

# PLAY IT AGAIN, BABBAGE! – A FRAMEWORK TO EXPLOIT MUSICAL REPETITION FOR HIGH-QUALITY AUDIO COMPRESSION

Stuart Cunningham, Vic Grout & John McGinn  
*Centre for Applied Internet Research (CAIR), University of Wales  
NEWI, Plas Coch Campus, Mold Road, Wrexham, LL11 2AW, North Wales, UK  
{s.cunningham | v.grout | j.mcginn}@newi.ac.uk*

## ABSTRACT

The field of audio compression has been of huge significance over the last decade, especially with the proliferation of the popular MP3 file format for storing music, and distributing audio over data networks, in particular, the Internet.

This paper proposes new methods of compression based on the frequently fundamental compositional element of repetition within music. By exploiting the musical content of a piece of audio, data can be discarded that is perceptually redundant, without the removal of frequency components and variable rate encoding found in other compression techniques, such as MP3.

Frameworks for a new method of compression are presented and several techniques considered, as possible solutions to the problem of implementing an effective process of data reduction.

## KEYWORDS

Audio, compression, waveform analysis, perception.

## 1. INTRODUCTION

In recent years the use of compression methods to distribute and store digital audio, and in particular music, has increased significantly. These techniques for compression have managed to achieve good to high levels of audio quality whilst maintaining a reasonable trade-off against the amount of data generated, which has been one of the main factors contributing to the success of these methods (Coleman, 2005).

The ideal scenario when storing and transmitting musical audio data would be to use an uncompressed digital audio format, such as WAVE, thereby preserving the quality, bit-depth, and sample rate of the original digitised sound. The generally desired properties are, but are not restricted to, a 44.1 kHz sampling rate and 16-bit depth, or ‘CD quality’ encoding. However, in using this ‘best’ representation, large amounts of data are generated, making compression desirable (Rumsey, 1996). This requirement has produced a multitude of techniques that can be used to reduce the data sizes of digital audio files, including formats such as MP3, WMA and RealAudio. These compression systems, and others, have been successful since they provide significant ratios of compression (in the region of 10:1 – 15:1) whilst still maintaining a perceptibly high quality of audio clarity and response. However, the main identifiable shortfall of these contemporary methods is that they all employ a system of lossy compression, based on the removal of information which is psycho-acoustically redundant.

Therefore, a data reduction technique which offered comparable ratios of compression whilst still maintaining the original bit-depth and sample rate of digital audio would be of interest to many. Such a system would not only allow high-quality audio to be stored in a reduced size, but would also be useful for applications in musicology, archiving, composition, and distribution, to name but a few.

This paper introduces the framework for a compression technique which exploits redundancy in musical structure to achieve data reduction, without affecting the parameters of bit-depth or sample rate.

## 2. REUSABLE AUDIO SAMPLES

The structure and composition of most popular genres of music almost always involves some form of repetition. Viewed at the macroscopic level this could be the observable structure, such as AABA, for example. However, at a more microscopic level it could be viewed as the recurrence of the same chords, musical notes, or phrases, which will often be highly likely to occur more than once in a given piece of music. This repetition can be taken advantage of for the purposes of compression. If two or more identical, or even perceptively similar, sequences of music occur in a song, then it should not be necessary to encode all occurrences, when one could suffice. It is upon this basic principle that the methods described in this paper are based. This exploitation of repeated data is a technique that is successfully used already in the image compression domain, and algorithms used in JPEG and MPEG-2 make use of such run length encoding. How similar two music extracts should be is a matter to be determined through investigation, though this could be a user-defined compression parameter. Figures 1 and 2 present two different waveform extracts from a piece of music “*The Song of the Clyde*” (Gourlay et al, 1995). Each waveform is a piano sequence that occurs several times in the song.

Figure 1. Piano Sequence 1 (occurring at 0 seconds)

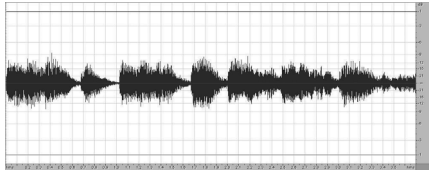
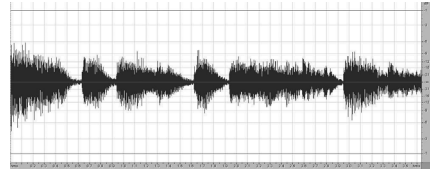


Figure 2. Piano Sequence 1 (occurring at 52 seconds)



Although Figure 1 and 2 are not identical, they clearly demonstrate a similarity in their structure and pattern. When listening to these individual extracts, a listener would detect a high-degree of similarity between the two sounds, although there is, of course, some noticeable difference. This system will exploit human hearing deficiencies and psychoacoustics to identify sequences of music that are perceived to be identical or similar. A previous pilot study into the capability of hearing differentiation of reusable audio samples has been carried out, and showed promising results in this area (Cunningham, 2005).

Through analysis of an audio file (potential techniques are discussed later), it should be possible to detect sections of audio which are repetitive throughout the file. Therefore, if the file is searched for matching blocks of audio data, a list can be compiled of matching sequences, a measure of the difference between matches, and the point(s) at which each sequence occurs. Once this has been achieved it would be necessary only to encode one of each matching block, plus any remainder of the file, along with the information required to create a representation of the original audio from the blocks retained. Finally, it would be desirable to package all these components into a single file.

Therefore, although this system could not be termed lossless, the data blocks which preserved are not subjected to any loss. However, this is not to rule out a method which then encoded remaining blocks into MP3 format, for example. Such a combination could provide previously unattained ratios of data reduction under favourable circumstances.

## 3. RELATED & PREVIOUS WORK

Many methods have been examined to attempt to extract patterns and sequences from music, although these have generally used music notation as the basis for pattern analysis and extraction could well have a distinct advantage in terms of the processing time required to produce results.

Dannenberg & Hu have showed positive results in detecting similar sequences of melody in monophonic music and have also experimented in the polyphonic domain, primarily using pitch detection and chroma quantisation (Dannenberg<sub>a</sub> et al, 2002, Dannenberg<sub>b</sub> et al 2002). Another technique considered by Mazzoni and Dannenberg was using a pitch contour matching technique against a database of modified MIDI songs which had assigned pitch contours, alleviating problems introduced by quantisation (Mazzoni et al, 2001). However promising, these methods rely on pitch detection, normally based either on frequency analysis, or by using musical notation and MIDI. In the system proposed here, it is necessary to account for not only the pitch, but the entire polyphonic spread of sounds, after all, though the proposed system is

intended to be used in music compression, there is no reason why it should be in any way restricted in its use in the audio domain.

Chai (Chai, 2003) and Chai and Vercoe (Chai et al, 2003) present work more considerate towards polyphonic, complex audio. Their research demonstrates excellent and highly-positive results in the detection of musical structure within musical pieces. The defining differences between the system proposed here and that of Chai is that Chai's system tends to look for high level music structure (e.g. ABABA) whereas our system will detect structure at as small a level as the user wishes to define. The other main difference is that Chai uses single-channel audio, low sample rates and low bit-depth. Our system aspires to operate at any sample rate, bit-depth, and across multiple channels, provided the search algorithms are scaled correctly.

Peeters, La Burthe, and Rodet have also considered the issue of detecting patterns and structure in polyphonic music using signal analysis, for the purposes of music summarization and employ a similarity matrix approach which can be visually depicted (Peeters et al, 2002). Their work also presents the results of such analysis, and the visual indicators for several very successful tests across a varied genre of musical compositions. Particularly, their system is focused on genres of music where repetition plays a significant role in the piece.

It is interesting to bear in mind a set of compression requirements Hacker identified (Hacker, 2000). These are the requirements, which any format hoping to supersede MP3, would have to meet:

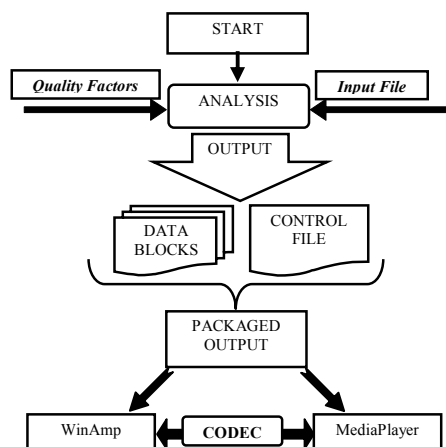
- Smaller file sizes
- Superior audio quality
- Free and unprotected format

Though these are obvious requirements, they serve as a guideline and reminder of the goals that should be achieved.

The most significant issue to note when examining these previous works is that compression is rarely mentioned. The concept of exploiting musical structure and repetition for compression is occasionally hinted at, but with no other detail other than possible application areas. Similarly, the notion of using pattern comparison between musical pieces for copyright and legal applications is only ever suggested.

Initial requirements and draft framework for this compression system was identified and proposed in detail in a previous work, by the authors of this paper (Cunningham, 2005). This particular system is summarised in Figure 3.

Figure 3. Overview of Compression System



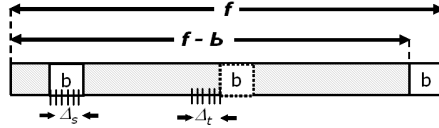
In this previous investigation, data savings of between 13.7% and 32.3% have been made in initial, and very rough tests on polyphonic waveform music, whilst still retaining the original sampling rate and bit-depth of the original audio. Given the meticulous analysis that a computer program can provide, it would not be unreasonable to expect the amount of data saved to increase, especially for genres of music where repetition occurs frequently.

#### 4. FRAMEWORK FOR WAVE ANALYSIS

Given the length of a file  $f$ , the file would then be iteratively searched using a search block  $b$ , which would increase in size through the lifetime of the search process. Therefore, a minimum and maximum size of search block is established before searching,  $b_{min}$  and  $b_{max}$ , and an increment value between these limits of block sizes is also set,  $\Delta_b$ . We Assume  $\Delta_b$  divides  $b_{min}$  and  $b_{max}$ .

A search increment across the file, the interval between one search position and the next, is also defined  $\Delta_s$ , and effectively will allow a fineness or depth facility to the function. Similarly, a final incremental parameter,  $\Delta_t$ , provides an offset for the target block where the match for  $b$  is sought. This is illustrated in Figure 4.

Figure 4. Assignment of Search Components



If the complexity or number of steps required to match one block of size  $b$  can be generically defined as function  $X(b)$ , then for a given set of values of file size, maximum and minimum block size and block size, search block start position and target block start position increments, the complexity of an exhaustive search routine considering all possible matches of  $b$  ( $b_{min} < b < b_{max}$ ) is given by:

$$C(f, b_{min}, b_{max}, \Delta_b, \Delta_s, \Delta_t) = \sum_{b=\frac{b_{min}}{\Delta_b}}^{\frac{b_{max}}{\Delta_b}} \sum_{s=0}^{\frac{f-\Delta_b b}{\Delta_s}} \sum_{t=0}^{\frac{f-\Delta_b b}{\Delta_t}} X(\Delta_b b) \quad (1)$$

Since the length of the file is effectively static at the start of the compression process, the five parameters ( $b_{min}$ ,  $b_{max}$ ,  $\Delta_b$ ,  $\Delta_s$ , and  $\Delta_t$ ) are all in effect measures of quality' or effectiveness of the search process, which could be defined at the beginning of any search. However, the end user of such a system is unlikely to want to specify five different values each time they save a file, although an 'Advanced User' set of options might allow this. It would be much better to give an overall, single 'Quality' rating factor to the end user, where they could specify the percentage of quality value of the final compressed file. Internally however, it is expected that the setting of this value would scale the previously defined quality parameters by the required factors to facilitate an overall quality factor.

To provide simplification, these parameters are consolidated into one value:  $\Delta = \Delta_b = \Delta_s = \Delta_t$ . To provide additional simplification the assumption is made that  $b_{min} = \Delta b = \Delta$  and  $b_{max} = f/2$ , since in most cases it would be impractical to search for a block which is greater than or equal to, half the length of the file. Given these values, the expression in (1) then simplifies to:

$$C(f, \Delta) = \sum_{b=1}^{\frac{f}{2\Delta}} \sum_{s=0}^{\frac{f-\Delta b}{\Delta}} \sum_{t=0}^{\frac{f-\Delta b}{\Delta}} X(\Delta b) \quad (2)$$

Finally, if we can approximate  $X(\Delta b)$  by an invariant term  $X$ , which is not dependent on  $b$ , and take  $f \gg b$ , so that all  $(f-\Delta b)/\Delta$  terms reduce to  $f$  then this reduces the expression to:

$$C(f, \Delta) \approx \frac{f^3 X}{2\Delta^3} \quad (3)$$

From these expressions it is clear that performing such a search would be highly dependant on the cost or complexity of the search itself  $X$ . The time required for the search process will depend on the particular pattern discovery techniques being employed. Three proposed techniques to be tested are presented later in this paper, in Section 6.

## 5. OPTIMISATION OF FRAMEWORK

Figure 5 and Figure 6 plot the number of steps required against the increasing incremental values  $\Delta_s$  and  $\Delta_t$ , respectively using the expression given in (1). This is done with the aim of demonstrating that beyond certain limits there is an insignificant, even nonexistent, need to search beyond certain upper values.

Figure 5. Complexity with Variance  $\Delta_s$

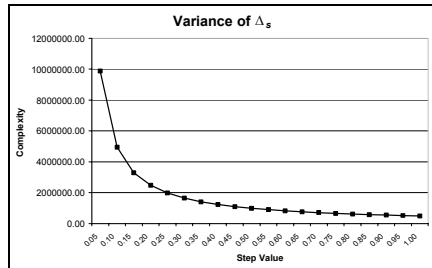
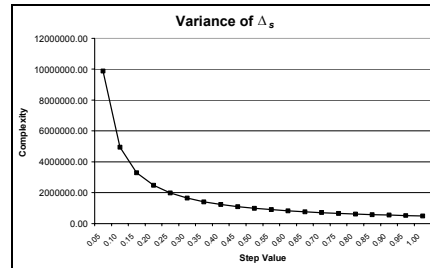


Figure 6. Complexity with Variance  $\Delta_t$



From the information presented in Figures 5 and 6, curves can be clearly detected in the graphs, indicating there would be an upper value level that search parameters could be restricted to. These results have potential ramifications into the refinement and optimisation of search algorithms at implementation.

It is also reasonable to expect, from these results, that a single, consolidated incremental parameter, identified as  $\Delta$  in (2) and (3) would have a similar, if perhaps scaled, reduction curve, therefore providing comparable optimisation to the upper bound of any search.

Since the system will be primarily used for compression of music, search parameters could be refined within the limits of the musical characteristics of a given composition. For example, if the tempo of the piece is known this information could be used to inform configuration of search parameters.

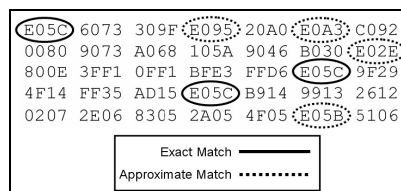
## 6. MATCHING SEQUENCES OF AUDIO

It is expected that any method applied to provide the waveform matching feature (defined as  $X$  earlier) would use the search procedures defined in (1), (2), and (3), earlier in the paper. The method which will be ultimately used is not yet clear, and as such, several methods are proposed at this stage. It could even be considered that in the final system multiple methods of pattern detection could be used, thereby providing maximum efficiency and ratification of any blocks of data to be extracted or encoded.

### 6.1 Binary & Numerical Matching

A binary or similar numeric pattern search would concentrate on examining the numerical contents of a file. The raw data sections could be searched as described earlier, in an attempt to find a match within desired parameters, whether or not an exact or approximate match is required. Although looking for exact matches would be simple, work is required to determine what variations from an original block would be acceptable as matches within the confines of the binary data, as raw binary numbers do not visually translate into a sound that can be easily perceived. Figure 7 demonstrates a graphical example of how such a technique might work.

Figure 7. Numerical Pattern Detection & Matching



However, this approach could well have a distinct advantage in terms of the processing time required to produce results, as this would be a likely candidate as being the least computationally intensive search procedure.

A simple correlation or difference of squares measure could also be used to test for similarity in this situation. If we wish to compare two blocks of  $n$  pieces of data,  $x_1, x_2, \dots, x_n$  and  $y_1, y_2, \dots, y_n$ , then the similarity could be calculated as:

$$M_1 = \frac{\sum_{i=1}^n x_i y_i}{\sqrt{\sum_{i=1}^n x_i^2 \sum_{i=1}^n y_i^2}} \quad (4)$$

Or, for difference of squares method, more simply as:

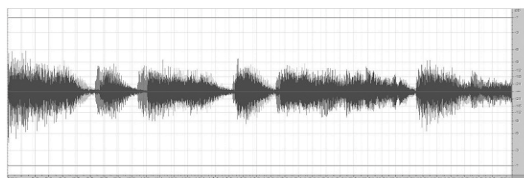
$$M_2 = \frac{1}{n} \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (5)$$

It should be noted at this point, however, that both these expressions involve the computationally intensive square root operation, which may increase the overall time to perform this kind of search operation.

## 6.2 Graphical & Image Matching

Graphical pattern matching is perhaps a more novel approach to solving the problem. Here chunks of waveform trace, spectral or harmonic graphs could be employed as the material for analysis, extracting sections from these visual representations and then attempting to overlay a section on any other part of the image. Referring back to Figures 1 and 2, it is clear to see a general similarity between the two waveforms. From observation, it would be reasonable to hypothesise that were two graphical waveforms compared using a graphical comparison technique that identical and/or similar matches would be detected. To further exemplify the possibility for such a technique, Figure 8 shows a merge between the images Figures 1 and 2, overlaid. In this image the darkest parts of the image are where the waves show similarity and overlap, where the lighter, greyer areas show only one waveform being present. Although this particular example is effectively conceptual, it shows a definite similarity and potential as a method for future experimentation.

Figure 8. Overlaid Merge of Two Waveforms



Like the binary approach, this method is almost purely visual and does not provide indication about how the waveform sounds, although the visual element does at least give an indication of the acoustic fundamentals of a sound. Though graphical matching may seem an effective method, care is required to ensure that the audio content is not in any way lost during a search and extraction process. Consideration should also be given to the fact that image matching can be computationally heavy and deliberation must be given to the properties associated with images, such as resolution, dimensions, and number of colours. These factors will impinge on the overall effectiveness and computational load of a graphical matching system.

## 6.3 Fourier Analysis for Matching

Audio matching is the most logical, effective, and comprehensive method of searching a wave file for matching blocks, especially since this technique should fully exploit the fact that we are dealing with digital

audio signals. Such a method could be implemented by analyzing the amplitude and frequency components of a signal and carrying out comparisons of this, in the remainder of a file. This approach, and modified versions of this, has been used extensively in similar previous work with good levels of success (Mazzoni, 2001, Chai, 2003, Peeters et al, 2002). To demonstrate potential for this technique Figures 9 and 10 show Fourier graphs for the two piano sections of “*The Song of the Clyde*” (Gourlay et al, 1995) presented earlier.

Figure 9. Fourier Analysis of Piano Sequence 1

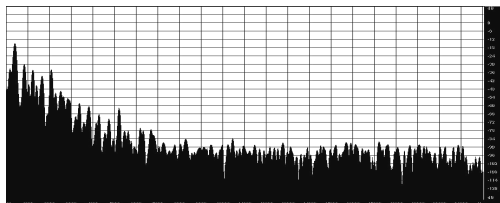
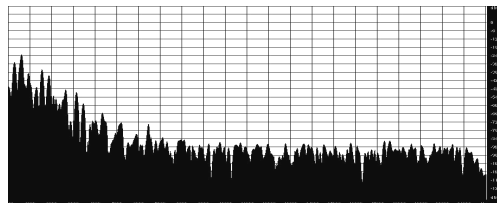


Figure 10. Fourier Analysis of Piano Sequence 2



Though these examples are not identical, there are definable trends and similarities. Obviously, the use of Fourier analysis and the Fast Fourier Transform (FFT) would play a large part in this type of approach. Consequently, these methods could be prone to suffering large computational loads due to the nature of the brute-force searches described in this paper. However, a possible solution to this problem is the recent advances in improving the speed of Fourier Transforms. In particular, the Very Fast Fourier Transform (VFFT) (Shepherd, 2004) displays great potential in helping to overcome any significant delays.

## 7. FUTURE WORK

Aside from the obvious work indicated in this paper in the field of evaluating the best method to use when searching for patterns within music, there are two additional issues that arise.

The first is to ensure that such a system is truly viable and that the listening experience will not be detrimentally affected by using the compression procedures proposed in this paper.

Secondly, the optimisation of the search procedure should be considered in order to make the system more viable and likely to succeed in the competitive market of audio compression technologies.

### 7.1 Human Perception Testing

Testing the parameters of human hearing and psychoacoustics, particularly slanted to this system of compression, will be an important factor in future research and development. After all, human listeners are expected to be the primary targets of this technique. The key questions invoked by this work and previous investigation (Cunningham, 2005) are as follows:

1. *What do we define as a match of audio chunks?*
2. *How do we measure the distance between chunks?*
3. *What is an acceptable difference between chunks?*

Through further investigation into hearing, a concise and effective system can be developed which will be truly suited for its purpose, and could be implemented widely with confidence. Such information could also assist in informing the development of search algorithms, to provide a solution of high performance.

### 7.2 Search Algorithms

Clearly a brute-force, exhaustive search will be effective and is useful for initial testing of theories and applications. However, upon further testing and final implementation it will be desired to refine this process.

It is expected that in future development the search algorithms could be improved greatly, and refined to provide higher performance. Initially, common performance techniques may be applied, such as associating a weighting or cost with each potential match of block and implementing the method of least cost. This leads to

fields of experimentation in the areas of dynamic programming, learning algorithms, heuristics and methods based on Hidden Markov Models (Goldberg, 1989, MacDonald et al, 1997).

As well as attempting to devise new and novel methods of improving the search procedure, the tremendous amount of work already available in the field of pattern detection and recognition, across a variety of media, will surely be able to inform and expand the functionality of this system.

## 8. CONCLUSIONS

The methods proposed in this paper are novel and have initially displayed positive results in enabling data reduction in high-quality audio music files. Again, however, it should be stressed that the application of the techniques here should not be limited purely to the field of musical audio.

Though compression is the primary goal of our research, a successful pattern recognition and detection system could have a variety of uses in the world of music and musicology. These methods could be used in legal situations to settle differences about how similar or possibly copied one piece of music was to another, such as in copyright disputes. Another function for this system in copyright might be to detect the use of music samples within audio compositions or collages. Blocks extracted from waveforms could be compiled into a central library, which could aid such applications as audio searching by humming or whistling, i.e. content-based searches. Such libraries could also be used for composition, providing a system of 'Object-Oriented Music Composition' for computer music students and musicians alike.

Undoubtedly, there is still a lot of work to be done, across multidisciplinary fields, of how to implement this system, as well as to evaluate any implementation, through intensive testing. Nonetheless, we are confident that a system to rival the quality of MP3 and its successors can be developed effectively.

## REFERENCES

- Chai, W., 2003. Structural Analysis of Musical Signals via Pattern Matching, *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Hong Kong, China.
- Chai, W., Vercoe, B., 2003. Music Thumbnailing via Structural Analysis, *Proceedings of the 11th ACM International Conference on Multimedia*, (pp. 223-226), Berkeley, California, USA.
- Coleman, M., 2005. *Playback: From the Victrola to MP3, 100 Years of Music, Machines, and Money*, Da Capo Press, Cambridge, MA, USA.
- Cunningham, S., 2005. Waveform Analysis for High-Quality Loop-Based Audio Distribution, *Proceedings of the 20th International Conference on Computers and Their Applications* (pp. 19-25), New Orleans, Louisiana, USA.
- Dannenberg, R. B., Hu, N., 2005. Pattern Discovery Techniques for Music Audio, *Proceedings of ISMIR 2002 Conference on Music Information Retrieval* (pp. 63-70), IRCAM, Paris, France.
- Dannenberg, R. B., Hu, N., 2002. Discovering Musical Structure in Audio Recordings, *Proceedings of Music and Artificial Intelligence: Second International Conference* (pp. 43-57), ICMAI, Edinburgh, Scotland, UK.
- Goldberg, D. E., 1989. *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison Wesley, Boston, USA.
- Gourlay, I., Bell, R.Y., *The Song of the Clyde*, Performed by Sam Cunningham & John Drake, 1995. Scotland by Request, Ess-Jay Music.
- Hacker, S., 2000. *MP3: The Definitive Guide*, O'Reilly, UK.
- MacDonald, I. L., Zucchini, W., 1997. *Hidden Markov and Other Models for Discrete-valued Time Series (Monographs on Statistics & Applied Probability)*, Chapman & Hall / CRC, FL, USA.
- Mazzoni, D., Dannenberg, R. B., 2001. Melody Matching Directly from Audio, *Proceedings of ISMIR 2001 Conference on Music Information Retrieval* (pp. 73-82), Indiana, USA.
- Peeters, G., La Burthe, A., Rodet, X., 2002. Toward Automatic Music Audio Summary Generation from Signal Analysis, *Proceedings of ISMIR 2002 Conference on Music Information Retrieval* (pp. 94-100), IRCAM, Paris, France.
- Rumsey, F., 1996. *The Audio Workstation Handbook*, Focal Press, Oxford, UK.
- Shepherd, S. J., 2004. The Very Fast Fourier Transform ~ A Tool for Science in the 21st Century, *Proceedings of 10th International Conference on Electrical and Electronic Engineering X ICEEE 2004* (pp. 6-11), Acapulco, Mexico.