

USING AUDIO FEATURE EXTRACTION FOR INTERACTIVE FEATURE-BASED SONIFICATION OF SOUND

Sam Ferguson

Creativity and Cognition Studios
 School of Software
 Faculty of Engineering and IT
 University of Technology, Sydney
 Samuel.Ferguson@uts.edu.au

ABSTRACT

Feature extraction from an audio stream is usually used for visual analysis and measurement of sound. This paper seeks to describe a set of methods for using feature extraction to manipulate concatenative synthesis, and develops experiments with reconfigurations of the feature-based concatenative synthesis systems within a live, interactive context. The aim is to explore sound creation and manipulation within an interactive, creative, feedback loop.

Index Terms— Interactive Sonification, Concatenative Synthesis

1. INTRODUCTION

In this paper, we seek to discuss and explore approaches to live interaction with sonifications of sound. Sonification of the characteristics of sound have been undertaken in the past using various methods [1, 2, 3], but most of these have dealt with offline, static, processing of recorded sound. In this study, we investigate ways to explore and interact with sound as it is produced, or as it is played. This provides methods for:

1. exploring the characteristics of recorded sound rapidly and interactively;
2. responding to characteristics of instrumental sound in a feedback loop;
3. manipulating and mutating sampled sound in an interactive manner;
4. creating new responsive sounds.

Sonifications of sound, while seemingly a tortology, are in fact an incredibly sensible application of sonification, and one that should be expected to hold strong potential. When one wishes to understand a sound recording it is common to

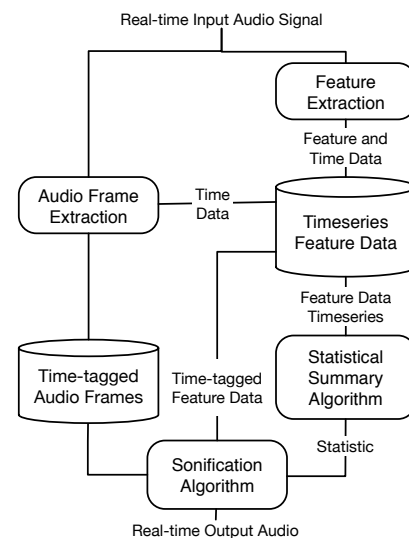


Figure 1: *Brief Overview of Sonification System*

listen to it carefully, replaying sections of interest and making comparisons with other sections. Thinking generically, this process could be compared to accessing a dataset, reading one particular number and then comparing that with another number within the dataset. However, despite data analysis commonly involving much more sophisticated techniques than simple comparisons or readings of datasets, techniques for listening to an entire recording or listening to specific algorithmically chosen parts of a recording are limited or non-existent. Sonification techniques, partnered with granular or concatenative synthesis, provide a solution to fill this gap, and this has been explored by Ferguson et al. [4]. Summative numerical results of feature extraction from audio signals can obscure the divergent nature of different audio signals, as feature extraction algorithms are naturally reduc-



This work is licensed under Creative Commons Attribution Non Commercial 4.0 International License. The full terms of the License are available at <http://creativecommons.org/licenses/by-nc/4.0>

tionist, but the process of representing sound data in the auditory modality can help to place audio characteristics in their proper context and balance the precision of abstract numerical quantities with the ground truth of auditory sensory perception. Using the sound material, reorganised or transformed in some way using methods that mimic typical visualisation techniques, to *re*-represent the extracted sound data mean that typical analysis approaches can happen in the auditory domain rather than visual domain or happening completely analytically [5, 4].

This paper extends this concept by investigating approaches to *live interaction* with sonifications of sound. Modern digital signal processing have facilitated the creation of real-time versions of audio feature extraction algorithms that previously required offline processing. The real-time nature of this processing significantly increases the set of uses that the results of feature extraction can be applied to, most notably feature extraction can be used to act as a control for real-time sound manipulation.

2. BACKGROUND

Sonification has been used for many years to represent generic numerical data in an analogous way to visual graphing, and there is some evidence that it is more effective than visualisation in particular contexts, especially for monitoring real-time data (eg. [6]). Statistical representations has been sonified in the past for various purposes - sonifications have been used for representing probability densities [7, 8], statistical representations [9, 10] and for listening for abnormal sounds or statistical anomalies in a stream of data [11, 12, 13].

The concept of *Adaptive* digital audio effects (A-DAFX) was introduced by Verfaillie et al. [14] and extends an audio effect using static control values by employing features extracted from the input audio signal as control inputs for the audio effects being applied to the signal. As Verfaillie et al. point out [15], a compressor or limiter incorporates a feedback loop to use the level of the input audio (a feature) to control gain change in a systematic way. Similarly, processing by ‘auto-tune’ algorithms corrects pitch inaccuracy by assessing the extracted pitch against the closest correct pitch, and applying a varying pitch shift. How A-DAFX differ from these examples is that the input feature is not specific to a particular audio effect, but is arbitrary and modular. In a similar fashion but with a slightly different purpose, Park et al. have also theorised this idea as Feature Modulation Synthesis [16, 17].

These approaches are strongly associated to the work on concatenative synthesis [18, 19, 20, 21], but reapplied to an exploration and representation purpose. Further, Schwarz has recently investigated interaction with ‘sound spaces’ as a method of playing concatenative synthesis systems [22].

Performing with a traditional musical instrument often

involves practising the instrument, whether for a scored work or for improvisation, and repeating tones and practical manoeuvres in performance that have been precisely learnt during practise. For instance Carey’s *derivations* system [23] is an improvisational computer system that responds to musical sound, while Johnston et al. [24] have discussed the process of designing conversational interaction with digital systems. Of course before these more recent systems, many computer systems have been designed that are responsive and improvisational, or at least give that impression, including Lewis’ *Voyager* [25], and Rokeby’s *Very Nervous System* [26].

3. METHOD

There are several processes that make up this framework, namely, a) feature detection, b) manipulation description, c) manipulation application, and d) interaction.

Altering sounds in adaptive ways that differ from traditional input-output sound processing requires a second pathway to be introduced to the pipeline. This is developed by adding a feature detection stage to create a secondary data stream running parallel to the audio stream. This requires rapid real-time calculation of features to generate feature data for manipulation purposes, as well as a memory buffer to store recent audio data in a convenient format alongside the feature data. The two datasets are indexed by their time tags, so they can be related directly to each other. The second component of the system is the manipulation of the digital audio based on a transformation of the feature data into some type of function or re-organisation scheme that can be applied to audio data. Thirdly, there is the process of applying this manipulation to the audio stream in an efficient manner. Finally, the process of interacting with each of these stages is a basic issue that limits the applicability of methods of this nature. Obviously, interaction with an audio stream brings the crucial issues of causality and latency.

3.1. Feature Extraction

When considered abstractly, although feature detection algorithms describe sound characteristics in many distinct and different ways, they fall into a small number of particular formats. The simplest format is for a feature detector to take a frame of sound (often approximately 20-50 ms), analyse it, and then return a single numeric value as a response (see Figure 2). For each frame (of for instance 2048 samples) of contiguous sound, a numeric value is produced by the feature detector algorithm, and a time-series data trace is built from these new changing values. This type of feature detector is very common and easily used for building sonifications of sound [4], as the feature data output is completely predictable (for every frame of sound a single numeric value

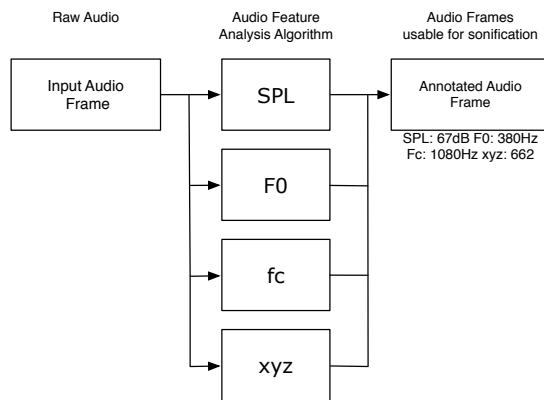


Figure 2: Feature detection algorithms that produce a single real number for a frame of sound can easily be treated as a black box.

is returned). This data format allows many different features (harmonic, brightness, pitch, loudness, etc.) to be treated by algorithms in an identical manner, although the characteristics investigated are likely to be very different. Once a set of audio frames and time-tagged numeric feature data are collated, the various statistical algorithms can be applied to the numeric data and audio frames at the same time.

Many audio features that do not conform to this simple format, and do not output a single value per frame. Some analysis algorithms result in a set of numbers being produced from one frame, as for instance, the Fourier transform, the mel frequency cepstral transform, or octave band analysis do. Furthermore, some other feature detectors may be unpredictable, in that they may create an unknown number of values (including 0) from an audio frame, depending on the content of the sound. In this paper we will focus mainly on the implications for datasets made up of single timeseries features, however other types of feature could be incorporated in further study.

3.2. Time-series Statistics

After the feature detection algorithm has been applied to the audio, a new time-series is created that consists of the feature time-series data. This numeric data is then mapped to a sonification algorithm that uses audio frame data, and various processes exist by which this may be done. Using statistical methods the feature time-series can be summarised as a value using typical descriptive statistics methods, for instance the median or maximum value. Running the statistical analysis at each addition to the time-series during real-time analysis means that the statistical analysis is also a parallel time-series, but one which represents the characteristics of the

rapidly varying feature data in a summative or indicative manner. A statistical indicator of this nature can then be used as an input to the frame selection method that follows this stage. It is likely it would play a role as, for instance, determining the centre of a range from which to select frames of the same pitch.

A further approach is to use the statistical time-series of the feature time-series to find a second order statistical time-series. A difference between the current value and an extreme (e.g. the minimum or maximum value), would search for sounds close to the upper reaches of the feature - in the case of pitch, when differencing against the maximum pitch value, the difference would be smallest when the pitch was closest to that maximum value.

3.3. Frame Selection Method

The statistical analysis of the feature detector time-series essentially creates another time-series. The method used to apply this to the audio stream can be one of many alternatives, based somewhat on the purpose of the sonification. Playing frames of sound rapidly has the effect of physically representing the statistics of the sound [27], and so links well with the statistical analysis examples described in the previous section. An example of this could be the playing of frames of the sound produced when a flute plays the note A. If the frames were drawn from recordings of a performer with precise tuning, then the average sound created when they are rapidly presented together will be a precisely tuned A. However, if the performer plays an A with various tunings, or perhaps with a vibrato, then the average sound will represent this information by blurring the tuning across a pitch range, but also giving a general impression of the mean pitch.

In statistical terms, the concatenated sound, when temporally blurred, represents the dispersion of the feature data extracted from the sound. Similarly, if the performer has excellent precision but has low accuracy tuning (plays the same, inaccurate tone repeatedly), then this will also be represented. Statistically, this would have a comparatively low dispersion, but a high deviation of the central tendency of the distribution from the correct tone. The simplest way of looking at a feature time-series is by using descriptive statistics (Figure 3), each of which can be turned from a numerical value into a simple sound by selecting the appropriate frames of audio from the sample (see Figure 4).

A more significant application of the manipulation time-series is to drive the selection of frames to be blurred with the current frame. Where the value of the feature time-series is close to the values of recent frames, those frames can be 'blurred' with the current frame to create a textural sound composed of audio that is similar within one feature dimension. This will create a 'simple' sound, where the frames are highly similar, and a 'complex, muddy' sound

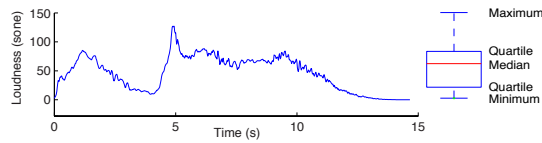


Figure 3: A time-series may be summarised with descriptive statistics, and visualised with a box plot.

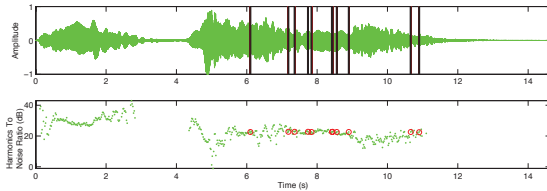


Figure 4: Median feature frames being drawn from a sample. Again the feature detector is arbitrary, and in this case is Harmonics to Noise Ratio.

tending towards noise, where there is a significant difference between the characteristics of the frames. This effect can be seen easily where one blurs multiple frames of a piano playing a single tone, compared with multiple frames of a singer singing a tone with vibrato, the change in pitch caused by the vibrato is shown clearly in the resulting blurred tone - which deviates across the pitch range traversed by the vibrato.

In this work the term concatenative synthesis will be used to describe the process of re-synthesizing sound from the recording, in order to link this research with previous work that has been based on feature extraction and then audio frame concatenation. In fact, this technique also has a lot in common with typical granular synthesis methods (see Roads' Microsound [28] for a review). Granular synthesis, however, in most instances does not make use of feature data in the selection and playback of grains of sound – it is usually based on random frame choice guided by parameters such as grain duration, grain window function, grain transposition and grain density (how many grains are selected at one time). By contrast, concatenative synthesis tends to use set methods for most of these parameters, uses randomisation sparingly, and is more concerned with the selection of optimal frames of sound in order to match a target, or to match the path closest to a target sound.

3.4. Interaction

Given that the system is not time-invariant, the sound that is an input to the system also acts as an interaction input, as the features produced by the musician are transformed into

control data. This means that by playing their instrument into the system the musician has a stronger form of control over the way that the system is controlled than if they were using a linear time-invariant system (such as a reverberation or a delay effect).

A simple example is to create a system that lowers the gain for notes that are not precisely consonant with a specified temperament system. That is, the instrument is altered so that notes that are 'out of tune' are softer than notes that are 'in tune'. As pointed out by authors in the past [2], this means that the visual modality need not be used to experience auditory material (i.e. a musician doesn't have to look at a meter or dial to receive information about whether they are in tune).

A similar feedback loop exists where the audio feedback is not controlled by using gain only, but by replacing (or augmenting) the natural audio feedback with sound produced with concatenative synthesis. This technique is different to natural audio feedback because it allows the use of the sound's recent history to be compared with the current sound. That is to say that the sound produced by concatenative synthesis can be composed of frames of sound recorded in the very recent history, reorganised systematically to represent an 'average' or mean sound. The selection of which frames of sound to use is crucial, and will determine what type of sound is received as feedback.

4. EXAMPLES

Examples of this framework will help demonstrate it in use in various contexts. The following examples are different configurations of the same basic concepts.

4.1. Listening to a descriptive statistic of a feature

In this example the system - 1) calculates the feature extraction, 2) calculates the running statistic (the median in Figure 5) of the feature time-series, 3) which is then used as the basis for the criterion for the selection of output audio frames. These frames are selected randomly within a range around the statistic and then concatenated for output.

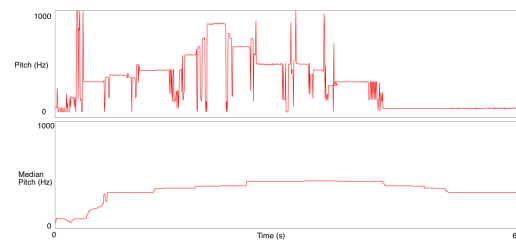


Figure 5: The feature data (upper pane) is filtered by a running median filter (lower pane).

The median is an interesting statistic to follow, but is calculated in the same way that the percentiles, quartiles and maximum and minimum are (the median is also the 50th percentile), so in effect the statistic itself is also a parameter that can be interacted with. One may choose to control the statistic using any of many interaction methods, that could be obtained in real-time to form part of the performance practice of the musician, thereby creating a new musical interface.

4.2. Listen to peaks from a feature histogram

This configuration takes the previous example, and replaces the median extraction with a histogram, the crucial difference between the two being that a histogram is a multidimensional method of describing a distribution of a times-series, whereas the median is one dimension only. The advantage of a histogram is that multiple areas of activity can be located, rather than only one. These multiple peaks can then be used as inputs to the frame selection criterion. This means that, for instance, when using pitch as a feature input, if one wished to play two notes simultaneously, one would play each note for a long duration, and the histogram would show two peaks at each pitch, which could then be used to select frames from the audio containing those two pitches. To change the notes that are selected one would simply play another note for a longer duration, and the histogram would change accordingly (see Figure 6 for an example).

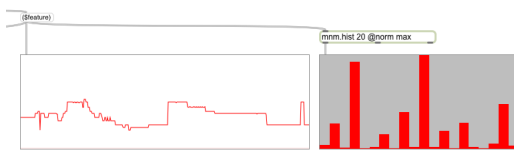


Figure 6: *In this example, a feature time-series is recorded (eg. pitch, left pane), and the feature is statistically analysed to build a histogram (right pane) that shows which values (notes) continued for the longest.*

A configuration of this nature allows the creation of polyphonic chordal sound, that is closely related to the input sound. This means the input musical melody can be reframed as a method of playing notes for chordal outcomes rather than melodic, requiring a rethinking of the way that improvisation is envisaged.

4.3. Using sound level to control pitch range

Although the previous examples use only one feature as an input to their configuration, it is also of course possible to use two feature inputs and map them to different parameters of the same frame selection criteria. In this case we use the

pitch of the sound to choose the pitch of the frames that are selected, but also use sound level to control the size of the pitch range from which frames are selected.

This opens the possibility to different levels of control - a basic type of control may exist where the feature is used in a mapping that follows the same contour directly (eg. the pitch of the input audio being used to direct the pitch of frames selected). Alternatively, a mapping may non-linearly respond to a feature - in this example the feature range used in the frame selection can be constant for values of sound level that fall below a threshold, but rapidly expand when the threshold is exceeded, providing both predictability when appropriate and rapid change when necessary.

4.4. Implementation

The implementation of this system was completed using the Max/MSP platform¹, alongside FTM and Gabor extensions [29, 30] as the basis for the feature extraction (using the *Yin* pitch algorithm [31]), as well as the MNM extensions performing the statistical processing of the feature time-series [32].

5. DISCUSSION

This paper addressed methods of exploring the characteristics of sound and performing with sound through sonification of feature data extracted from the sound. It identified methods by which the statistics of sound could be explored in realtime, and as the sounds were being produced.

The basis of this framework is to

- apply feature detection to an audio signal, to create a feature time series;
- apply statistical analysis to the feature time-series to create a value or set of values that can be used;
- create a criterion for frame selection that is based on the feature time-series statistical analysis
- use the selected frames in the application of concatenative or granular synthesis.

Using the features of a created sound as an interaction method is not a common approach to musical interaction. Interaction inputs tend to be thought of as ‘controls’, implying that the user of the system has complete knowledge of what action they wish the system to undertake, and that the system is purely deterministic following the user’s command. Many musical contexts, however, rely on communication and reflection between musical participants for the musician’s purpose to be fully realised, with the concept of ‘jamming’ a common one.

Nevertheless, while the framework is designed to be reflective rather than one-way, the fact that the system is based

¹<http://www.cycling74.com>

on simple statistical methods, rather than opaque neural networks or machine learning techniques, means that there is the likelihood of a musician *learning the system* – and with enough knowledge of the configuration of a method they even be able to subvert the intentions of a system and achieve novel outcomes. For instance, consider a musician repetitively playing two notes an octave apart into a configuration that is seeking and replaying the median pitch. A system that responds to musical output in a predictable, but still complex fashion allows for new types of creative opportunities.

Clearly, the re-representation of feature data by re-playing the sound that was analysed to create it means that the feature under investigation is linked inextricably to the sound produced. There are hundreds of defined feature extraction algorithms that can be used in the feature detection stage of the system (see [33] for software that implements a wide array of them). As they often have exactly the same data format (frame of sound in, single numeric value out), many of them are completely interchangeable in this framework (except that real-time implementations in the target platform may not be easily obtained). However, such a reconfiguration of the feature detection may offer creative possibilities that are unpredictable or unexpected, as they can have quite idiosyncratic characteristics.

Statistical methods are often used in data analysis for their ability to find patterns and draw out the ‘nature of things’. Despite being applied to musical feature data in real-time they still have this ability, and thus they act as an immediate reflection of the characteristics of the sound over a period in recent time. Used in an appropriate manner they have the ability to allow listeners to examine the nature of steady sound compared with changing sound, and to listen to the way that sounds change over time. They can also be used to make comparisons and to assess the range of variation within a feature rapidly.

6. CONCLUSION & FUTURE RESEARCH

This paper has described an approach towards the use of feature extraction and feature data analysis for creative and exploratory musical possibilities. We have defined a simple framework for the sonification of sound played into a computer system, based on the statistical characteristics of the feature time-series data extracted from the audio in real-time. Examples of the configuration of the system are presented to demonstrate the variety of ways the system can be configured.

There are many opportunities for future work aligned with this research direction. The modularity of the framework, and the way in which the stages may influence each other is an important element to be investigated. Also, characterising the statistics of different feature detectors, in terms of their noise, precision and reliability may help when choosing

appropriate methods of input signal analysis. The element of time and rhythm is essentially ignored in the statistical processes described above, but is likely to be able to make an important contribution of the musicality of this system. Finally, a user study with practising musicians is likely to lead to important findings about the system being used in practice.

7. REFERENCES

- [1] D. Cabrera, S. Ferguson, and R. Maria, “Using sonification for teaching acoustics and audio,” in *1st Australasian Acoustical Societies’ Conference*, Christchurch, New Zealand, 2006.
- [2] S. Ferguson, “Learning musical instrument skills through interactive sonification,” in *New Interfaces for Musical Expression (NIME06)*. Paris, France: IRCAM – Centre Pompidou, 2006, pp. 384 – 389.
- [3] D. Cabrera and S. Ferguson, “Auditory display of audio,” in *120th Audio Engineering Society Convention*, Paris, France, 2006.
- [4] S. Ferguson and D. Cabrera, “Exploratory sound analysis: sonifying data about sound,” in *14th International Conference on Auditory Display*, Paris, France, 2008.
- [5] S. Ferguson, “Exploratory sound analysis: Statistical sonifications for the investigation of sound,” Ph.D. dissertation, 2009.
- [6] W. T. Fitch and G. Kramer, “Sonifying the body electric: Superiority of an auditory over a visual display in a complex, multivariate system,” in *Auditory Display*, G. Kramer, Ed. Santa Fe Institute Studies in the Sciences of Complexity, Addison-Wesley, 1994, vol. 18, pp. 307–325.
- [7] J. Williamson and R. Murray-Smith, “Sonification of probabilistic feedback through granular synthesis,” *IEEE Multimedia*, vol. 12, no. 2, pp. 45–52, 2005.
- [8] ———, “Granular synthesis for display of time-varying probability densities,” in *International Workshop on Interactive Sonification*, Bielefeld, Germany, 2004.
- [9] J. H. Flowers, D. C. Buhman, and K. D. Turnage, “Cross-modal equivalence of visual and auditory scatterplots for exploring bivariate data samples,” *Human Factors*, vol. 39, no. 3, pp. 341–351, 1997.
- [10] J. H. Flowers and T. A. Hauer, “The ear’s versus the eye’s potential to assess characteristics of numeric data: Are we too visuocentric?” *Behaviour Research Methods, Instruments and Computers*, vol. 24, no. 2, pp. 258–264, 1992.

- [11] T. Hermann, C. Niehus, and H. Ritter, “Interactive visualization and sonification for monitoring complex processes,” in *Proceedings of the 9th International Conference on Auditory Display*, Boston, USA, 2003.
- [12] T. Hermann, G. Baier, U. Stephani, and H. Ritter, “Vocal sonification of pathologic eeg features,” in *Proceedings of the 12th International Conference on Auditory Display*, London, UK, 2006.
- [13] J. Edworthy, E. Hellier, K. Aldrich, and S. Loxley, “Designing trend-monitoring sounds for helicopters: methodological issues and an application,” *Journal of Experimental Psychology Applied*, vol. 10, no. 4, pp. 203–18, 2004.
- [14] V. Verfaillie, U. Zolzer, and D. Arfib, “Adaptive digital audio effects (a-dafx): A new class of sound transformations,” *Ieee Transactions on Audio Speech and Language Processing*, vol. 14, no. 5, pp. 1817–1831, 2006.
- [15] V. Verfaillie, M. M. Wanderley, and P. Depalle, “Mapping strategies for gestural and adaptive control of digital audio effects,” *Journal Of New Music Research*, vol. 35, no. 1, pp. 71–93, 2006.
- [16] T. H. Park, J. Biguenet, Z. Li, C. Richardson, and T. Schar, “Feature modulation synthesis (fms),” in *International Computer Music Conference*, Copenhagen, Denmark, 2007.
- [17] J. B. Tae Hong Park, Zhiye Li, “Not just more fms: Taking it to the next level,” in *International Computer Music Conference*, Belfast, Northern Ireland, 2008.
- [18] D. Schwarz, “A system for data-driven concatenative sound synthesis,” in *COST G-6 Conference on Digital Audio Effects (DAFX-00)*, Verona, Italy, 2000.
- [19] —, “The caterpillar system for data-driven concatenative sound synthesis,” in *6th International Conference on Digital Audio Effects*, London, UK, 2003.
- [20] —, “Data-driven concatenative sound synthesis,” Ph.D. dissertation, 2004.
- [21] D. Schwarz, R. Cahen, and S. Britton, “Principles and applications of interactive corpus-based concatenative synthesis,” in *Journ??es d’Informatique Musicale (JIM’08)*, Albi, 2008.
- [22] D. Schwarz, “The sound space as musical instrument: Playing corpus-based concatenative synthesis,” in *New Interfaces for Musical Expression (NIME’12)*, Ann Arbor, USA, 2012.
- [23] B. Carey, “Designing for cumulative interactivity: The _derivations system,” in *New Interfaces for Musical Expression*, Ann Arbor, Michigan, 2012.
- [24] A. Johnston, L. Candy, and E. Edmonds, “Designing and evaluating virtual musical instruments: facilitating conversational user interaction,” *Design Studies*, vol. 29, no. 6, pp. 556–571, 2008.
- [25] G. Lewis, “Too many notes: Computers, complexity and culture in voyager,” *Leonardo Music Journal*, vol. 2000, no. 10, pp. 33–39, 2000.
- [26] D. Rokeby, “Transforming mirrors: Subjectivity and control in interactive media,” *Leonardo Electronic Almanac*, vol. 3, no. 4, p. 12, 1995.
- [27] S. Ferguson and D. Cabrera, “Auditory spectral summarisation for audio signals with musical applications,” in *10th International Society for Music Information Retrieval Conference*, Kobe, Japan, 2009.
- [28] C. Roads, *Microsound*. Cambridge: MIT Press, 2001.
- [29] N. Schnell, R. Borghesi, D. Schwarz, F. Bevilacqua, and R. Müller, “Ftm - complex data structures for max,” in *International Computer Music Conference*, Barcelona, 2005.
- [30] N. Schnell and D. Schwarz, “Gabor, multi-representation real-time analysis/synthesis,” in *COST-G6 Conference on Digital Audio Effects (DAFx)*, Madrid, 2005.
- [31] A. de Cheveigné and H. Kawahara, “Yin, a fundamental frequency estimator for speech and music,” *Journal of Acoustical Society of America*, vol. 111, no. 4, 2002.
- [32] F. Bevilacqua, R. Muller, and N. Schnell, “Mnm: a max/msp mapping toolbox,” in *International Conference on New Interfaces for Musical Expression (NIME05)*, Vancouver, BC, Canada, 2005.
- [33] D. Cabrera, S. Ferguson, and E. Schubert, “Psysound3: Software for acoustical and psychoacoustical analysis of sound recordings,” in *Proceedings of the 13th International Conference on Auditory Display*, Montreal, Canada, 2007.