Human Motion Classification using 3D Accelerometers: Comparison between Neural Network and Hidden Markov Model Approaches

B. Bukvic*, P. Schaller*, P. Celka** and T. Klinger***

* Carinthia Tech Institute/Electronics and Signal Processing Group, Villach, Austria
** Griffith University/School of Engineering/Electronics and Signal Processing Group, Gold Coast, Australia
*** Carinthia Tech Institute/Electronics and Signal Processing Group, Villach, Austria

E-Mail Address: e0114buba@edu.fh-kaernten.ac.at

Abstract: The aim of this project was to develop algorithms for analysing and classifying daily life human motion activities using acceleration signals. Capturing human motion is crucial for reporting pathological motion behaviours of humans in their daily practice and can be used for further diagnosis by medical doctors. The acceleration signals are acquired with accelerometers placed on human chest. The classes of activity of interest are: resting, running, walking flat, walking upstairs, and walking downstairs. All other activities that may occur are classified as "others". Two different classifiers, with different feature sets, are used for the classification: Neural Network and Hidden Markov Model. Comparing the values for sensitivity and specificity can be noticed that all human motion activities can be classified with sensitivity higher than 80%. The choice of the classifier depends on the particular application and on the available computing resources.

Introduction

Human motion has been studied for a long time by the medical community in order to understand the biomechanics of movements. Nowadays, the most important and well recognised health problems in the occidental world are cardiovascular diseases from which obesity, stress and depression are the main active factors. Capturing motion is crucial for reporting pathological motion behaviours of humans in their daily practice and can be used for further diagnosis by medical doctors.

In order to realise the human motion classification a Pattern Recognition System (PRS) has been developed (Figure 1) applying the machine learning approach. The main goal in machine learning is to extract necessary information from a set of input data in order to build an appropriate system. By developing the system it is advantageous to have good priori knowledge about the problem to be solved. The Pattern Recognition System consists basically of two main parts [1]:

- Feature extraction from a data input;
- Classifier.

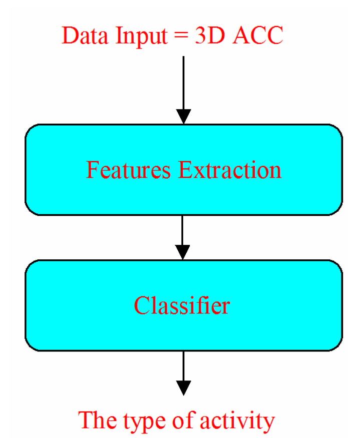

Figure 1: Pattern Recognition System

The main task of the *Feature Extraction* is to find significant features for different human motion activities using the provided acceleration signals (3D ACC). Significant features make the distinguishing between different human motion activities/classes possible. That is, their values are very similar for objects which belong to the same class, and very different for objects in different classes.

This fact is very important for the following task of the *Human Motion Classification*, because significant features are crucial for a successful classification. Two different classifiers were developed in this step, Neural Network and Hidden Markov Model. Each of them uses a specific set of extracted features which are passed over by the feature extractor. The task of the used classifier is to make the final decision and to classify the human motion activities which were performed. We have been interested in the following classes: 1) Resting, Running, Walking Flat (WF), Walking Upstairs (WUP) and Downstairs (WD).

The motion classification is subject independent, but is not suitable for people with abnormal gaits or people with other physical problems like clinical obesity.

Materials and Methods

Signals and Data: For recording the data a 3D accelerometer sensor (Figure 2), developed and built at

 ISSN: 1727-1983 © 2005 IFMBE

the Griffith University, was used. The sensor is microprocessor controlled (Hitachi H8/300H family microprocessor) and consists out of two 2D accelerometers (Analog Device ADXL 202E) [2].

Figure 2: 3D – accelerometer device

The two accelerometers allow the 3D motion recording and thus, accelerations signals from three different directions are available:

- Lateral – along the shoulders;
- Vertical – along gravity;
- Frontal – perpendicular to the thorax.

The device records the data at a sample rate of 150 Hz and stores it to an on-board 16 MB+ memory card [2]. The data can be transferred to a PC using the serial interface (RS232). For the classification, the recorded signals were resampled to a sample rate of 40 Hz. During the recording process the sensor was fixed to the person's chest using two flexible belts so that the sensor followed the movement without slipping or waggling. The data was stored in ".csv" format so that it can simply be loaded to MATLAB ®.

The signals of 10 subjects, including male and female, aged 20-40 years old were recorded. All subjects used the same recording protocol which included all six activities of interest.

Before extracting features, the recorded signals have to be divided into equal long parts because it is not possible to detect features from a whole data file at once. This process is called windowing. The length of a window depends on the dynamic of the given process and has a big impact on the results of the classification. On the one hand a large window has a small time resolution. That means that activities that took place for a short time (shorter then the length of the window) will hardly be found. On the other hand if the window is too short, there will be a high time resolution but also a high error rate because some of the features don't work well with a small window. In this project two windows with the length of ten and fife seconds have been implemented.

Features: After recording data the first step for a successful activity classification was to extract significant features from the recorded data. Features are signal properties that allow deciding between different activities. Those features have to be robust, immune to noise and subject independent. Because each classifier has another approaches two different feature sets had to be developed:

- For the NN: main peak location of the power spectral density (PSD), main peak amplitude of the PSD, spectral entropy, peak to peak rhythmical amplitude and lowpass filtering of the lateral acceleration signal.
- For the HMM: variance, spectral entropy, peak to peak rhythmical amplitude and lowpass filtering of the lateral acceleration signal.

The PSD [3] describes the distribution of the power of a signal over its frequencies. The PSD $P_{SD}(n)$ for a given discrete-time signal $x(n)$ can be calculated as:

$$\tilde{x}(n) = DFT(x(n)) \tag{1}$$

$$P_{SD}(n) = \frac{1}{N}\sum_{n=0}^{N-1}(|\,\tilde{x}(n)\,|)^2 \tag{2}$$

Detecting the location of the *main peak in the PSD* (i.e. the frequency at which the activities highest power expenditure takes place) can thus be used as a feature, because the main peak for different activities will be located at different frequencies. While, for the Running, the PSD consists of a main peak at 3 Hz (Figure 3) the PSD for the activity WF has its main peak at 2 Hz and some other slightly peaks at frequencies above and below the main peak's frequency. The PSD for activities WUP and WD looks similar to the WF PSD.

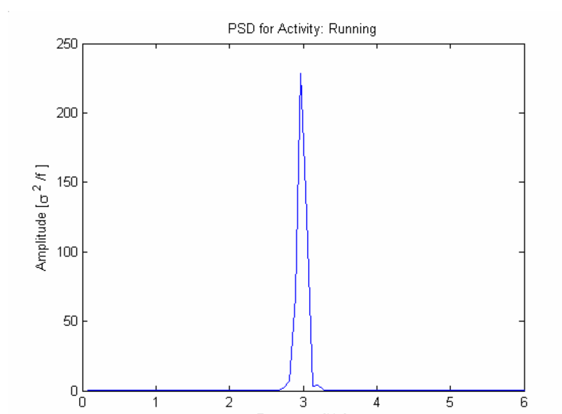

Figure 3: PSD for activity Running

There is not only a difference in the PSD's frequency location of the main peak, but also in its *main peak amplitude*. Activities with high energy expenditure also show high energy peaks in their PSD. Resting is the activity with the lowest energy expenditure. There is a significant difference in the amplitudes between the activity Resting on one side and the remaining activities on the other side. Therefore is this feature used to separate Resting from other activities.

Spectral entropy [4] is one of the information's theory concepts. It is used to measure the information

 ISSN: 1727-1983 © 2005 IFMBE

content and to find some redundancy in the signal in order to distinguish rhythmic and non-rhythmic activities. To calculate the spectral entropy of a given signal window the PSD of the signal has to be calculated first (1), (2). The next step is to calculate the probability density function PDF which is for a PSD of the length L defined as:

$$P_{DF}(n) = \frac{P_{SD}(n)}{\sum_{n=1}^{L} P_{SD}(n)} \tag{3}$$

Once computed the PDF the spectral entropy is calculated as following:

$$SE = -\sum_{n=1}^{L} P_{DF}(n) * \log_2(P_{DF}(n)) \tag{4}$$

Rhythmical signals have lower entropy values, while non rhythmical signals have higher values. The unit used for the measurement of the entropy depends on the base used for the algorithms. In this case the base is two, therefore is the unit in bits.

The feature *lowpass filtering of the lateral acceleration signal* is based on the specifics of human movement. When somebody walks upstairs the person's upper body moves to the left and to the right much more than by any other activity that should be classified. That can be shown by using a low pass filter by the lateral acceleration signal with a cut-off frequency of 0.1Hz. After lowpass filtering, a significant difference in amplitudes between walking activities was detected. Concrete, the amplitude value of the activity WUP is significantly higher then by the remaining walking activities (Figure 4).

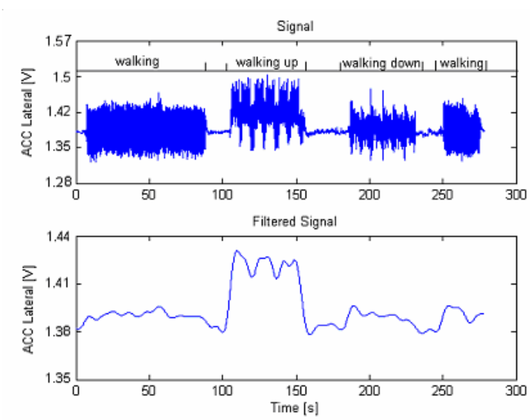

Figure 4: Lateral signal before and after applying lowpass filtering

When somebody is walking downstairs the recorded signals show some specific characteristics because the motion has to be stopped at each stair which leads to higher amplitudes in the vertical acceleration signal. That can be shown with the help of the feature *peak to peak rhythmical amplitude*. First all maxima and minima for the considered window have to be located. After locating the minima and maxima the computing algorithm detects the two highest maxima and the two lowest minima of them. The difference between the average of the two maxima and the two minima is the value used for this feature.

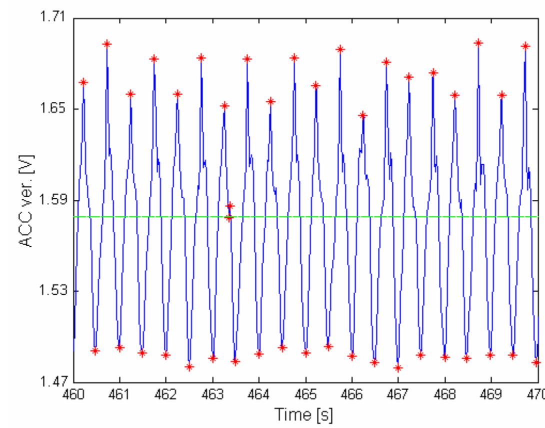

Figure 5: An example for the feature *peak to peak rhythmical amplitude*

Variance is the last feature of the signal *x(n)* which is used for the classification. It is the measure of the variability of *x(n)*. It is computed in the following way:

$$\sigma^2 = \frac{\sum (X-\mu)^2}{N} \tag{5}$$

*Classifier***:** In order to classify the desired activities, two different machine learning approaches are used "Neural Network" (NN) and the "Hidden Markov Model" (HMM).

Neural network models are algorithms inspired by the biological nervous systems [5, 6]. Neural Networks are composed of several layers, where each layer consists of several simple elements, the so called neurons. The numbers of layers and the number of elements in each layer depend on the problems these networks have to solve. Neural Networks can be trained both supervised and unsupervised. The network used in this project consists out of three layers - an input (I) a hidden (H) and an output layer (O).

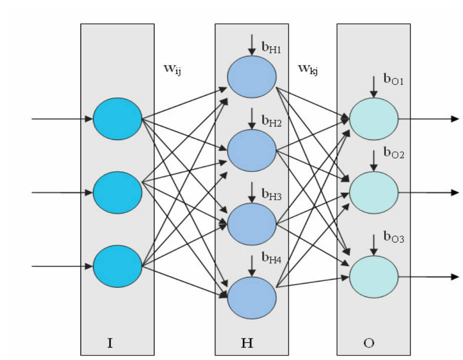

Figure 6: Three layer neural network

The input layer consists of three inputs, the hidden layer of four neurons and the output layer of three neurons. The network is fully connected which means that each unit in each layer is connected to each unit in the layer above. Each connection is weighted with a specific weight (w_{ij}, w_{kj}) and each neuron has an additional weighted input – the bias (bH1 – bH4, bO1 – bO3). Out of the three dimensional input vector the network creates a three dimensional output vector (Figure 6).

Backpropagation is the most general method used for supervised training of neural networks. In the backpropagation-algorithm, input training vectors and a corresponding target vector are used to train the network to approximate a function [6]. The backpropagation-algorithm involves two phases: The forward phase and the backward phase. Before the learning algorithm can start, the weights and biases have to be randomly initialised. In the forward phase an input vector is presented to the input layer. The outputs of the units in the input layer equal the values in the input vector. Now each neuron in the hidden layer adds up the weighted sum of all its inputs and creates an output signal. The same step happens again between the neuron of the hidden and the output layer. The output, created by the output layer neurons is saved. During this phase of the backpropagation algorithm all weights and biases don't change. The forward phase ends by calculating the output error which is the squared difference between the output target and the actual output, over all output units. In the backward phase the output error is propagated back to update the weights and biases and thus to minimize the output error.

The first-order discrete HMM *M* is a stochastic generative model for time series [1, 7]. A HMM consists of a finite set of hidden states $\omega_i(t)$ which can not be direct considered but indirect over their visible states $v_i(t)$. Each of the hidden states $\omega_i(t)$ emits one visible states $v_i(t)$ for each time *t*. In our case hidden states are represented by the activities i.e. they are six of them.

Visible states are represented by *regions/clusters* built using previous extracted features. Once represented by features Entropy and Variance and once represented by features Lowpass filtering of the lateral signal and Peak to Peak Amplitude (Figure 7). The cluster separation is done using linear discriminant analysis. For every window feature vector, consisting of all used features, is computed and according to its location in the feature space a visible state is detected. Thus, an observation sequence *O* is produced.

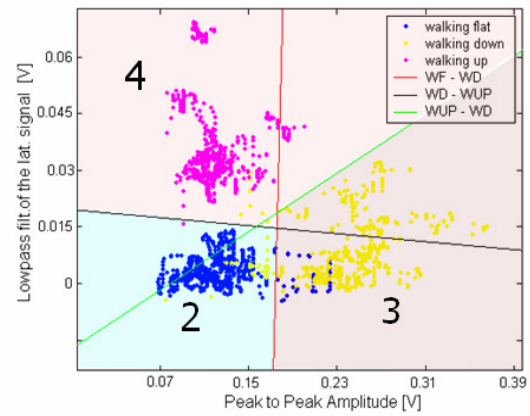

Figure 7: Visible states for *Walking Upstairs (WUP)*, *Walking Downstairs* (WD) and *Walking Flat (WF)*

Basic problems in HMM's are solved using following algorithm [1]:

- Evaluation problem – Forward algorithm
- Decoding problem – Viterbi algorithm
- Training/Learning – Baum-Welch algorithm

Results

The classifiers where tested against each others with the same sets of data. Both classifiers show very good results. Comparing the values for sensitivity and specificity you will notice that all activities can be classified with sensitivity higher than 80%. There are some activity specific differences in the quality of the classification results but the overall performances of the two classifiers are pretty much the same. The following table shows the results of the tests:

Table 1: Specificity and Sensitivity

Activity	HMM		NN	
	Spec. [%]	Sens.[%]	Spec.[%]	Sens.[%]
WF	0.96	0.86	0.95	0.94
WUP	0.98	0.90	0.99	0.96
WD	0.98	0.81	0.96	1
Running	1	1	1	1
Resting	0.96	0.95	1	0.86
Others	0.97	0.81	0.99	0.82

Spec: Specificity; Sens: Sensitivity;

Figure 8 visualizes the results of the classification of a data file that contains all six activities of interest. The classification algorithm is able to classify all six activities successfully. The calculated "Good Detection Rate" (GDR) for this example is 89.03%.

ISSN: 1727-1983 © 2005 IFMBE

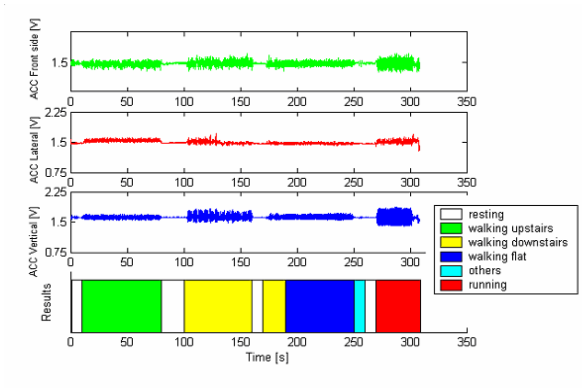

Figure 8: Results example

Discussion

The results presented in this paper may give the impression that both classifiers are equally applicable in the field of human motion classification but that is not true. The choice of the classifier depends on the particular application and on the available computing resources. The HMM classifier doesn't only work with state, but also with the time information. That implies that in an HMM approach the results of the current activity classification (at time t) has an impact onto the next classification (at time t+1). It is also very important to have a sufficient knowledge about the problem to solve. This helps to set initial values of the HMM's parameters before training. Thus, a faster convergence of the training algorithm is possible. A possible application would be the monitoring of a sportsman workout during a interval training.

The NN classifier does not include the time information into the classification process and can therefore be used in a widely spread field of applications. A possible application would be a 24 hour monitoring of a medical patient to record his daily amount of activities and to show the influence of his usually daily exercise level on his illnesses.

All the described features and algorithms have been fully developed and tested, but there still are some possibilities to improve the classification, like dividing the class "others" into some subclasses.

Conclusions

Capturing motion can be used in today's medicine to report pathological behaviours or other sickness linked to the skeleton and muscles. But motion classification can also be used in sport to improve training methods or to optimize motion sequences.

For a successful motion classification finding good and significant features is essential. These features have to be robust, subject independent and immune to noise. In this project six different features have been implemented. The size of the signal window has a big impact on the accuracy of the algorithm. There is always a trade-off between the size of the signal window and the desired time resolution of the algorithm. A large window creates accurate results because most of the features create better results, but there is a low time resolution. Two different signal windows have been implemented.

Hidden Markov Models own powerful and efficient algorithms which make the human motion classification possible. Advantages of those algorithms are especially their relative simple mathematical fundamentals and they make a successful implementation in the MATLAB programming environment possible. Those relatively simple mathematical fundamentals allow also fast computation and thus, also easier implementation by real applications (there are no integrals, derivates, squares and roots).

Multi layer neural networks and their possibility to create non linear output mappings showed themselves to be a very powerful tool for this kind of decision making and created very good results. Backpropagation, the most widely used method for supervised training of neural networks, has thus been implemented to train the network.

Acknowledgements

We would like to thank Dr Daniel A. James and M. Andrew Wixted for providing the 3D accelerometer device.

References

[1] R. O. DUDA, P. E. HART, and D. G. STORK (2001): 'Pattern Classification', 2nd Edition ed: 'Wiley Interscience'
[2] D. A. JAMES, N. DAVEY, and L. GOURDEAS (2003): 'An integrated platform for micro sensor data acquisition, storage and analysis', 'Griffith University'
[3] W. C. S. TODD and K. MOON (2000): 'Mathematical Methods and Algorithms for Signal Processing', Prentice Hall
[4] G. E. POWELL and I. C. PERCIVAL (1979): 'A Spectral entropy method for distinguishing regular and irregular motion of Hamiltonian systems'
[5] B. MÜLLER and J. REINHARD (1990): 'Neural Networks - An Introduction', 'Springer Verlag'
[6] H. DEMUTH and M. BEALE (2004): 'Neural Network Toolbox - For use with Matlab', 4 ed:, 'The MathWorks'
[7] P. BALDI and S. BRUNAK (2001): 'Bioinformatics', pp. 81-91,165-222.

ISSN: 1727-1983