Mathematical Problems of Computer Science 40, 44---54, 2013.

Time Domain Feature Extraction and SVM Processing for Activity Recognition Using Smartphone Signals

Sahak I. Kaghyan[1] and Hakob G. Sarukhanyan[2]

[1]Armenian-Russian (Slavonic) University
[2]Institute for Informatics and Automation Problems of NAS RA
e-mail: sahak.kaghyan@gmail.com, hakop@ipia.sci.am

Abstract

Automatic classification of human movement is a feature that is desired for a multitude of applications and mobile phone technology continuously evolves and incorporates more and more sensors to enable advanced applications. Combining these two concepts we can deal with "an activity recognition via smartphone sensors" problem where sensors of these devices play a core role when we deal with personalized activity tracking systems. In this paper we give an overview of the recent work in the field of activity recognition from mobile devices that can be attached to different parts of the body (pocket, wrist, and forearm). We focus on the technique of feature extraction from raw acceleration signal sequences of smartphone (mean, standard deviation, minimal and maximal signal values, correlation, median crossing). Further processing of these data allowed to classify the activity performed by the user. The core classification stage of the current approach was based on the method of "learning with the teacher" where the features of signal sequences were analyzed using the support vector machines (SVM) learning method.

Keywords: Smartphone, Accelerometer, Activity recognition, Time domain feature, SVM.

1. Introduction

Nowadays the Internet and personal computer are the most common ways to connect people, allowing them to share information with each other. On the other hand, none of them is able to reach each person anywhere and anytime like the cell phone does. Moreover, concerning the mobile technologies in general, now they are becoming ubiquitous all over the world, changing the way we communicate, conduct trade, and provide care and services. Certainly, some of the most compelling benefits of mobile technologies are in the areas of disease prevention, chronic disease management and improving healthcare delivery. For all the advances that occur in mobile healthcare (mHealth), its full potential for a very large group of beneficiaries – the elderly and those who support them – is only starting to emerge. The ability to monitor the physical state of a person leads us to the concept of personalized healthcare system implementation. One of the ways to help people with diseases is to give the doctors an opportunity to remotely monitor their patients' life activity via cellular phones and smartphones they care.

Human activity recognition (HAR) has matured in recent years which will enable many health promotions and intervention applications. There are no standardized performance evaluation strategies. Recent efforts on designing public datasets might be one of the approaches to address this problem. Generally, activity recognition (AR) aims to identify the actions taken by a person. Three main classes of activity recognition are considered including coarse location tracking, video stream analysis and inertial navigation systems (INS) such as accelerometers.

Sensor data are typically communicated from sensors to servers for further processing. Alternatively signal processing can be performed in mobile devices such as smart-phones. Many authors usually don't use standard tests for accuracy rate checks and validity of most reported results depends on testing specifics. There is no consensus even on a standard list of activities, but most of the reports include "walking", "sitting", "jogging" and "standing" patterns. Recognition can be accomplished, for example, by exploiting the information retrieved from inertial sensors such as accelerometers [4]. In some smartphones these sensors are embedded by default and we benefit from this to classify a set of physical activities (standing, walking, laying, , walking upstairs and walking downstairs) by processing inertial body signals through a supervised Machine Learning (ML) algorithm for hardware with limited resources. So, in general, activity recognition algorithms can be divided into two major categories. The first one is based on supervised and unsupervised machine learning methods. Supervised learning requires the use of labeled data upon which an algorithm is trained.

2. Related Works

Recognizing a predefined set of activities is a recognition (classification) task: features are extracted from the space-time information collected by sensors and then used for classification. Feature representations are used to map the data to another representation space with the intention of making the classification problem easier to solve. In most cases, a model of classification is used that relates the activity to sensor patterns. Learning of such models is usually done in a supervised manner (human labeling) and requires a large annotated datasets recorded in different settings. Smart phones include various sensors such as gyroscopes, accelerometers, proximity sensors and have become affordable and ubiquitous. Convenient user interfaces make them attractive for all population groups.

Oner et al. in [1] presented an early work on a pedometer mobile application that was coupled with e-mail to notify medical assistants or family members. Their purpose was to use a mobile smart phone to detect the fall event regardless of the phone position or orientation. The algorithm that was introduced in the article was based on the acceleration peak detection and was tested for different conditions.

Das et al in [3] introduced an attempt to recognize the activity using Motorola Droid smartphone. Activity classification was done through several stages: data acquisition, signal processing, feature extraction and classification. Using the nearest neighbor classifier the program could predict patterns or activities with 93% accuracy after it had been calibrated for a particular user.

One of our early works [4] introduced a general method of classification that used the nearest neighbor method and showed 80% of accuracy.

3. Feature Extraction Concept

There are many terms, used almost interchangeably, or the process of extracting the important features from a set of data, including “feature extraction,” “feature (subset) selection,” “data mining,” “information content” and “(feature) dimension reduction.” Whatever the term used, the process of feature extraction can be defined as “*a process of identifying valid, useful and understandable patterns in data*”. The large amount of time-series data that is generated in a gait analysis study is called “high dimensional” data. High dimensional data can be thought of simply as lots of data (there are lots of dimensions to the data). Fast developing gait measuring techniques and methodologies generate more and more data. This in turn results in what is known as the “curse of dimensionality” (more and more data to manage, more dimensions).

The objective of feature extraction is to keep all the useful features of the data and discard all the redundant parts of the data. There are two required outcomes to this process: (1) reduce the amount of data to a manageable level (dimension reduction), and (2) keep the most important features of the data and eliminate all the redundant features of the data (feature selection). The idea is to provide a “summary” which can be used to give a meaningful interpretation of the data. The objective of the first step towards the feature selection is a dimension reduction, to reduce the search space to a lower, more manageable dimensionality and this has to be achieved in a way that retains relevant features of the data and removes irrelevant features. That is, the feature selection selects “m” relevant features from the entire set of “n” features such that m<n. Noting the inherent redundancy in data, ideally m<<n, and ideally m contains all the relevant features and no irrelevant features of the data. Reducing the amount of data and extracting the key features are most often done in one of two ways in gait analysis, namely *time-series parameterization* (selecting key features from time-series data/graphs) or *data transformation* (transforming the original data set to a smaller set of numbers while preserving as much information about the original data set as possible, e.g., by Fourier transform or wavelet transform).

4. Learning Method Classes and Training Algorithm Stages

On the other hand, all approaches have limitations and efficiency strengths depending on sensor “hardware” that was used during information retrieving and used algorithms. A special focus of the paper is on mobile devices that are inherently wearable and equipped with GPS, accelerometers and so on that can be used to assess activity. Unsupervised learning is based on unlabeled data and applies:

(1) acquire unlabeled sensor data,
(2) aggregate and transform them into features,
(3) model data by e.g. clustering techniques.

The second broad category exploits logical modeling and reasoning. In this case we do the following:

(1) use a logical formalism to explicitly define and describe a library of activity models,
(2) aggregate and transform sensor data into logical terms,
(3) perform logical reasoning based on observed actions, which could explain the observations.

The algorithms and models for supervised learning and activity recognition include Hidden Markov Models (HMM), dynamic and naive Bayes networks [9-11], decision trees [12], nearest neighbor [4,13] and SVM [18] approaches. HMMs and Bayes networks are currently the most commonly used methods in activity recognition even though they require extensive computational resources. Multicore computers and clusters are typically used for these types of classifications. The number of machine learning models that have been used for activity recognition varies almost as greatly as the types of activities that have been recognized and types of sensor data that have been used. Solutions range from naive Bayes classifiers to support vector machines [1-6].

5. Feature Extraction and SVM Implementation for Acceleration Raw Signal Sequences

Many activity recognition systems use one or several wearable sensors attached to different parts of human body in order to collect data and transfer them to a nearby server station or to head web server through the Internet. Our current research does a recognition process based on a machine learning method. This means that training sets can be individually calibrated by user in order to increase the prediction accuracy. SVM classifier was applied to extracted feature vectors acquired from raw signal sequences. For each of primitive activities (walking, sitting, jogging, etc.) the user can input the number of patterns matching personally to his actions. These patterns were asynchronously stored in SQLite mobile database of smartphone (Fig. 1). The user is able to decide for how long to retrieve signals and what reading frequency to set (mobile application, designed for this purpose, has this options). Consider that we have a finite set of labels representing each activity – A = {$A_1 \ldots A_N$}. Each of training sets for each activity performed for Δt seconds. Let us denote as D the set of feasible distances between two neighbor points of two different instances of single activity A_i as follows: D = {$d_1, \ldots, d_N$}. Each activity has its own set of trained sequences, necessary for calibration. Let us denote them as follows:

$$TS\_A_i = \left\{TS^1_{A_i}, \quad \mathrm{K} \ , \quad TS^M_{A_i}\right\}, \ M \ge 0, 1 \le i \le N \ .$$

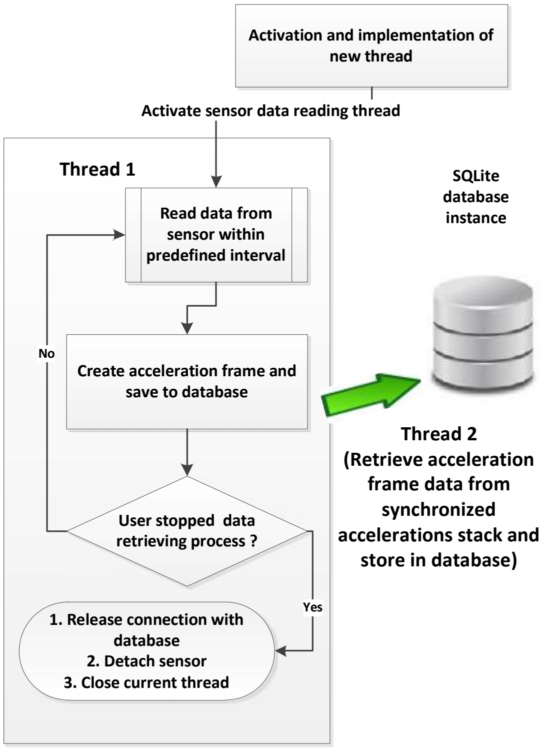
Activation and implementation of new thread
Activate sensor data reading thread
Thread 1
Read data from sensor within predefined interval
Create acceleration frame and save to database
User stopped data retrieving process ?
No
Yes
1. Release connection with database
2. Detach sensor
3. Close current thread
SQLite database instance
Thread 2 (Retrieve acceleration frame data from synchronized accelerations stack and store in database)

Fig. 1. Asynchronous process of raw acceleration signal saving

Each $TS^c_{A_i}, 1 \le c \le M$ element represents a sequence of r-dimensional vectors (r depends on sensor, from which data is acquired and feature vectors are calculated). Generally, each activity can hold different number of training sets, but for ease of marking, let us assume that all activities have the same signal length. Thus, we can represent the whole training set of activities with the following N×M matrix:

$$TS_A = \begin{pmatrix} TS^{1}_{A_1} & \Lambda & TS^{M}_{A_1} \\ \mathrm{M} & \mathrm{O} & \mathrm{M} \\ TS^{1}_{A_N} & \Lambda & TS^{M}_{A_N} \end{pmatrix}, \tag{1}$$

where every variable $TS^{j}_{A_i}$ has the following structure:

$$\begin{cases} TS^{j}_{Ai} = \left(p^{i\,j}_{1},\, p^{i\,j}_{2}, \mathrm{K}\ ,\, p^{i\,j}_{l}\right) \\ p^{i\,j}_{k} = \left(v^{i\,j}_{1\,k}, v^{i\,j}_{2\,k}, \mathrm{K}\ , v^{i\,j}_{r\,k}\right) \\ 1 \le i \le N, 1 \le k \le l, 1 \le j \le M\ , \end{cases} \tag{2}$$

So, here we deal with data cubes and each of the variables $v^{i\,j}_{s\,k}, 1 \le s \le r$, denotes a single feature, calculated from raw acceleration sequence of one axis. Once data has been collected and transferred on server, a feature extraction must be implemented. Unlabeled pattern must be processed through the following stages:

(1) noise reduction,
(2) feature extraction,
(3) learning and inference,
(4) activity recognition.

In order to do noise reduction we used the median filter with 10 sequential neighbor points comparing to eliminate noisy points (10-fold window). Generally, we must isolate certain time and frequency domain characteristics from the raw data. But in current research we focused on time frequency domain characteristics extraction. As we deal with tri-axial accelerometer signals, so during each time frame we acquired (x, y, z) – 3-dimensional vector of real values and our whole signal sequence can be represented as three arrays of *l* length or as a $3 \times l$ dimensional matrix. From this sequence we shall extract the following time domain features:

1. *Min* – minimum value of the selected array;
2. *Max* – maximum value of the selected array;
3. *Mean* – mean value of the selected array («);
4. *Standard deviation* – determines standard deviation of the selected array (±);
5. *Correlation* – correlation between pair of accelerometer axes (°);
6. *Median crossing* – number of zero crossings of the mean of the selected array.

So, we have 6 feature characteristics, each one of which is calculated from one data of one array. Thus, in our case we gain 18-dimensional feature vector based on time domain features (*r*=18). It is clear how the *Min* and *Max* functions for each axis calculate. The remained characteristics are represented as follows:

$$\text{«}_x = \frac{1}{l}\sum_{i=1}^{l} x_i\ , \tag{3}$$

$$\pm_x = \frac{1}{l-1}\sum_{i=1}^{l} (x_i - \text{«}_x)^2\ , \tag{4}$$

Variable $\circ_{xy}$ determines a correlation between the accelerometer X and Y axes. For the other two variables the value is calculated by the same formula (5):

$$\circ_{xy} = \frac{\sum_{i=1}^{l}(x_i - \text{«}_x)(y_i - \text{«}_y)}{\sqrt{\sum_{i=1}^{l}(x_i - \text{«}_x)^2 \sum_{i=1}^{l}(y_i - \text{«}_y)^2}}, \tag{5}$$

Here «_x and «_y are the mean values X and Y arrays, respectively.

Median crossing value is calculated in two steps. Firstly we get the median value of array and after that we calculate the number of zero crossings in a new array:

Step 1: Determine the *median* of array along x-axis.

1.1. Sort the array: $S_x = Sort(X)$,
1.2. Get the length of the array: $L_x = Length(X)$

$$median_x = \begin{cases} \frac{1}{2}\left(S_{x1} + S_{x2}\right), \ x1 = \left[\frac{L_x}{2}\right], \ x2 = \left[\frac{L_x}{2} + 1\right], if \ L_x \ array \ length \ is \ even \\ \frac{1}{4}\left(S_{x1} + 2S_{x2} + S_{x3}\right), \ x1 = \left[\frac{L_x}{2} - 1\right], \ x2 = \left[\frac{L_x}{2}\right], x3 = \left[\frac{L_x}{2} + 1\right], otherwise \end{cases} \tag{6}$$

Step 2: Calculate zero crossings.

2.1. Subtract the median value from the original array: $X_{Sub} = X - median_x$,
2.2. Count the number of zero crossings in new X_{Sub} array.

Having the list of sequences representing the given activity we constructed an average weighted sequence for each labeled activity using the *D* set of feasible bounds between the neighbor points as a limitation tool. As a result we gain one sequence of an average signal which represents an activity and each point in sequence, also has its own weight. So, when the unlabeled/test activity is compared with every instance of average activities set, along with the distance of points the importance of the comparing points will be calculated depending on the weight of the average point. Here we denote as L_{AVG} the set of average sequences representing each activity:

$$\begin{cases} L_{avg} = \left\{L^1_{avg}, \mathrm{K}\ , L^N_{avg}\right\}, \\ L^i_{avg} = \left(P^{i\ 1}_{avg}, \mathrm{K}\ , P^{i\ l}_{avg}\right), 1 \le i \le N \\ P^{i\ k}_{avg} = \left(q^{i\ k}_{avg}, p^{i\ k}_{avg}\right), 1 \le k \le l \\ p^{i\ k}_{avg} = \left(v^{i\ k}_{1\ avg}, v^{i\ k}_{2\ avg}, \mathrm{K}\ , v^{i\ k}_{r\ avg}\right), \end{cases} \tag{7}$$

Here L^i_{avg} is *N*-dimensional vector of objects, representing average instances for each labeled activity. Every component, in its turn, is a combination of $q^{i\ k}_{avg}$ non-negative weight value and $p^{i\ k}_{avg}$ *r*-dimensional point. Variable *r* denotes the number of feature vectors that define the selected activity (as well as the feature space dimension). Coordinates of average point calculated as mean of proper coordinates of the same time frames along each of axes and the average weight of specified point calculates below using the formula (8):

$$p^{ik}_{avg} = \frac{1}{M}\sum_{m=1}^{M} p^{ik}_{m} = \left(\frac{1}{M}\sum_{m=1}^{M} v^{ik}_{1m}, \quad \Lambda\ , \quad \frac{1}{M}\sum_{m=1}^{M} v^{ik}_{rm}\right), \tag{8}$$

$$q^{ik}_{avg} = \frac{1}{M-1}\sum_{m=1}^{M-1} sumw(p^{ik}_{m}, p^{i+1k}_{m}), \tag{9}$$

Here the sum of an average weight for neighbors of activity in given *i* time frame calculates as follows:

$$sumw(p_m^{i\ k}, p_m^{i+1\ k}) = \begin{cases} £_i\,, if \quad dist(p_m^{i\ k}, p_m^{i+1\,k}) > d_i \\ 1\,, \quad otherwise \end{cases} \tag{10}$$

where the value of variable 0≤ $£_i$ ≤1 is the weight decrease coefficient of those two neighbor points which have a distance more than feasible bound for the given activity class A_i. The distance itself is calculated as an Euclidean distance between the vector points as follows:

$$dist(p_m^{i\ k}, p_m^{i+1\,k}) = \sqrt{\sum_{c=1}^{r}(p_{cm}^{i\ k} - p_{cm}^{i+1\,k})^2}\ , \tag{11}$$

Fig. 2 illustrates the average graphic for one specified activity, based on calculations of mean points for each time frame of different training instances. Result graphic was estimated according to the values of signals of several labeled instances representing a specified activity.

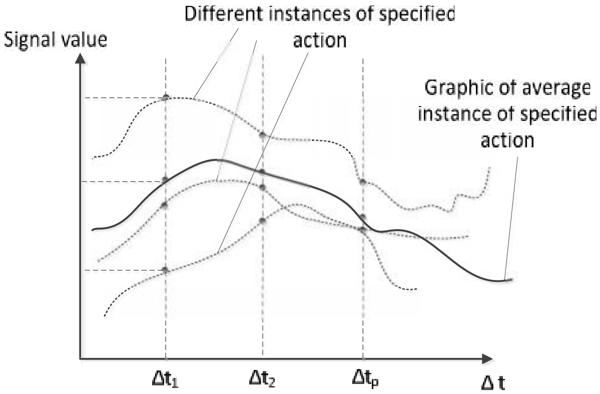

Fig. 2. Gaining the average instance of specified (labeled) activity based on all patterns of the same class. Generally, it represents average of the given activity type graphic of all patterns.

After that the unlabeled activity must be sequentially compared with average activities in order to find the most matching one:

$$weighted\_dist\,(L_{avg}^{i}, L_{unlabeled}) \rightarrow \min \tag{12}$$

On classification stage we used SVM algorithm [7, 18]. Having a set of predefined patterns allowed us to create hyper planes in order to distinguish different activities from each other using SVM implementation (Figure 3). Here is illustrated a binary classification case, when two sets are linearly separable (this is used when we compare a test set with sitting/standing/idle activities).

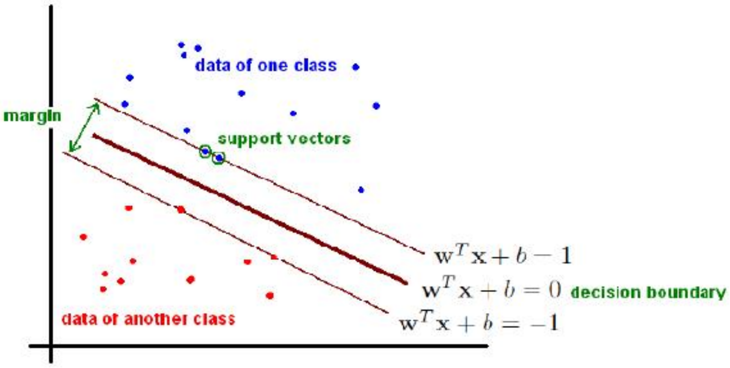

Fig. 3. General principia of decision boundary calculation. Binary classification with linearly-separable signal sets.

In that case we do the following steps:

1. Take two L_{avg}^{s} and L_{avg}^{h} labeled set S:

$$S=\left\{(x_i\ ,\ y_i)\,|\,x_i\in\{L_{avg}^{s}\cup L_{avg}^{h}\}=\{p_{avg}^{t\,i}\}_{t\in s,\,h}\overset{def}{=}P\ ,\ y_i\in\{-1\,,+1\},1\le i\le|\,L_{avg}^{s}\,|+|\,L_{avg}^{h}\,|\overset{def}{=}|\,P\,|\right\},$$

2. Consider that we have point “clouds” and we want to find the optimal function which describes the separation hyperplane:

$$f\,(x)=w^{T}x+b\ , \tag{13}$$

Here the vector *w* is an unknown vector and *b* is the bias. So, all points that are above hyperplane will belong to one class and the rest to the other (Fig. 3), thus, the following is correct for *w*:

$$\begin{cases} w^{T}x_i+b\ge+1, & for\quad y_i=+1\\ w^{T}x_i+b\le-1, & for\quad y_i=-1\end{cases} \tag{14}$$

So, the main goal is to maximize the margin from both sets by minimizing the norm of *w*:

$$\min_{(w,b)}\frac{1}{2}\,\|\,w\,\|^{2},\ subject\ \ to\ \ y_i\,(w^{T}x_i+b)-1\ge0,\forall i\ , \tag{15}$$

Here we transform the current problem to quadratic optimization problem (16) by applying the Lagrange multipliers $\sim_i\ \ge 0$ for $\forall i$ point from the target set to Lagrangian form (L_P) in (17):

$$\min_{(w,b)}\max_{\sim\ge0}\left\{\frac{1}{2}\,\|\,w\,\|^{2}-\sum_{i=1}^{|P|}\sim_i[\,y_i\,(w^{T}x_i+b)-1]\right\}, \tag{16}$$

$$\begin{aligned} L_P&=\frac{1}{2}\,\|\,w\,\|^{2}-\!\sim[\,y_i\,(w^{T}x_i+b)-1]\\ &=\frac{1}{2}\,\|\,w\,\|^{2}-\sum_{i=1}^{|P|}\sim_i[\,y_i\,(w^{T}x_i+b)-1]\\ &=\frac{1}{2}\,\|\,w\,\|^{2}-\sum_{i=1}^{|P|}\sim_i y_i\,(w^{T}x_i+b)+\sum_{i=1}^{|P|}\sim_i\ , \end{aligned} \tag{17}$$

After differentiating L_P with respect to *w* and *b* and setting derivatives equal to zero:

$$\frac{\partial L_P}{\partial w}=0\Rightarrow w=\sum_{i=1}^{|P|}\sim_i y_i x_i\,,\qquad \frac{\partial L_P}{\partial b}=0\Rightarrow\sum_{i=1}^{|P|}\sim_i y_i=0\ , \tag{18}$$

we get the Dual Form (L_D) from our Primary Form L_P:

$$\begin{gathered} L_D=\sum_{i=1}^{|P|}\sim_i-\frac{1}{2}\sum_{i,j}^{|P|}\sim_i\sim_j y_i y_j x_i x_j=\sum_{i=1}^{|P|}\sim_i-\frac{1}{2}\sum_{i,j}^{|P|}\sim_i H_{ij}\sim_j\,,\\ s.t.\ \sim_i\ \ge0\ and\ \sum_{i=1}^{|P|}\sim_i y_i=0,\ where\ H_{ij}=y_i y_j x_i x_j\ , \end{gathered} \tag{19}$$

As a result of solving the dual problem, we can compute the *w* from $\sim_i$ terms as follows:

$$w=\sum_{i=1}^{|P|}\sim_i y_i x_i \tag{20}$$

3. For comparing the rest of the activity types, i.e. walking, running and so on, we must extend the linear separable case by adding non-negative $\P_i$ (1≤i≤|P|) slack variables and use the “Soft Margin” method, introduced by V. Vapnik and C. Cortes in [20], and use a non-linear kernel function (radial basis function – RBF in our case).

$$\begin{cases} w^T x_i + b \ge +1 - \P_i, & for \quad y_i = +1 \\ w^T x_i + b \le -1 + \P_i, & for \quad y_i = -1 \\ \P_i \ge 0 \;\; for\ \forall i \end{cases} \tag{21}$$

So, here we adapt the objective function, represented in (22), and formulate the problem as shown in (23):

$$\min_{(w,b)} \frac{1}{2} \| w \|^2 + C \sum_{i=1}^{|P|} \P_i, \; subject \;\; to \;\; y_i (w^T x_i + b) - 1 + \P_i \ge 0, \forall i \, , \tag{22}$$

$$\min_{(w,\P,b)} \max_{\sim,\text{«} \ge 0} \left\{ \frac{1}{2} \| w \|^2 + C \sum_{i=1}^{|P|} \P_i - \sum_{i=1}^{|P|} \sim_i [y_i (w^T x_i + b) - 1 + \P_i] - \sum_{i=1}^{|P|} \text{«}_i \P_i \right\}, \tag{23}$$

where the parameter *C* controls the trade-offs between the slack variable penalty and the size of the margin between dissecting hyperplanes. Reformulating as a Lagrangian, which as before we need to minimize the computations with respect to variables *w, b,*¶ and to maximize with respect to variables $\sim \ge 0$ and $\text{«} \ge 0$:

$$L_P = \frac{1}{2} \| w \|^2 + C \sum_{i=1}^{|P|} \P_i - \sim [y_i (w^T x_i + b) - 1 + \P_i] - C \sum_{i=1}^{|P|} \text{«}_i \P_i \; , \tag{24}$$

and can compute the result *w*.

As SVM was originally designed for a binary classification, it cannot deal with a multi-class classification directly. The multi-class classification problem is usually solved by a decomposition of the problem into several two-class problems. In this paper, we used One-Versus-One Strategy (OVO), where a set of binary classifiers are constructed using the corresponding data from two classes. After the weighted distances between all the classified pairs have been calculated, we used the voting strategy of “Max-Wins” to produce the output.

6. Conclusion

Smartphone is a new category of mobile phones that can perform computing just like a personal computer, but with smaller resources capability. Many sensors are already embedded smartphones which, thus, can be considered as a perfect tool for short-term physical activity recognition. Thus, these wearable devices can be used for unobtrusive activity recognition. Smartphones are also able to provide a wide range of connectivity option in one integrated device. Ultimately, these devices have been very personalized in human’s daily life so that the implementation using a smartphone will relieve the users to carry wearable sensors creating discomfort. Our approach introduces a method of classifying raw signals transferred from a smartphone. We constructed the set of feature vectors from raw acceleration sequences. Then we built the set of average signal sequences for the given activity (one average instance for each activity) and generated weights for each average point of each instance. Finally, we applied SVM algorithm to the constructed data and gained a classified activity. Our future work is targeted to process the classification stage straight on the mobile device and implement the real-time classification by using not only time domain features, but also frequency domain features in order to increase the classification accuracy.

References

[1] M. Oner, J. A. Pulcifer-Stump, P. Seeling and T. Kaya, “Towards the run and walk activity classification through step detection – An android application”, *34th Annual International Conference of the Engineering in Medicine and Biology Society*, electronic journal, doi: 10.1109/EMBC.2012.6346344, San Diego, CA, 2012.

[2] V. N. Vapnik, *Statistical Learning Theory* , New York: John Wiley & Sons, 1998.

[3] S. Das, L. Green, B. Perez, M. Murphy and A. Perring, “Detecting user activities using the accelerometer on Android smartphones”, Carnegie Mellon University, CMU, Technical Report, 2010.

[4] S. Kaghyan and H. Sarukhanyan, “Activity recognition using K-nearest neighbor algorithm on smartphone with Tri-axial accelerometer”, *International Journal of Informatics Models and Analysis (IJIMA)*, ITHEA International Scientific Society, Bulgaria, pp. 146-156, 2012.

[5] J. Parkka, M. Ermes, P. Korpipaa, J. Mantyjarvi, J. Peltola and I. Korhonen, “Activity classification using realistic data from wearable sensors”, *IEEE Transactions on Information Technology in Biomedicine*, vol. 10, no. 1, pp. 119-128, 2006.

[6] A. Mannini and A. M. Sabatini, “Machine Learning Methods for Classifying Human Physical Activities from on-body sensors”, *Sensors 10*, pp. 1154–1175, 2010.

[7] Z. He and L. Jin, “Activity recognition from acceleration data based on discrete cosine transform and SVM”, *Proceedings of the 2009 IEEE International Conference on Systems, Man and Cybernetics*, San Antonio, USA, pp. 5041-5044, 2009.

[8] S. Chernbumroong, A. S. Atkins and H. Yu, “Activity classification using a single wrist-worn accelerometer”, *5th IEEE International Conference on Software, Knowledge, Information Management and Applications*, Benevento, Italy, pp. 1-6, 2011.

[9] J. Yang, B. N. Schilit and D. W. McDonald, “Activity recognition for the digital home”, *IEEE Computing and Processing*, vol. 41, pp. 102-104, 2008.

[10] A. A. Efros, A. Berg, G. Mori and J. Malik, “Recognizing Action at a Distance”, *Proceedings of the International Conference on Computer Vision*, Nice, France, pp. 726-733, 2003.

[11] E. Tapia and S. Intille, “Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart rate monitor”, *International Symposium on Wearable Computers (ISWC)*, pp. 37-40, 2007.

[12] S.-W. Lee and K. Mase, “Activity and location recognition using wearable sensors”, *IEEE Pervasive Computing*, vol. 1, no. 3, pp. 24-32, 2002.

[13] T. Huynh and B. Schiele, “Unsupervised discovery of structure in activity data using multiple eigenspaces”, *2nd International Workshop on Location- and Context-Awareness (LoCA)*, 2006.

[14] N. Ikizler and P. Duygulu, “Human Action Recognition Using Distribution of Oriented Rectangular Patches. In Human Motion”, ICCV’07, Rio de Janeiro, Brazil, pp. 271-284, 2007.

[15] M. Fikri Azli bin Abdullah, A. Fahmi Perwira Negara, S. Md. Sayeed, D. Choi and K. Sonai Muthu, “Classification algorithms in human activity recognition using smartphones”, *International Journal of Computer and Information Engineering 6*, pp. 77–84, 2012.

[16] P. D. Thompson, “Exercise and physical activity in the prevention and treatment of atherosclerotic cardiovascular disease”. *Statement from the Council on Clinical Cardiology (Subcommittee on Exercise, Rehabilitation, and Prevention) and the Council on Nutrition, Physical” Arteriosclerosis, Thrombosis, and Vascular Biology*, vol. 23, pp.42-49, 2003.

[17] S. Kaghyan, H. Sarukhanyan and D. Akopian, “Human movement activity classification approaches that use wearable sensors and mobile devices”, *IS&T/SPIE Electronic imaging symposium, Conference on multimedia and mobile devices*, vol. 8667, Electronic Journal, doi: 10.1117/12.2007868, Burlingame, CA, 2013.

[18] Z.-H. He and L.-W. Jin, “Activity recognition from acceleration data using AR model representation and SVM”, *Proceedings of the Seventh International Conference on Machine Learning and Cybernetics*, Kunming, pp. 12-15, 2008.

[19] C. Burges, “A tutorial on support vector machines for pattern recognition”, *Data Mining and Knowledge Discovery*, pp. 121-167, 1998.

[20] C. Cortes and V. Vapnik, “Support-vector networks”, *Machine Learning*, vol. 20, pp. 273-297, 1995.

Submitted 22.08.2013, accepted 04.10.2013.

Շարժման ճանաչումը բջջային հեռախոսի ազդանշաններից ժամանակային հատկանիշների առանձնացման և SVM մեթոդի կիրառման միջոցով

Ս. Կաղյան և Հ. Սարուխանյան

Ամփոփում

Ժամանակակից շարժական սարքերը հնարավորություն են տալիս լուծելու տարատեսակ խնդիրներ։ Այդպիսի խնդիրներից մեկն է՝ «բջջային հեռախոսի միջոցով ճանաչել և դասակարգել մարդու շարժումը» խնդիրը։ Այդ գործում մեծ նշանակություն ունեն այն տվիչները, որոնցով համալրված են սարքերը։ Ներկա հոդվածում ներկայացված է վերջին տարիների որոշ աշխատանքների վերլուծությունը։ Այնուհետև ներկայացված է աքսելերոմետրի «հում» ազդանշանների հաջորդականությունից ժամանակային հատկանիշների առանձնացման մեթոդը։ Ստացված հատկանիշների հետագա մշակումը թույլ տվեց կատարելու մարդու կողմից կատարված շարժումների դասակարգումը։ Հատկանիշների վերլուծման և վերջնական ճանաչման համար կիրառվել է «ուսուցանում ուսուցչի հետ» մոտեցումը՝ հենքային վեկտորների մեթոդի օգտագործմամբ։

Распознавание движений с помощью сигналов датчиков мобильных устройств путем выделения временных характеристических значений и применением метода опорных векторов

С. Кагян и А. Саруханян

Аннотация

Современные мобильные устройства дают возможность решать самые разные задачи. Одной из таких является задача распознавания и классификации человеческой активности при помощи сенсоров мобильных устройств. В этом случае непосредственную важность имеют сами сенсоры, с которых происходит считывание данных. В данной работе представлен сравнительный анализ исследований в данной области за последние несколько лет. Затем, представлен метод из “сырых” последовательностей сигналов акселлерометра мобильного устройства выделения временных характеристических данных, в число которых входят такие величины, как среднее по координатным осям, стандартное отклонение, коэффициент корреляции и т.д. На основе полученных величин была сделана дальнейшая обработка характеристических величин с целью конечной классификации. Сам процесс окончательной классификации был сделан по принципу “обучение с учителем” – с применением метода опорных векторов (SVM алгоритма).