Low-cost step aerobics system with virtual aerobics trainer

Alejandro Rosa-Pujazón[1], Isabel Barbancho[1], Lorenzo J. Tardón[1], and Ana M. Barbancho[1]

Universidad de Málaga, Andalucía Tech, ATIC Research Group,
ETSI Telecomunicación, Campus de Teatinos s/n, 29071 Málaga, SPAIN,
alejandror@uma.es, ibp@ic.uma.es, lorenzo@ic.uma.es, abp@ic-uma.es

Abstract. In this paper a low-cost step-aerobics instructor simulation system is presented. The proposed system analyses a given song to identify its rhythmic pattern. Subsequently, this rhythmic pattern is used in order to issue a set of steps-aerobics commands to the user, thus simulating a training session. The system uses a Wii Balance Board to track exercises performed by users and runs on an Android smartphone. A set of tests were conducted to assess user experience and opinion on the system developed.

Keywords: Human-computer interaction, music information retrieval, signal processing

1 Introduction

Thanks to the advances in information and communication technologies in recent years, our everyday life is fully integrated with the use of high-processing devices, ranging from desktop computers to smartphones. Nowadays, we are quite familiar with these devices, and there are more and more possibilities for new types of applications and experiences that improve over what was previously had. In particular, the addition of these technologies to the field of music provides new ground to explore through the use of innovative interaction paradigms.

Research in this field has showed that the combination of music and technology into interactive applications offers more enriching experiences, even creating experiences that would not be accessible with more traditional tools, such as new types of musical instruments (Jordà, 2010), musical expression through body movement (Antle, Droumeva, & Corness, 2008)(Khoo et al., 2008)(Castellano, Bresin, Camurri, & Volpe, 2007)(Halpern et al., 2011), modification of visual patterns by using sung or speech voice (Levin & Lieberman, 2004), etc. The use of advanced human-computer interfaces (Trail et al., 2012) (Rosa-Pujazón, Barbancho, Tardón, & Barbancho, 2013b) (Rosa-Pujazón, Barbancho, Tardón, & Barbancho, 2013a) also provides more accessible paths to music, lowering the barriers inherent to the abstract nature of music and allowing naïve users to enjoy musical expression at its fullest. It has also been shown that interactive

Fig. 1: System overview.

music applications can supply the limitations of the current traditional model for musical rehearsal and practice, such as for example providing a virtual simulation of the role of a conductor in an ensemble (Baez, Barbancho, Rosa-Pujazón, Barbancho, & Tardón, 2013), and the use of these applications for even mere leisure can improve and foster users' interest towards music learning (Gower & McDowall, 2012)(Wang & Lai, 2011).

In this paper, following the idea in (Baez et al., 2013), we present the research conducted towards a system that provides users with a virtual simulation of an instructor role. Concretely, the system proposed simulates the role of an step-aerobics instructor, by processing a given input signal and extracting the beat times and the commands to issue to the trainees. The system has been conceived to run on a smartphone and uses off-the-shelf devices (i.e. Wii Balance Board) to implement the interaction paradigm, thus providing a low-cost affordable alternative to complement real-life step-aerobics classes. Section 2 portrays a description of the system implemented and the signal processing techniques considered, while section 3 illustrates the results from an experiment conducted with the current implementation of the system in order to gather data on its impact on user experience. Finally, the conclusions derived from the work performed are presented, as well as a proposal of future lines of study.

2 System description

The idea behind the system implemented is to emulate a step aerobics instructor, so that the final application could issue aerobics-like commands to the user according to the rhythm of the song played. Therefore, users would perform exercises according to the commands given by the application, and the application itself would indicate to users whether their exercises were being performed in the right way or not.

In order to achieve this functionality, the structure of the system was organised in the form of several function-specific modules, as depicted in Fig. 1. The application is designed to run on an Android smartphone. Thus, it was written in Java, using the Android SDK tools.

The Human-Computer Interface module provides the application interface (menus, step-commands displayed, etc.). The Step-Tracking module refers to the device used to gather information on how the user is performing the exercise.

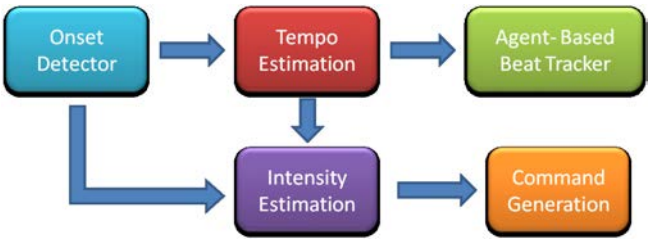


Fig. 2: Beat-Tracker block structure.

This functionality is implemented using a Wii Balance Board device, which was used as a proxy for the standard step board.

The Song Analyser module implements the main functionality of the application. Its purpose is to read the audio data from the song file and analyse it in order to extract a data descriptor of the rhythm of the song. This rhythmic data is used by the application to both create the exercise instruction commands and to determine whether the user exercises are performed in synchrony with the rhythm or not. The Song Analyser offers two different functionalities: first, it allows the system to track the beat times of the song analysed; second, it divides the song in segments, each one labelled accordingly to the rhythmic intensity of that particular segment, so that an adequate set of step-aerobics commands can be issued according to the rhythm of the piece considered.

The block structure of the Song Analyser is depicted in Fig. 2:

- an *onset detector*, which is in turn comprised of two elements: a spectral analysis module that implements a Short-Time Fourier Transform (STFT), and an element that subsequently determines the spectral energy flux of the PCM data.
- a *tempo estimation* component that determines the most likely tempo values for the song processed.
- an *agent-based beat tracker* which calculates the position of the rhythm beats according to the tempo hypothesis provided.
- a rhythmic intensity estimator, to determine which segments in a given clip are played with higher or lower rhythmic patterns.
- a steps-commands generator to issue the corresponding commands according to the beat patterns detected and the overall rhythmic intensity of the song at each segment.

The main two functionalities of the system are described in further detail in the following subsections.

Beat tracking function While it may be feasible to directly process the audio data (Barbancho, Barbancho, Tardón, & Urdiales, 2009), an onset detection function was implemented in order to implement automatic feature extraction of musical pieces. The reason behind this decision is that an onset function is simpler and faster to work with than raw PCM data, while keeping the rhythm

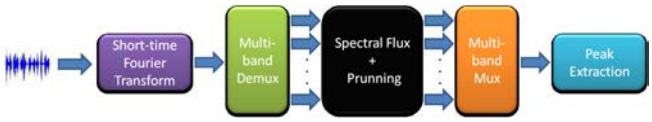


Fig. 3: Onset detector block diagram.

data of the original musical piece. Therefore, by using the onset detector, tempo estimation and effective beat-tracking tasks are drastically reduced in complexity. In our implementation of the onset detection function, we decided to use a spectral domain based analysis; in particular, we decided to use the spectral energy flux of the audio signal, as described in the works of Dixon (Dixon, 2006) and Alonso et al. (Alonso, Richard, & David, 2004). Once again, spectral flux was chosen as the onset detection function because it is quite a proficient method with regards as factors such as onset detection accuracy, simplicity of programming, and execution speed (Dixon, 2006).

Concretely, the onset detector was implemented following this algorithm (illustrated in Fig. 3): first, an STFT is applied, using a Hamming window of size of 2048 samples with a 50% overlay between successive chunks of windowed data. The resulting signal is then divided into eight different components according to a frequency multiband division (Table 1), and for each component, the corresponding spectral flux is calculated. Each of the spectral fluxes obtained is rectified and pruned, so that for each spectral flux sample, that sample is set to 0 if it lies below a given threshold value, and kept otherwise. The different threshold values for each sample are dynamically set by finding the average of the spectral flux value in a window of 1 second centred on the i-th sample processed, setting the i-th threshold value to 1.5 times this average.

The resulting eight pruned spectral flux functions are recombined into a single signal. A peak signal is then defined by extracting the peaks in this combined signal, and finally, the onset signal is obtained by filtering the peak signal so that every pair of consecutive peaks are spaced by at least 50 milliseconds (if a pair of peaks does not fulfil this criteria, the peak with lower amplitude is deleted).

Frequency range
0Hz-250Hz
250Hz-500Hz
500Hz-1KHz
1KHz-2KHz
2KHz-4KHz
4KHz-8KHz
8KHz-16KHz
16KHz-max

Table 1: Frequency bands for onset detection

The tempo estimation component and the agent-based beat-tracker were implemented according to the algorithm described in (Dixon, 2001). Tempo induction is achieved by analysing the inter-onset intervals (time intervals between pairs of rhythmic events) and clustering them in order to create a list of tempo hypothesis (created and ordered according to a score value related to the frequency that each inter-onset interval has in the rhythm structure of the onset function).

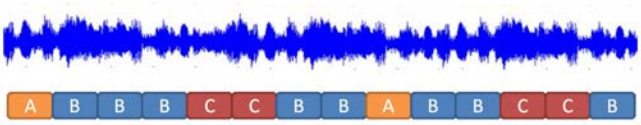
These tempo hypothesis are later used as the basis of the agent-based beat-tracking algorithm, which in turn estimates the temporal location of each musical rhythm beat in the onset signal, and thus in the original song. The algorithm creates a set of agents, each of them taking a tempo hypothesis as a initial inter-onset interval value. Then, according to the actual position of the rhythmic events in the onset signal, the different agents dynamically update their interval value as well as their actual score (according to the accuracy of their predictions on each rhythm event position). In the end, the highest scoring agent's history of predictions is taken as the list of tracked beats to consider in the application. This information is stored in the phone in the form of a file with *.btr* extension, in order to further increase speed of execution in later stages of the application.

Maximum and minimum values for the tempo estimations were fixed at 240 and 30 BPM respectively (a sufficiently wide enough range of values for step aerobics music), and agents that failed to find a rhythmic event for a period of 3 seconds were discarded.

Intensity estimation As previously indicated, the system is capable of dividing the audio excerpt analysed into segments according to their rhythmic intensity. In order to do so, the input signal is split in chunks, each chunk having a size $2T_c$ seconds. Chunk division is then achieved by applying a $2T_c$ window to the audio data, with an overlay of 50% between successive windows. T_c parameter is defined as the number of seconds required to cover a total of 24 beat times played at the original song's tempo.

After fully compartmentalising the audio signal into chunks, a density score is determined by dividing the total number of actual beats collected at each chunk by the total amount of beats in the overall piece. The aforementioned density score is used to discriminate different density levels in the son. Concretely, a k-Nearest Neighbours unsupervised machine learning process is performed over the beat-density signal resulting from the concatenation of each chunk's density score. At the end of this process, each chunk is labelled into one of the k potential levels of intensity found.

For each of the k categories considered, the average number of beats per chunk for each category is determined, and the resulting value is used as a score to order the different categories increasingly. The most commonly found category (hereafter category l) among chunks is assumed to represent the standard level of rhythmic intensity. Every other chunk's intensity is set according to the ordering previously computed, taking category l as a reference. An example for a fictitious case with $k = 3$ categories is portrayed in Fig. 4.



(example for k=3 categories)	A	B	C
Average number of Beats per category	15	23	47
Intensity label	Low	Standard	High

Fig. 4: Category labelling for $k = 3$ classes: the chunks are labelled according to the k-NN classifier, and the intensity for each category is set according to the average number of beats per chunk, taking the most common category (B) as the standard rhythmic intensity

Finally, once the different chunks of the song have been assigned an equivalent intensity level, the system randomly generates a pattern of step exercises to output while playing the song as commands given by a virtual trainer. Each chunk is then associated with a set of exercises according to the level of intensity identified. Thus, while the song is playing, the commands corresponding to the particular chunk in play in that moment will be presented to the user in synchronization with the rhythmic patterns identified previously by the beat tracker module.

3 Experiments and Results

We conducted an experiment aimed at collecting data regarding user experience when using the designed virtual aerobics trainer. Concretely, a total of 10 participants took part in the experiment. All of the participants were male, with an average age of 29.4 and two of them had a strong musical background. Only one of them had attended previously a class of steps-aerobics.

Each participant was asked to "attend" one training session with the virtual instructor, which consisted on one song being played along with the corresponding commands. For the purpose of this training session, the application was run on a personal computer, and thus the different commands were presented to the users through a standard desktop display. The songs considered for this test were all extracted from compilations of steps music, concretely they were: Feel this Moment (Pitbull), Gangnam Style, Kiss N Tell (Kesha), Can't Hold Us (Macklemore) and Don't Stop the Party. Each song had a corresponding BPM value of 136, 130, 144, 146 and 128 respectively. Each participant was randomly assigned one song, so that each song were used in two different experimental sessions (hence covering the 10 participants). For the purpose of this experiment, we considered $k = 3$ levels of rhythmic intensity, which in the case of all the 5 songs considered yielded one average level of intensity, one of higher intensity, and one of lower.

Prior to the virtual step aerobics session, each participant was presented with the set of exercises from which the system would randomly select the different exercise patterns. Participants were given as much time as possible to get familiar with the different kinds of steps that could be issued. The different exercises considered were appropriately labelled as low, standard and high intensity exercises, attending to the complexity of the exercises considered. All of the standard and low intensity exercises were performed in 4 beat times, while the high intensity exercises required 8 beat times to be performed. During each experimental session, for each beat time detected the system would issue a command according to the level of rhythmic intensity in the segment of the song currently in play (e.g. Basic Left Step), and the following commands would just keep the count of beat times associated to the exercise (e.g. 2, 3, 4) until its end. After that, a new command is issued in the next beat time, and the process is repeated until the song ends. For the purpose of this experiment, we used the database of step exercises available in (*Aerobics-Steps dictionary*, 2014).

At the end of the experimental session, each participant was asked to fill in a questionnaire in which they assessed their experience with the virtual instructor. Concretely, the users were asked to evaluate the following items:

- Utility perceived.
- Satisfaction with the experience.
- Novelty of the application.
- Ease of use.
- Synchronization between commands and rhythmic intensity.

For the particular case of the "Ease of use" item, participants were explicitly asked to take out of consideration perceived difficulties because of having to memorize the different exercises previously. In addition to the questionnaire, additional data was extracted from an informal interview between the participants and a researcher concerning their overall perception of the system. The results obtained are summarized in Fig. 5

Overall, the application had a warm welcome for the most part, with the most valued aspects being its novelty and the synchronization between rhythmic intensity and the commands issued. Satisfaction was the item that received the worst critics, and in fact several participants stated that they found that having to read the commands from the display detracted from the overall experience. This suggest that including pre-recorded voice commands is a much needed improvement, but the overall response was still quite positive. Some participants expressed their concerns regarding the complexity of the commands given, as they reckon some of the combinations of exercise patterns proved to be too difficult to follow. In fact, the system randomly selects exercise patterns taking only their difficulty into consideration, but does not evaluate whether a given succession of exercises is more or less adequate to conform a global exercise. In order to solve this issue, a more detailed analysis must be performed regarding how the different exercises are selected, ideally with the assistance of a professional trainer.

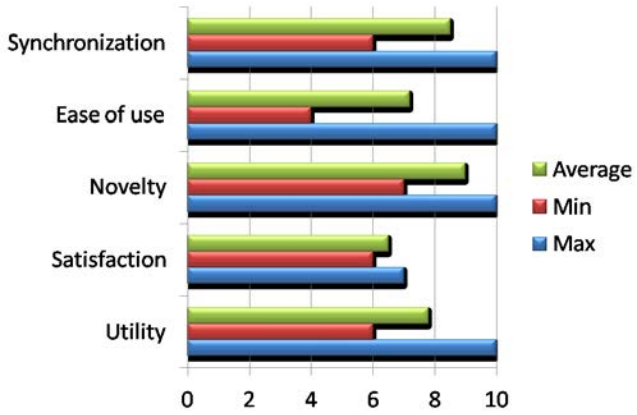


Fig. 5: Results from participants' survey.

4 Conclusions and Future Works

A low-cost system for step-aerobics practice has been implemented. The system makes use of off-the-shelf devices such as an Android smartphone and a Wii Balance Board to implement an affordable human-computer interface. The application developed can analyse a song rhythmic structure to extract the beat times and issue step-aerobics commands according to the rhythmic intensity of each part of the audio clip. An experiment has been conducted showing that user experience was mostly positive and that the system can indeed qualify as an useful tool for step-aerobics practice. In general terms, participants have found the application to be an appealing emulation of a real step-aerobics trainer. Yet, there some issues where the system can be improved to provide an even more satisfying experience. The most relevant short-comings identified by the participants were the need of having to read the commands and the lack of coherence in some of the combinations of exercises, which proved to be inappropriately demanding.

Regarding future works, we aim to address these two short-comings specifically, by adding the use of pre-recorded voice commands and the assistance of a professional instructor. Additionally, even though this paper has not addressed it directly, we hope to assess the usefulness of this system for rehabilitation purposes, as it has been proven that step-aerobics exercises can have a positive impact in this regard (Clary, Barnes, Bemben, Knehans, & Bemben, 2006).

Ackknowledgements This work has been funded by the Junta de Andalucía under Project No. P11-TIC-7154. The work has been done in the context of Campus de Excelencia Internacional Andalucía Tech, Universidad de Málaga.

References

Aerobics-steps dictionary. (2014, June). Retrieved from `http://www.turnstep.com/moves.html`

Alonso, M. A., Richard, G., & David, B. (2004). Tempo and beat estimation of musical signals. In *Proceedings on international conference on music information retrieval, ismir 2004* (pp. 158–163).

Antle, A., Droumeva, M., & Corness, G. (2008). Playing with the sound maker: do embodied metaphors help children learn? In *Proceedings of the 7th international conference on interaction design and children* (pp. 178–185).

Baez, R., Barbancho, A. M., Rosa-Pujazón, A., Barbancho, I., & Tardón, L. J. (2013). Virtual conductor for string quartet practice. In *Smac 2013 - stockholm music acoustics conference 2013* (pp. 292–298).

Barbancho, A. M., Barbancho, I., Tardón, L. J., & Urdiales, C. (2009). Automatic edition of songs for guitar hero/frets on fire. In *Multimedia and expo, 2009. icme 2009. ieee international conference on* (pp. 1186–1189).

Castellano, G., Bresin, R., Camurri, A., & Volpe, G. (2007). Expressive control of music and visual media by full-body movement. In *Proceedings of the 7th international conference on new interfaces for musical expression* (pp. 390–391).

Clary, S., Barnes, C., Bemben, D., Knehans, A., & Bemben, M. (2006). Effects of ballates, step aerobics, and walking on balance in women aged 50-75 years. *J Sports Sci Med*, *5*(3), 390–399.

Dixon, S. (2001). Automatic extraction of tempo and beat from expressive performances. *Journal of New Music Research*, *30*(1), 39–58.

Dixon, S. (2006). Onset detection revisited. In *Proc. of the int. conf. on digital audio effects (dafx-06)* (pp. 133–137).

Gower, L., & McDowall, J. (2012). Interactive music video games and children's musical development. *British Journal of Music Education*, *29*(01), 91–105.

Halpern, M., Tholander, J., Evjen, M., Davis, S., Ehrlich, A., Schustak, K., ... Gay, G. (2011). Moboogie: creative expression through whole body musical interaction. In *Proceedings of the 2011 annual conference on human factors in computing systems* (pp. 557–560).

Jordà, S. (2010). The reactable: tangible and tabletop music performance. In *Proceedings of the 28th of the international conference extended abstracts on human factors in computing systems* (pp. 2989–2994).

Khoo, E., Merritt, T., Fei, V., Liu, W., Rahaman, H., Prasad, J., & Marsh, T. (2008). Body music: physical exploration of music theory. In *Proceedings of the 2008 acm siggraph symposium on video games* (pp. 35–42).

Levin, G., & Lieberman, Z. (2004). In-situ speech visualization in real-time interactive installation and performance. In *Npar* (Vol. 4, pp. 7–14).

Rosa-Pujazón, A., Barbancho, I., Tardón, L. J., & Barbancho, A. M. (2013a). Conducting a virtual ensemble with a kinect device. In *Smac 2013 - stockholm music acoustics conference 2013* (pp. 284–291).

Rosa-Pujazón, A., Barbancho, I., Tardón, L. J., & Barbancho, A. M. (2013b). Drum-hitting gesture recognition and prediction system using kinect. In *I simposio español de entrenimiento digital seed 2013* (pp. 108–118).

Trail, S., Dean, M., Tavares, T., Odowichuk, G., Driessen, P., Schloss, W., & Tzanetakis, G. (2012). Non-invasive sensing and gesture control for pitched percussion hyper-instruments using the kinect.

Wang, C., & Lai, A. (2011). Development of a mobile rhythm learning system based on digital game-based learning companion. *Edutainment Technologies. Educational Games and Virtual Reality/Augmented Reality Applications*, 92–100.