) based approaches. In this paper we therefore attempt to bridge the above research gap by using the advanced machine learning algorithms for event classification related to the usage of mobile phones based on multiple sensor data fusion. As a proof of concept we have selected some simple use cases that can be identified based on the fusion of the readings of two particular sensors, namely, the gyroscope and the accelerometer [5]. We later infer that the proposed concepts and the framework can be extended for fusing data from a larger number of sensors that can result on identifying more complicated scenarios.

For clarity of presentation this paper is divided into several sections. Apart from this section which provided an insight into the problem domain and identified relevant research gaps, section-2 introduces the reader to the research background that includes information on sensors to be utilized, some key definitions to be used and the machine learning approaches to be investigated. Section-3 introduces the reader to the experiments carried out and information on data collection/capture. Section-4 analyses the experimental results and evaluates the performance of different classification algorithms. Finally section-5 concludes with an insight to future research that can originate from this preliminary study.

## 2. BACKGROUND

### 2.1 The Co-ordinate System

The de-facto coordinate system used in relation to a mobile handheld device uses six different degrees of freedom. This means that the device is free to move forward/backward, up/down and left/right. These can be used to represent the device's pitch "Tilting forward and backwards", roll "Tilting side to side" and yaw "Turning left to right" as illustrated in the figure 1 [7].

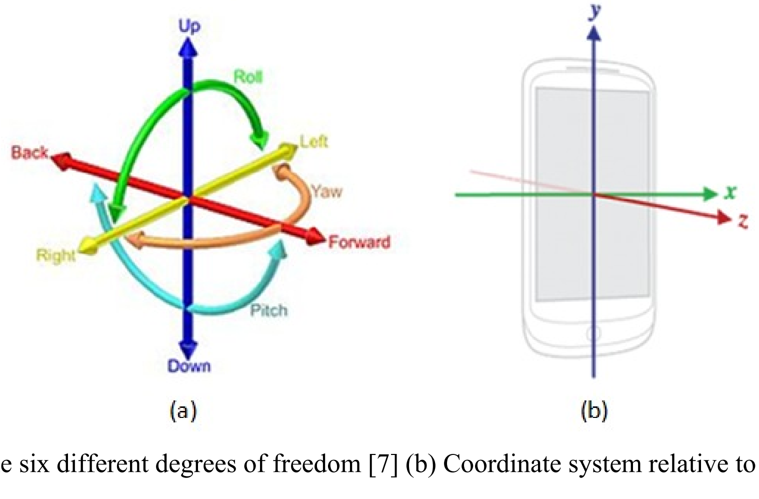


Figure 1. (a) The six different degrees of freedom [7] (b) Coordinate system relative to a device [5]

The coordinate system used by the sensors is defined with respect to the device's screen as illustrated in the figure 2. It is noted that even if the physical orientation of the device changes, the directions of the x, y and z axes are not changed w.r.t. the devices screen. It is also important to note that a typical mobile device's default orientation is 'portrait' and in most cases tablets have the 'landscape' orientations as its default. It is vital to know that a sensor's coordinate system relies on the default orientation of the device in which it's being used [5].

### 2.1 Sensors

Within the present research context we make use of two different sensors which are amongst the most widely used within a mobile device, namely, the Accelerometer and the Gyroscope. Detecting the motion of a handheld device is done using the Accelerometer. It measures the linear acceleration of the performed move. The Gyroscope measures the angular rotational velocity. The data captured from the Accelerometer and the Gyroscope is used within software based Rotation Sensors for tracking any twists or rotations of the phone. It is noted that both the Accelerometer and the Gyroscope measure the rates of change.

In a typical mobile device the accelerometer and the gyroscope produces three readings each, being the readings in the x, y and z directions w.r.t. the device as depicted in figure 2. For the experiments carried out within the context of the proposed research the corresponding six readings were utilized.

### 2.2 Modelling device behavior

In the proposed research we use advance machine learning algorithms to model the mobile device behavior and classify the usage into specific events. In particular we make use of the popular Neural Network based classifier, Multi-layer Perceptron [8] and the Ensemble Classifiers, Random Forest [6] and Bagging [1] Ensemble Learning [8] is an approach that uses different classification techniques to build up a model. The data is divided into multiple sets depending on the classifier used. Each set will go through the learning/ training process and a model will be generated. In other words, each set will generate a model based on the base classifier. The model will go through a testing option, which is in the proposed work, is ten-fold Cross Validation. The best model will be chosen based on a voting/ average technique.

The following section presents the proposed methodology including the details of experiments conducted and results captured.

## 3. EXPERIMENTS & DATA CAPTURE

A number of experiments were designed with regards to a specified set of scenarios of mobile phone usage. Experiments representing each scenario were repeated ten times to gather data from the two sensors. The data gathered were stored on the mobile device and later transferred to a computer for further analysis using machine learning algorithms. The following sections describe each stage in more detail.

### 3.1 The Experiments

Seven different scenarios of mobile phone were investigated. These include the user siting down and holding the phone stationary as if to pay attention to the screen content, the user sitting down and shaking the phone randomly with no intention to read the content, user holding the phone paying attention to the displayed content but walk with it, the user walking and shaking the phone with no intention of paying attention to the content, the user running with the phone but holding the phone in hand tightly, the user running with the phone but also and shaking it and finally standing still, holding the frame and reading the display content. The above scenarios were conducted in succession in the above order. Before the first experiment, the user originates data collection and once the last experiment is over the user stops data collection.

### 2.2 Data Capture

For the above experiment, a Samsung galaxy s4 mobile phone that consists of the two selected sensors as in-built, was used. An android based mobile application was developed, which enabled the capturing multiple sensor values. Even though this experiment focuses on the Accelerometer and Gyroscope sensors only, the readings from a number of other in-built sensors were also captured for future research. The captured data is stored in a .CSV file and later transferred to a PC for further analysis. Note that each of the two sensors has three values each, being their corresponding readings in the X, Y and Z directions.

## 4. RESULTS & ANALYSIS

Figure 2 illustrates the gyroscope and accelerometer readings captured. A detail visual analysis of figure 2 reveals that the relative magnitudes of the three sub readings of the accelerometer and the gyroscope could be used for event classification. For example from point 1 to point 97, the user was sitting holding the phone still. This is indicated by the very low gyroscope values. Only the X and Z components of the accelerometer indicates any significant value. The events that occur between the points marked 97 and 193, 449 and 543, 641 and 737, represent the scenarios where the phone was shaken. This is indicated by the high values of the accelerometer readings in all directions and somewhat strong readings obtained from the gyroscope. Between points indicated by 385 to 449, the user was walking indicating somewhat changing/noisy nature of the accelerometer and gyroscope readings.

However it is also clear that coming up with manually defined thresholds for the six sensor readings that will allow the classification of the seven events will be still a tedious task. Further the raw signals captured by the sensors are noisy and will therefore have to be cleaned prior to further analysis.

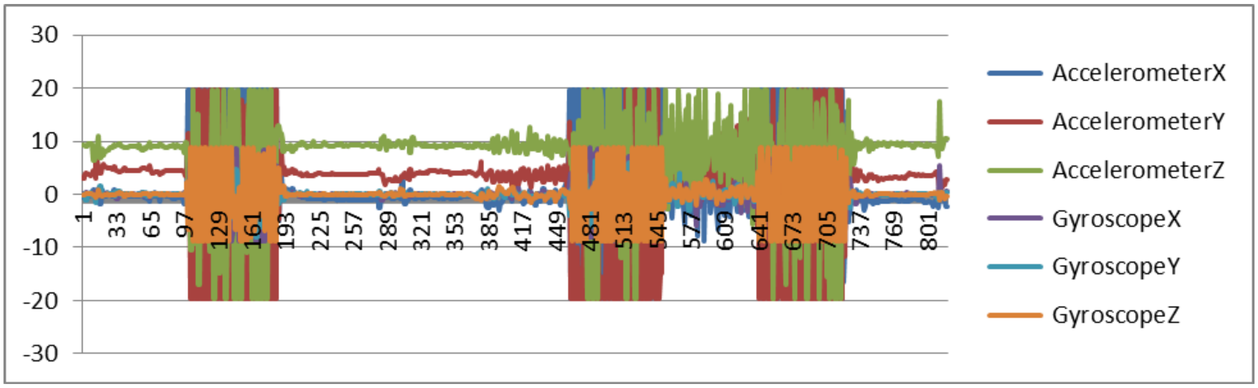
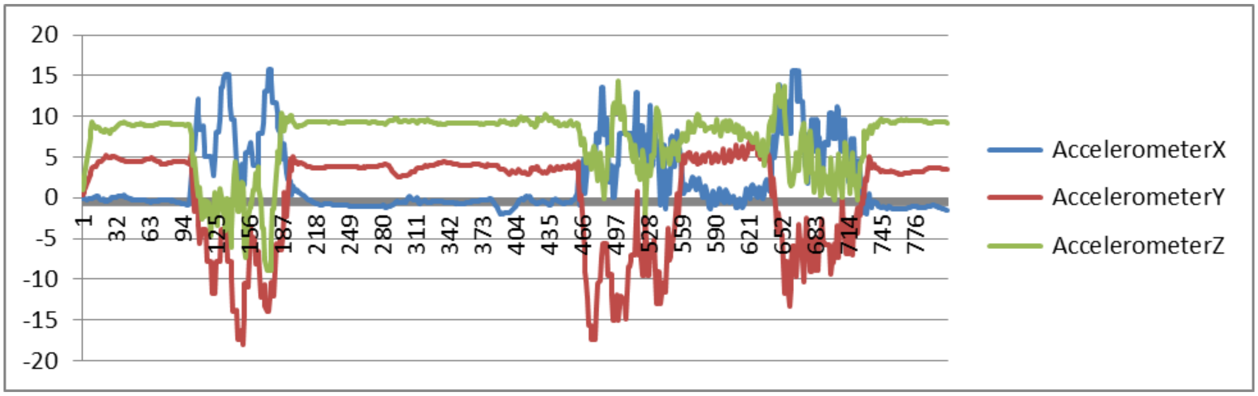
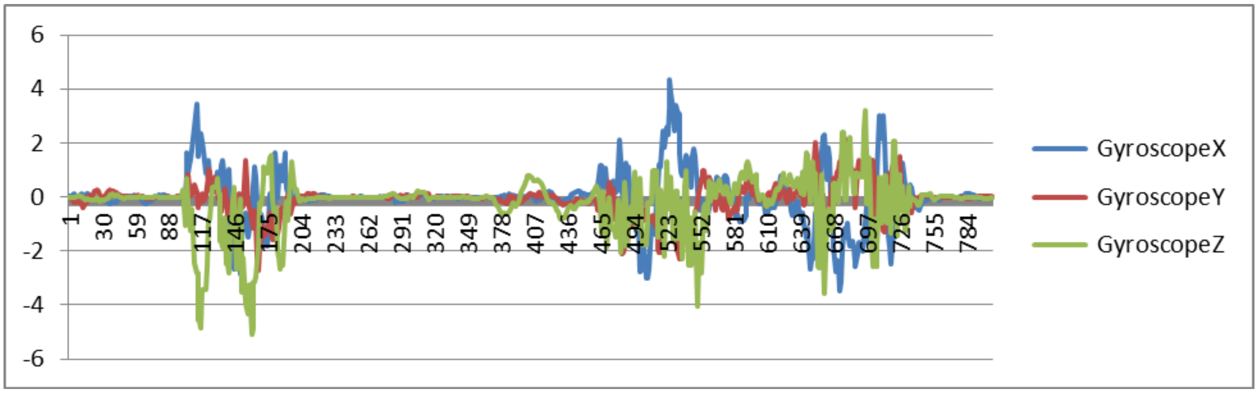


Figure2. Captured, Raw Accelerometer and Gyroscope Data

To smooth the data and minimize noise from the captured raw data of both sensors, a Moving Average Filter was used. Figure 3 illustrates the filtered signals of the accelerometer and gyroscope readings. Both signals now have a significantly reduced amount of noise and should be ready for further processing.



(a) Accelerometer data after noise reduction



(b) Gyoscope data after noise reduction

Figure 3. Accelerometer and Gyroscope Data after using a Moving Everage Filter for Noise

For the purpose of classification of events four classification algorithms were used as implemented within WEKA [2] data mining tool. These are Multi-Layer Perceptron (MLP), Random Forest, and Bagging. It is noted that MLP is a single layer classifier wherewas Random Forest and Bagging are ensemble classifiers. As bagging invloves the use of a base classifier, it was experimented with two base classifiers, namely Random Forest and MLP.

In order to investigate the impact of noise reduction in overall accuracy, figure 4 illustrates a plot that indicates the performance of the four classifiers when the moving average filter was and was not used on the raw data captured from the two sensors. The vertical axis depicts the correctly classified percentage of all events tested. The biggest impact on noise reduction has been when the MLP was used as a single classifier and when used within the ensemble classifier, bagging. This is indicated by the 15% approximate performance improvement. No significant improvement has been obtained when using Random Forest as a classifier indicating its robustness to the presence of noise.

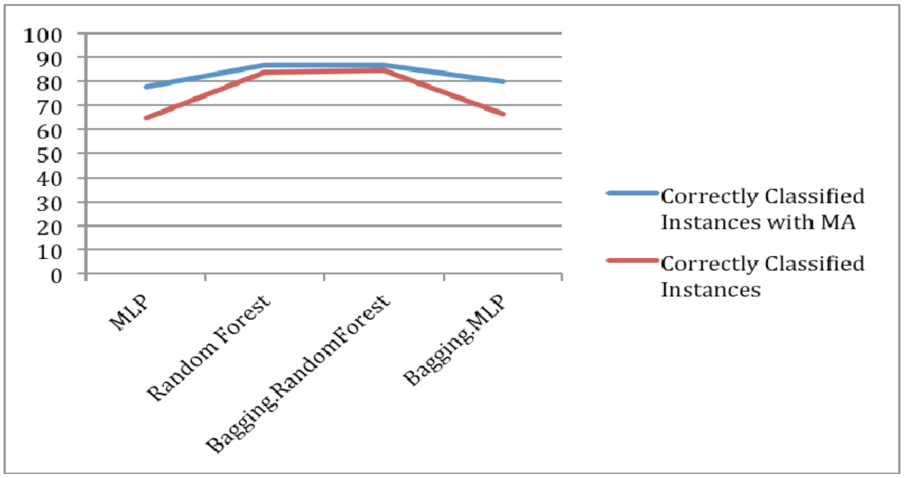
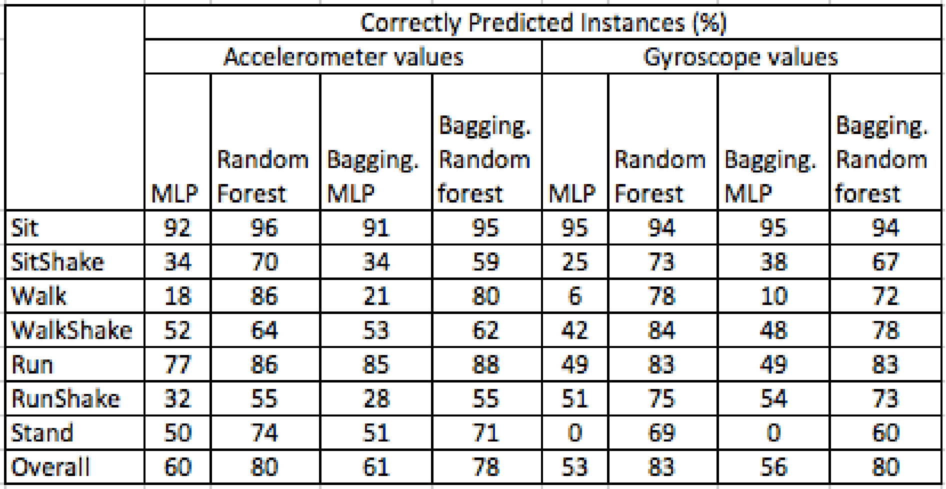


Figure 4. Performance comparison with and without noise reduction - % correct total event predictions

Table 1, illustrates the ability of the four classifiers to identify the seven events based on the individual classifier readings, i.e. the accelerometer and gyroscope readings considered independently and individually. The percentage accuracies of the predictions are tabulated. The highest accuracy is obtained for identification of sitting with all classification approaches indicating accuracies of over 90%. Running can also be classified relatively accurately. When the sensors are considered independently it is the less complex tasks that can be identified more accurately. This is well indicated by the dismal performance of the classifiers when identifying events that includes shaking as an added complication. Further observation is that the MLP as a single classifier has the worst average performance and bagging with random forest as the base classifier has the highest accuracy of performance.

Table 1. Percentage correctly predicted instances when using the Accelerometer and Gyroscope independently

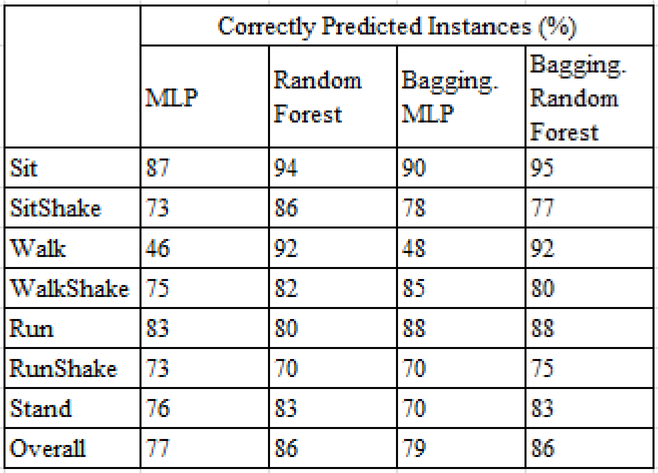


| | Correctly Predicted Instances (%) | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | Accelerometer values | | | | Gyroscope values | | | |
| | MLP | Random Forest | Bagging. MLP | Bagging. Random forest | MLP | Random Forest | Bagging. MLP | Bagging. Random forest |
| Sit | 92 | 96 | 91 | 95 | 95 | 94 | 95 | 94 |
| SitShake | 34 | 70 | 34 | 59 | 25 | 73 | 38 | 67 |
| Walk | 18 | 86 | 21 | 80 | 6 | 78 | 10 | 72 |
| WalkShake | 52 | 64 | 53 | 62 | 42 | 84 | 48 | 78 |
| Run | 77 | 86 | 85 | 88 | 49 | 83 | 49 | 83 |
| RunShake | 32 | 55 | 28 | 55 | 51 | 75 | 54 | 73 |
| Stand | 50 | 74 | 51 | 71 | 0 | 69 | 0 | 60 |
| Overall | 60 | 80 | 61 | 78 | 53 | 83 | 56 | 80 |

Table 2 illustrates the percentage of the correctly predicted instances when two readings of the two sensors were combined and used in association with each of the four classification algorithms. A significant improvement in overall accuracy is clearly noticeable when results of table 2 are compared with those tabulated in table 1. The more common and simpler activities such as sitting, walking, running and standing, indicates accuracy levels of over 80% when using all classification approaches. Sitting seems to be the easiest to be identified amongst the rest. The overall percentage of the correctly predicted instances is the same for Random forests and Bagging with Random forests when classifying simpler tasks such as sitting, standing, walking and running. However in classifying more complex scenarios that involve shaking the two said classifier performance is mixed. However overall bagging with random forest provides the best classification accuracy. MLP on the other hand, as a non-ensemble classifier, has generally poor accuracy results, when compared to the rest of the models investigated. Even though when MLP is used as the base classifier within Bagging the accuracy shows slight improvement it is still worse when compared to the accuracy obtained from the random forst based approaches.

When comparing the results tabulated in Table 1 and 2, the overall results for Bagging with random forests has increased from 80% to 86%. The accuracy in using Bagging with MLP for classification indicates a greater improvement in accuracy from 60% to 80%. Random forests have generally given the same accuracy results of above 80%. Finally the accuracy obtainable with MLP as a single classifier increased from 60% up to 77%. In summary it can be concluded that when the two sensor readings are combined the models can predict with higher rates of accuracy.

Table 2. Accuracy of the Correctly Predicted Activities combining Accelerometer and Gyroscope data

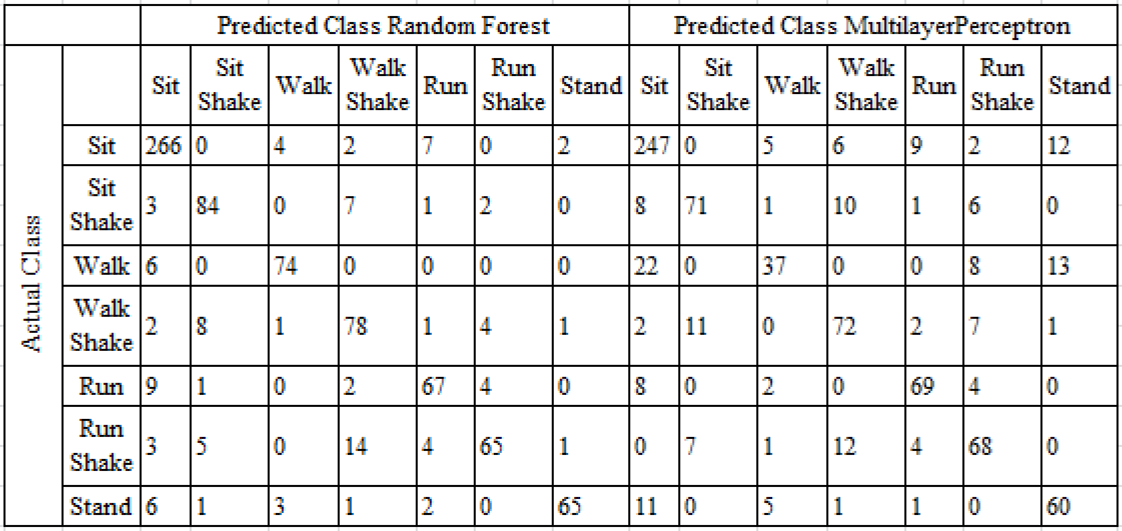


| | Correctly Predicted Instances (%) | | | |
|---|---|---|---|---|
| | MLP | Random Forest | Bagging. MLP | Bagging. Random Forest |
| Sit | 87 | 94 | 90 | 95 |
| SitShake | 73 | 86 | 78 | 77 |
| Walk | 46 | 92 | 48 | 92 |
| WalkShake | 75 | 82 | 85 | 80 |
| Run | 83 | 80 | 88 | 88 |
| RunShake | 73 | 70 | 70 | 75 |
| Stand | 76 | 83 | 70 | 83 |
| Overall | 77 | 86 | 79 | 86 |

Table 3 illustrates the confusion matrix obtained when MLP and Random Forest are used an single classifiers. It indicates the presence of prediction errors specifically when recognising certain scenarios, more than some others. When using the Random Forest classifier, 14 known Run-Shake scenarios were classsfiied wrongly as Walk-Shake. 8 Walk-Shake cases have been incorrectly predicted as Sit-Shake. There was a slight confusion between the Stand and Sit scenarios where 6 occurances of Stand have been wrongly classified as Sit.

The predicted class for MLP had more incorrect predictions compared to Random Forests. Major inaccuracies of classification were with Walk as it was incorrectly predicted as Sit, 22 times and as Stand, 13 times. Similar to the case of using Random Forests, there was a confusion between identifying Walk-Shake and Run-Shake scenarios as both activities include shaking the phone. It was also slightly harder to identify if the user was walking or runing. In conclusion, Random Forests gives an overall better result as compared to MLP.

Table 3. Confusion Matrix for Random Forests and MultilayerPerceptron

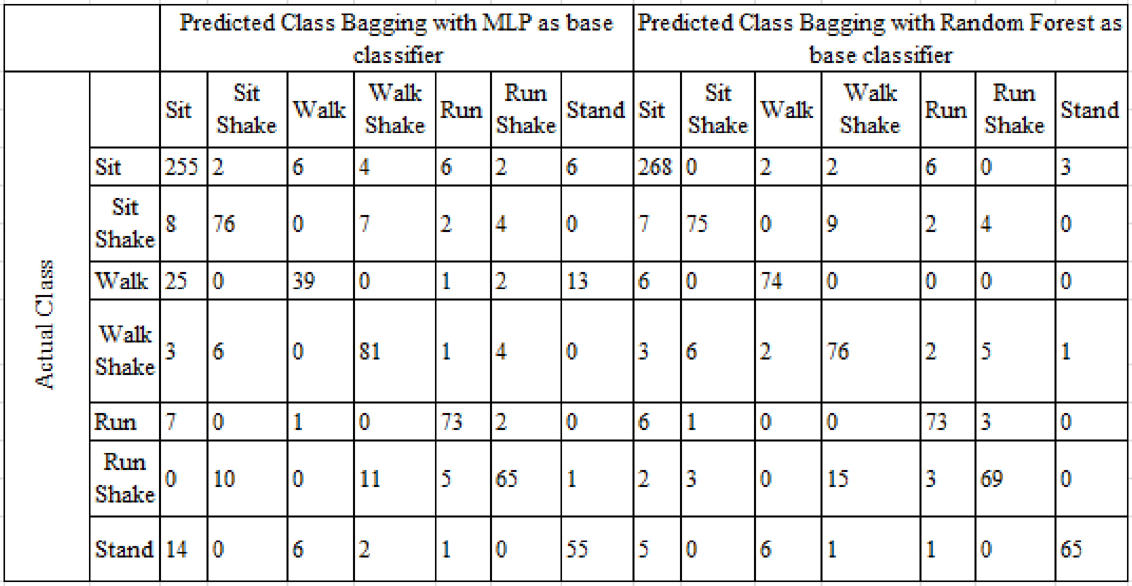


| | | Predicted Class Random Forest | | | | | | | Predicted Class MultilayerPerceptron | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Actual Class | | Sit | Sit Shake | Walk | Walk Shake | Run | Run Shake | Stand | Sit | Sit Shake | Walk | Walk Shake | Run | Run Shake | Stand |
| | Sit | 266 | 0 | 4 | 2 | 7 | 0 | 2 | 247 | 0 | 5 | 6 | 9 | 2 | 12 |
| | Sit Shake | 3 | 84 | 0 | 7 | 1 | 2 | 0 | 8 | 71 | 1 | 10 | 1 | 6 | 0 |
| | Walk | 6 | 0 | 74 | 0 | 0 | 0 | 0 | 22 | 0 | 37 | 0 | 0 | 8 | 13 |
| | Walk Shake | 2 | 8 | 1 | 78 | 1 | 4 | 1 | 2 | 11 | 0 | 72 | 2 | 7 | 1 |
| | Run | 9 | 1 | 0 | 2 | 67 | 4 | 0 | 8 | 0 | 2 | 0 | 69 | 4 | 0 |
| | Run Shake | 3 | 5 | 0 | 14 | 4 | 65 | 1 | 0 | 7 | 1 | 12 | 4 | 68 | 0 |
| | Stand | 6 | 1 | 3 | 1 | 2 | 0 | 65 | 11 | 0 | 5 | 1 | 1 | 0 | 60 |

Table 4 illustrates the Confusion Matrices when using Bagging with base classifier Random Forests and MLP. Bagging with Random Forests has given its best prediction in recognising Walking as it was mistakenly predicted as only Sitting, 6 times. It wasn't cofused with any other activity. The same prediction accuracy was indicated for Walking activity when using Random Forests classifier as a single classifier (see Table 3). When the phone was shaken, there were more incorrect predictions as compared to when it was not shaken in carrying out the more simpler activities on their own.

When using Baggin with MLP as the base classifier, there were a noticable number of incorrectly classified walking activities. It has been confused 25 time with siting and 13 times with standing. Run-Shake was incorrectly predicted 10 times as Sit-Shake and 11 times as Walk-Shake. The common activities shows a large number of incorrectly classified instances as compared to when Random Forest was used with Bagging for predeiction. The activity with the leaset number of classification errors is Run activity. However, Bagging with MLP as a base classifier has given better overall accuracy results when compared to MLP used as a single classifer.

Table 4. Confusion Matrix for Bagging with MLP and Random Forests as the base classifiers



| | | Predicted Class Bagging with MLP as base classifier | | | | | | | Predicted Class Bagging with Random Forest as base classifier | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Sit | Sit Shake | Walk | Walk Shake | Run | Run Shake | Stand | Sit | Sit Shake | Walk | Walk Shake | Run | Run Shake | Stand |
| Actual Class | Sit | 255 | 2 | 6 | 4 | 6 | 2 | 6 | 268 | 0 | 2 | 2 | 6 | 0 | 3 |
| | Sit Shake | 8 | 76 | 0 | 7 | 2 | 4 | 0 | 7 | 75 | 0 | 9 | 2 | 4 | 0 |
| | Walk | 25 | 0 | 39 | 0 | 1 | 2 | 13 | 6 | 0 | 74 | 0 | 0 | 0 | 0 |
| | Walk Shake | 3 | 6 | 0 | 81 | 1 | 4 | 0 | 3 | 6 | 2 | 76 | 2 | 5 | 1 |
| | Run | 7 | 0 | 1 | 0 | 73 | 2 | 0 | 6 | 1 | 0 | 0 | 73 | 3 | 0 |
| | Run Shake | 0 | 10 | 0 | 11 | 5 | 65 | 1 | 2 | 3 | 0 | 15 | 3 | 69 | 0 |
| | Stand | 14 | 0 | 6 | 2 | 1 | 0 | 55 | 5 | 0 | 6 | 1 | 1 | 0 | 65 |

## 5. Conclusion and Future Work

In this paper we have investigated the possibility of activity analysis and recognition in relation to the use of a mobile phone, based on fusing the data captured from multiple sensors. A proof of concept that machine learning algorithms can be utilized for this purpose has been provided based on fusing the data captured from the accelerometer and the gyroscope of a mobile phone. The use of four different machine learning algorithms namely the well-known Multi Layer Perceptron, Random Forests, Bagging with MLP and Bagging with Random Forests have been investigated in detail for the classification of seven common events associated with the typical use of a mobile device. It has been shown that the ensemble classifier Bagging when used with the base classifier Random Forests performs best giving accuracy figures of beyond 90% in identifying all events tested. The ability of the ensemble learning algorithms to accurately identify events when the scenarios increase in complexity has been proven.

We are currently working on integrating a number of other in-built sensors in the above process, allowing more detailed and complex scenarios to be identified accurately. The computational cost of each approach will also be investigated to study practicalities of implementation.

# REFERENCES

[1] Breiman L, "Bagging Predictors. Machine Learning," Kluwer Academic Publishers 24, 123-140 (1996).

[2] Weka, The University of Waikato, "Data Mining Software in Java", 2013, <http://www.cs.waikato.ac.nz/~ml/weka/ > (18th March 2015).

[3] Hinckley K, Pierce J, Sinclair M, Horvitz E, "Sensing Techniques for Mobile Interaction," ACM 1-58113-212-3/00/11 (2000).

[4] Kwapisz J, Weiss G, Moore S, "Activity Recognition using Cell Phone Accelerometer," ACM 978-1-4503-0224-1, (2010).

[5] Sensors Overview, Android Developers, <http://developer.android.com/guide/topics/sensors/sensors_overview.html> (18th March 2015).

[6] Breiman L, "Random Forests. Machine Learning," Kluwer Academic Publishers 45,5-32 (2001).

[7] Wikipedia, "Six degrees of freedom." (2014) < http://en.wikipedia.org/wiki/Six_degrees_of_freedom#Robotics > (23rd February 2015).

[8] Ian H. Witten, Eibe Frank and Mark A. Hall, [Data Mining] , Morgan Kaufmann Publishers, Burlington, USA, 232-241(2011).