A Personalised Body Motion Sensitive Training System Based on Auditive Feedback

Gerold Hoelzl

Johannes Kepler University, 4040 Linz, Austria
gerold.hoelzl@gmail.com

Abstract. In this paper the architecture and functionality of a personalized body motion sensitive training system based on auditive feedback is discussed. The system supports recognition of body motion using body worn sensors and gives the user feedback about his or her current status in adaptively selecting audio files accompanying the speed and path of exercise.

1 Introduction

1.1 Motivation

Being an enthusiastic sportsman, jogging (running) is one of my favourite sports. During the training sessions it is fun to listening to music, e.g. from an mp3 player, this also makes the training more amusing.

One problem is that the rhythm of the played audio file doesn't always fit to the running frequency of the exercising person. This can be very distracting.

The solution described in this paper is a system that adaptively selects audio files fitting to the rhythm of the runner. If the running rhythm changes during training (e.g. due to a power-up, exhaustion or change in terrain), the played music is automatically adapted to the new running rhythm.

Additionally the system can provide the user with status information like pulse, speed & distance or warn the user if predefined limits in those parameters are exceeded.

The aim of the project was to design a mobile, wearable system capable of fulfilling the above mentioned characteristics and to show its technical feasibility. The paper is structured as follows:

Part 2 identifies necessary system tasks needed to build such a system. Each task is described in detail and a framework is presented showing the integration of the different tasks into the whole system.

Part 3 focuses on developing a prototypical system. Requirements for hardware- and software components are defined and the prototype and its hardware components are presented.

Part 4 discusses and analysis the experimental results gathered from testing the prototype.

Part 5 concludes the paper giving a summery of the work and an outlook on future enhancements of the system.

S. Ystad et al. (Eds.): MobiCASE 2009, LNICST 35, pp. 12–25, 2010.

1.2 Related Work

Being an enthusiastic sportsman and engineer as mentioned in 1.1, I'm always having a look at actual developments at the sports tool sector.

One interesting product from a cooperation of Nike and Apple is the so called "Nike+iPod" (http://www.apple.com/at/ipod/nike/). It supports the runner giving feedback via an iPod about the current running speed, distance and practicing time while the sportsman can listen to his favourite songs. It consists of special designed running shoes from Nike, an iPod from Apple and a sensor that is used to determine the running speed. The major disadvantages of the product are that on the one hand at least an iPod is needed for the audio feedback and on the other hand only a handful shoes from Nike support the needed sensor. (Runners mostly have their favourite shoes and don't want to change them).

Playlists and song playback have to be generated manually and don't adapt to changes of the context of the sportsman. There also is no support to monitor the heart rate (ECG) or the running route (GPS) and a MAC-computer is needed for further analysis of the data.

Especially for sportsmen it is important to get information about their biosatus during training. Actual projects like AMON, A Wearable Medical Computer for High Risk Patients [1] show that it's no problem to continuously monitor, analyse and log multi sensor biodata like heart rate, blood pressure, blood oxygen saturation and temperature even for high risk patients in a wearable watch like form.

This project didn't require to measure all physiological parameters and to use a device qualified for medical purpose. To get the most interesting biofeedback parameter for sportsmen, the heart rate, a Polar ecg-sensor was used (3.1).

A substantial amount of research has been performed in the area of wearable computing and context recognition. Many independent researchers have demonstrated the suitability and excellent further potential of body worn sensors for automatic context and activity recognition [2].

The available scientific literature reports about successful applications of such sensors to various types of activities, ranging from the analysis of the simple walking behaviour [3] to more complex tasks of everyday life like e.g. the recognition of Wing Chun movements [2] and even workshop assembly [4].

Many past works have demonstrated 85% to 95% recognition rates for ambulation, posture and other activities using acceleration data. Advances in miniaturization will permit accelerometers to be embedded withing adhesive patches, belts, wrist bands and bracelets and to wirelessly send data to a mobile computing device that can use the signals to recognize user activities [5]. This made the use of an acceleration sensor for analyzing the movement of a person the best option. An overview of projects using accelerometers to detect user activity is given in [5].

A more advanced, actual project dealing with sports activity is the "wearable trainer" for nordic walking mentioned in [6].

The aim of the project is to monitor user motions and ensure that the user gets the maximum benefit of the exercise while minimizing risk factors such as joint damage or overextension. It consists of unobtrusive body fixed sensors and correlates sensor signals, terrain data and user position. Inferring Motionpatterns from the collected data enables the possibility of teaching the practicing person doing the correct motion sequence and getting most out of its training.

So the "wearable trainer" tries not only to react on the behaviour of the user but rather to teach the user doing the right motion sequence and giving tips and analysis on how to improve the motion.

2 System Design

In order to adaptively select audio files corresponding to the actual running rhythm of the user the system has to handle three main tasks:

- analyzing the movement of the practicing person (2.1)
- analyzing the audio files for classifying them(2.2)
- mapping the movement-feature to the audio-feature to be able to select the correctly fitting music samples (2.3)

The development framework used for the system is described in section 2.4 showing how the above mentioned parts are working together.

2.1 Movement Analysis

The goal of the movement analysis is to calculate a feature corresponding to the running rhythm of the user.

According to Hay [7], a footstep can be split up into 13 sections (Figure 1). The important phases for analyzing the movement are phase 7, 8 and 9. During this sections defined as "front support phase (7)" and "rear support phase (8,9)" the ground-pressure of the foot leads to a peak.

Monitoring the footstep with an acceleration sensor [8] placed rear hip (on backbone origin) shows that the vertical acceleration during a footstep in a "support phase" passes the 1,8g gravitational acceleration. In all other phases ("non supported phases") it stays far beneath as seen in figure 2.

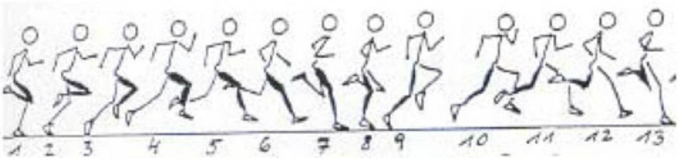


Fig. 1. Movement sections

The acceleration data shown in figure 2 were collected during a walk with moderate speed at around 4 km/h. The sensor was placed rear hip (on backbone origin). The faster the tempo is, the higher the acceleration value gets. This makes a detection of a step more easier because the spread of gravitational acceleration between a "support phase" and a "non support phase", indicating a footstep, is increasing.

The effect of different heights and weights of different people is minimal and only influences the spread of gravitational acceleration between a "support phase" and a "non support phase". The more lightweight a person is, the smaller the spread gets but easily passes the 1,8g acceleration value.

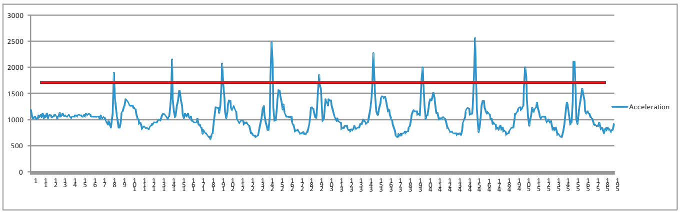
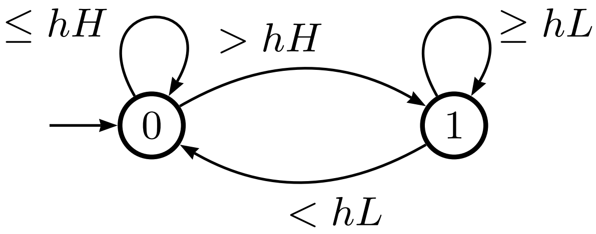


Fig. 2. Vertical acceleration

Based on knowing the characteristics of the acceleration values during a footstep, filtering the raw acceleration sensor data using a threshold filter with a hysteresis to avoid oscillations between states (as shown in figure 3) extracts the single footsteps.

Data analysis showed that a threshold value of 1,75g performs well in extracting the single footsteps. To avoid oscillations, the threshold values of the filter were set to 1,8g for high state (hH) and 1,7g for low state (hL). Figure 4 shows the raw acceleration sensor data overlaid with the extracted footsteps using a threshold filter as shown in figure 3, adjusted to the above mentioned threshold values.



- **hH**. . . threshold for high state
- **hL**. . . threshold for low state

Fig. 3. Schema of the used threshold filter with hysteresis

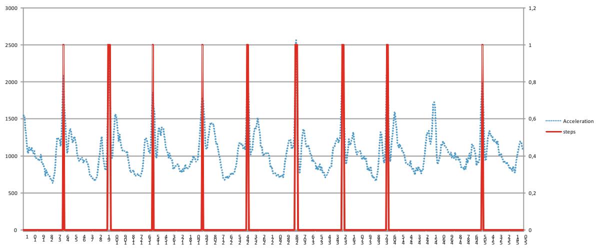


Fig. 4. Raw acceleration data overlaid with extracted footsteps

Having extracted the single footsteps out of the raw acceleration sensor data (figure 4) it is possible to define a feature that classifies the running rhythm over a specified time frame.

The feature defined as the *mean footsteptime mft[ms]* (1) is the arithmetic mean over the time between the single footsteps of the user over a specified time frame (illustrated in figure 5).

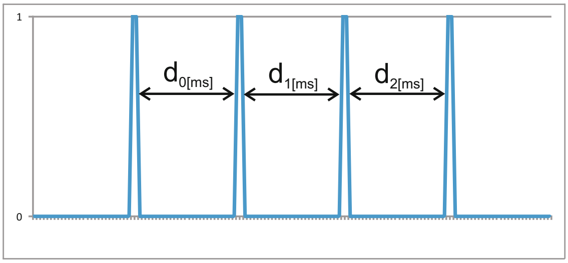


Fig. 5. Distances between footsteps to calculate mft

$$mft_{[ms]} = \frac{1}{n} \sum_{i=0}^{n-1} d(i)_{[ms]} \tag{1}$$

where:
n . . . number of distances [ms] between the single footsteps
$d(i)$ . . . distance between footsteps [ms].

2.2 Audio Analysis

To be able to map the running rhythm to music, a feature has to be extracted from the music that correspondences with its speed.

One feature that fulfills this requirement is the beats-per-minute feature [9]. It calculates how many beats per minute (bpm) occur in a given music sample. The beats-per-minute feature is typically used to measure the tempo of the music.

Humans perceive the beat as a binary regular pulse underlaying the music [9]. This qualifies the feature for mapping it to the running rhythm of the practicing person because the time between the beats can be multiple or part of the the time between the footsteps of the user.

Figure 6 shows schematically the detected beats out of an audio signal. An overview of techniques for beat tracking in music is given in [10].

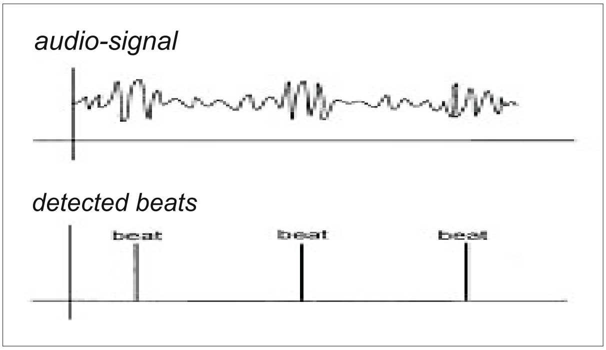


Fig. 6. Detected beats in audio sample

2.3 Mapping

Having a feature corresponding with the running-rhythm of the user (mean footsteptime (2.1)) and a feature corresponding with the tempo of the music (beats-per-minute (2.2)) it is possible to map one feature space into the other (mft↔bmp, bpm↔mft).

Therefore a mapping function has to be defined that maps the mft-feature to the bpm-feature. This enables selecting music fitting to the running rhythm of the user.

The mapping function F (2) is defined as:

$$\mathbf{O}_{[bpm]} = F(\mathbf{M}, f_{[mft]}) \tag{2}$$

where:
$\boldsymbol{M} = \{m_1, \ldots, m_n\}$ ... *set of music files*
$f_{[mft]}$ ... *mft-feature-value*
$O_{[bpm]} \subseteq \boldsymbol{M}$ ... *set of corresponding music files.*

Given a set **M** of music samples and a **mft-feature value** the mapping-function generates a set **O** of music samples with the corresponding **bpm-featue value**.

2.4 Development Framework

The concept framework for developing the system is shown in figure 7.

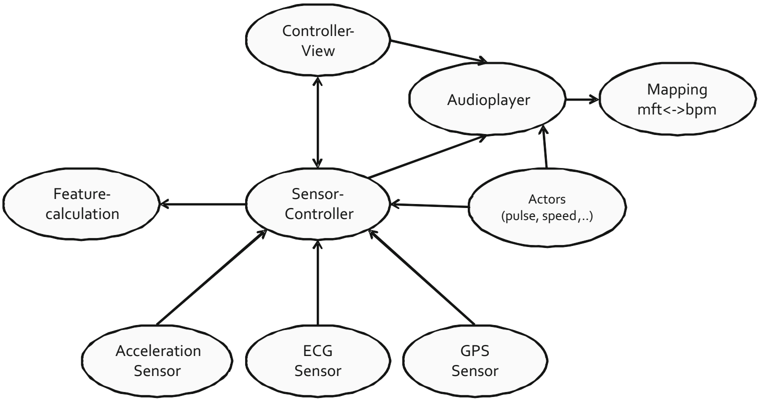


Fig. 7. Development framework

The key element of the framework is the *sensor controller*. It manages the attached sensor nodes (e.g acceleration sensor for movement analysis, GPS sensor for position, speed and direction, and an ECG-sensor). The sensor controller is expandable so new sensors can be attached easily to extend the system.

Further the sensor controller is responsible for managing the storing of the collected data from the attached sensor nodes and for periodically starting the calculation of the mft-feature value (2.1) from the acceleration-sensor data.

The mft-feature is sent from the sensor controller to the *audio-player* where the mapping (2.3) of the mft-feature to the fitting set of music files and the playing of the audio files itself takes place. The audio player is capable of playing audio files parallel and overlaid, so status information from the actors like exceeding pulse limits can be directly played into the current played music file.

The so called *actors* are bound to the data of a specific sensor (e.g. ecg) and perform checking of it's parameters in specified time intervals. If e.g. predefined limits in these parameters (e.g. heart rate) are exceeded the actors can raise warnings using the audio-player. This enables the system to inform or warn the user about different informations on different parameters of different sensors if necessary.

The *controller-view* enables the user to control the audio player (select audio files, change volume,..) and view status information from the sensors (heart rate, speed,..).

3 Prototype

To build a prototypical setup of the system, showing its technical realisation, components are needed that fulfill the special requirements of a wearable (mobile) system used during sports activity [11]. The components have to be selected considering especially the following aspects:

From the view of computing [12]:

- limited processing (computing) power and memory
- limited bandwidth
- limited user input, output
- limited power consumption

From the view of wearability [11]:

- placement (where on the body it should go)
- form (defining the shape)
- human movement (consider the dynamic structure)
- proxemics (human perception of space)
- attachment (fixing forms to the body)
- weight
- thermal (issue of heat next to the body)
- aesthetics (perceptual appropriateness)

Concluding the prototype system has to achieve two main requirements from the view of usability. It must be *unobtrusive* and *unrestrictive* meaning that it must not restrict the movement of the user and its wearing should be impalpable [11].

So it's necessary to use:

- as small and light sensors as possible with low power consumption (long operating time, avoiding heat)
- wireless communication between the components to not restrict the users movement with wires
- running on a device the user takes with him anyway (like mobile phone), minimizing the number of components the user has to take with him additionally

3.1 Hardware

During the past few years, devices like mobile phones or mp3 player became smaller and more light weight. They are frequently already integrated in one single device, equipped with communication technology like bluetooth [13], and their processing power was increased dramatically.

Many people take a mobile phone with them while doing outdoor sports to be on the one hand reachable and on the other hand able to make a call if something happens (like an injury). This and the ability of playing audio files and having more and more different sensors like GPS integrated in nowadays mobile phones make it the ideal hardware platform for hosting such a system.

Taking the requirements from 3 and the above mentioned circumstances in consideration the prototype focuses on being hosted on a mobile phone using its integrated components like the audio player or the gps-sensor being extended by components not offered by the host platform like an acceleration or heart-rate sensor using bluetooth [13] for interconnecting them.

To build up the prototypical system the following hardware components where selected:

- Processing unit (controller)
  For running the application a state of the art sony-ericsson mobile phone (www.sonyericsson.com), the C702 is used. Relevant features for the prototype are the built in gps sensor and the newest version of the javaME Platform (3.2) from sony-ericsson for mobile phones (Java Platform 8 (JP8)) (3.2). Users owning a mobile phone running this platform or higher don't have the need to take a separate processing unit with them as mentioned in 3.
- Sensors
  - acceleration sensor
    The *SparkFun WiTilt v3.0* sensor is used to measure the acceleration during a footstep. The sensor is set to 100hz sampling frequency and measures an acceleration up to 6g. This gives enough reserve for measuring the running movement when the sensor is placed rear hip (on backbone origin) [5].
  - ecg-sensor
    To enable the prototype of giving a feedback about the user's biostatus a Polar ecg sensor (http://www.polar.fi/en/) is used to measure the heart rate.
    For receiving and interpreting the signal of the polar ecg sensor the "polar heart rate monitor interface (HRMI)" from Active-Robots (www.active-robots.com) is used.
  - gps-sensor
    For getting location information like position, speed and direction the built in gps sensor of the processing unit is used.
- Component Interconnection
  For interconnecting the components of the prototype, the processing unit (controller) and the different sensors, a Bluetooth connection [13] is used.
  Bluetooth uses radio technology to establish wireless short-range communication between devices creating a "wireless personal area network" or an "ad hoc" connection [13].
  Bluetooth is de de facto standard to establish an instant wireless connection between locally distributed devices. Especially its high level of integration, very low power consumption and its reliability make it the ideal choice for connecting the components of the prototype [14].

Components of the prototype not equipped with Bluetooth capabilities (e.g. HRMI) were connected to a BlueNiceCom IV Bluetooth module from Amber Wireless (http://amber-wireless.de/) to be able to be connected to the prototype.

More and more mobile phones are equipped with built in acceleration sensors (e.g. iPhone). Use of them would be possible to analyse movement (2.1) and free users of taking a separate acceleration sensor with them. The disadvantage of this solution would be that it restricts the users of using the device for doing something else (e.g. show route). This is because for analysing the acceleration data, the sensor has to be placed and fixed in a defined position. If the position is changed, analysing is not possible any more. So a separate sensor is used to avoid this restriction.

Figure 8 shows the selected hardware components of the prototype.

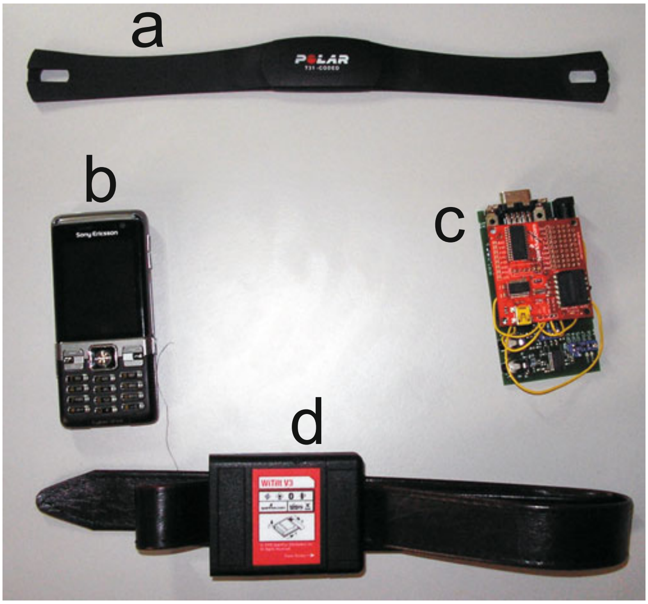


Fig. 8. Components of the prototype: a) polar-ecg-sensor, b) processing unit (Sony-Ericsson C702 JP8), c) HRMI connected to Bluetooth module (can be integrated and miniaturised), d) acceleration sensor fixed on a belt to be worn rear hip (on backbone origin)

3.2 Software

For implementing the prototype JavaME (http://java.sun.com/javame/) was selected. The reasons for selecting this application platform were the broad and free availability, being almost standard on nowadays mobile phones and the libraries especially supporting the needs of the system:

- JSR82: Java APIs for Bluetooth (http://jcp.org/en/jsr/detail?id=82)
- JSR179: Location API for J2ME (http://jcp.org/en/jsr/detail?id=179)
- JSR135: MobileMedia API (http://jcp.org/en/jsr/detail?id=135)

Sony-Ericsson, as other manufacturers as well, offers an implementation of JavaME for mobile phones called Java Platform in the actual version 8 (JP8) enabling it's mobile phones using JSR82 for connecting components using Bluetooth, JSR179 for accessing the built in GPS-Sensor and JSR135 for dealing with multimedia content. The schema of the prototype is shown in section 2.4.

Extracting the beat (2.2) out of the music is a computationally intensive work. For the prototypical setup the beat was extracted on a separate machine and stored as meta data to the music files.

For extracting the beat the software Mixxx [15] based on [9] was used.

4 Experimental Results

The prototype was tested in both naturalistic and laboratory environments. The laboratory tests where done to determine the performance and accuracy of the system. Testing the prototype under naturalistic circumstances should point out it's usability in ordinary conditions expected when using the system.

To determine the accuracy of the calculated mft-feature-value 2.1 a metronome was used to produce a defined, steady pulse to help the testing participant to get into the right running rhythm. The so collected data showed a very high accuracy of the calculated mft-feature-value (table 1). The time derivation in the measured values is explainable due to how exact the testing participants could adjust to the defined running rhythm.

Table 1. Accuracy of calculated mft-feature-value

expected [ms]	*measured[ms]*
400	380-420
500	475-530
700	660-730
900	870-935
1100	1050-1160

To get the performance of the system, the time needed to calculate the mft-feature was measured.

Therefore the possible options of connected sensors (acceleration, ecg and gps), the time frame of steps to calculate the mft-feature for, and the use of the audio player (ua→using audioplayer, no→no audioplayer) were varied (table 2).

Table 2. Duration of mft-feature calculation [ms]

	TIME FRAME FOR CALCULATING MFT-FEATURE					
	last 5 sec		last 10 sec		last 20 sec	
USED SENSORS	na	ua	na	ua	na	ua
acc	*4-6*	*6-15*	*10-40*	*25-70*	*20-70*	*90-140*
acc+ecg	*4-7*	*7-17*	*10-50*	*20-75*	*20-80*	*95-150*
acc+ecg+gps	GPS sensor causes crash of bluetooth connections					

Summarizing table 2: the system has a good response time (for this purpose) in calculating the mft-feature value even having a time frame of 20 sec

($\approx$ 2000 stored values) and the audio player is used. The time derivation in the calculation-time of the mft-feature-value is mainly explainable due to how many steps are recorded in the time frame, but also background tasks of the mobile phone affect the calculation time.

The use of the audio player significantly increases (doubles) the time needed for calculating the mft-feature, showing that for the parallel decoding of a mp3-file a lot of resources are needed, and it seems that the mp3-decoding is not being implemented in a separate hardware component.

The use of a second sensor (ecg) didn't increase the calculation time much. This can be explained because of the low sampling frequency the ecg-data is collected (750ms) and no further calculation intensive processing of the data is performed.

Unexpected behavior of the system was the crash of the bluetooth connections when the built in GPS-sensor was used. Using the GPS-sensor resulted in crashing both bluetooth connections (gps-sensor was still working) and the need to power off and restart the system (phone) to get bluetooth working again. This is difficult to explain but looks like a bug in the JavaME implementation.

After performing the tests, the mft-featue-calculation interval was set to 250ms to get a real time response of the system, and the time frame of steps to calculate the mft-feature for was set to 8 sec.

Test participants told that a longer time frame smooths the mft-feature too much when changing the running rhythm having a too long adjustment time to the new running rhythm. Contrariwise having a shorter time frame results in a too sensitive reaction of the system.

The calculation interval of 250ms was selected to get on the one hand a "feeled" realtime response and on the other hand not to fully load the system. A fully loaded system results in an instable gathering of the acceleration sensor data because the system is fully utilized doing the calculation of the mft-feature-value.

The calculation interval of 250ms and the time frame of 8 sec results in a processor utilization between 70-90% giving the system enough reserves for doing "mobile phone internal" stuff (e.g. managing bluetooth connection, network or broadband connections,make a call,..).

Testing the prototype under natural circumstances (running on the street and on loosely ground (crushed stone road)) showed the same good results as the testings in the laboratory environment.

5 Conclusion and Future Work

In this paper a novel approach for adaptively selecting music fitting to the running rhythm of a user was presented.

It solves the problem of listening to music having a distracting rhythm during training. Three main tasks, (1) analysing the movement of the practising person, (2) analysing the audio files for classifying them and (3) the mapping function between the movement-feature and the audio-feature were described.

A Framework on how to build such a system was shown and a prototype demonstrating the technical realisability on off the shelf hardware was developed.

Using and testing the developed prototype showed that its hard to develop a mapping function that fits to all users because the rhythm of the music is sensed very subjective by each person. Therefore an enhancement of the system would be to enable the learning of the mapping function on the fly during training using techniques from machine learning like decision trees [16] or nearest neighbour classification [17].

Another interesting focus for further research is not only to select music fitting to the running rhythm but rather build training programs forcing or retaining the sportsman in selecting pushing or assuasive music according to predefined parameters like e.g. pulse-limits.

Enabling the system of being tracked and monitored using a remote computer as shown in figure 9 enables a trainer or a third person to watch the users and their status in real time. Also competitions over e.g. continents are thinkable if the users run the "same" track (e.g. 400m in a stadium) and the systems can be tracked and synchronized comparing the position of one to each other.

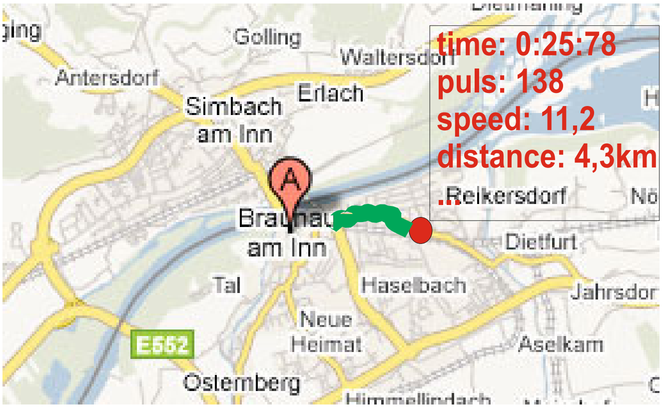


Fig. 9. Tracking and monitoring the system using a remote computer

Acknowledgements

This work was done as a master project at JKU Linz[1]. My thanks go to Prof. Gabriele Kotsis[2] for her encouragement and helpful suggestions during this work.

---

[1] Johannes Kepler University Linz, Austria, Altenbergerstraße 69, 4040 Linz, http://www.jku.at

[2] Professor Gabriele Kotsis, Johannes Kepler University Linz, Austria, Altenbergerstraße 69, 4040 Linz, Email:gabriele.kotsis@jku.at

References

1. Lukowicz, P., Anliker, U., Ward, J., Troester, G., Hirt, E., Neufelt, C.: Amon: a wearable medical computer for high risk patients. In: Wearable Computers (ISWC 2002). Proceedings of the Sixth International Symposium on Wearable Computers, pp. 133–134 (2002), ISBN: 0-7695-1816-8
2. Heinz, E.A., Kunze, K., Gruber, M., Bannach, D., Lukowicz, P.: Using wearable sensors for real-time recognition tasks in games of martial arts. In: Proceedings of the 2nd IEEE Symposium on Computational Intelligence and Games (CIG), pp. 98–102. IEEE Press, Los Alamitos (2006)
3. Lee, S.-W., Mase, K.: Recognition of walking behaviors for pedestrian navigation. In: Control Applications (CCA 2001). Proceedings of the 2001 IEEE International Conference on Control Applications, pp. 1152–1155 (2001), ISBN: 0-7803-6733-2
4. Lukowicz, P., Ward, J.A., Junker, H., Stäger, M., Tröster, G., Atrash, A., Starner, T.: Recognizing workshop activity using body worn microphones and accelerometers. In: Ferscha, A., Mattern, F. (eds.) PERVASIVE 2004. LNCS, vol. 3001, pp. 18–32. Springer, Heidelberg (2004)
5. Bao, L., Intille, S.S.: Activity recognition from user-annotated acceleration data. In: Ferscha, A., Mattern, F. (eds.) PERVASIVE 2004. LNCS, vol. 3001, pp. 1–17. Springer, Heidelberg (2004)
6. Lukowicz, P., Hanser, F., Szubski, C., Schobersberger, W.: Detecting and interpreting muscle activity with wearable force sensors. In: Fishkin, K.P., Schiele, B., Nixon, P., Quigley, A. (eds.) PERVASIVE 2006. LNCS, vol. 3968, pp. 101–116. Springer, Heidelberg (2006)
7. Hay, J.G.: The biomechanics of sports techniques. Prentice-Hall, Englewood Cliffs (1978), ISBN: 0-13-077164-3
8. Beigl, M., Krohn, A., Zimmer, T., Decker, C.: Typical sensors needed in ubiquitous and pervasive computing. In: Proceedings of the First International Workshop on Networked Sensing Systems (INSS 2004), pp. 153–158 (2004)
9. Jensen, K., Andersen, T.: Beat estimation on the beat. In: Applications of Signal Processing to Audio and Acoustics, 2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 87–90 (2003), ISBN: 0-7803-7850-4
10. Hainsworth, S.W.: Techniques for the automated analysis of musical audio. Signal Proscessing Group, Department of Engineering, University of Cambridge, Tech. Rep. (December 2003)
11. Gemperle, F., Kasabach, C., Bauer, J.S.M., Martin, R.: Design for wearability. In: ISWC 1998: Proceedings of the 2nd IEEE International Symposium on Wearable Computers, pp. 116–122. IEEE Computer Society, Los Alamitos (1998)
12. Estrin, D., Culler, D., Pister, K., Sukhatme, G.: Connecting the physical world with pervasive networks. IEEE Pervasive Computing 1(1), 59–69 (2002)
13. Dideles, M.: Bluetooth: a technical overview. Crossroads 9(4), 11–18 (2003)
14. Krassi, B.A.: Reliability of bluetooth. In: Proceedings of the 12th Conference on Extreme Robotics, RTC, St. Petersburg (2001)
15. Andersen, T.H., Andersen, K.: “Mixxx” (2009), `http://www.mixxx.org/`
16. Bringmann, B., Zimmermann, A.: Tree 2 - decision trees for tree structured data. In: Jorge, A.M., Torgo, L., Brazdil, P.B., Camacho, R., Gama, J. (eds.) PKDD 2005. LNCS (LNAI), vol. 3721, pp. 46–58. Springer, Heidelberg (2005)
17. Cover, T., Hart, P.: Nearest neighbor pattern classification. IEEE Transactions on Information Theory 13(1), 21–27 (1967)