AUDITORY SPECTRAL SUMMARISATION FOR AUDIO SIGNALS WITH MUSICAL APPLICATIONS

Sam Ferguson
Faculty of Design, Architecture and Building
University of Technology, Sydney
samuel.ferguson@uts.edu.au

Densil Cabrera
Faculty of Architecture, Design and Planning
The University of Sydney
d.cabrera@arch.usyd.edu.au

ABSTRACT

Methods for spectral analysis of audio signals and their graphical display are widespread. However, assessing music and audio in the visual domain involves a number of challenges in the translation between auditory images into mental or symbolically represented concepts. This paper presents a spectral analysis method that exists entirely in the auditory domain, and results in an auditory presentation of a spectrum. It aims to strip a segment of audio signal of its temporal content, resulting in a quasi-stationary signal that possesses a similar spectrum to the original signal. The method is extended and applied for the purpose of music summarisation.

1. INTRODUCTION

Graphical display is the predominant approach to conveying musical sound analysis information to people, including via spectrograms, spectra, waveform graphics and musical manuscript. While the visual system is dominant in many information transfer contexts, sonification (the representation of information through non-speech sound) offers many (often complementary) possibilities for information transfer [1]. As audio and music are data that are experienced primarily in the auditory domain, sonification would appear to be an appropriate method for analysis and representation of audio data, as it sidesteps the translation process from the auditory domain to the visual domain that is inherent in using visual representations.

A variety of simple techniques for sonification of sound in the context of audio education have been proposed by Cabrera and Ferguson [2, 3], and Ferguson has developed techniques for techniques for statistical sonifications of audio in his Ph.D thesis [4]. These sonification methods provide auditory analogues to common statistical visual displays (such as cumulative distribution functions and box plots), but with much richer information than visual charts. One of the solutions proposed in the thesis is a method of displaying spectral data, which is the focus of this paper.

Spectral analysis is one of the most fundamental, powerful, and widely used methods for the investigation of audio. This paper discusses an approach to spectral display that does not use Fourier analysis, and exists completely within the auditory domain. Instead of a Fourier or related transform implemented through signal processing, the method uses the spectral analysis of the human auditory system. While almost all listening could be thought of as involving auditory spectral processing, in listening that is focused on spectral features, temporal features are distractions that should be removed. Such features include rhythm, prosody, language, and more generally, the time structure of the sound being analysed. Put simply, the technique blurs temporally fluctuating audio signals to create quasi-stationary signals with almost identical spectra envelopes to the original signals, but without any semblance to the original time-dependent fluctuation. This technique is rooted in the theories of Gabor [5, 6] and granular synthesis [7], and has been strongly influenced by the recent advances in concatenative synthesis by Schwarz [8, 9].

Information visualisation literature has focused on methods for presenting data in ways that present large overviews of data, but allow a user to 'zoom and filter' the representation to find information that is important [10]. Fry's *Computational Information Design* outlines a method for developing interactive information representation systems [11]. A sonification method that would improve on visual methods may; use the original audio as the sound material for the analytic representation; filter the content of that audio in some way; maintain context and meaning of the audio; and draw relationships and present pertinent contrasts.

Schenkerian analysis of musical works is well-known and features in many undergraduate music curricula [12]. This graphical analysis method based on musical manuscripts allowed Schenker to reveal the various layers of a composition. The spectral sonification method described in this paper has the potential to be used in a similar manner allowing a scaling of perspective from large to small scale structures depending on the periods analysed.

2. SPECTRAL SUMMARISATION ALGORITHM

The core technique this paper presents is a method for spectrally summarising a larger audio signal. A representation that can convey the spectrum using audio without frequency domain signal processing can be built using a simple digital algorithm based on fragmenting the time signal into potentially overlapping windows, and recombining them in a way that (i) maintains a roughly constant power spectral envelope that matches the signal's long term spectrum; and (ii) removes distracting (non-spectral) features. This can be achieved by concatenating short windows (or grains) of audio by averaging a large proportion, but not all, of the original signal windows. A number of unique but spectrally similar windows need to be concatenated together, since if a single audio grain is repeated the resulting sound will be dominated by amplitude modulation related to the repetition rate. A systematic explanation of a process to create and concatenate unique but spectrally similar grains is as follows:

1. For a user-selected window length w_n samples, randomly select a window length w_{nr} from the range of values between $w_n - \frac{w_n}{2}$ and $w_n + \frac{w_n}{2}$.
2. Randomly select windows of length w_{nr} from the signal to be averaged to create a set $\{w_1, w_2, ...w_m\}$ of m unique windows.
3. Sum this set of windows, and divide by the square root of the number of windows ($\frac{\{w_1+w_2,...+w_m\}}{\sqrt{m}}$) to produce a single frame of w_{nr} samples duration.
4. Repeat steps 1, 2 and 3 (re-randomising each time) until enough unique but spectrally similar audio frames are produced to build a stationary sound of a chosen duration.
5. Concatenate these audio frames, using overlapping and adding with a custom window function. Ramps taken from either side of a Hanning window function are applied only to the overlapping proportion (typically only 5-10% of w_n) to maintain a constant sum between concatenation boundaries (see Figure 3).

This method is simple, but it is successful at creating a quasi-stationary sound with a spectral profile that matches the original file, while keeping the time variance to a minimum.

2.1 Validation, Tradeoffs and Limitations

To validate the appropriateness of the averaging process we undertook a comparison of spectra created by this spectral averaging method against the spectrum of the unmodified sample. A distinction worth mentioning is that through mixing we are amplitude (pressure) averaging, rather than power (pressure squared) averaging. Summing a large number of randomly selected signals as described above may be considered to be an operation on incoherent signals, which is why the square root of the number of windows is used in the denominator of the algorithms third step. Hence, the power spectrum of the sonification approximates the power spectrum of the original wave. While there is some potential for a substantial discrepancy between the power of the resulting spectrum and a true long term power spectrum, tests have shown that discrepancies are not severe for realistic signals if the averaging method uses a window size larger than 1024 samples.

There is a significant smearing of energy when using shorter window lengths. We demonstrated this by comparing a spectrally averaged sine tone at 2kHz, using a range of window lengths, to the original signal. Figure 1 demonstrates this effect. Generally, window lengths of 1024 samples or greater decrease smearing and increase spectral representation quality significantly. Subjectively, short window lengths tend towards extremely noise-like signals bearing little resemblance to the tonal spectra expected.

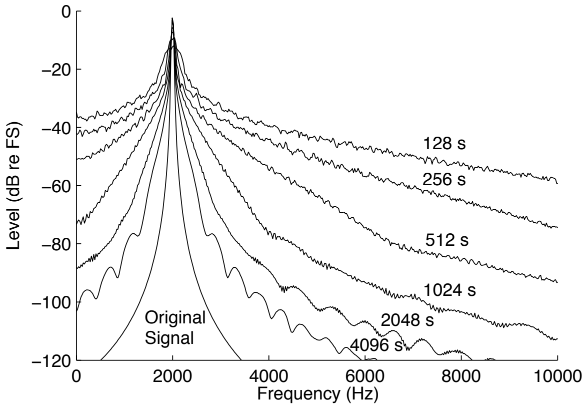

Figure 1. A 2kHz sine wave is spectrally averaged using a variety of window sizes, and compared with the original sine wave signal. The larger window sizes result in less spectral smearing.

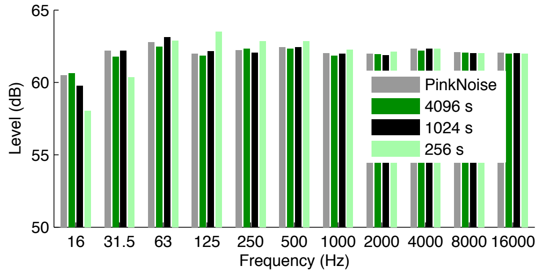

Figure 2. A pink noise recording is compared against spectral representations of pink noise using various window lengths - longer window length result in less spectral deviation.

The window length used in the spectral summarisation algorithm has a small effect on the low frequency range of the spectrum reproduced. To investigate this we compared a spectrum of a sample of pink noise (with a 48000 Hz sample rate) against three spectral summarisations, one using a 4096 point window, one with a 1024 point window, and one with a 256 point window. The length of the window determines the frequency below which the spectral representation begins to attenuate – at 4096 points there is little effect, but at 256 points it starts to become more significant and further reductions in window length result in the low cutoff frequency increasing proportionally. The cutoff frequency (fc) is apparently based on the largest wave period that can be represented by a specific window length (w_n), summarised by the relationship:

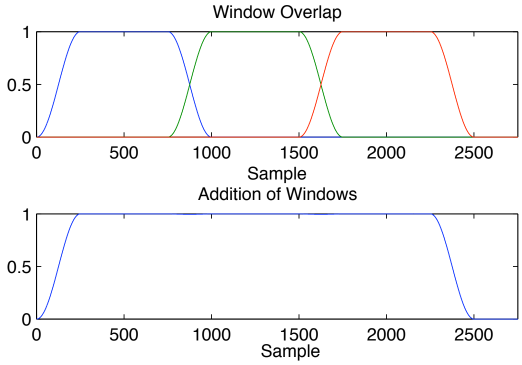

Figure 3. A custom window is designed to maintain a constant power sum between concatenated windows of audio.

$$f_c = \frac{f_s}{w_n} \tag{1}$$

This investigation tends to support the use of window sizes of at least 1024 samples and upwards for this spectral averaging technique. Larger window sizes will tend to allow more temporal fluctuation, depending on the temporal fluctuation present in the original signal, so there is a tradeoff present, however window sizes smaller than 1024 samples seem likely to significantly alter the spectrum to a degree where it is unrecognisable and non-representative. These parameters are likely to be experienced interactively, and therefore there is probably a subjective element to a user's selection for the most appropriate window size.

The windowing and overlapping at the concatenation stage must not introduce either discontinuities (clicks) or amplitude modulation, and thus we have designed a custom window shape that incorporates a large plateau (similar to a Tukey window) as well as a *Hanning* window function's ramps at either end. The proportion of the window devoted to the ramp is determined by the number of overlapping samples, and the resulting window shape maintains a constant sum at the window boundaries (see Figure 3). Furthermore, the randomisation around a central window length, ameliorates amplitude modulation effects that may arise out of periodic selection window length.

3. HARMONY ANALYSIS APPLICATIONS

Harmony analysis typically requires a familiarity with reading musical manuscripts, a difficult skill that is analogous to learning a new language. This, of course, places an immediate barrier to those users without these skills, but it also presumes a level of expertise in cross-modal perception in those users who possess skills in this language. A user who is presented with a harmony analysis on a manuscript is expected to ascertain the auditory meaning of the symbols and their consequences within the musical structure. This is not necessarily straightforward, and many users will 'interact' with the manuscript by using a piano to play back the pitches and compare their significance, while other users develop skills in producing auditory images of the various pitches. Methods that bridge the gap between symbolic representations of pitch relationships and auditory pitches are possible alternative solutions for these issues. Generally, the idea of this exploration is to make the patterns within music clearer than they are in a typical musical recording, so that users may understand harmonic patterns at multiple structural levels, and in intuitive manners.

The problem of producing a sonification of the harmony within the audio recording is therefore one of filtering the audio recording to contain less information, with an emphasis on that information which would be included in a harmonic representation. Such information may include musical elements like the fundamental frequency of the bass notes, and other notes presented either loudly or for comparatively long periods, while avoiding short decorative notes, or quick scalar passages. It would seek to remove, generally speaking, the temporal presentation of the notes, as well as their amplitude envelopes, resulting in a stationary sound with each important pitch presented simultaneously to build a chordal sound, accentuating the harmonic contribution each note makes, and diminishing each note's individual quality.

A structural representation would also need to describe how each section of the music relates to each other. The form of the piece is a crucial element in musicology, but it can be difficult to understand music at the formal scale from reading a musical manuscript, or from listening to an audio recording. Snyder [13] describes three levels of musical memory: the early processing level – which deals with characteristics of single notes, the short-term memory level – which holds musical phrases and rhythmic pattern, and the long-term memory level – which deals with formal sections, movements or entire pieces. Snyder also describes how long-term or formal memory deals with sections of music that are too long to be understood in the present, and their order needs to be consciously reconstructed as they do not automatically retain their time-order.

Simplifications or shorter versions of the musical sample can be used to describe the form of the piece in an amount of time that can be held within short-term memory. By presenting an auditory representation that is shorter than the original audio recording, but is proportional to the original, form can be more appropriately presented. Elements like key changes and voice ledaing become more obvious, as the ear can compare the short term memory of the old key to the new.

3.1 Voice Leading and Chord Patterns

The algorithm for building an averaged chord progression is as follows:

1. Get time information to use for time value boundaries – this may be based on extracted symbolic musical information, rhythmic information, timeseries descriptor peaks or various other time markup methods.
2. Find the first time boundary and the second time boundary to be averaged across and find the audio

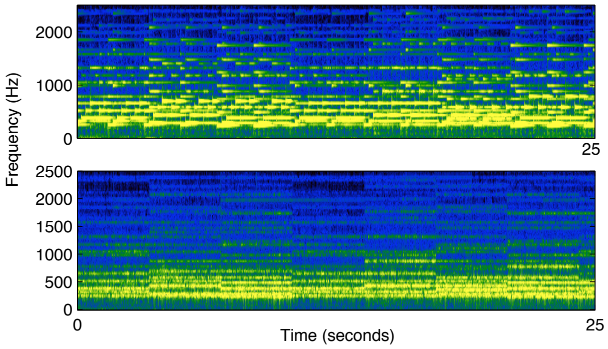

Figure 4. Comparing the sonification (top) and the original audio (bottom) we can see that the sonification attempts to blur the spectral components from each bar into a stationary sound. This sound is changed at every bar-line, approximately once every 3 seconds. The graphic only demonstrates the first 25 seconds of the piece.

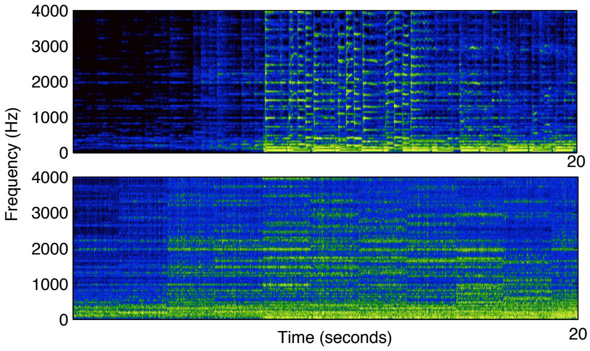

Figure 5. A second chord sonification example that demonstrates the structure of a Beatles song, *Norwegian Wood*. The upper graph is a spectrogram of the original audio, and the lower is the spectrally averaged sonification. The graphic shows only the first 20 seconds.

data in between the two times.

3. Apply the averaging method to this time interval to create the same duration of averaged audio, or perhaps a duration altered by a constant factor.
4. Place the resulting audio data at the corresponding sample numbers of the output audio, using appropriate overlapping.
5. Repeat the process after stepping forward to the second and third boundary, and continuing to step forward until the entire recording has been averaged and the output sonification built.

A simple method for finding time boundaries with which to segment the chordal structure is to extract the beats and assume that chord changes will be synchronised with beats, or more likely with bars. Depending on the meter of the piece (3/4, 4/4 or 6/8 commonly) we use particular beats as time boundaries, and in the following examples we have manually set the meter based on listening to the music, but advanced beat tracking algorithms may correctly estimate it as well. We also need to set an *anacrusis* value, that describes whether the piece begins on the first beat of the bar. Beat tracking is well-researched, and we use Dixon's *Beatroot* algorithm [14].

3.2 Harmonic Pattern Examples

We will attempt to use this algorithm to represent the long-term structure of some pieces of music.

One piece that is defined primarily by its chordal content (as opposed to its melodic or rhythmic) is Bach's Prelude No 1. from 'the Well-tempered Clavier'. A Schenkerian analysis has also been published for this piece [12]. By applying the algorithm to the audio we produce a sonification that is presented in Figure 4. The sonification created is not a completely stationary sound, like a set of tones, nor is it a sound that has discernible starting or ending notes. It demonstrates characteristics of the timbre of the instrument, but primarily it presents the notes that have sounded. The quality of the sound is similar to the sound that would be produced if the pianist stopped at the end of the bar and held down all the keys played in the bar.

The averaging across the bar is particularly appropriate for this particular piece, due to the manner in which Bach presents a single chord per bar. For other situations this may result in chords blurring into other chords, resulting in strong dissonance. Despite this, there are a large majority of pieces where this simple scheme would be sufficient. The remainder may be dealt with using more sophisticated methods, that employ harmonic and rhythm based pre-processing to carefully avoid averaging across chord changes incorrectly.

One purpose of the blurring of the audio is to be able to place one bar's harmonic content temporally adjacent to the next's. This should allow each harmonic change to be understood in terms of the notes within each chord, and to which notes they each move. An example of a pattern that might be uncovered through this process is the bassline in this prelude. While these notes are strongly sounded at the beginning of the bar, they decay by the end of the bar, and other higher notes are dominant by this stage. The blurring applied places each of these sounds adjacent to each other, yielding a *legato* bassline.

The other useful process possible by using the blurring of the audio is that the speed of the example can be arbitrarily altered. The blurred audio has no temporal content, so a bar's worth of sound may be presented over 3 seconds or in half a second. By setting an arbitrary compression factor for the duration, we can proportionally change the duration of the piece while maintaining the formal structure. This can be used to alter a 3 minute piece, whose structure can be 'remembered', into a 20 second piece, whose structure can be 'heard'.

In the structural sonification of *Norwegian Wood* (Figure 5), we hear a clear structure of descending melodic notes that define the chord structure. The structure is a lot simpler than that of the Prelude, and each formal section can be clearly heard.

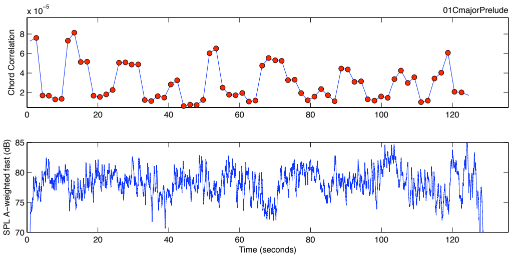

Figure 6. There is a 15dB SPL range over the time period of the musical example. High levels can be heard to correspond to the chords which have the least correlation and are the most dissonant.

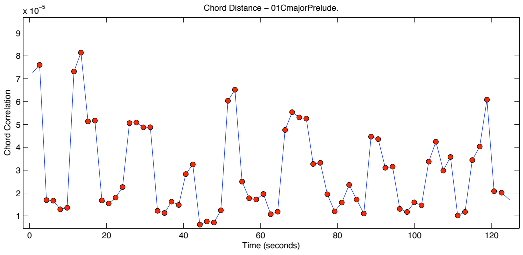

Figure 7. The correlation between various chords seems to follow a predictable pattern. The gain function sonification makes that pattern more apparent by exaggerating it, and highlighting low correlation values. Low values of correlation are analogous to high values of chord distance.

3.3 Chordal Patterns and Context

While we wish to maintain the temporal order of the various chords, due to their importance to the overall direction and purpose of a piece of music, it may be interesting to annotate the sonification in terms of the values of other parameters. A simple parameter to investigate is the sound pressure level (SPL). From listening to the sonification we can hear that often tonic chords are quieter compared with the chords that lead into them. A comparison of SPL against chord estimates shows the decrease in SPL that accompanies every return to the tonic (see Figure 6). We may wish to accentuate this further. By normalising the level of audio from of each section, and then mapping the SPL extracted from the audio to an expanded gain function that is then re-applied to each section, we can experience the structural implications of the performer's use of dynamics more clearly.

While this is a straightforward example, the use of sound pressure level as the mapping target is arbitrary, and many other such targets exist. Another candidate parameter to base a gain function sonification on is the harmonic distance the chord is from the tonic. One can attempt to approach this from a chord recognition perspective, but in this case we will use the chroma pattern only and will compare it against the first (tonic) chord.

The virtual pitch algorithm of Terhardt takes a template matching approach to finding pitches in the sound [15]. It applies a peak picking algorithm to find the points in the spectrum that are peaks. These are then applied to a successive template matching algorithm that attempts to place the peaks under a pitch template. The pattern of pitches across the audible range can be constrained to create a chroma pattern, representing the strength of each of the 12 pitches within an octave, regardless of pitch-height.

Using this method we calculate an average chroma pattern for each bar, and then multiply and sum those chroma pattern vectors to create a number representing the correlation between each bar and the first chord. A high number represents a high number of similar notes, or low chord distance, while a low number represents a large amount of difference. These values are clearly a useful target for mapping to a gain function. With larger chord distances associated to greater SPL, and smaller chord distances associated to low SPL, the effect should be similar to typical musical approaches.

An alternative approach to expanding parameters such as sound pressure level is to use a parameter as the basis of temporallycutting and reorganising the harmonic units or bars. Any measurable parameter that can be derived from a steady-state spectrum could be used for this purpose. The time periods (in this case bars) that are used to average from in the algorithm described in section 3.1 are then associated with a median parameter value taken for their time period. The median parameter value determines the reordering, and then the audio units are rearranged based on the new order. If the chord correlation values mentioned above are used to order the bars in ascending order, then the sonification progresses from chords that are generally dissimilar to the tonic, to chords that are more consonant – giving an overall impression of a long cadence. During this process particular chords can be assessed within the overall scheme of chords.

3.4 Beyond Chords

Analysis of sections of music larger than individual chords or bars is easily implemented using this method. The long term spectra of entire pieces can be sonified into a short sound, so that, for example, the average spectra of each of Bachs Preludes and Fugues (one in each major and minor key) can be quickly compared by ear. The overall spectrum of the entire collection of preludes could then, for example, be sonified so as to be compared by ear to those of other similar collections of preludes or etudes played on piano (such as those of Chopin or Listz).

4. CONCLUSIONS

We have presented an algorithm for spectral summarisation, and have applied it to the problem of music summarisation. This method is based on concatenative and granular synthesis and aims to strip musical audio of its fine temporal content while maintaining the spectral shape and energy. We have described methods for applying this basic spectral summarisation technique to the analysis of harmony and voice leading, and for using it to compare chords against the tonic chord of a musical piece.

4.1 Applications, Limitations and Future Work

These techniques are useful for assessing musical samples in a semi-automated manner - in a way that hopefully falls somewhere between listening to the entire audio file, and an abstract information retrieval algorithm. In this way it may be possible to apply this method in the design or checking of music information retrieval algorithms. This representation method may also be useful in education contexts, for the assessment of spectra, and to introduce ideas of structure and tonality. Its application to auditory browsing, for instance of digital archives of musical recordings, is also worth consideration.

Short signals highlight a limitation of this method – a certain amount of audio data is required to reliably build an average window from. For short signals the use of large windows is also difficult, leading to the tradeoff between spectral smearing and window length described in 2.1. This method is likely to be experienced in an interactive context, due to its reliance on computer technology. Investigation into good ways to provide interactive user access to this algorithm is likely to greatly improve its usefulness. Lastly, the suppression of the audio's temporal information throws away a lot of temporal qualities that are fundamental in musical practice. Modifications of this method that seek to systematically explore aspects of music apart from only the spectral and harmonic qualities are worth careful consideration.

It is easy to forget how powerful auditory analysis can be when visual and textual presentation of data are overwhelmingly common. Sonification of audio is more than a tautology, and extends beyond the trivial case of merely playing the original audio recording. This paper examines one simple technique for the sonification of sound recordings which focuses on spectral features. One of the attractive features of this technique is that it does not employ any spectral analysis using digital signal processing instead the spectrum analysis is achieved in the ear, and the purpose of the technique is to prepare the audio so as to provide a sound that focuses attention on spectral features. In other work we have examined other spectrum sonification techniques that do use Fourier transforms, such as exaggerating spectral features through auto-convolution (raising the spectrum to an integer power) [2].

Applications of this technique extend beyond conventional harmony-based music, and beyond music. Broadly speaking, it is applicable to audio recordings that have medium or long term spectral features of interest (including harmony, timbre) that might be difficult to clearly discern without the removal of temporal structure and/or the compression of duration.

4.2 Acknowledgements

This research was supported by an Australian Postgraduate Award and a Departmental Supplementary Scholarship.

5. REFERENCES

[1] G. Kramer, B. N. Walker, Terri Bonebright, P. R. Cook, J. H. Flowers, Nadine Miner, and John G. Neuhoff. Sonification report: Status of the field and research agenda. Technical report, NSF, 1997.

[2] D. Cabrera, & S. Ferguson. Auditory Display of Audio. In *120th Audio Engineering Society Convention*, Paris, France, 2006.

[3] D. Cabrera & S. Ferguson. Sonification of sound: Tools for teaching acoustics and audio, In *13th International Conference on Auditory Display*, Montreal, Canada, 2007.

[4] S. Ferguson. *Exploratory Sound Analysis: Statistical Sonifications for the Investigation of Sound*. Ph.D Thesis, University of Sydney, 2009.

[5] D. Gabor. Theory of communication. *J. Inst. Elec. Eng.*, 93:429–457, 1946.

[6] D. Gabor. Acoustical quanta and the theory of hearing. *Nature*, 159:591–594, 1947.

[7] C. Roads. *Microsound*. MIT Press, Cambridge, 2001.

[8] D. Schwarz. *Data-driven Concatenative Sound Synthesis*. Ph.D Thesis, University of Paris 6 Pierre et Marie Curie, 2004.

[9] D. Schwarz, R. Cahen, and S. Britton. Principles and applications of interactive corpus-based concatenative synthesis. In *Journées d'Informatique Musicale (JIM'08)*, Albi, 2008.

[10] B. Shneiderman. The eyes have it: a task by data type taxonomy for information visualizations. In *IEEE Symposium on Visual Languages*, Boulder, CO, USA, 1996.

[11] B. Fry. *Computational Information Design*. Ph.D Thesis, MIT, Cambridge, MA, 2004.

[12] H. Schenker and F. Salzer. *Five graphic music analyses (Funf Urlinie-Tafeln)*. Dover Publications, New York, 1969.

[13] B. Snyder. *Music and Memory: An Introduction*. MIT Press, Cambridge, MA, 2001.

[14] S. Dixon. Automatic extraction of tempo and beat from expressive performances. *J. New Music Res.*, 30(1), 2001.

[15] E. Terhardt, G. Stoll, and M. Seewan. Algorithm for extraction of pitch and pitch salience from complex tonal signals. *J. Acoust. Soc. Am.*, 71(3):679–688, 1982.