Soundscape-Sensing in Social Networks

João Cordeiro[1], Álvaro Barbosa[1,2,], Bruno Afonso[1]

[1] *School of Arts / Research Centre for Science and Technology of the Arts. Portuguese Catholic University – CRP, Rua Diogo Botelho, 1327, 4169-005 Porto – Portugal. Email: {jcordeiro}, {abarbosa}, {bafonso}@porto.ucp.pt*

[2] *Faculty of Creative Industries – University of Saint Joseph, Macao, China. Email: abarbosa@usj.edu.mo*

Abstract

In this paper we present a status report of the design and development stages of an online social network system based on Soundscapes. Additionally, a detailed description of the auditory scene analysis module of the system is made.

The term Soundscape is used to describe the relation and interaction between the acoustic environment of a place and its inhabitants, namely on how these perceive and judge the sound. From this premise we developed a mobile application that senses and shares information regarding the Soundscapes of the places inhabited by each user during his/her daily life. Thus, each user acts as a terminal of a sensor network linked through the use of sound.

The goal of this research is to assess the use of Soundscapes in social network interactions and consequently promote sound awareness among people.

Introduction

What turns Environmental Sound into Soundscape is somehow a human interpretation of the first, comprised of all the meaning(s) emerging from the interaction between the listener, the place and the sound. From this definition, we have designed a mobile application that senses the sound environment where individual users are located and shares this information within their social network, impregnating the data with social significance, turning this analysis into a Soundscape analysis.

Soundscape studies have shown [14] that Environmental Sound is a rich resource for understanding the social context of a place, its dynamics, problems and virtues, assuming sound as a resource rather than waste. The Soundscape approach to solve noise annoyance problems has proven to be a valid solution [15], since it approaches sound from an holistic perspective, which takes meaning and context in consideration rather than focusing solely on sound level measurements, as usually found in Community Noise strategies [2]. Nonetheless, this approach is still confined to geographic places, whether interior or exterior, private or public, natural or artificial. In our system we propose a redefinition of the concept *place* by shifting the point-of-audition from a static geographical *place* to a dynamic *place*: the user.

The User and the Added Value for Social Interactions

During their daily life, people travel through a sea of sounds, eventually without even changing their geographic location. Passively or actively, Soundscapes vary along the hours, days, months and years, characterizing in each moment the sonic context of a place. By extending the analysis time span, sonic patterns about the place are unveiled, contributing to the characterization of its sonic profile. This data can then be extrapolated to other layers of significance, mainly when correlated with data gathered through multi-modal analysis (geolocation, time, weather conditions, etc.).

Shifting the *place* from static to dynamic, we are characterizing not a geographical location but a node of a social network, which drifts in space, time and network position. With the proposed system, the user is able to keep track of his/her personal sonic profile. If more data is collected, more detailed and accurate is the profile. This information when evaluated in context can be of great relevance for daily social interactions, since the short-term and long-term analyses show different (but complementary) aspects of the social behaviour of the nodes of a network.

Why Sound

The auditory human system is a quite sophisticate sensor, able of great accuracy when discriminating between different sound events in the physical world. In familiar situations, humans are able to identify who is talking, walking or snoring just by listen to the resulting sounds from these actions; and when trained, the human ear proves to be a valuable tool in the work environment, both for prognostic tasks (mechanical, medical applications) and artistic creation (music). Unlike vision, audition captures omnidirectional stimuli and operates even during sleep time. Such exquisite properties have saved radio broadcast when TV arrived (by pushing it into the automobile), and recently has granted an increasing value to sonic interaction research, mainly in the field of sonification [7]. Moreover, from a computational perspective, sound is less demanding in terms of processing power and disk space, being also easier to capture (due to its omnidirectional nature). By approaching Environmental Sound from the Soundscape perspective, we fill the conceptual gap between the physical and social worlds.

Overview of the system

The system is comprised of a mobile application, a web application and a webserver combined in a typical client-server configuration (**Error! Reference source not found.**). The mobile application is responsible for input and output tasks, covering the visual and auditory display of real-time information as well as the soundscape sensing. The web application displays the sonic profile (corresponding to long-term information) and the webserver saves and retrieves information based on forms/queries (represents a functional piece of the system, potentially invisible for the user).

The mobile application includes four modules:

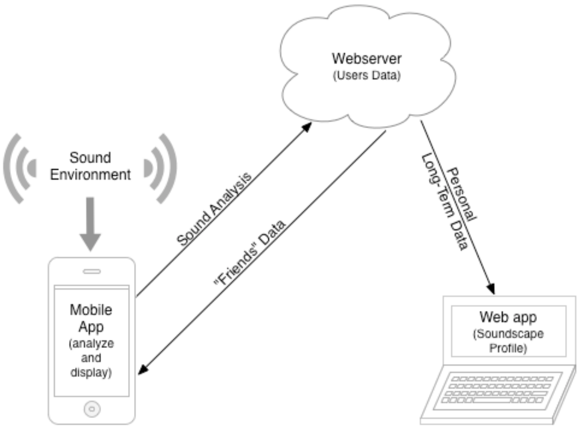

Figure 2 - Client-Server Configuration

1) **A soundscape-sensing module** that performs a sound level measurement of the acoustic environment and classifies it according to three general categories: music, speech or environmental sound (this module is described later in further detail);

2) **A client-server database module** used to store and distribute the captured information through the "friends" of the user's social network. It is built with MySql open-source database system and queried by PHP scripts bridging to connect the mobile application and the database. The application running on de mobile device (iOS) communicates with the PHP script on the server through http forms.

3) **A visualization module** that displays the current soundscape information of user's "friends". This visualization presents the information to the user in a fast, accurate and meaningful way. In the design of our visualization, we have associated a representation of a sound wave to each "friend" on the social network. The colour and waveform of the sound wave vary according to the current soundscape of the "friend" (Figure 2).

- Grey Sine Wave – Music
- Black Noise Wave – Environmental Sound
- Blue Sine Wave with AM – Speech

The amplitude of the sound waves represented in the GUI varies according to the measured sound level. When the sound is perceived as silence (below 5 dB) a pink line is displayed. On the left end of the line is displayed the name of the "friend" in grey, while the name of the user appears in green. When "friends" are offline, the colour of their sound waves turn grey, so user know that the representation regards the last captured/analysed soundscape and not the current soundscape of his "friend". When user turns from online to offline, all the sound waves turn grey, meaning that he/she is no longer receiving real-time updates.

4) **A sonification module** with an auditory display intended to convey information regarding the actual sonic status of the social network. Passing two fingers on the screen from top to bottom triggers this sound, which is the only afforded active interaction between the user and the application.

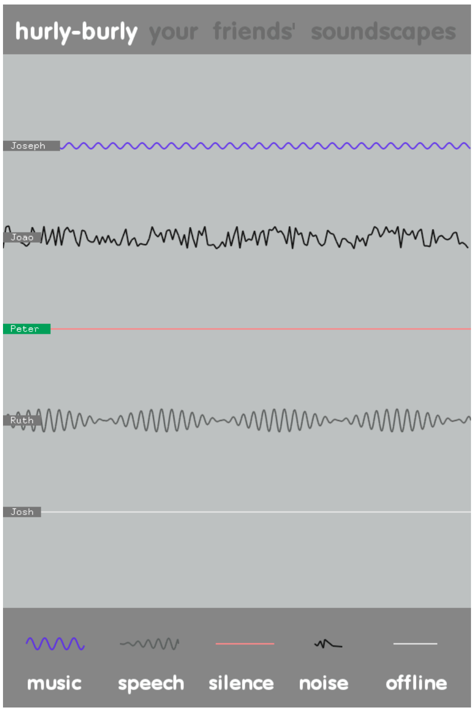

Figure 1 - GUI of the application

This sonification process is called a model-based sonification [6], since the produced sound is a result of the system reaction to a physical interaction and is based on the properties of sonified data.

The web-based application

The web-based application is meant to display the history of users personal soundscapes. The main goal is to unveil patterns on user' sonic activity. This module was not yet implemented.

Soundscape Sensing Module

The Soundscape Sensing module runs in the mobile device and is comprised of a two-step analysis: 1) sound level and 2) sound classification. Since a mobile device has limited processing power when compared with high-end desktop units, the analysis task required a compromise between accuracy/precision and processing resources.

The analysis is always performed on ±2,79 seconds mono audio samples (44.1kHz, 16bit), recorded every 3 minutes using the built-in microphone of the mobile device. This module was built using visual programming language Pure Data and embedded in system using a wrapper called LibPD.

Sound Level Measurement

The sound level measurement of the recorded sound is accomplished using LAeq (2,79 s), which is regarded as standard metric for environmental sound level. Although being tailored for more continuous sounds and derived from the 40dB Equal Loudness Contour Curve, preliminary tests proved that such metric was more convincing than a linear Leq, which didn't respond well for quieter sounds. This version of the application does not take in consideration the microphone frequency response, which varies from model to model.

Sound Classification Module

The recorded sound is classified in three major classes: musical sounds, speech sounds and environmental sounds. This three major groups where chosen regarding the differences on the production processes (sound sources / activity) and underlying emotional potential.

Speech is a human activity and is usually associated with the exchanged of information. When speech label is activated, it means that the background noise is low. User is potentially listening or talking in a quiet environment (at school, at home watching TV/radio, at the office, having a conversation, etc.)

Music is also resultant from human activity and is usually related with leisure activities. Generally, music is found on the acoustic environment when festive activities occur (parties, celebrations), in artistic events (concerts), at private places (car, home), in the workplace (musicians) and in commercial places (shopping centres, stores, etc.). Music classification is based mainly on the pitch content of the audio material. When the music label is activated it means that there is a low level of background noise and music with a defined pitch content is playing.

Environmental Sounds can be described as audible acoustic events which are caused by motions in the ordinary human environment, have real events as their sources (and thus are meaningful), are usually more "complex" than laboratory sinusoids and are not part of a communication system [16]. In the scope of our application, environmental sounds are all the sounds that are not considered music nor speech or, in some cases, when more then one category is present.

The sound classification task occurs in two-steps:

1) Discrimination between speech and non-speech sounds

For this tasked we have used Spectral Irregularity, Sigmund~ (pitch) and Sigmund~ (envelope) as audio features and a k-NN algorithm as classifier. The ±2,79 seconds sample (122880 audio samples at 44.1kHz) is divided in three smaller files with 40960 samples, which are analysed with a Hann window size of 4096 samples and a hop size of the same value. These proportions remain equal for all features.

Spectral Irregularity inspects a spectrum and assesses how much each frequency bin compares to the immediate neighbours. Spiky spectra will have higher Spectral Irregularity then smooth spectra [8].

Sigmund~ is a Pure Data object created by Miller Puckette which analyses an incoming sound into sinusoidal components. The output is two-fold: pitch estimation and envelope tracking, describing roughly the mean amplitude of the sample.

2) Separation of non-speech sounds into musical sounds and environmental sounds.

The second stage of the classification task uses sigmund~ (pitch) SpecSpread, SpecKurtosis and SpecFlatness audio features, and a k-NN classifier.

Spectral Spread is a measure of the concentration of a spectrum's energy around its spectral centroid as defined in MPEG-7 standard [9]. A single sine wave shows a bandwidth of zero and white noise is close to infinite bandwidth. It is extracted by taking the root-mean-square (RMS) deviation of the spectrum from its centroid in each frame. This same measure is named Bandwidth elsewhere [11].

Spectral Kurtosis measures the flatness of a distribution around its mean value, indicating the peakedness and flatness of a distribution [13]. In other words, it is the smoothness of the spectrum compared to a Gaussian distribution (4th moment) [12]. The Spectral Kurtosis value for a sinusoid will be much higher than for white noise.

Spectral Flatness is the ratio of the geometric mean of magnitude spectrum to the arithmetic mean of magnitude spectrum [13], it measures the similarity of the spectrum to a white noise [12]. The spectrum of white noise should have a high flatness value, close to 1.0.

The feature design used in the classification was based on literature review [10] and trail and error tests. More then ten audio features were tested, including: spectral brightness, spectral flatness, spectral roll-off, spectral flux, spectral centroid, zero crossing rate, mel-frequency cepstral coefficients, bark-frequency cepstral coefficients, spectral magnitude, spectral skewness and spectral kurtosis [3].

The first step of the classification task (speech and non-speech) was accomplished using a 1st order k-nearest neighbor (k-NN) machine-learning algorithm, which was chosen regarding its easy implementation, low demand of processing power and satisfactory results. Other more complex algorithms (as Neural Networks, Support Vector Machine or Hidden Markov Models) were not tested but are regarded as an option for these type of task [1, 4, 5]. A set of 649 sound samples was analysed and clustered in two groups: Speech (0-444) and Non-Speech (445-648). From this analysis resulted a dataset, which was used as training example on the feature space.

When the audio sample is analysed, a confidence value is outputted by the k-NN (the tID Pd object [3]). If confidence level is <=2 the sample is rejected and a new one is captured, else it is classified as speech or non-speech.

The second stage of the classification separates non-speech sounds into music or environmental sounds. The first step is to extract the audio features into a feature vector, which is routed to a k-NN classifier. This classifier was trained with 445 sound samples, clustered in two groups: Environmental Sounds (0-223) and Music (224 - 445). If the confidence level reported by the classifier is =>1 then the classification is accepted, else a decision logic takes place, based on the pitchness of the sound reported by sigmund~ object. If the quantity of successfully recognized pitches is <= 8 then the sound is classified as environmental sound, else, if the mean of the recognized pitches is >= 59, the sound is classified as environmental, else it is classified as music.

The classification task was successfully integrated in the iOS application.

Preliminary tests

Preliminary tests were conducted in laboratory in order to assess the success of the classification algorithm. A database of 106 sounds (not present in the training examples) was used. The test was accomplished in a desktop unity and inputted as digital signal (without using a microphone, nor a iOS device). From the 106 samples, 10 were rejected (9%) and 96 were accepted (91%), from which 92 (87%) were classified correctly and 4 (4%) were misclassified.

Discussion

Preliminary tests in laboratory proved that the classification algorithm was successfully implemented. When running on the iOS device, the algorithm works but the overall sensation is that some non-musical sounds are classified as such. This suggests that more tests need to be accomplished, mimicking real-life scenarios (capturing real-life sounds with the device microphone). Energy resource tests show a considerable increase on battery usage, which may refrain users from running the application continuously.

Future Work

Future work includes the tuning of the classification algorithm and conduction of long-term tests with multi-users, in order to access the underlying goals of the system.

Acknowledgments

The analysis module was implemented using Pure Data visual programming language developed by Miller Puckette and embedded in the system using LibPd developed by Peter Brinkman, turning Pd into an embeddable library usable as a sound engine in mobile phone apps. The C++/openFrameworks code runs in the iOS device through the use of ofxiPhone, an add-on for openFrameworks 006+, developed by Memo Akten, Lee Byron, Zach Gage and Damian Stewart in coordination with the core OF team (Zach Lieberman, Theo Watson and Arturo Castro). The sound measure block integrates code from the PureMeasurment developed by Matthias Blau for Pure Data. The sound classification block uses code from TimberID, by William Brent.

Research sponsored by Fundação para a Ciência e Tecnologia, Portugal (SFRH / BD / 47844 / 2008).

References

[1] Álvarez, L. et al. 2011. Application of neural networks to speech/music/noise classification in digital hearing aids. *Recent Advances in Signal Processing, Computational Geometry and Systems Theory. Proceedings of the 11th WSEAS International Conference on Signal Processing, Computational Geometry And Artificial Vision. Proceedings of the 11th WSEAS International Conference* (Florence, 2011), 97–102.

[2] Berglund, B. et al. 1999. *Guidelines for community noise*.

[3] Brent, W. 2010. A Timbre Analysis And Classification Toolkit for Pure Data. *International Computer Music Conference Proceedings* (Ann Arbor, 2010), 224–229.

[4] Gaunard, P. et al. 1998. Automatic classification of environmental noise events by hidden markov models. *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on* (1998), 3609–3612.

[5] Guo, G. and Li, S.Z. 2003. Content-based audio classification and retrieval by support vector machines. *IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council*. 14, 1 (Jan. 2003), 209–15.

[6] Hermann, T. 2011. Model-Based Sonification. *The Sonification Handbook*. T. Hermann et al., eds. Logos Publishing House. 399–427.

[7] Hermann, T. et al. 2011. *The Sonification Handbook*. Logos Publishing House.

[8] Jensen, K. 1999. *Timbre models of musical sounds*. University of Copenhagen.

[9] Kim, H. et al. 2006. *MPEG-7 audio and beyond: Audio content indexing and retrieval*.

[10] Lu, L. et al. 2001. A robust audio classification and segmentation method. *Proceedings of the ninth ACM international conference on Multimedia - MULTIMEDIA '01* (New York, New York, USA, 2001), 203–211.

[11] Mitrovic, D. et al. 2010. Features for Content-Based Audio Retrieval. *Advances in Computers: Improving the Web*. Elsevier. 71–150.

[12] Pachet, F. and Roy, P. 2009. Analytical Features: A Knowledge-Based Approach to Audio Feature Generation. *EURASIP Journal on Audio, Speech, and Music Processing*. 2009, (2009), 1–23.

[13] Peeters, G. 2004. A large set of audio features for sound description (similarity and classification) in the CUIDADO project. *CUIDADO IST Project Report*. 54, version 1.0 (2004), 1–25.

[14] Schafer, R.M. 1977. *The Soundscape: Our Sonic Environment and the Tuning of the World*. Destiny Books.

[15] Schulte-Fortkamp, B. et al. 2007. Soundscape: An Approach to Rely on Human Perception and Expertise in the Post-Modern Community Noise Era. *Acoustics Today*. 3, 1 (2007), 7.

[16] VanDerveer, N.J. 1979. *Ecological Acoustics: Human Perception of Environmental Sounds*. Cornell University, August.