International Journal on Recent and Innovation Trends in Computing and Communication
Volume: 2 Issue: 5

ISSN: 2321-8169
1312 – 1317

Activity Recognition System using Smartphone

Priti Moon M.E.(Student)
Dept. of Computer Engineering
Sinhgad Academy Of Engineering, Pune University
Pune, India
moon.priti@gmail.com

Assistant Prof. B. B. Gite
Dept. of Computer Engneering
Sinhgad Academy Of Engineering, Pune University
Pune, India
bbgite.sae@sinhgad.edu

***Abstract*—** Human activity monitoring is becoming more automated gradually. In near future it would be required more advanced electronic systems to assess physical activities. Today's day most of the doctors and scientists use their memory or match data from text books for monitoring activities for checking dissimilarities of experimented data. Though for automated monitoring some digital modern instruments are working in some sophisticated places also. In this paper we will describe how to extract human walking features and patterns with Smartphone built in accelerometer. This will initiates the work to create model of different subjects of normal physique like short, average and long heights and walking of fast, medium and slow in speed. Walking features of healthy person can be compared with walking features of any patient with leg injury or some leg related diseases and result would give the nature of defects.

Keywords-*Activity Recognition System,Feature Server, Ground Truth data, Run Time Data*

*****

I. Introduction

Activities are the actions over a period of a person and denoted by sequences of states of a time domain. Automated recognition of any human manual activities like: walking, sitting, running, playing, sleeping, kicking, taking medicine, moving neck and arms, exercising etc. is known as Activity Recognition System [1]. Recognizing human activities with wearing sensors on to the body has become an important research area, aiming to create or improve innovative applications providing activity monitoring. An emerging application for wireless body sensor networks contributes their use in medical care. In a hospital or clinic, outfitting every patient with tiny, wearable wireless sensors would allow medical representative like doctors, nurses and other caregivers to continuously monitor the status of their patients. In health care field, long term analysis of human activity could be helpful in early detection of diseases or even to encourage people to improve their activity level.

The new generation of Smartphone is being considered by users as an important personal device, together with an exponential availability. These devices have an increased potential for an adequate mean of gathering motion data, to use for building human activity prediction systems. The perceptions of their benefits are becoming commonplace, as users have become accustomed to their ubiquity. Smart phones are equipped with a wide range of internal sensors, including accelerometers and gyroscopes, which can be used to monitor human daily activities. By using Smartphone human activity can be sensed and prepare digital data by boosting up and filtering of noise values. With the help of statistical and computational process raw data is transformed to model of digital activity.

If patient suffering from diseases like: Wilson, Perkinson, Dystonia (affecting a single body part), Trunk Dystonia (abnormal posture), Tremor, RLS(Restless Leg Syndrome) and any damages in legs cause disorder in motion and walking [1] then walking pattern of a patient can define the state of patient as critical, average or normal. Walking features of healthy person can be compared with walking features of any patient mentioned above and result would give the nature of defects. This is called as feature analysis and matching computing problem.

Applications for human activity monitoring are emerging, not only for ones who want to attend their relatives and could not be present, but also for people who want to check the physical activity and improve it if necessary. These types of applications are named as context aware monitoring system. Gesture and posture of human being is named as Context. Patients with above mentioned diseases needs to keep check on their health regularly for recovery and context monitoring will help them for maintaining their status. Also this can help them to save their time and money of continuously visiting doctor for their regular checkups. This system is useful for doctors also as at present, in most of the cases, doctors and scientists use their memory or match data from text books for activity monitoring and goes for checking dissimilarities of experimented data, though for automated monitoring some digital modern instruments are working in some sophisticated places also.

II. Related Work

This Section describes the state of the art related with the activity recognition. Some previous methods in this area are also presented. The main technology concepts and resources will be used in the project are also explained.

A. Activity Recognition

Many applications perform classification on body sensor nodes but have no mobile. It only works on-body aggregator for sensor control and activity recognition feedback. In past many works [4] [5] uses multiple on-body sensor motes to detect user activities, body posture, or medical conditions, but such motes are only used to collect and store data with analysis performed offline. Such experiments limit subject mobility due to periodic communication with a fixed base station and also lack real time analysis and feedback to the subject.

In some work [3] Android Smartphone has been used with TinyOS motes, they have combine the sensing power of

 ISSN: 2321-8169
1312 – 1317

on-body wireless sensors with the additional sensing power, computational resources, and user-friendly interface of an Android Smartphone. But in such circumstances, sensors need to be on body which is uncomfortable for the subject. Also this can be performed in experimental lab for better result; with daily life usage this method will be more obtrusive for the result as well as for the user.

The whole process for activity recognition begins with gathering the raw data, specifically, motion data. Inertial sensors are an adequate solution to detect motion of the subject. These inertial sensors respond to stimuli by generating signals that can be analyzed and interpreted [7]. Mainly, sensors are placed next to the body and should be comfortable for the user**.** The new generations of Smart phones are well equipped with a wide range of inertial sensors, including accelerometers and gyroscopes, which can be used to monitor subject's daily activities. These devices are practical, small and unobtrusive, becoming a perfect platform for an activity recognition system. Other most important desirable features are the possibility to be wearable also works in real-time and be used for long-term monitoring.

B. Sensors

Sensors are mainly used for the detection of human activity. There are three main concerns regarding sensors as which type of sensors needs to use? Where will be location of sensor? How many sensors need to be used for the experiment (number of the sensors)? The majority of context-aware systems have used inertial sensors, particularly accelerometers, to estimate the inclination of the body from the vertical also to determine the orientation and movement of the user [7]. Accelerometers use transducers for measuring linear acceleration. "An accelerometer behaves as a keeping mass on a spring. When the accelerometer experiences acceleration then the mass is displaced to the point that the spring is able to accelerate the mass at the same rate as the casing. This displacement is then measured to give the acceleration"[8].

The signal obtained with accelerometers has two main components, one is a gravitational acceleration component (static) that provides information on the postural orientation of the subject, and another is a body acceleration component (dynamic) that provides information on the movement of the subject. A 3D accelerometer measures the acceleration along x (lateral), y (vertical) and z (longitudinal) axes relative to the screen of the phone as illustrated in Figure 1.

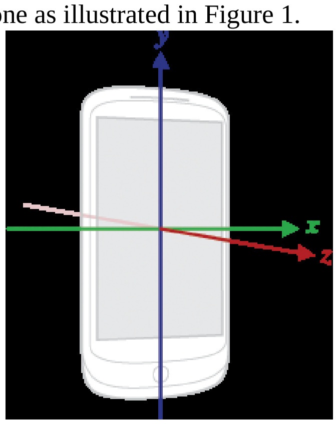

Fig 1: Smartphone axis orientation

Nowadays researchers are more interested in using accelerometers because users consider accelerometers less intrusive than other sensors such as microphones or cameras [10]. Other advantages which makes this type of sensor useful for human activity recognition, as their low cost and small size.

C. Smartphone

The new generation of Smartphone is important personal device for users, together with an exponential availability. There is increased potential for these devices as an adequate mean of gathering motion data, to use for building human activity recognition systems. One of the most important applications of human activity recognition system is health care.

Smart phones are well equipped with a wide range of internal sensors as described in Table 1, including accelerometers and gyroscopes, which can be used to monitor human daily activities.

TABLE I
NUMBER OF SENSORS IN SOME RECENTLY LAUNCHED SMARTPHONE'S

Sensor type	iPhone 4s	Galaxy S3	HTC One X	Droid 4	Lava Xolo 900
Accelerometer	•	•	•	•	•
Gyro	•	•	•	•	•
Dig. Compass	•	•	•	•	•
Proximity Sensor	•	•	•	•	•
Ambient Light Sensor	•	•		•	•
Barometer		•			

Android-based Smartphone [6] have been chosen as the platform for this project because the Android operating system is free, open-source, easy to program, and availability in people daily living. The most important thing is low cost of the Smartphone that have accelerometers incorporated are also part of people nowadays routine.

III. TECHNOLOGY

Once the dataset is prepared, a classification algorithm will be implemented. As previously mentioned, human activity recognition is normally treated as a classification problem, using techniques of machine learning based on probabilistic and statistical reasoning. In 1959, Arthur Samuel defined machine learning technique as a "field of study that gives computers the ability to learn without being explicitly programmed". Samuel has correctly specified about the basis of machine learning as it builds a model and a classifier which is capable of learning from unseen data. The model would represent the data instances where each instance represents a data window with fixed size. Also the model would represent functions of these instances in the training step and ultimately the classifier could generalize for unseen data. Basically, Machine learning is a branch of computer science concerned with induction problems for which an underlying model for predictive or descriptive purposes has to be discovered, based on known properties learned from the training data. Conceptually, Machine learning algorithms can be divided in several categories:

- **Supervised learning technique** makes use of labeled data upon which an algorithm is trained and followed by training the algorithm to classify unknown data [7].
- **Unsupervised learning technique** [7] tries to directly construct recognition models from unlabeled data. The basic idea of this technique is to manually assign a probability to each possible activity and to predefine a stochastic model that can update these likelihoods according to new observations and to the known state of the system.
- **Semi-supervised learning technique** [9] is a class of supervised learning technique that also makes use of unlabeled data for training. Basic idea of this technique is to have small amount of labeled data along with a large amount of unlabeled data. Semi-supervised learning lies between unsupervised learning and supervised learning.
- **Ensemble learning technique** uses multiple models to obtain better predictive performance than could be obtained from any of the constituent models. It trains multiple learners to solve the same problem.

Summarization of the machine learning process that will also be implemented in this project is described in Figure 2. In this process there are mainly 5 steps. After data acquisition and preprocessing steps, the features need to be extracted in order to train the classification algorithm that will be finally evaluated using confusion matrices.

Fig 2: Schematic representation of a machine learning process

A. Data Collection

Data collection is described in the previous section under Sensor and Smartphone. Many Smartphone in market are coming up with inbuilt inertial sensor such as accelerometer.

B. Preprocessing

The raw data usually needs to be pre-processed. Accelerometer raw data needs to be divided in sequential windows, to be pre-processed. For online applications, the window has to be defined in parallel with data collection and for offline applications; the window is defined prior to data collection [7]. Raw data needs to be split into training set and test set. The training set will be used to train the activity recognition algorithm and the test set will then be used to evaluate the recognition algorithm after training.

C. Feature Extraction

In feature extraction for each window, some features are extracted to characterize the signal. These extracted features are then used as input for the activity recognition algorithms, to associate each window with an activity. Conceptually, there is Time-domain and frequency-domain features that can be extracted from motion data.

- **Time-domain features** are basic mathematical and statistical metrics used to extract basic signal information from raw data. These features are simple to compute.
- **Frequency-domain features** capture the respective nature of a sensor signal. For frequency-domain features, the sensor data window has to be transformed into frequency domain, using fast Fourier transform (FFT), which is a spectral representation of the signal.

D. Classification

Once extracted features are prepared then machine learning techniques will be applied in order to construct a classifier. Main task of Classification is to choose the correct class label for a given input. There are many classification methods available some are described below,

- Basically, **Decision tree** is flowchart like tree structure in which for each attribute, one branch per each possible result of a test is generated. Finally, algorithm stops when it finds the leaf which represents a class.
- **Naive Bayes classifiers** determine which label should be assigned to a given input value. Label for an input value can be generated by calculating the prior probability of each label, which is determined by checking the frequency of each label in the training set. Then whichever label have highest probability is assigned to the input value.
- **SVM** method is used for classification of both linear and nonlinear data. In this classification method, separating hyperplane is located in the class space and classifies points in that space. Main concern is to find the maximum margin hyperplane separating two classes. The instances with minimum distance to the hyperplane are defined as support vectors.

ISSN: 2321-8169
1312 – 1317

- The **back-propagation** algorithm performs learning on a multilayer feed-forward neural network. It iteratively learns a set of weights for prediction of the class label of tuples. A multilayer feed-forward neural network consists of an input layer, one or more hidden layers, and an output layer.

E. Evaluation

There are many different ways to evaluate the performance of activity recognition algorithm some are listed below.

- Accuracy is the simplest metric that can be used to evaluate a classifier. It measures the percentage of inputs in the test set that the classifier correctly labeled.
- **Precision** is the how many inputs that we identified were relevant. **Recall** is how many of the relevant inputs that are identified.
- The **F-Measure** is combination of the precision and recall to give a single score which is mentioned to be the harmonic mean of the precision and recall.
- When performing classification tasks with multiple labels then result can be represented in an n-n matrix of n classes. This is termed as **Confusion Matrix**.

IV. Recording Manual Activities

Aims of this project are (a) to prepare digital model of walking of different subjects, (b) uploading these models in Local Server and Feature Server, (c) collecting RTD of different patients of pathological deformation in legs and (d) recognizing percent of mismatching of walking pattern of patients.

Positional features are recorded within a mobile phone and processed with the help of application software. Through wireless transmitter data are transmitted to PDA/Laptop. Feature Server contains features of all activities of various subjects of different age, weight and height. This data stored in the feature server is called as Ground Truth Data. Local Server can process all local data and if required can connect with the Feature Server for accessing Ground Truth Features (GTF).

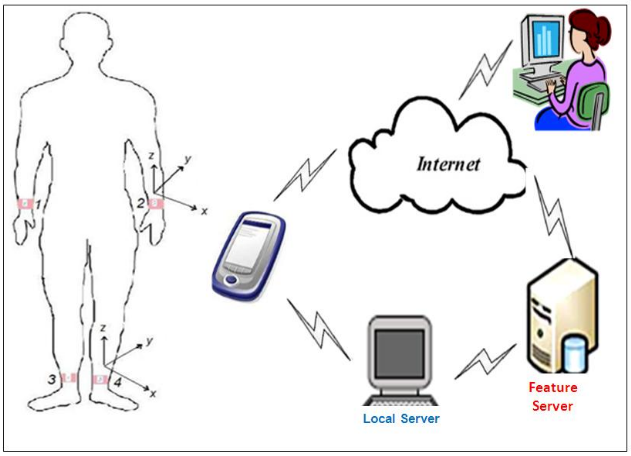

Fig 3: Experimental setup of Activity Recognition System

Once Run Time Data of patient is ready then by using motion disorder detection calculate average % of error in motion of body or walking disorder compared to features of GTD.

Generally Subject is divided according to their heights as *short (<150cm. approximately), medium (>=150cm. and <180cm. approximately)* and *long (>=180cm. approximately).* In real life, a subject may walk in three ways as *slow, medium and fast* walking. Positional features are recorded within a Smartphone and processed with the help of application software. Four sensor nodes are attached at four positions like: ankle of left and right legs, and wrist positions of left and right hands of the subject.

If we consider a human body is a 3D volume, the logic is that when a medium height and size person stands normally the separation of two hands' wrist position is in average is 35.5cm. Same logic applied for considering separation of two legs' ankles is 17.8cm. By this logic, considering left hand's wrist position as origin point (0,0,0), we add 35.5cm with the data of right hand. Similarly, we add 17.8cm with the right leg's data.

TABLE II
MAXIMUM RANGES OF DISPLACEMENT OF LEGS FROM (0,0,0) COORDINATE TO THE POSITIVE DIRECTION OF THE AXES

	Slow Walking			Medium Walking			Fast Walking		
	Medium	Long	Short	Medium	Long	Short	Medium	Long	Short
X	25	40	15	30	45	20	40	48	25
Y	17	22	10	20	25	15	24	30	21
Z	10	14	6	12	18	10	15	20	12

It is required to quantify these fuzzy data as shown in Table-2, where the maximum ranges of displacements for three subjects of each three activities are shown. In Fig.4, three subjects' maximum stepping displacements along x axis are shown as an example.

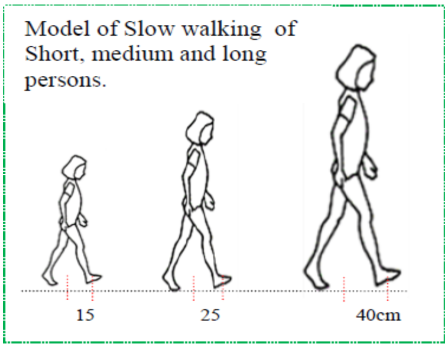

Fig 4: Leg's displacement for short, medium and long persons, displacement along x axis from the centre point of the body is shown here for slow walking model.

The total work flow of recording GTD and extracting features from that is shown in ALGORITHM-1. To make a prototype model, we generate random values within the range mentioned in Table-2. Total 1000 record have been generated which is called as GTD (ground Truth Data). We mix these 1000 records following the rule of coordinators.

We consider that for a medium height person the distance between wrist and ankle is about 2.5 ft. or 76.2cm. Suppose that left hand wrist position is (0,0,0), then left leg ankle's coordinate will be (0,0,0 + 76.2cm) i.e. the length from left wrist of hand to left ankle of leg. By considering this view, the right hand's wrist coordinate will be (0, 0 + 35.56, 0). In the same way, the right leg ankle's coordinates with reference to left wrist is (0, 0 + 35.56, 0 + 76.2cm). In this way, data will be parsed and GTD (ground truth data) will be ready.

ISSN: 2321-8169
1312 – 1317

ALGORITHM-1(algorithm of GTD)
#Input is x,y,z values of the GTD table.
#Output is the feature table.
#existing data is guessed and some positions are filtered.
#Then, features are calculated.
1. Count size of the table as row=number of rows in GTD
2. Count no. of columns as col=no. of column
3. Assume high=1.1; low=0.01;boost=0.03;
4. For c= 1 to col
5. Calculate mx= maximum value available in data table.
6. For r= 1 to row
7. #filter high data and boostup very low data of GTD
*8. If GTD(r,c) is >mx*high then*
9. GTD(r,c)=mx
*10. Else if GTD(r,c) is < low*mx then*
*11. GTD(r,c)=boost*mx*
12. End for
13. End for
14. Make k numbers of clusters from GTD
15. Calculate features from each cluster.
16. Compute statistical feature values
17. Save all features as GF(Ground Feature)
18. End

Where r=row and c=column position of any data present in GTD matrix.

V. CLUSTERING

K – means algorithm is used for clustering. The *k*-means algorithm takes the input parameter, *k*, and partitions a set of *n* objects into *k* clusters so that the resulting intra-cluster similarity is high but the inter-cluster similarity is low. Cluster similarity is measured in regard to the *mean* value of the objects in a cluster, which can be viewed as the cluster's *centroid*.

The *k*-means algorithm proceeds as firstly, it randomly selects *k* of the objects, each of which initially represents a cluster mean or center. For each of the remaining objects, an object is assigned to the cluster to which it is the most similar, based on the distance between the object and the cluster mean. It then computes the new mean for each cluster. This process iterates until the criterion function converges. All points nearer to those centroids are segregated into clusters based on some formula of distances (Euclidean, Manhatan, K-L divergence etc.). We used Euclidean method. The result of clustering is shown in figure 5 as an example for medium walking and long height person.

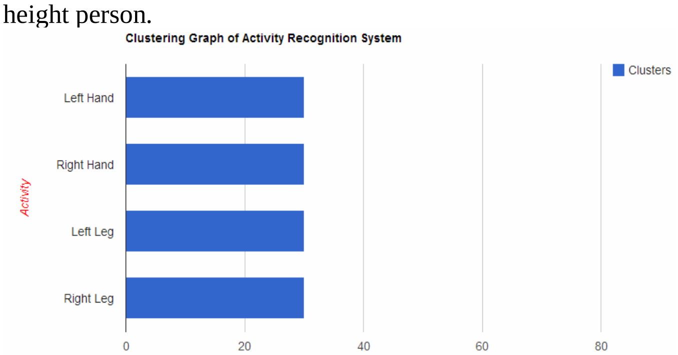

Fig 5: Clustering of medium walking and Long Height Person

VI. FEATURE EXTRACTION

Time domain and frequency domain features are extracted and the resultant table is as shown in Table-3. Some statistical standard formulas are used to calculate features of GTD data table. Statistical formulas like: variance, standard deviation, mean, correlation coefficient etc. values are extracted from Ground Truth Data (GTD). Correlation coefficient(r) is a matrix of 3x3 for x,y,z vectors and it is [r_{xx}, r_{xy}, r_{xz} ;r_{yy}, r_{yx}, r_{yz}; r_{zz}, r_{zx}, r_{zy}]. There are N samples of data for each x, y and z vectors, $d_i(t)$ is the data value, r_{ij} is the correlation coefficient value when i=x, j=y (say) and s_i, s_j are the standard deviation of vector x(t) and y(t) respectively. Final equation of correlation coefficient is:

$$r_{ij} = \frac{1}{N-1}\sum_{t=1}^{N}\left(\frac{d_i(t)-\bar{d}_i}{s_i}\right)\left(\frac{d_j(t)-\bar{d}_j}{s_j}\right)$$

Other formulas used to extract features from clusters are:
Average distance from cluster heads is:

$$d_1 = \frac{\text{total distance from cluster head}}{\text{total members in cluster}}$$

Total distance index from cluster head is:

$$d_2 = \frac{\text{total distance from all cluster heads}}{\text{total members in operation}}$$

Normalized maximum value is calculated as:

$$\text{NMaxX} = \frac{\text{Max X} - \text{MeanX}}{\text{Max X}}$$

In the same way NMaxY and NMaxZ are calculated and used as feature values.

C1, C2, C3, C4 are the four clusters.

TABLE III
FEATURE VALUE FROM GTD

Sl.	Features	Sl.	Features
1	Avg. distance From C1	14	Variance of X
2	Avg. distance From C2	15	Variance of Y
3	Avg. distance from C3	16	Variance of Z
4	Avg. distance From C4	17	STD of X
5	Members in Clusters1	18	STD of Y
6	Members in Clusters2	19	STD of Z
7	Members in Clusters3	20	NMax X
8	Members in Clusters4	21	NMax Y
9	Total distance index	22	Nmax Z
10	Max. of X	23	Mean of X
11	Max. of Y	24	Mean of Y
12	Max. of Z	25	Mean of Z
13	Correlation coefficient of X, Y and Z.		

VII. MOTION DISORDER DETECTION

Suppose S is a set of sensor nodes.
S = {S1, S2…Sn}
Where n = number of nodes

In each sensor there are 'k' types of sensors, so, total number of sensors (TS) working in experiment model will be,
TS = n * k
A = {A1, A2 …Am}
Where A = set of activities
m = number of activities to be monitored for set of subjects

B= {B1, B2…Bi}
Where B = set of subjects

i = number of subjects

So, number of activities (ACT) to be recorded will be,
ACT = m * i

$$\text{Total Data} = A_m \times B_i$$

Random data is generated within range of x,y,z as mentioned in Table-2. Then, 5% errors are seeded on GTD and this file is known as Run Time data (RTD). After seeding errors, we check the boundary values of clusters and readjust the values by updating the existing values through some filtration techniques.

Algorithm 2. (Motion disorder detection)
#input is database of GDT(Ground Truth Data)
#output is average error in any activity recognition.
1. Seed errors within 5% positions of GTD data file,
(Which we call runtime data or RTD)
2. Say window-size=r
3. Read r numbers of records from RTD data file.
4. Make clustering for k numbers of clusters
5. Calculate clusters feature values and
6. Calculate statistical features
7. Make the feature table as RF_i*(Runtime Feature)*
8. Calculate percent of error in each feature as
$e_i=100*(GF_i - RF_i)/GF_i$
9. If $e_i > \theta$ *then goto step 3.*
10. Calculate average % of error in motion of body or walking disorder compared to features of GTD.
11. Infer suitable decision or predict nature of diagnosis to be required for the particular patient.

Where, GF_i *is ground truth feature,* θ *is threshold value preassigned by user, say 5%.*

VIII. Conclusion

Feature server data will be accessible in LAN (Local Area Network), WAN (Wide Area Network) or in internet for public use. For comparison of an activity every time it is not required to generate GTD. With use of Smartphone is very easy to deploy this application in daily life activity. By this application, percent of matching or mismatching may be observed by the patient also.

References

[1] Dulal Acharjee, Dr. Amitava Mukherjee and Prof. Nandini Mukherjee, "Activity Recognition System using Body Sensor Network and Feature Server" in *IETE summer 2012*.
[2] L. Bao & S. S. Intille (2004). "Activity Recognition from User Annotated Acceleration Data", LNCS 3001, Springer, pp. 1-17.
[3] Matthew Keally and et.al, PBN: Towards Practical Activity Recognition Using Smartphone-Based Body Sensor Networks, SenSys-11, ACM 978-1-4503-0718-5/11/11.
[4] K. Lorincz, B. Chen, G. Challen, A. Chowdhury, S. Patel, P. Bonato, and M. Welsh. "Mercury: A Wearable Sensor Network Platform for High-Fidelity Motion Analysis". In *SenSys '09, pages 183–196.* ACM, 2009.
[5] M. Quwaider and S. Biswas. Body Posture Identification using Hidden Markov Model with a Wearable Sensor Network. In *Bodynets '08*, pages 19:1–19:8. ICST, 2008.
[6] Raquel Cerqueira da Silva "Smartphone Based Human Activity Prediction" July 2013.
[7] Wilde, A. (2010). "An overview of human activity detection technologies for pervasive systems". Seminar paper. Pervasive and Artificial Intelligence Group of University of Southampton.
[8] Bird, Steven, Klein, Ewan, & Loper, Edward. (2009). "Natural Language Processing with Python" - Chapter 6: O'Reilly Media.
[9] Lopes, A., Mendes-Moreira, J., & Gama, J. (2012). "Semi-supervised learning: predicting activities in Android environment." Workshop on Ubiquitous Data, 1-5.
[10] Huynh, Duy Tam Gilles. (2008). "Human Activity Recognition with Wearable Sensors. Dissertation".
[11] Nikhil Raveendranathan, Stefano Galzarano, Vitali Loseu, Raffaele Gravina, Roberta Giannantonio, Marco Sgroi, Roozbeh Jafari, and Giancarlo Fortino, From Modeling to Implementation of Virtual Sensors in Body Sensor Networks, IEEE Sensors Journals, vol.12, no.3, March 2012.
[12] Liang Wang and et.al, Real-time Activity Recognition in Wireless Body Sensor Networks: From Simple Gestures to Complex Activities, The Sixteenth IEEE International Conference on Embedded and Real- Time Computing Systems and Applications, IEEE 2011
[13] Karandeep Malhi, Subhas Chandra Mukhopadhyay, *Fellow, IEEE*, Julia Schnepper, Mathias Haefke, and Hartmut Ewald, A Zigbee-Based Wearable Physiological Parameters Monitoring System, IEEE Sensors Journal, vol. 12, no. 3, March 2012