Acoustic Analysis of Music Albums

Jensen, Karl Kristoffer

Published in:
Machine Learning Techniques for Adaptive Multimedia Retrieval

DOI (link to publication from Publisher):
10.4018/978-1-61692-859-9

Publication date:
2011

Document Version
Early version, also known as pre-print

Citation for published version (APA):
Jensen, K. K. (2011). Acoustic Analysis of Music Albums. In C.-H. Wei (Ed.), *Machine Learning Techniques for Adaptive Multimedia Retrieval: Technologies Applications and Perspectives* (pp. 322-340). IGI global. https://doi.org/10.4018/978-1-61692-859-9

Chapter 15
Acoustic Analysis of Music Albums

Kristoffer Jensen
Aalborg University Esbjerg, Denmark

ABSTRACT

Most music is generally published in a cluster of songs, called an album, although many, if not most people enjoy individual songs, commonly called singles. This study proposes to investigate whether or not there is a reason for assembling and enjoying full albums. Two different approaches are undertaken in order to investigate this, both based on audio features, calculated from the music, and related to the common music dimensions rhythm, timbre and chroma. In the first experiment, automatic segmentation is done on full music albums. If the segmentation is done on song boundaries, which is to be expected, as different fade-ins and –outs are employed, then songs are seen as the homogenous units, while if the boundaries are found within songs, then other homogenous units also exist. A second experiment on music sorting by similarity reveals findings on the sorting complexity of music albums. If the sorting complexity is high, then the albums are unordered; otherwise the album is ordered with regards to the features. A discussion of the results of the evaluation of the segment boundaries and sorting complexity reveals interesting findings.

INTRODUCTION

Music can be enjoyed on different time scales, going from the individual notes, to the riffs, as popularized in for instance ring tones, through choruses and full songs, which are popularized through the single format, and to albums, that many consider cannot be listened to other than at full length. This applies in particular to the concept albums. The investigations of albums will lead to the analysis of segmentation of music, to the analysis of sorting of music, and to the analysis of the theories of music perceptions.

DOI: 10.4018/978-1-61692-859-9.ch015

Theories of what homogenous units are to be found in music can be found in the music theory, for instance by the grouping theory of Lerdahl & Jackendoff (1983). Results from memory research (Snyder 2000) can also be used as the ground reference. Snyder refers to echoic memory (early processes) for event fusion, where fundamental units are formed by comparison with 0.25 seconds, the short-term memory for melodic and rhythmic grouping (by comparison up to 8 seconds), and long-term memory for formal sectioning by comparison up to one hour. Snyder (2000) relates this to the Gestalt theory grouping mechanisms of proximity (events close in time or pitch will be grouped together. Proximity is the primary grouping force at the melodic and rhythmic level (Snyder 2000, p 40). The second factor in grouping is similarity (events judged as similar, mainly with respect to timbre, will be grouped together). A third factor is continuity (events change in the same direction, for instance pitch). These grouping mechanisms give rise to closure, that can operate at the grouping level, or the phrase level, which is the largest group the short-term memory can handle. When several grouping mechanisms occur at the same time, intensification occurs, which gives rise to higher-level grouping. Other higher-level grouping mechanisms are parallelism (repeated smaller groups), or recurrence of pitch. The higher-level grouping demands long-term memory and they operate at a higher level in the brain, as compared to the smaller time-scale grouping. The higher-level grouping is learned while the shorter grouping is not. Snyder (2000) further divides the higher level grouping into the objective set, which is related to a particular music, and the subjective set, which is related to a style of music. Both sets are learned by listening to the music repeatedly. Snyder (2000) also related the shorter grouping to the 7±2 theory (Miller 1956), that states that the short-term memory can remember between five to nine elements.

Recently, the chunk has been appointed as an important element of music (Kühl 2007, Godøy 2008). A chunk is a short segment of a limited number of sound elements, corresponding to the working memory of approximately 3 seconds. A chunk consists of a beginning, a focal point (peak) and an ending. Both Kühl and Godøy seems to believe that the chunk is fundamental in music, but while Kühl mainly relates the chunking to the cognition, in particular the memory, Godøy also relates chunking to the action, i.e. physical gestures. Kühl (2007) extends the chunks to include microstructure (below 1/2 second), mesostructure (the present, approximately 3 seconds) and macrostructure (approximately. 30-40 seconds).

Automatic segmentation using dynamic programming has been proposed previously (Jensen et al 2005, Jehan 2005). In Jensen (2007), the dynamic programming is done of self-similarity matrices, created from the original features (rhythm, chroma or timbre) by comparing each time vector to all other time vectors. The dynamic programming will cluster the time vectors into segments, as long as the vectors are similar. By varying the insertion cost of new segments, segment boundaries can be found at different time scales. A low insertion cost will create boundaries corresponding to micro-level chunks, while a high insertion cost will only create few meso-level chunks. Thus, the same segmentation method can create segments of varying size, from short to long.

With the advent of downloaded music on personal computers, the necessity of assisting users choosing music among thousands of songs has arisen. Such a choice can be random (Shuffle play), by automatic playlist generation, or relate to a degree of similarity between songs. Playlist generation can be done on audio features (Foote 1997), for instance based on one song, or audio input, as in the query-by-humming systems (McNab 1996, Rolland et al 1999, Ghias *et al* 2001*)*. Playlist generation can also be based on meta-data (Pauws & Eggen 2002), and collaborative filtering.

Album vs. Single

What is an album? It is a collection of music tracks, that most often has some unifying content, by the composer, musicians, or otherwise. An album has a total duration above half an hour, and generally contains more than 10 tracks. Originally, the album term was designating a bound container of several 78-RPM discs, which individually could contain up to 10 minutes of music.

Assumably, before the start of technology for storing and distributing music, all genres would be performed in a length corresponding to an album or longer, i.e. in excess of half an hour. The technology up to around 1950 (cylinders and discs; acoustic devices from 1877 (the invention of recordings by Edison) to 1925, and electric devices thereafter) did not allow storing more than a couple of minutes on each support (Schoenherr 2005). Only with the advent of the 33-RPM vinyl record was it possible to store a full album on one support. As the cost of producing the 45-RPM single was lower than producing the larger 33-RPM, it retained popularity among the younger and dance-oriented record buyers. But the CD in the 1980ies supplanted the vinyl, and it does not contain the same production cost advantages for the single format, and thus single sales declined (The Sydney Morning Herald 2004). Lately, singles have regained popularity in the internet-based music sales, represented by the iPod/iTunes combination. In addition, different internet download opportunities currently put thousand of songs on many user music players. It seems clear that technology plays a role in the relative popularity of the album as compared to the single. It is only natural that technology also plays a role in helping users choose music in large databases, through the development of automatic playlist generators. These systems generally work by identifying the similarity between songs. The question remains, however, whether similarity is a good measure for grouping songs. Furthermore, are songs in an album homogenous individually, or do album tracks structure differently, either as one more homogenous group, or as smaller sections that are more homogenous that full songs.

It appears that one of the reasons for listening to music is mood regulation. This applies in particular for the adolescent music listeners (Behne 1997, Saarikallio & Erkkilä 2007). Behne (1997) studied changes in adolescent's development of Musikerleben (basically the sum of psychic processes that accompany active music listening). By asking children questions using different questionnaires, the mood managing (compensating) was the main listening mode found. Children with more problems have higher vegetative or sentimental listening style. Different mood management approaches was also found, i.e. coping with anger either by listening to aggressive music; "living out all feelings of anger", or by listening to slow music; "seek consolation in the music". Gender differences were also found; stimulative listening is more pronounced for boys, while sentimental listening is more pronounced for girls. Saarikallio & Erkkilä (2007) develops a theoretical model through group interviews and forms, which describes music mood regulation as a process of satisfying personal mood-related goals. Both Behne (1997) and Saarikallio & Erkkilä (2007) found a number of listening styles/regulatory strategies that are actively used by listeners to compensate for a problem or satisfy mood regulatory goals. It is not clear from these experiments, however, whether shorter or longer music excerpts are necessary or used in mood regulations. Other issues that may determine the choice of single versus album are the lyrics, and the issues of music and identity (Gerstin 1998).

While other studies are necessary in order to understand the full implications of singles vs. albums in mood regulations and identity creation, the work presented here attempts to draw some initial conclusions and replies to these questions through the analysis of the acoustic content of selected albums. First, features related to the common musical dimensions rhythm, timbre and chroma are calculated. Secondly, two experiments are performed on the selected album, using these features. The first experiment involves determining the homogeneity of songs in an album by merging the features of full albums, and then analyzing whether segment boundaries obtained by automatic segmentation fall within or between songs. The second experiment involves determining whether acoustic similarity is related to song position in an album. To determine this, songs in the selected albums are sorted, and the distance to the original order (the sorting complexity) is analyzed, to find if and how each album is ordered.

The work is organized as follows; first, the feature calculations are presented, and then the dynamic programming are presented along with analysis of the result of segmentation experiments. A second experiment regarding the sorting of songs is presented thereafter, along with a discussion of the significance of the results.

FEATURE ESTIMATION

In this work, features corresponding to music perception have been used in order to perform a good segmentation or sorting of the songs, and also to be able to assess the results to some aspects of human listening. Three different features are investigated here; the rhythmic feature (the *rhythmogram*, Jensen 2005) is based on the autocorrelation of the perceptual spectral flux (PSF, Jensen 2005). The PSF has high energy in the time position where perceptually important sound components, such as notes, have been introduced. The timbre feature (the *timbregram*) is based on the perceptual linear prediction (PLP), a speech front-end (Hermansky 1990), and the harmony feature (the *chromagram*) is based on the chroma (Bartsch & Wakefield 2001), calculated on the short-time Fourier transform (STFT). The Gaussian Weighted Spectrogram (GWS) is performed in order to improve resilience to noise and independence on block size for the *timbregram* and *chromagram*. More information of the feature estimation used here can be found in (Jensen 2007). A speech front-end, such as the PLP alters the STFT data by scaling the intensity and frequency so that it corresponds to the way the human auditory system perceives sounds. The chroma maps the energy of the FFT into twelve bands, corresponding to the twelve notes of one octave. Using the rhythmic, timbre, and harmonic features to identify the structure of the music, thus some of the different aspects of music perception are taken into account.

Rhythmogram

Any model of rhythm should have as basis some kind of feature that reacts to the note onsets. The note onsets mark the main characteristics of the rhythm. In a previous work (Jensen 2005), a large number of features were compared to an annotated database of twelve songs, and the perceptual spectral flux (PSF) was found to perform best. The PSF is calculated with a step size is 10 milliseconds, and the block size is 46 milliseconds. As the spectral flux in the PSF is weighted so as to correspond roughly to the equal loudness contour, both low frequency sounds, such as bass drum, and high frequency sounds, such a hi-hat are equally well taken into account.

This frequency weighting is obtained in this work by a simple equal loudness contour model. The power function is introduced in order to simulate the intensity-loudness power law and reduce the random amplitude variations. These two steps are inspired from the PLP front-end (Hermansky 1990) used in speech recognition. The PSF was compared to other note onset detection features with good results on the percussive case in a recent study (Collins 2005). In order to obtain a more robust rhythm feature, the autocorrelation of the feature is now calculated on overlapping blocks of 8 seconds, with half a second step size (2 Hz feature sample rate), Only the information between zero and two seconds is retained. The autocorrelation is normalized so that the autocorrelation at zero-lag equals one. If visualized with lag time on the y -axis, time position on the x-axis, and the autocorrelation values visualized as intensities, it gives a fast overview of the rhythmic evolution of a song. This representation, called *rhythmogram* (Jensen 2005), provides information about the rhythm and the evolution of the rhythm in time. The autocorrelation has been chosen instead of the fast Fourier transform FFT, for two reasons. First, it is believed to be more in accordance with the human perception of rhythm (Desain 1992), and second, it is believed to be more easily understood visually. The *rhythmogram* gives information about the tempo of the song, along with the strength of the tempo, and also to a slighter degree it gives information about the time signature. In a task where different songs are compared for segmentation or for sorting, the tempo difference will probably dominate the calculations.

Timbregram

The timbre is understood here as the spectral estimate and done here using the perceptual linear Prediction, PLP (Hermansky 1990). This involves using the bark (Sekey & Hanson 1984) scale, together with an amplitude scaling that gives an approximation of the human auditory system. The PLP is calculated with a block size of approximately 46 milliseconds and with a step size of 10 milliseconds. The *timbregram* is a feature that is believed to capture orchestration of the music mainly. In the *timbregram*, information about which instruments are participating in the music at the current time step is given, along with indications of what dynamic level the instruments are played. As the *timbregram* gives information on a very small time scale, it is rather noisy. In order to eliminate some of the noise, and improve the use of the *timbregram* data, smoothing is performed, by summing each bin across the full time-scale using a Gaussian that is localized in time. This is called the Gaussian Weighted Spectrogram, GWS (Jensen 2007). Using the GWS, all segments are used at all time steps, but the current block values are weighted higher than the more distant blocks. By averaging, using the Gaussian average, no specific time localization information is obtained of the individual notes or chords, but instead a general value of the time area is given. In this work, the averaging is done corresponding to a −3 dB window of approximately 1 second. After the GWS, the *timbregram* has a stepsize of ½ second.

Chromagram

Note estimation is notoriously error-prone even if a lot of progress is made in the domain currently. There exists one estimate that is robust and related to the note values, the chroma, which is used here. In the chroma, only the relative content of energy in the twelve notes of the octave is found. No information of the octave of the notes is included. The chroma is calculated from the STFT, using a blocksize of 46 milliseconds and a stepsize of 10 milliseconds. The chroma is obtained by summing the energy of all peaks of 12 $\log_2$ of the frequencies having multiples of 12. The *chromagram* gives information about the note value, without information about the octave. This is a rather good measure of which chords are played, and also of the musical scale and tonality. The *chromagram* gives information of each note played, along with information of unvoiced signals that add some noise to the *chromagram*. As with the *timbregram*, the *chromagram* is smoothed in time by the Gaussian Weighted Spectrogram. This removed information about the individual note values, and it retains better information about the tonality and the long-term presence of note values. The averaging using the GWS is done corresponding to a −3 dB window of approximately 1 seconds. After the GWS, the *chromagram* has a stepsize of ½ second.

As an example of the features, the *rhythmogram*, *timbregram* and *chromagram* of Alicia Keys – Songs in A Minor (J Records, 2001) is shown in Figure 1.

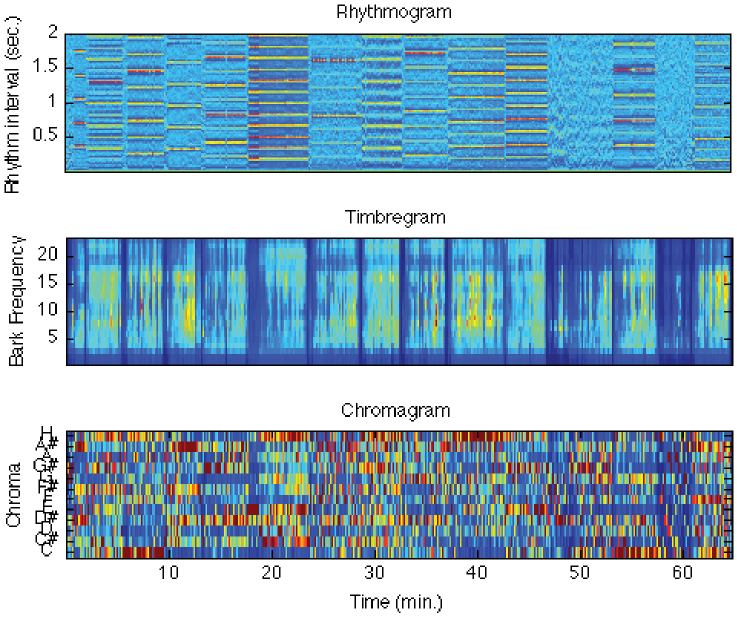
In the *rhythmogram*, it is clear from the visual analysis that each song has a different tempo. Of course, whether a tempo is a certain value or the double or half is difficult to ascertain, but the *rhythmogram* certainly gives information about the tempo, along with some information about the rhythm employed in each song. As for the *timbregram*, all songs have energy high up in the trebles, except one or two in the second part of the album. Most songs have a fade-in and fade-out in the beginning and the end of each song. It also seems many songs end up with some sort of crescendo.

AUTOMATIC MUSIC SEGMENTATION

With the features available for full albums, it is now possible to investigate whether automatic segmentation will find segmentation boundaries in between songs, or inside songs.

Automatic segmentation using dynamic programming or other methods has been proposed previously (Foote 2000, Bartsch & Wakefield 2001, Goto 2003, Chai & Vercoe (2003), Jensen 2005, Jensen *et al* 2005, Jensen 2007). These studies have shown that the acoustics of music permit to segment the songs into smaller segment, corresponding to the chorus/verse, or other structures. In an automatic segmentation task, adjacent blocks are grouped together, forming segments. This can for instance correspond to

Figure 1. Rhythmogram (top), timbregram, and chromagram (bottom) of the album 'Song in A Minor' by Alicia Keys



the chorus/verse structure found in most rhythmic music, or to changes in the rhythmic pattern, in the orchestration or in the notes played.

The dynamic programming used here based on the shortest-path algorithm (Cormen *et al* 2001) and done on self-similarity matrices, created from the original features (rhythm, chroma or timbre, Jensen 2007) by calculating the L2 norm of each time vector compared to all other time vectors, using a sequence of *N* vectors of each song that should be divided into a number of segments. In order to do this, let the *cost c(i,j)* of a segment from block *i* to *j* be the weighted sum of the self-similarity and the cost of a new segment be a fixed cost *α*. In order to compute a best possible segmentation, an edge-weighted directed graph *G* is constructed with the set of nodes is all the block of the song. For each possible segment an edge exists. The weight of the edge is *α + c(i, j)*. A path in *G* from node *1* to node *N + 1* corresponds to a complete segmentation, where each edge identify the individual segments. The weight of the path is equal to the total cost of the corresponding segmentation. Therefore, a shortest path (or path with minimum total weight) from node *1* to node *N + 1* gives a segmentation with minimum total cost. Such a shortest path can be computed in time $O(N^2)$.

The dynamic programming will cluster the time vectors into segments, as long as the vectors are similar. By varying the insertion cost α of new segments, segment boundaries can be found at different time scales. The same segmentation method can create segments of varying size, from short to long, from the grouping to the form of the music. Kühl (2007) related the different segment sizes in music to the notion of chunks. According to him, the chunk is an important element of music. A chunk is a short segment of a limited number of sound elements; a chunk consists of a beginning, a focal point (peak) and an ending. Kühl (2007) extends the chunks to include microstructure (below 1/2 sec), mesostructure (the present, appr. 3-5 secs) and macrostructure (Superchunks, Kühl and Jensen 2008) (at 30-40 secs). Using the shortest-path segmentation method a low insertion cost will create boundaries corresponding to micro-level chunks, while a high insertion cost will only create few meso-level chunks.

SONG OR CHUNK SEGMENTATION

In this section, a number of full albums are segmented into the same number of segments as there are songs in the album, using the automatic segmentation method based on the shortest path dynamic programming algorithm (Jensen *et al.* 2005). The idea is that if all or most of the segmentation boundaries fall in between songs in the album, then the songs are homogenous between them, while if many automatic segmentation boundaries fall inside songs, then other homogenous areas besides the songs exist.

Database

In order to investigate the issues of album, song or chunk preference, a small collection of representative albums have been collected. In this context, only rhythmic, popular music has been considered. The general idea is to consider three types of music, the original rock-n-roll, concept album and R&B/pop genres. The hypothesis is that there would be a difference in where the segment boundaries are found, dependent on the genre. It is expected that the concept album genre has less boundaries found at song boundaries, and the pop/dance genre has most. The same albums are also used for the experiment on whether similarity is a good measure of playlist ordering, by considering each album a playlist of its own. The original rock-n-roll genre consists of two albums, The Beatles, Sgt Peppers Lonely Hearts

(Parlophone/Capitol, 1967) and Rolling Stones, Exile On Main Street (Atlantic 1972). The concept albums also consist of two albums, Magma - Mekanik Destruktiw Kommandoh (A&M 1973) and Alice Cooper - The Last Temptation (Epic 1994). Finally the pop/R&B genre also consists of two albums, Shakira - Laundry Service (Epic 2001) and Alicia Keys - Songs In A Minor (J Records, 2001). These albums can also be divided into early rhythmic music, from 1967, 1971 and 1972, and recent rhythmic music, from 1994, 2001 and 2001. A second experiment is done using the full Beatles discography.

Segmentation

In order to sort the music, the *rhythmogram*, *timbregram* and *chromagram* features are calculated for each song individually, and then merged together into full album features. The self-similarity matrix is then calculated for each feature, and the shortest-path algorithm is employed using a Newton optimization method to iteratively find the same number of clusters, as there are songs. In this approach, first a low alpha value corresponding to a low cost of inserting new segment and a high alpha value corresponding to a high new segment insertion cost are used to calculate the extreme segmentation boundaries. Then, the average of the two alpha values are used to calculate the segmentation boundaries in between. Now, if the mean segment length is higher than the researched length, the low alpha is set to the mean value, otherwise the high alpha limit is set to the mean value. This is repeated until the alpha limits difference is below a threshold. As the number of segments is decreasing continually with alpha value (Jensen *et al* 2005), this method is guaranteed to converge. Unfortunately, the number of segments often changes in jumps of more than one, which sometimes makes it impossible to find the correct number of segments.

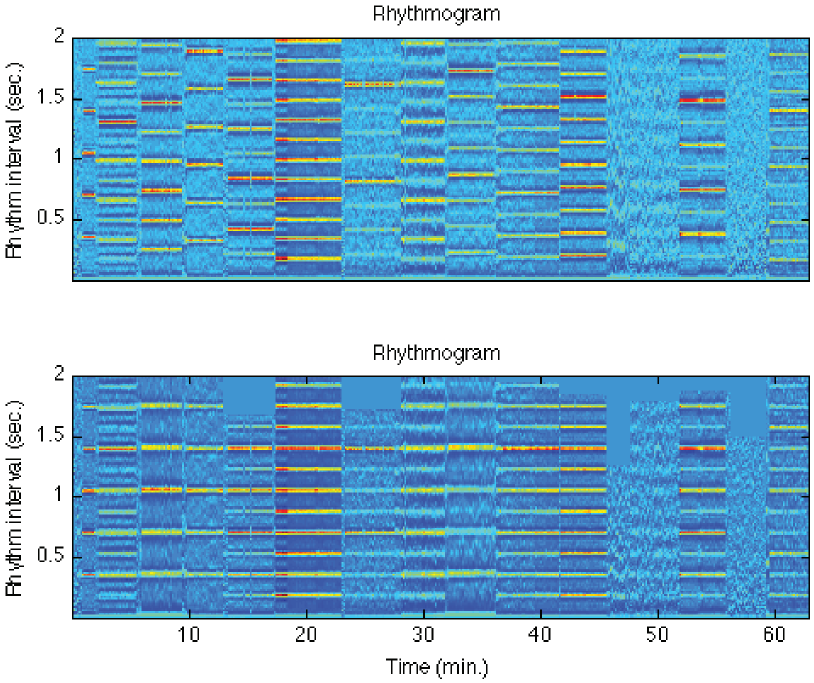
Tempo Normalization

As it is visible that tempo is one of the most changing features between songs, the tempo of the *rhythmogram* is normalized in order to ignore this effect in the segmentation task. Indeed, tempo changes only occur between songs generally, and it would have been dominant in the task of determining the automatic segmentation boundaries for the *rhythmogram*.

The tempo normalization is done as follows. First, the mean *rhythmogram* is calculated for each song of an album. Then the tempo difference of the first and the second song is determined. This tempo difference is estimated by calculating the minimal L2 difference between the mean *rhythmogram* of the two songs, where the second *rhythmogram* is interpolated until it corresponds best with the first song. This is done between a positive and negative tempo difference limit, at 43% of two seconds. The maximum of the L2 difference for all interpolations between the limits is found, and this is the tempo difference between the two songs. For all remaining songs, this same method is used, with the difference that the L2 norm is calculated between the new song and the mean of all songs up to the previous song. As an example of this tempo normalization, the *rhythmogram* of Alicia Keys – Song in A Minor is shown in Figure 2.

In many cases, it is not clear whether a tempo octave error has occurred, and for two songs at the end, there is not much rhythmic presence at all. However, this method of tempo normalization will certainly remove most of the tempo effect on the segmentation.

Figure 2. Original (top) and tempo normalized (bottom) rhythmogram of Alicia Keys – 'Songs in A Minor'

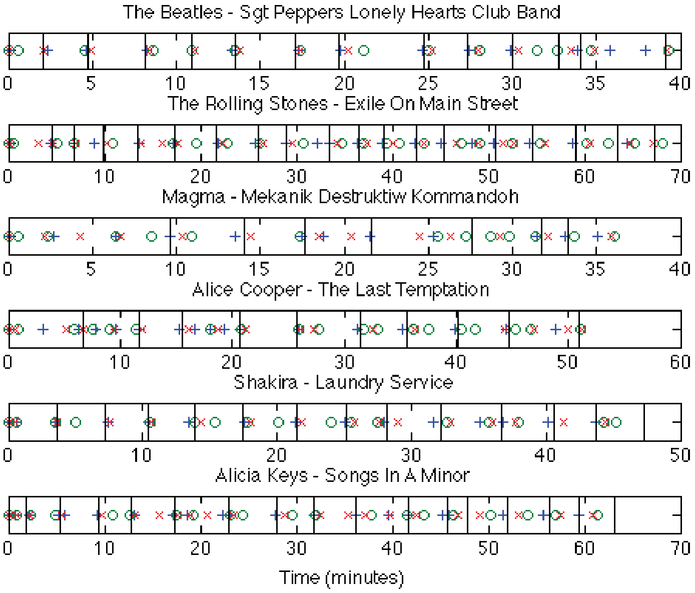


Segmentation Analysis

It is not necessarily an easy task to assess whether the music is homogenous in an album or in each song, or in shortest segments within songs. As a first approach, standard information retrieval performance measures are used in the comparison between the automatic segmentation boundaries and the song boundaries. This will give a first approach to whether the automatic segmentation renders boundaries corresponding to the song boundaries or not. Unfortunately, the automatic segmentation boundaries do not match the song boundaries. For a reasonable distance threshold of 4 seconds, and the mean of the six albums, 3.50 matches were found, of 13.83 song boundaries (Recall=25.30%) & 16.33 *rhythmogram* segments (Precision=15.91%), F_1=0.24, and (for the *timbregram*-based segmentation) 2.17 matched, of 13.83 song boundaries (Recall=15.66%) & 17.83 automatic segments (Precision=9.85%), F_1=0.14. Finally, the information retrieval measures are 2.00 matched, of 13.83 song boundaries (Recall=14.46%) & 17.50 *chromagram* segments (Precision=9.09%), F_1=0.13. As an initial conclusion, the *rhythmogram* boundaries match the song boundaries significantly better than the other features. This, notwithstanding that the *rhythmogram* has been normalized for each song, so as to have the same tempo throughout an album. The six albums have an F_1 value of 0.22, 0.13, 0.16, 0.16, 0.18 and 0.16, respectively. The original rock albums thus have both the best and the worse matches than the concept and pop/R&B albums. These scores are not impressive, though.

All the automatic segmentation boundaries are shown for the six albums together with the song boundaries in Figure 3. The *rhythmogram* boundaries are denoted'+', the *timbregram* 'o', the *chromagram* 'x'. The song boundaries are denoted with vertical lines. If the '+', the 'o', or the 'x' fall on the vertical

Figure 3. Segmentation boundaries for six albums. Rhythmogram boundaries are denoted '+', timbregram 'o', chromagram 'x', and the song boundaries are shown as vertical lines. Time is on the x-axis in minutes, corresponding to the length of each album

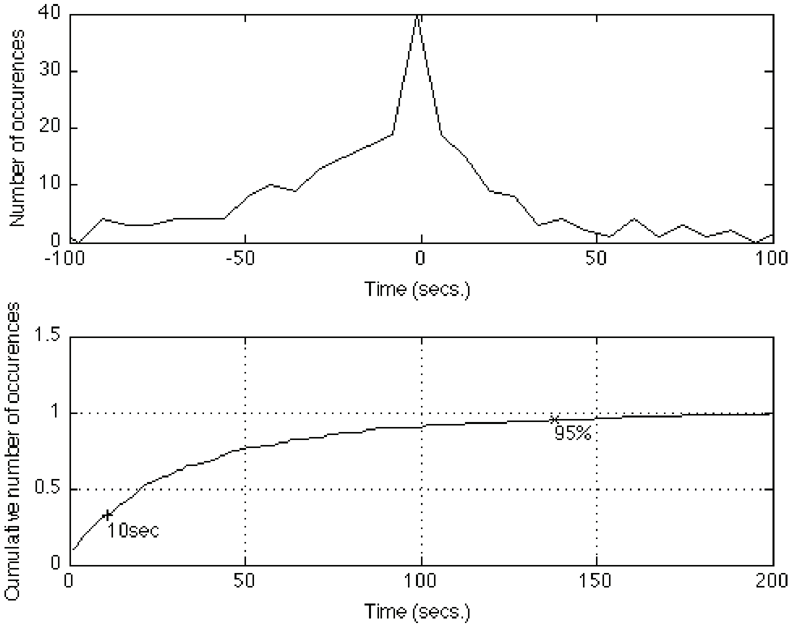


lines, then the corresponding automatic segmentation boundary is a match. The Beatles and Shakira albums seemingly have better matches, which are also reflected in the F_1 values.

While the recall and precision, and the associated F_1 measure give an estimate of the accuracy of the matching between the automatic segmentation boundaries, and the song boundaries, it is believed that these measures are not necessarily giving a full answer to the problem presented here. Instead, the distance of each automatic segmentation boundary to the nearest song boundary is calculated for each automatic segmentation boundary for all features and songs. This measure is presented in a histogram in Figure 4 (top).

If the absolute values of the histogram of distances between the automatic and song boundaries are cumulative summed and divided with the absolute sum, then a measure of the percentages of the automatic segmentation boundaries that are matched to a song boundary within each distance is obtained. This is presented in Figure 4 (bottom) for all six albums and all features. A 95% matching rate is obtained for distances up to 138 seconds. As the mean length of the songs on the six albums is 234 seconds, it would be expected that the automatic boundaries should be matched within half of this. As this is not the case, it can be said that song boundaries is not the main acoustic discriminatory place in music albums. Additionally, the bad recall, precision and F_1 scores also give the same indications. However, giving the intro and outros of the music under test, it may be the case that most automatic segmentations fall within these limits, in which case these boundaries still can be said to belong to the song boundaries. The intro and outro time has been estimated to have an average length of 10 seconds. This includes mainly fade-ins and outs, but also instrumental additions/removals, rhythm changes and occasionally chroma changes. If the matching measures are done on 10 sec thresholds, a better result is obtained, of course. The features match 6.5, 3.5, and 2.5 respectively for the *rhythmogram*, *timbregram* and *chro-*

Figure 4. Histogram of distances to nearest song boundary for all albums and features (top) and cumulative relative number of occurrences (bottom). The 10 seconds mark and 95% matched mark has been indicated in the lower subplot



magram. The corresponding F_1 values are 0.43, 0.22, and 0.16. The F_1 values for the six albums are 0.41, 0.27, 0.16, 0.21, 0.29 and 0.26. Significantly better results, but still not so impressive. With the 10 second matching threshold limit, the concept albums have the highest matching scores, which was also the expected result.

Finally, by calculating all distances for all features and songs, 32.13% of the matches are fond within 10 seconds. Thus, approximately one third of the boundaries found from automatic segmentation are matched to the song boundaries.

MUSIC SORTING

Playlist generation is a common topic in music information retrieval today, as it gives technological-based applications to the problem of identifying songs to play in large music databases. Indeed, thousands of songs can exist on music players, with little meta-data (artist, song, genre, etc) to support the retrieval of songs. Playlist generation can be based on either the acoustics of the sound, or some measure of collaborative behavior, or on the metadata belonging to the music. Foote (1997) used the Mel Frequency Cepstral Coefficients (MFCC) and a supervised greedy decision tree growing method and a correlation distance measure on the histogram of the leaf probabilities to sort male and female speech, percussive sounds and music. Other uses of audio features are for instance genre matching (Tzanetakis & Cook 2002). A particular application of music retrieval using acoustics is the so-called query-by-humming systems, in which the singing input is transcribed and pattern matched (McNab 1996, Rolland et al 1999, Ghias *et al 2001)* to the best match in a transcribed music database. Feng *et al* (2003) uses mood as a qualifier for music retrieval. Pauws (2002) uses meta-data of music in an interactive playlist generation process that creates a playlist of similar songs, based on one selected song. In an experiment, Pauws (2002) found indications that this playlist generation created playlist with high rated music than random playlist.

Other uses of meta-data is in the Pandora.com music genome project, in which music experts hand-label the music for better retrieving results. Collaborative music recommendation systems exist in many variants. One example is the iTunes Genius recommendation system.

The approach used here for the experiment on whether album order follows audio similarity or not is based on the Travelling Salesman Problem, which is also used in (Pohle *et al* 2005) for playlist generation. Audio features, the *rhythmogram*, *timbregram* and *chromagram* are calculated, but only the mean value and the standard deviation of each song are retained. Afterwards, the self-similarity matrix is calculated using the L2 norm. Finally, the travelling salesman problem (TSP) solution is calculated using the concorde software (Concorde 2009). As the optimal solution is NP-hard, and solutions so far are limited to a relatively low number of cities, the Lin-Kernighan (1973) heuristics is used to find a good solution. A travelling salesman problem is related to finding the minimum length that visits each 'city' exactly once. In our case, the cities are songs, and the distances are the L2 norm distance between each song and all other songs. The songs are ordered by the TSP solution in a manner, so as to minimize the total cost from the first song to the last song.

Sorting Complexity

If the songs in an album are sorted in a manner so as to minimize the difference between each adjacent song according to some feature, then the album is considered ordered. If not, then it is considered unordered. In order to verify this, a simple exchange-sorting algorithm is used. This sorting is done in $O(n^2)$. The maximum number of sorting steps is determined to be $0.5n(n-1)$. As the TSP solution may be inversed, this is tested to see if the sorting by the inverse list is done faster.

The TSP algorithm is now used to sorts each album, and the results are compared to the album sequence (*1…n*), the inverse sequence and all possible circular shifted sequences, i.e. (*2…n, 1*; *3…n, 1, 2*, etc). The number of steps necessary to sort the result is divided by the maximum number of sorting steps for randomly ordered sequences, and the square root is taken,

$$C = \sqrt{\frac{Ns}{0.5 \cdot n(n-1}}, \tag{1}$$

where *Ns* is the number of sorting steps, *n* is the number of songs in the album, and *C* is the sorting complexity of the album. The sorting complexity is equal to one for the inverse order, which is therefore considered ordered. For any random permutation, it does not typically exceed 0.7. The TSP-based sorting and the estimation of the sorting complexity are done for each feature (*rhythmogram*, *timbregram* and *chromagram*) individually. If the sorting complexity (*C*) is above 0.7 then the complexity is maximal, while if it close to zero, the sorting complexity is minimal. The sorting complexity *C* can be approximately said to be linear with respect to the randomness of the order of the result of the TSP solution. The sorting complexity corresponds to the percentage of an ordered sequence that is scrambled. I.e., if the sequence has *N* elements, and the first *N/2* elements a permuted randomly, then $C \approx 0.35$. The alternating sequence, i.e. first odd, then even elements, or on a general level, first multiples of *n*, then the other elements has sorting complexity *C<=0.5*, with the *C=0.5* for *n=2.* The total sorting complexity for all six test albums are shown in Table 1.

As a first conclusion, all albums have rather high sorting complexity. However, if the sorting complexity is significantly above *0.7* then it has some order different from a random permutation. This is not the case for the ordering found here. Indeed, the mean sorting complexity for all six albums and three features is *C=0.53*, which corresponds for instance to the case where approximately 76% of the songs are in random order. The Rolling Stones album has relatively higher sorting complexity for all three features, while the concept albums have the lowest sorting complexity. In particular, the concept albums have low sorting complexity for the *chromagram*- and *timbregram*-based TSP sorting result. Overall, the *chromagram* has the lowest sorting complexity, with the *timbregram* having the second-lowest sorting complexity.

Finally, if the songs of all six albums are merged together, then the sorting complexity is *0.63*, *0.52*, and *0.63* for the *rhythmogram-*, *timbregram-*, and *chromagram*-based TSP sorting results. If the circular shift sorting complexity maximum is found instead, then the full six albums have a complexity of *0.78*, *0.85*, and *0.78*. This is above the sorting complexity for completely random order, and there must therefore be some hint of a sorting system in the albums. This can be an alternating, an up-down sequence, or a something else, which renders sorting complexity above *0.7*. It is also interesting to observe that the same feature, which renders the lowest sorting complexity for the merged six albums, i.e. the *timbregram*, also renders the highest. This is also an indication that there is some system in the sorted order, in particular perhaps using the *timbregram*.

While it is difficult from the current study to say exactly how the order of the album songs is chosen, it is clear that it is not a random order sequence.

A second experiment using the same method has been performed on the complete Beatles discography. The sorting complexity result of this is shown in Table 2.

In the case of the Beatles discography, the sorting complexity is approximately the same as the six albums of varying genres. The mean sorting complexity is *0.55*, *0.56*, and *0.55* for the *rhythmogram*, *timbregram* and *chromagram*, respectively. While the *timbregram* renders the lowest sorting complexity for 10 out of 13 albums, it does not have lower overall mean sorting complexity. No distinctive difference among the albums or features is otherwise immediately discernible. The merged 13 albums have a sorting complexity of *0.64*, *0.54*, and *0.63*. The maximum sorting complexity within the reversed

Table 1. Sorting complexity values for the TSP sorting results for six albums and three features

Album/Feature	**No songs**	*Rhythmogram*	*Timbregram*	*Chromagram*	**Mean**
Beatles	13	0.58	**0.45**	0.59	0.54
Rolling Stones	18	**0.58**	0.62	0.60	0.60
Magma	7	0.53	**0.38**	0.44	0.43
Alice Cooper	10	0.56	0.49	**0.42**	0.49
Shakira	13	0.57	0.52	**0.51**	0.53
Alicia Keys	16	0.54	0.53	**0.52**	0.53
Mean	12.83	0.56	0.51	**0.50**	0.53

Table 2. Sorting complexity for the Beatles full discography

Album/Feature	**No**	*Rhythmogram*	*Timbregram*	*Chromagram*	**Mean**
Please Please Me	14	0.57	**0.50**	0.58	0.55
With The Beatles	14	0.58	0.56	**0.44**	0.53
A Hard Days Night	13	**0.47**	0.59	0.62	0.56
Beatles for Sale	14	0.57	**0.56**	0.58	0.57
Help!	14	0.57	**0.52**	0.55	0.55
Rubber Soul	14	**0.52**	0.55	0.56	0.55
Revolver	14	0.58	**0.50**	0.60	0.56
Sgt Peppers	13	0.58	**0.45**	0.59	0.54
The White Album 1	17	**0.51**	**0.51**	0.55	0.52
The White Album 2	13	0.57	**0.48**	0.53	0.53
Yellow Submarine	13	0.60	**0.44**	0.45	0.50
Abbey Road	17	0.55	**0.49**	0.61	0.55
Let It Be	12	**0.55**	0.59	0.63	0.59
Mean	14	0.55	0.56	0.55	0.55

and circular shifted TSP order is *0.77*, *0.84*, and *0.77*. As with the six albums previously, the *timbregram* has both the lowest and the highest possible sorting complexity.

The Actual Order of Albums

While it is clear now, that the albums under test are not in an order that can be recreated with the TSP sorting using the features, the actual order has not been determined. Apparently, according to the sorting complexity, the earlier albums are sorted more on the timbre, mainly, and the rhythm feature, while the later albums are sorted more in accordance with the chroma feature, although the differences between the sorting complexity for the different features are not that big.

While the order is probably not inherently random, it is difficult with the methods presented here, to identify whether there is system in the original order. The resulting TSP order for the six albums of varying genres is shown in Figure 5.

An immediate analysis of the TSP sorting order reveals that most often, the order is shifting up and down with a frequency of 2-6, i.e. 1 to 3 up and 1 to 3 down alternately. The Rolling Stones albums has most up-down order, while the Alice Cooper album has four up, one down, and then six up, with only two inversions for the rhythmogram, and some very similar for the *timbregram*, while the *chromagram* reveals four down and then six up in perfect order. Magma also have more order, this confirming the hypothesis that the concept albums are more homogeneous that the other albums. Overall, however, most albums have alternating up and down order.

The full Beatles discography album TSP order is shown in Figure 6.

The Beatles albums possibly also show a more systematic order in the *timbregram*-based TSP order, in particular for the later albums, which the sorting complexity values also showed previously. For instance, Yellow Submarine (Album 11) has an almost original order, with only few inversions.

Figure 5. Resulting TSP order for six albums, using the rhythmogram (top), the timbregram, and the chromagram (bottom). The x-axis is the original order, and the y-axis is the TSP order. Plus signs at the x-axis denote the album shifts

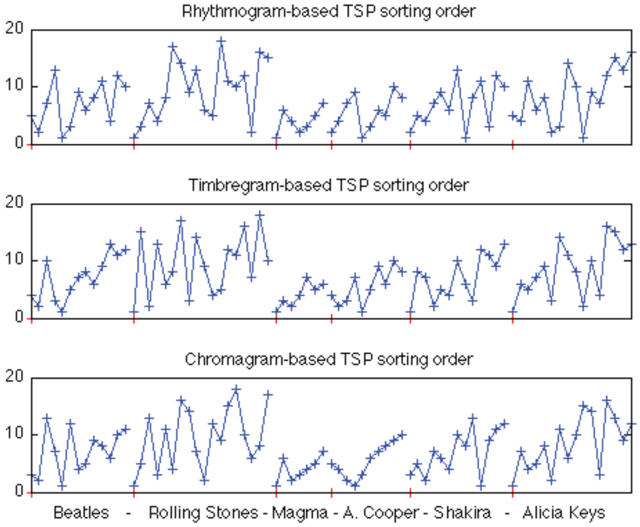
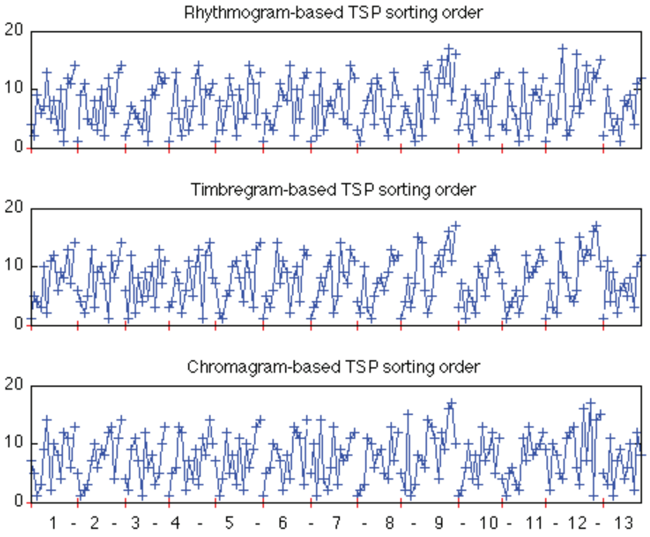


Figure 6. Resulting TSP order for full Beatles discography albums, using the rhythmogram (top), the timbregram, and the chromagram (bottom). The x-axis is the original order, and the y-axis is the TSP order. Plus signs at the x-axis denote the album shifts. Please refer to Table 2 for the album names



An initial investigation reveals preponderance for a periodicity of four in the TSP result order for the *rhythmogram-* and *chromagram*-based sorting, while the *timbregram*-based sorting reveals a periodicity of three. While this periodicity is not very strong in the autocorrelation, it is nonetheless repeated for both datasets used here. The sample size is too limited, however, to ascertain a truth in this matter with certitude. The order of the albums is neither random, nor ordered, with the exception of the concept albums. The hypothesis of alternating up/down order is plausible given the observations made here.

CONCLUSION

Whether music is sold in individual songs (singles) or through collections of songs (albums) has been more a matter of technology and commercial issues than a matter of user preferences due to the acoustic content of albums according to the literature. Singles were preponderant before approximately 1950 because of technological limitations in the support, and also afterwards, because of the 45 RPM single was cheaper than the 33-RPM album. Only with the CD pressing technology did single lose its price-advantage, which also decreased the relative sale of singles. With the internet-based sales, the price-advantage of singles was restored, and single sales increased again. As music listening is often related to mood regulation or identity building, the actual order of the songs in albums is potentially very important.

This study has investigated whether the acoustic content of music has an influence in the structure of albums. On the question of where the homogeneity is in albums, an experiment has been performed that reveals that approximately half of the segments found by automatic segmentation belong to song boundaries, while the other half is found inside songs. The experiment was based on three features related to music dimensions; the *rhythmogram*, *timbregram* and *chromagram*, that were merged for full albums. Six albums belonging to three genres, rock, concept albums and pop/R&B were used. For each album and feature, the self-similarity was calculated and the shortest-path through the self-similarity distances was calculated using dynamic programming. The intro and outro average duration was determined to be approximately 10 seconds, and the automatic segmentation boundaries was matched to the song boundaries using standard information retrieval measures. The resulting F_1 values below 0.5 are certainly not impressive, and they reveal that a lot of changes inside songs are more important than the changes between songs. Therefore, the conclusion must be that approximately half the homogeneity ruptures in music occurs within the songs, and the other half in between songs. However, this is very dependent on the music genre. The pop music in this study has more distinct songs than e.g. rock and concept albums.

In order to determine the segmentation boundaries for the rhythm, novel tempo normalization has been presented. This allows the comparison of the rhythm between songs, without taking into account the tempo, which would otherwise be to dominant.

A second experiment has investigated the order of the albums. By calculating the mean and the standard deviation of the three features, and calculating the difference between each song, a travelling-salesman problem solution has been employed to sort the songs so as to minimize the total distance between them. In order to compare the novel order with the original order, a sorting complexity measure has been used to investigate whether an album song order is totally random, or if it contains some order. While the *rhythmogram* performs best in the segmentation task, the *timbregram* and *chromagram* performs best in the sorting task. For most more recent songs, the *chromagram* has the lowest sorting complexity, and for the older songs, the *timbregram*-based segmentation renders the order closest to the original order. Further analysis reveals that the concept albums have almost linear order with respect to most features. For most other albums, the order varies up/down with a frequency of two or three.

The preliminary conclusions that can be drawn from this work are the following; Music in commercial albums is not necessarily divided into individual homogenous songs. A lot of the changes happen within songs. Up to two-thirds of segment boundaries has been found inside songs. As for the order of musical albums, it has been found to generally be varied in a way so the rhythm and timbre content variation direction is changed every four songs on average, while the tonal content variation direction is changed every three songs. For the concept albums, a more linear order was found, however.

REFERENCES

Bartsch, M. A., & Wakefield, G. H. (2001). To Catch a Chorus: Using Chroma-Based Representations For Audio Thumbnailing. In *Proceedings of the Workshop on Applications of Signal Processing to Audio and Acoustics (CD).* Washington, DC: IEEE.

Behne, K. (1997) The Development of "Musikerleben" in Adolescence: How and Why Young People Listen to Music. In I. Deliége & J.A. Sloboda, (Eds.), *Perception and Cognition of Music,* (pp. 143–159). Hove, UK: Psychology Press.

Chai, W., & Vercoe, B. (2003). Music thumbnailing via structural analysis. In *Proceedings of ACM Multimedia Conference*, November.

Collins, N. (2005). A comparison of sound onset detection algorithms with emphasis on psychoacoustically motivated detection functions. In *Proceedings of AES 118th Convention*, Barcelona, Spain, May.

Concorde. (2009). Concorde TSP solver. Retrieved July 1, 2009, from http://www.tsp.gatech.edu/concorde.html

Cormen, T. H., Stein, C., Rivest, R. L., & Leiserson, C. E. (2001). *Introduction to Algorithms* (2nd ed.). New York: The MIT Press and McGraw-Hill Book Company.

Desain, P. (1992). A (de)composable theory of rhythm. *Music Perception*, *9*(4), 439–454.

Feng, Y., Zhuang, Y., & Pan, Y. (2003). Music information retrieval by detecting mood via computational media aesthetics. In *Proceedings of the 2003 IEEE/WIC International Conference on Web Intelligence*, Washington, DC.

Foote, J. (1997). A similarity measure for automatic audio classification. In *Proc. AAAI 1997 Spring Symposium on Intelligent Integration and Use of Text, Image, Video, and Audio Corpora*, Stanford, Palo Alto, California, USA.

Foote, J. (2000). Automatic Audio Segmentation using a Measure of Audio Novelty. *Proceedings of IEEE International Conference on Multimedia and Expo*, *I*, 452–455.

Gerstin, J. (1998). Reputation in a Musical Scene: The Everyday Context of Connections between Music, Identity and Politics. *Ethnomusicology*, *42*(3), 385–414. doi:10.2307/852848

Ghias, A., Logan, J., Chamberlin, D., & Smith, B. C. (2001). Query by humming - musical information retrieval in an audio database. In *Proceedings Multimedia,* (pp. 231-236).

Godøy, R. I. (2010). Chunking Sound for Musical Analysis. []. Berlin: Springer.]. *Lecture Notes in Computer Science*, *5493*, 67–80. doi:10.1007/978-3-642-02518-1_4

Goto, M. (2003). A chorus-section detecting method for musical audio signals. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, (pp. 437-440), April.

Hermansky, H. (1990). Perceptual linear predictive (PLP) analysis of speech. *The Journal of the Acoustical Society of America*, *87*(4), 1738–1752. doi:10.1121/1.399423

Jehan, T. (2005). Hierarchical Multi-Class Self Similarities. In *Proceedings of the WASPAA (CD), USA*.

Jensen, K. (2007). Multiple scale music segmentation using rhythm, timbre and harmony. *EURASIP Journal on Applied Signal Processing*, (Special issue on Music Information Retrieval Based on Signal Processing), 68–74.

Jensen, K., Xu, J., & Zachariasen, M. (2005). Rhythm-based segmentation of Popular Chinese Music. In *Proceeding of the ISMIR*, (pp. 374-380), London.

Kühl, O. (2007). *Musical Semantics*. Bern, Switzerland: Peter Lang.

Kühl, O., & Jensen, K. (2008). Retrieving and recreating Musical Form. [). Berlin: Springer-Verlag.]. *Lecture Notes in Computer Science*, *4969*, 263–275. doi:10.1007/978-3-540-85035-9_18

Lerdahl, E., & Jackendoff, R. (1983). *A generative theory of tonal music*. Cambridge, MA: M.I.T. Press.

Lin, S., & Kernighan, B. W. (1973). An Effective Heuristic Algorithm for the Traveling-Salesman Problem. *Operations Research*, *21*(2), 498–516. doi:10.1287/opre.21.2.498

McNab, R. J., Smith, L. A., Witten, I. H., Henderson, C. L., & Cunningham, S. J. (1996). Towards the digital music library: Tune retrieval from acoustic input. In *Proceeding DL'96*, (pp. 11-18).

Miller, G. A. (1956). The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological Review*, *63*, 81–97. doi:10.1037/h0043158

Pauws, S., & Eggen, B. (2002). PATS: Realization and user evaluation of an automatic playlist generator. In *Proceedings of the 3rd International Conference on Music Information Retrieval*, Ircam, France, (pp. 222-230).

Pohle, T., Pampalk, E., & Widmer, G. (2005). Generating similarity-based playlists using traveling salesman algorithms. In *Proceedings of Digital Audio Effects*, Madrid, Spain, (pp. 220-225).

Rolland, P. Y., Raskinis, G., & Ganascia, J. G. (1999). Musical content-based retrieval: an overview of the Melodiscov approach and system. *ACM Multimedia,* (1), 81-84.

Saarikallio, S., & Erkkilä, J. (2007). The roles of music in adolescents' mood regulation. *Psychology of Music*, *35*, 88–109. doi:10.1177/0305735607068889

Schoenherr, S. (2005). *Recording Technology History*. Retrieved July 1st, 2009, from http://history.sandiego.edu/gen/recording/notes.html

Sekey, A., & Hanson, B. A. (1984). Improved 1-bark bandwidth auditory filter. *The Journal of the Acoustical Society of America*, *75*(6), 1902–1904. doi:10.1121/1.390954

Snyder, B. (2000). *Music and Memory. An Introduction*. Cambridge, MA: The MIT Press.

The Sydney Morning Herald. (2004). Pop single still hits right note. Retrieved from http://www.smh.com.au/articles/2004/08/25/1093246622880.html, accessed July 1st 2009.

Tzanetakis, G., & Cook, P. (2002). Musical Genre Classification of Audio Signals. *IEEE Transactions on Speech and Audio Processing*, *10*(5), 293–302. doi:10.1109/TSA.2002.800560