Sound Synthesis based on Semantic Descriptors

Cesar Costa[1,3], Fábio Furlanete[1,2], Jônatas Manzolli[1], Fernando Von Zuben[3]

[1]Núcleo Interdisciplinar de Comunicação Sonora (NICS) - Universidade Estadual de Campinas (UNICAMP) Rua da Reitoria, 165. CP 6166 – Campinas, 13091-970, BRAZIL

[2]MUT – Universidade Estadual de Londrina (UEL)
Rod. Celso G. Cid, Km 375 Londrina, BRAZIL

[3]LBIC / DCA / FEEC / Universidade Estadual de Campinas (UNICAMP), Bloco G2 - Sala LE 14G, CP 6101 Campinas, 13083-970, BRAZIL

{cesar,ffurlanete,jonatas}@nics.unicamp.br,
vonzuben@dca.fee.unicamp.br

Abstract. *One of the main issues on sound design processes is how to describe the intended sound result, and how to use such description in the synthesis procedure. With the augmented interest on multimedia description notation, with standards like MPEG-7, we suggest the use of such scheme that aims at characterizing audio content to establish control reference in a synthesis process. In this paper we present an interactive sound design methodology with a user-defined semantic description paradigm. We briefly discuss audio notation and present our approach, implementation issues, and results.*

Resumo. *Um dos principais pontos do processo de design sonoro é como descrever o resultado sonoro desejado, e como usar esta descrição no processo de síntese. Com a ampliação do interesse em notação descritiva para multimídia, como o protocolo MPEG-7, sugere-se o uso deste tipo de mecanismo que tem como objetivo caracterizar o conteúdo do áudio como uma forma de estabelecer controle no processo de síntese. Neste artigo apresenta-se um método de design sonoro vinculado a um paradigma de descrição semântica fornecido pelo usuário. É discutida a notação de áudio e são apresentados a nossa abordagem, aspectos da implementação, e resultados obtidos.*

1. Introduction

Recently, multimedia content description has become an effervescent research area. The main worry is to establish how multimedia material could be described promoting general understanding and accessibility. Industry standard such as MPEG-7 has already emerged as a promising candidate, as emphasized by studies on multimedia metadata on the Web (van Ossenbruggen el all, 2005), the *Cuidado* music browser (Pachet et all, 2003), and the development of a music content semantic annotator like MUCOSA (Herrera et all, 2005).

Under the possibility of establishing a descriptive standard for audio, associating qualitative parameters for indexing and network retrieval, the same parameters can be used to describe the intended result in a sound synthesis process.

An ideal interface for sound synthesis would effectively adapt to the requests of the user, so that the obtained and the intended sound materials would express noticeable resemblance. However, for such accomplishment, sound description and expectations of the designer should be somehow absorbed by the software. This is actually done with acoustical or psychoacoustical feature analysis. But it is not only a translation or mapping problem here: the system should be able to deal with subjective sound evaluations. We have already proposed the use of existent sound material as reference (Costa et all, 2006), and this paper presents further development on this direction.

The idea of associating qualitative descriptions to sounds is not new. If we rewind at least to the 1950s we will find the efforts made by the Group d'Essay (Schaeffer, 1966) to build sound typologies and morphologies based on qualitative parameters (Smalley, 1986). Some works like (Nicol et all, 2006), which mapped FM synthesis into a timbre space defined by linguistic qualities, and (Johnson & Gounaropoulos, 2006), who also used linguistic expressions but now implementing a direct control of a computational audio processing engine, have dealt with this possibility. Nevertheless, we go further by not establishing a fixed semantic set but, as in early time of electroacoustic composition, we let it open to be adapted to designer’s sonic experience and to be associated with a creative project without using a direct parametrical control. Thus, we present a synthesis process named *Semantic Synthesis* where high-level qualitative parameters are used to control a generative process. Using artificial intelligence algorithms, including machine learning and adaptive systems, we implemented a synthesizer that adapts itself to user’s qualitative expressions toward a more intuitive interaction in a sound design process.

In this paper, the state-of-the-art on audio notation is firstly discussed. Next, we present a methodology to achieve user-orientation on a sound design process with a briefly exposition on how artificial intelligence techniques can be helpful. Further, the *Semantic Synthesis* architecture is described. Preliminary results and considerations on future directions and potential applications are included in the concluding remarks.

2. AUDIO NOTATION REVIEW

Recently, there is an increasing interest on developing a mature formalism on description and analysis of sound contains. It varies from physical characteristics retrieved by sound analysis methods such as (Loureiro & Serra, 1997), to human meaningful relations like: what’s being said – sound retrieval by speech recognition (Kurihara et all, 2006) - or who’s saying – audio notation techniques (Turnbull et all, 2006).

Audio notation techniques is based on the use of keywords related to sound parametrical analysis. It’s a tool for storing descriptive data originated from any kind of method, including both traditional low-level features and high-level contextual meaning.

Martínez (2004) has described the MPEG-7 standard as: formally named "Multimedia Content Description Interface", it is a standard for describing the multimedia content data that supports some degree of interpretation of the information meaning, which can be passed onto, or accessed by, a device or a computer code. It uses meta-data structures named Multimedia Description Scheme (MDS) defined with a Description Definition Language (DDL) based on XML and permits both text visualization humanly readable and compressed binary coded. MPEG-7 has a specific framework and basic semantics for commonly regarded low-level audio features such as spectrum envelope, attack time or harmonicity level. CLAM Library for Audio and Music (Amatriain et all, 2002) [http://mtg.upf.edu/clam/] includes data structure and tools for semantic analysis but let opened which features to be described.

3. USER ORIENTED SOUND DESIGN

Historically, synthesis processes have benefited from low-level analysis method. In fact, audio coding like MP3 is a direct appliance of analysis/synthesis practices. The association between a parameter and the audio signal is ruled by physical and mathematical relations, and therefore tend to generate universal models.

Such association earns complexity with high level features such as qualitative parameters. Hence, there are few models or tools for automatic analysis and even less regarding synthesis. Comparatively to other senses, auditory qualification is the only one that does not have a standardized qualitative set. In fact, other senses expressions like bright (visual) or rough (tact) are commonly borrowed. Not enough, the way different people link such expressions with real sound attributes seems to vary, being their culture references influenced by past experiences.

In this context, the Semantic Synthesis we introduced here aims to digitally generate sounds related and controlled by a model which entails a qualitative description. This description model consists in a set of fuzzy values generated by the user according his/her way of perceiving and qualifying the sound. Beyond a global psychoacoustic model, we have local sets of parameters individually shaped by the listeners experience and context. In order to active our goal and take advantage of artificial intelligence algorithms, the computational models used in our synthesis engine must be designed to (re)adapt themselves to each specific description set. The research area that deals with algorithms capable of improving themselves automatically through experience is known as *Machine Learning* (Mitchell 1997).

4. SYSTEM ARCHITECTURE

4.1 Semantic Sound Trajectories

The main concept of our system is named as "sound trajectories". The user starts by defining a set of parameters (terms) he/she wants to work with. It is a kind of personal dictionary. Them he/she will listen to sounds and notate a collection of samples according to the dictionary. It is done by scoring pre-defined sound windows with values in the interval [0..1] according to a personal evaluation which is based on the dictionary.

Let us consider a collection of sound $\mathbf{S = \{s_1, s_2, s_3, ..., s_N\}}$ with $\mathbf{N}$ samples and a set of keywords $\mathbf{W = \{w_1, w_2, w_3, ..., w_M\}}$ with $\mathbf{M}$ words pre-defined by the user. For each sound in $\mathbf{S}$ we define a *Semantic Sound Trajectory (SST)* $\mathbf{T^k}$ with k=1…$\mathbf{N}$ as follows:

$$\mathbf{T^k} = \{(\Delta t_1, G_1(n_1)), (\Delta t_2, G_2(n_2)), \ldots, (\Delta t_P, G_P(n_P))\} \quad \textbf{Eq.1}$$

where each pair (Δt_j, $G_j(n_j)$) for j = 1..**P** is provided by the user and they are named as "moments". Δt_j is a pre-defined time window and $G_j(n_j)$) is a score mark.

As previously said, it contains the user personal evaluation on how a time window (Δt_j) is related to a keyword from the dictionary **W**. Thus, from the user point of view, the design process consists on three steps:

- **Step 1:** start with a set of samples, define the keyword dictionary, create the SST for each sound. The SSTs will be the model to the system's data base associating keyword scoring to sound fragments, defining a space with **K** semantic trajectories $\mathbf{T} =\{ \mathbf{T}^1, \mathbf{T}^2,\ldots,\mathbf{T}^K\}$.
- **Step 2:** feed the systems data base with $\mathbf{T} =\{ \mathbf{T}^1, \mathbf{T}^2,\ldots,\mathbf{T}^K\}$. The trajectory correspondent to each notated sound file is automatically stored in the data base.
- **Step 3:** use the system to create new sound moments (Δt_j, $G_j(n_j)$). This synthesis process of new sound fragments as well as its interpolations in a new trajectory is handled by the system.

Using the SSTs our goal is to be able to describe sound qualities at several levels, from granular scope (microsound) to a whole sonic event (macrosound). Quality evolution in certain scale will influence the emergence of other qualities in a wider scale. For example, a fast evolution on grains energy determines a specific type of attack. Further, a fast attack followed by a slow decay and a little sustain defines the sonority of an individual sonic event. This is the very basis of sound morphology and typology.

The time evolution of a quality description can be defined as a trajectory in the quality space which is a sub-set of **T**. Trajectories characteristics may define other qualities in another sub-set, which could have its own trajectories as well. Figure 1 shows a graphical illustration of this multi-level interplay. Each screen represents a parametric quality space in a different level. The circles represent instantaneous quality points while the curves represent a trajectory in the space.

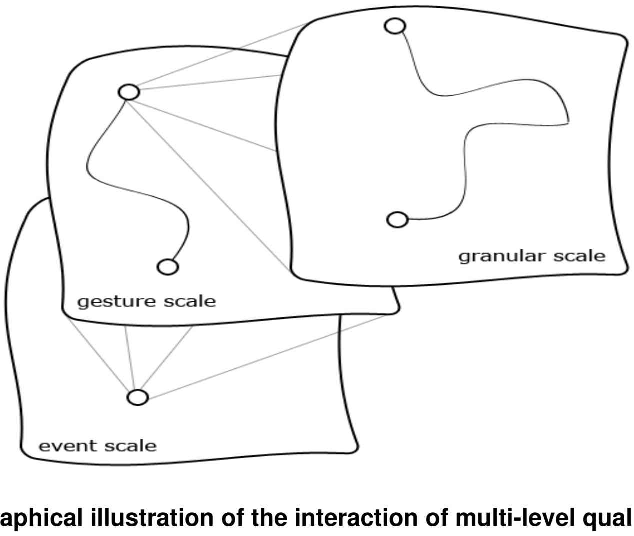

Figure 1. Graphical illustration of the interaction of multi-level quality spaces

4.2 Synthesis Process

The synthesizer has three main structures:

- *Description schema*: data structure that operates as an interface between semantic description and software;
- *Synthesis Engine*: general purpose audio synthesizer;
- *Translation Unit*: adaptive computational structure that for a given description returns parametric values for the synthesis motor.

The way these structures interact to synthesize audio is presented in Figure 2. The synthesizer receives a semantic description formatted into a description schema as input. The translation unit converts this description into control parameters of the synthesis engine in order to generate sound outputs.

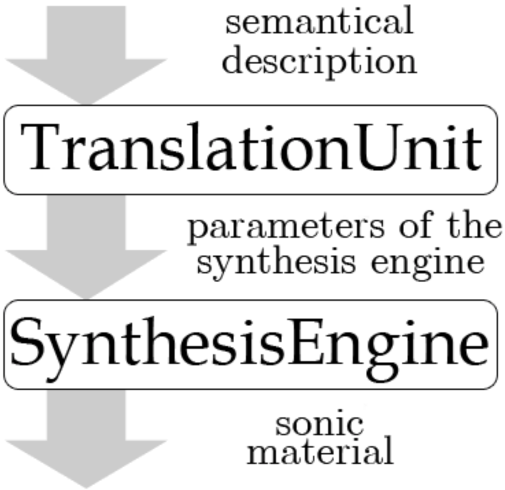

Figure 2. Synthesis Process

Both synthesis engine and description schema are pre-defined structures with well known behavior. The translation unit is an adaptive structure modified by learning machine algorithms, which performs statically during the synthesis process. It maps the description space into the synthesis motor parametric space. Due to mapping variation from user to user, the translation unit is always different.

To learn space mapping, the translation unit is trained with mapping samples, i.e., semantic descriptions and the sound material they represent. The learning procedure is presented in Figure 3.

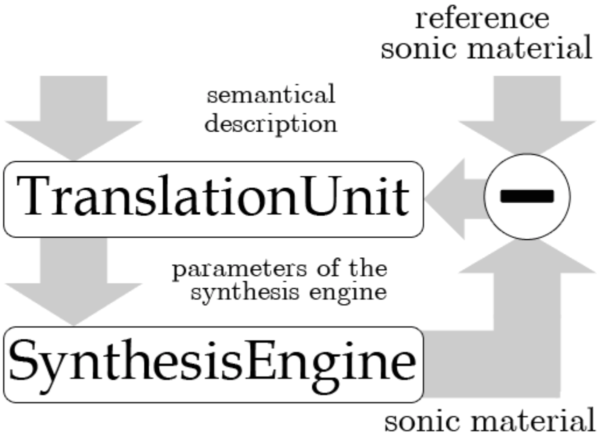

Figure 3. Learning Process

4.3 Implementation Considerations

A semantic synthesizer working with single-level granular scale trajectory has been fully implemented as a C++ shared library. It uses an abstract structure as synthesis engine for modular implementation of any general solution and an artificial neural network as the translation unit.

4.3.1 Description Schema

In a single-level quality space schema it could be structured parametrically. The schema is composed of a set of adjectives. A quality point is defined as a list of numerical values associated with the adjectives. For example, one may select the adjectives *heat, roughness, brightness* as the description schema and a quality point could be defined as (0.7, 0.5, 0.1), respectively for heatness, roughness and brightness.

4.3.2 Translation Unit

A multi-layer perceptron (MLP) artificial neural network with back-propagation to implement the learning algorithm is used as the translation unit. It is based on the fast artificial neural network (FANN) library (Nissen 2003). MLP works as a universal function interpolator, i.e., it maps a set of sampled points into a functional space. Here, the task is to approximate the function that relates a point from the qualitative parametric space to a point in the audio sample space.

4.3.3 Synthesis Engine

MLP is used to generate sound material directly from semantic description. However, MLP performance and capability to generalize any function is affected by space dimensionality and the quality of the available dataset. That is why we have adopted a Discrete Fourier Transform to implement the synthesis engine.

4.3.4 Trajectories Interpolation

User is expected to define the semantic trajectory by means of defining quality points. A quality point is composed of a sequence of qualitative parameters and the equivalent time stamp. A complete curve of qualitative values is generated by linear interpolation of such points in the parametric space.

4.3.5 Trajectories Translation

The output of the system is an audio stream that refers to the parametric trajectory. The translation is done in a granular size window scale, and the final stream is obtained using the overlap-and-add technique. The translation is done by presenting the instantaneous parameter to the MLP and converting the resultant floor and residue into a sound window.

5. Results

The system is expected to be capable of generalizing any given quality space to sound material mapping. This could be tested by two manners: with parametric synthesizers with well-known behavior; with a real human mapping and hearing sections procedures.

Using parametric synthesizers, it is possible to verify the system capability to learn a specific behavior. Their mapping must be sampled creating a training data set. This kind of experiment is not enough to estimate the capability to generalize human quality relations to sound material. However, it shows system capability to associate low-level audio behavior with some sort of parametrical control.

With human mapping there would be necessary that a designer specifies his/her own training set. After training, parametric specific data have to be synthesized and presented to designer for parametric classification. Then, human classification can be compared with the original parameter set that resulted on the sound material.

In the current implementation, with high dimensional synthesis data presented to the MLP, the experiments were limited to simple parametric synthesis. A four parameters synthesizer based on subtractive synthesis was programmed in MATLAB. Sound material could be controlled regarding white noise energy, fixed low-pass filter gain, fixed band-pass filter gain and fixed high-pass filter gain. A comprehensive training set was created to map the whole parametric space, with 625 samples presented to the network. Figure 4 shows some experimental results.

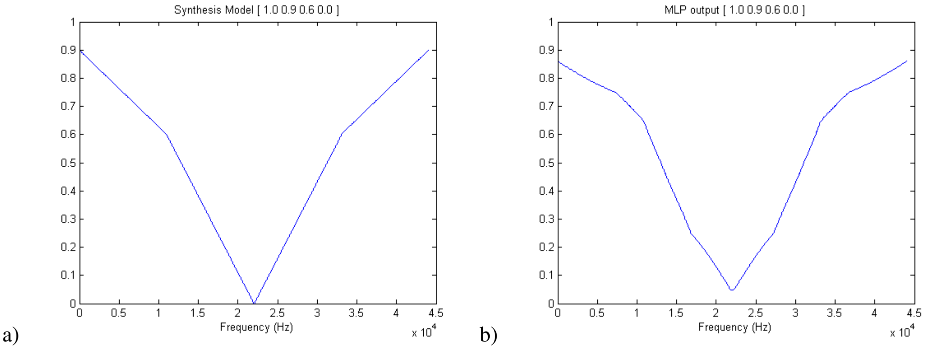

Figure 4. (a) Synthesis model output and (b); MLP output for the same parametrical stimulus

In this particular experiment, the MLP was capable of establishing a coherent approximated map. However, it has its limitations depending on the parametric behavior. For example a synthesizer with some sort of pitch parameters (such as FM), this kind of behavior would require an extensive training set since it needs an extremely connected behavior on the artificial network. On high dimensional data, it would be impossible to cover the whole synthesis space.

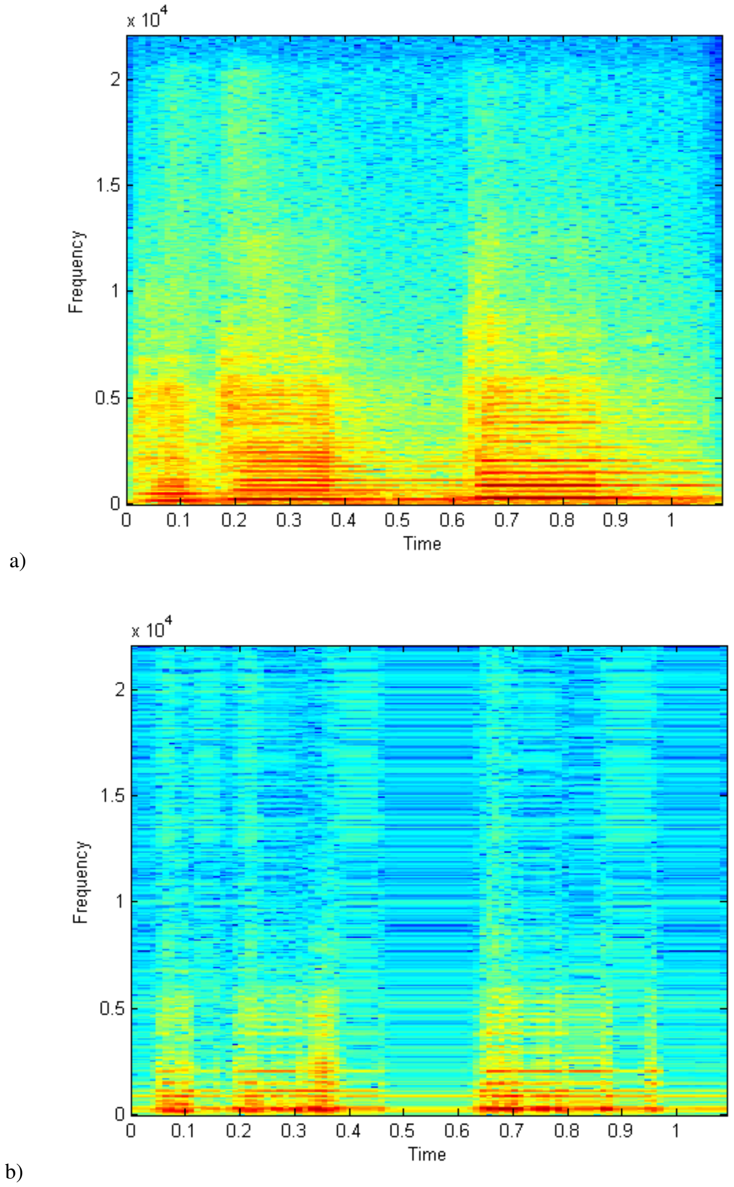

Figure 5. (a) Clarinet sample used for training and (b) system output

As a second experiment we covered the pitch problem. As input, a clarinet record (Figure 5a) was presented with parameters regarding to pitch and time position. Two different pitches have been presented. The output (Figure 5b) shows good response for the time positioning and energy mapping but, as expected, it did not generalized well the pitch parameter and spectral shape.

This problem extends to real world sounds and any given semantic description which is very likely to be dependent on complex signal behavior. Evolution solved this but molding specialized systems that process the raw signal into a lower dimensional and friendlier processed stimulus for the neural system.

Our approach is to use more specialized synthesis models, supported by psychoacoustic knowledge. An audio codec, as Vorbis (Moffitt 2001) can be used as the synthesizer motor. Vorbis is an open source audio codec. It works by spectral analysis, granular segregation, codification and further overlap-and-add reconstruction. It codifies the original signal window into spectral envelop (floor) and residual spectral density (residue). It also uses mapping system for multi-channel. Vorbis work as a dimensionality reducer. In the training module, audio window is first segregated into a floor and a residue, filtered by the Vorbis psychoacoustic model, and then presented to the MLP. Hence, in the synthesis module, the MLP provides as outputs floor and residue. Further they are joined in a regular audio window.

6. Conclusion and Further Works

The possibility of synthesizing sound material directly from semantic description of any ordinary mapping makes it a universal synthesizer. It can be used to override complex synthesis systems, to add new control paradigm to traditional methods or to generalize sound material mappings into a computational process.

It is expected that its generalization performance get enhanced with more specialized synthesis engines. The current implementation architecture has been projected for modular use, and other modules are being developed.

Regarding future directions, we first expect to have a functional implementation with other synthesis engines in hand for testing and validation. Further we aim to obtain better results with a more comprehensive description scheme. The capability of interaction with audio description standards like MPEG-7, or notation structures on audio frameworks like CLAM, would allow other programs to easily control our library.

Also, to allow the system to interact with more complex descriptions, including multi-level representation, it would be necessary to develop a translation unit based on highly structured computational models. Pieces of software like IQR (Bernardet et all, 2002) could be used to conceive such models. We are also going to consider the use of Support Vector Machines as an alternative (Cortes et all, 1995).

7. Acknowledgments

Authors would like to thank Brazilian Agencies CNPq and FAPESP for the financial support. Costa is supported by FAPESP, Furlanete by Capes, Manzolli and Von Zuben are supported by CNPq.

8. References

J. van Ossenbruggen, F. Nack, and L. Hardman. (2005) That Obscure Object of Desire: Multimedia Metadata on the Web (Part II). IEEE Multimedia, 12(1), January-March 2005.

Pachet, F., La Burthe, A. and Aucouturier, J-J, (2003) «The Cuidado Music Browser: An end-to-end EMD system» in Proc. 3rd International Workshop on Content-Based Multimedia Indexing, September 2003, Rennes (France)

Herrera, P., Celma, O., Massaguer, J., Cano, P., Gómez, E., Gouyon, F. & ,Koppenberger, F. (2005): MUCOSA: Proceedings of 6th International Conference on Music Information Retrieval; London, UK (pp. 77-83).

Costa, C. R., Manzolli, J., Von Zuben, F.J., (2006) "Population-Based Generative Synthesis: A Real-Time Texture Synthesizer based on Real-World Sound Streams" In Proceeding of 4th AES Brazil Conference, São Paulo, Brazil.

Schaeffer, P. (1966) Traité des objets musicaux: Essai interdiscipline. Le Seuil, Paris (France).

Smalley, D. (1986) Spectromorphology and Structuring Processes. In The Language of Electroacoustic Music, Org. Emmerson S., Harwood Academic Publishers, New York (USA).

Nicol, C., Brewster, S., Gray, P., (2006) "Designing Sound: Towards A System For Designing Audio Interfaces Using Timbre Spaces" Proceedings of ICAD 04-Tenth Meeting of the International Conference on Auditory Display, Sydney, Australia.

Johnson, C. Gounaropoulos, A. (2006) Timbre interfaces using adjectives and adverbs. In Proceedings of the 2006 Internacional Conference on New Interfaces for Musical Expression (NIME06), Paris, France.

Loureiro, R. Serra, X. (1997) A web interface for a sound database and processing system. In Proceedings of the Internacional Computer Music Conference, ICMC 1997.

Kurihara, K. Goto, M. Ogata, J. Igarashi, (2006). T."Speech pen: predictive handwriting based on ambient multimodal recognition". In Proceedings of the SIGCHI conference on Human Factors in computing systems: 851-860.

Turnbull, T. Barrington, L. Lanckriet, G., (2006) "Modelling Music and Words using a multi-class naïve Bayes approach". In proceedings of the 7th Internacional Conference on Musical Information Retrieval, Vitoria, Canada.

Martínez, J.M, (2004) MPEG-7 Overview (Version 10) ISO/IEC JTC1/SC29/WG11 Palma de Mallorca, October 2004.

Amatriain, X. , Arum, P. , and Ramírez, M. (2002) CLAM, Yet Another Library for Audio and Music Processing? In Proceedings of the 2002 Conference on Object Oriented Programming, Systems and Application (OOPSLA 2002), Seattle, USA, 2002. ACM.

Pierce, J.R., (1992) "The Science of Musical Sound" rev. Ed. W.H. Freeman and Company, New York.

Nissen, S., (2003) “Implementing Implementation of a Fast Artificial Neural Network Library (fann)”, Intern Report, Department of Computer Science, University of Copenhagen (DIKU).

Moffitt, J., (2001) “Ogg Vorbis—Open, Free Audio—Set Your Media Free” Linux Journal, Specialized Systems Consultants, Inc. Seattle, WA, USA, Vol. 2001, Issue 81es, Article No. 9.

Bernardet, U., Blanchard, M. Verschure, P. F. M. J.: (2002) “IQR: a distributed system for real-time real-world neuronal simulation.” Neurocomputing, 44-46: 1043-1048.

Cortes, C. and Vapnik, V., (1995) "Support-Vector Networks”, Machine Learning, 20.