Multiple scale music segmentation using rhythm, timbre and harmony

Jensen, Karl Kristoffer

Published in:
Eurasip Journal on Advances in Signal Processing

Publication date:
2007

Document Version
Publisher's PDF, also known as Version of record

Hindawi Publishing Corporation
EURASIP Journal on Advances in Signal Processing
Volume 2007, Article ID 73205, 11 pages
doi:10.1155/2007/73205

Research Article

Multiple Scale Music Segmentation Using Rhythm, Timbre, and Harmony

Kristoffer Jensen

Department of Medialogy, Aalborg University Esbjerg, Niels Bohrs Vej 6, Esbjerg 6700, Denmark

Received 30 November 2005; Revised 27 August 2006; Accepted 27 August 2006

Recommended by Ichiro Fujinaga

The segmentation of music into intro-chorus-verse-outro, and similar segments, is a difficult topic. A method for performing automatic segmentation based on features related to rhythm, timbre, and harmony is presented, and compared, between the features and between the features and manual segmentation of a database of 48 songs. Standard information retrieval performance measures are used in the comparison, and it is shown that the timbre-related feature performs best.

1. INTRODUCTION

Segmentation has a perceptual and subjective nature. Manual segmentation can be due to different attributes of music, such as rhythm, timbre, or harmony. Measuring similarity between music segments is a fundamental problem in computational music theory. In this work, automatic music segmentation is performed, based on three different features that are calculated so as to be related to the perception of rhythm, timbre, and harmony.

Segmentation of music has many applications such as music information retrieval, copyright infringement resolution, fast music navigation, and repetitive structure finding. In particular, the navigation has been a key motivation in this work, for possible inclusion in the *mixxx* [1] DJ simulation software. Another possibility is the use of the automatic segmentation for music recomposition [2]. In addition to this, the visualization of the rhythm, timbre, and harmony related features is believed to be a useful tool for computer-aided music analysis.

Music segmentation is a popular topic in research today. Several authors have presented segmentation and visualization of music using a self-similarity matrix [3–5] with good results. Foote [5] used a measure of novelty calculated from the selfsimilarity matrix. Cooper and Foote [6] use singular value decomposition on the selfsimilarity matrix for automatic audio summary generation. Jensen [7] optimized the processing cost by using a smoothed novelty measure, calculated on a small square on the diagonal of the selfsimilarity matrix. In [8] short and long features are used for summary generation using image structuring filters and unsupervised learning. Dannenberg and Hu [9] use ad hoc dynamic programming algorithms on different audio features for identifying patterns in music. Goto [10] detects the chorus section using identification of repeated section on the chroma feature. Other segmentation approaches include information-theoretic methods [11]. Jehan [12] recently proposed a recursive multiclass approach to the analysis of acoustic similarities in popular music using dynamic programming.

A previous work used a model of rhythm, the rhythmogram, to segment popular Chinese music [13]. The rhythmogram is calculated by taking overlapping autocorrelations of large blocks of a feature (the perceptual spectral flux *PSF*) that give a good estimate of the note onset. In this work, two other features are used, one feature that provides an estimate of the timbral content of the music (the *timbregram*), and one estimate that gives an estimate of the harmonic content (the *chromagram*). Both these features are calculated on a novel spectral feature, the Gaussian weighted average spectrogram (*GWS*). This feature multiplies and sums all the *STFT* frequency bins with a Gaussian with varying position and a given standard deviation. Thus, an average measure of the *STFT* can be obtained, with the major weight on an arbitrary time position, and a given influence of the surrounding time position. This model has several advantages, as will be detailed below.

A novel method to compute segmentation splits using a shortest path algorithm is presented, using a model of the cost of a segmentation as the sum of the individual costs of segments. It is shown that with this assumption, the problem

can be solved efficiently to optimality. The method is applied to three different databases of rhythmic music. The segmentation based on the rhythm, timbre, and chroma features is compared to the manual segmentation using standard IR measures.

This paper is organized as follows. First, the feature extraction is presented, then the self-similarity is detailed, and the shortest path algorithm outlined. The segmentation is compared to the optimum results of manually segmented music in the experiment section, and finally a conclusion is given.

2. FEATURE EXTRACTION

In audio signal segmentation, the feature used for segmentation can have an important influence on the segmentation result.

The rhythmic feature used here (the *rhythmogram*) [7] is based on the autocorrelation of the *PSF* [7]. 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 Gaussian weighted averaged perceptual linear prediction (*PLP*), a speech front-end [14], and the harmony feature (the *chromagram*) is based on the chroma [3], calculated on the Gaussian weighted short-time Fourier transform (*STFT*). The Gaussian weighted spectrogram (*GWS*) introduced here is shown to have several advantages, including resilience to noise and independence on block size. The *STFT* performs a fast Fourier transform (*FFT*) on short overlapping blocks. Each *FFT* thus gives information of the frequency content of a given time segment. The *STFT* is often visualized in the spectrogram. 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.

By using the rhythmic, timbral, and harmonic contents to identify the structure of the music, a rather complete understanding is assumed to be found.

2.1. 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 [7], 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 as

$$psf(n) = \sum_{k=1}^{N_b/2} W(f_k)\left\{(a_k^n)^{1/3} - (a_k^{n-1})^{1/3}\right\}, \quad (1)$$

where n is the feature block index, N_b is the block size, and a_k and f_k are the magnitude and frequency of the bin k of the short-time Fourier transform (*STFT*), obtained using a Hanning window. The step size is 10 milliseconds, and the block size is 46 milliseconds. W is the frequency weighting used to obtain a value closer to the human loudness contour. This frequency weighting is obtained in this work by a simple equal loudness contour model [15]. The power function is used to simulate the intensity-loudness power law and reduce the random amplitude variations. These two steps are inspired from the *PLP* front-end [14] 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 [16]. The *PSF* feature detects most of the manual note onsets correctly, but it still has many peaks that do not correspond to note onsets, and many note onsets do not have a peak in the *PSF*. 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),

$$rg_n(i) = \sum_{j=2n/f_{sr}+1}^{2n/f_{sr}+8/f_{sr}-i} psf(j)psf(j+i). \quad (2)$$

f_{sr} is the feature sample rate, and n is the block index. 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 colors, it gives a fast overview of the rhythmic evolution of a song.

This representation, called rhythmogram [7], 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 [17], and second, it is believed to be more easily understood visually. The rhythmograms for two songs, *Whenever, Wherever* by Shakira and *All of me* by Billie Holiday are shown in Figure 1. The recent Shakira pop song has a steady rhythm, with only minor changes in instrumentation that changes the weight of some of the rhythm intervals, without affecting the fundamental beat, while *All of Me* does not seem to have any stationary rhythm.

2.2. Gaussian windowed spectrogram

While the rhythmogram indeed gives a good estimate of the changes in the music, as it is believed to encompass changes in instrumentation and rhythm, while not taking into account singing and solo instruments that are liable to have influence outside the segment, it has been found that the manual segmentation sometimes prioritize the singing or solo instrument over the rhythmic boundary. Therefore, other features have been included that are calculated from the spectral content of the music. If these features are calculated on short segments (10 to 50 milliseconds), they give detailed information in time, too varying to be used in the segmentation method used here. Instead, the features are calculated on a large segment, but localized in time by using the average of many *STFT* blocks multiplied with a Gaussian,

$$gws_k(t) = \sum_{i=1}^{sr/2} stft_k(i)g(\mu,\sigma). \quad (3)$$

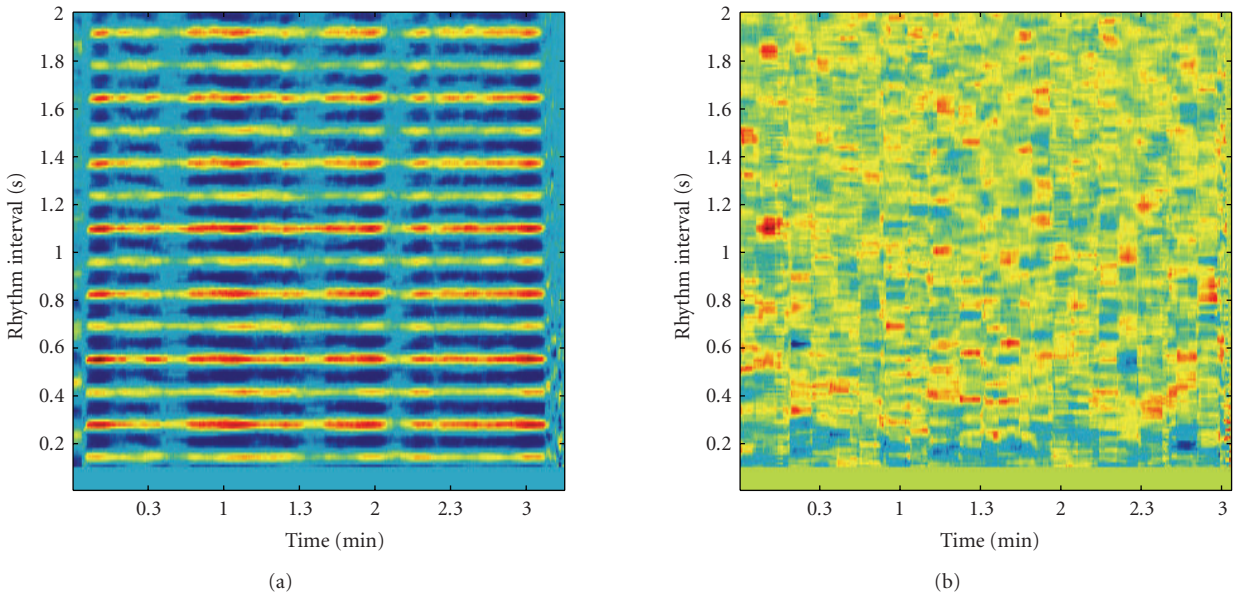

Figure 1: Rhythmogram of (a) *Whenever, Wherever* and (b) *All of Me*.

Here $stft_k(i)$ is the kth bin (corresponding to the frequency $f_k = k\,\mathrm{sr}/N_b$) of the ith block of the short-time Fourier transform, and $g(\mu, \sigma)$ is the Gaussian, defined as

$$g(\mu, \sigma) = \left(\frac{1}{\sigma\sqrt{2\pi}}\right) e^{-(t-\mu)^2/(2\sigma^2)}. \quad (4)$$

Thus, by varying μ, information about different time localizations can be obtained, and by increasing σ, more influence from the surrounding time steps can be included.

2.2.1. Comparison to large window FFT

The advantages of such a hybrid model are numerous.

- *Noise*

  Assuming the signal is consisting of a sum of sinusoids plus a rather stationary noise, this noise is smoothed in the *GWS*. Thus the voiced part will stand out stronger and be more pertinent to observation or subsequent processing.

- *Transients*

  A transient will be averaged out over the full length of the block, in case of the *FFT*, while it will have a strong presence in the *GWS* when in the middle of the Gaussian.

- *Peak width*

  The *GWS* has a peak width that is independent of the actual duration that the *GWS* encompasses, while the *FFT* has a decreasing peak width with increasing blocksize. In case of music with sligthly varying pitch, such as live music, or when using vibrato, a small peak width is advantageous.

- *Peak separation*

  In case of two partials at proximity, the partials will retain their separation with the *GWS*, while the separation will increase with the *FFT*. While this is not an issue in itself, the rise of space between strong partials that contain noise is.

- *Processor cost*

  The *FFT* has a processor cost of $O(N \log_2 N)$, while the *GWS* has a processor cost of $O(M(N_2 + N_2 \log_2 N_2))$, where M is the number of *STFT* blocks, and N_2 is the *STFT* blocksize. In case *FFT* is rewritten as $O(MN_2 \times \log_2(MN_2))$, to have the same total blocksize as the *GWS*, the *GWS* is approximately $\log_2(MN_2)/\log N_2$ faster.

- *Comparison to common speech features*

  While a speech feature, such as the *PLP*, has a better time resolution, it has no frequency resolution with regards to individual partials. The *GWS*, in comparison, still takes into account new notes in otherwise dense spectrum.

In conclusion, the *GWS* permits analyzing the music with a varying time resolution, giving noise elimination, while maintaining the frequency resolution at all time resolutions and at a lower cost than the large window *FFT*.

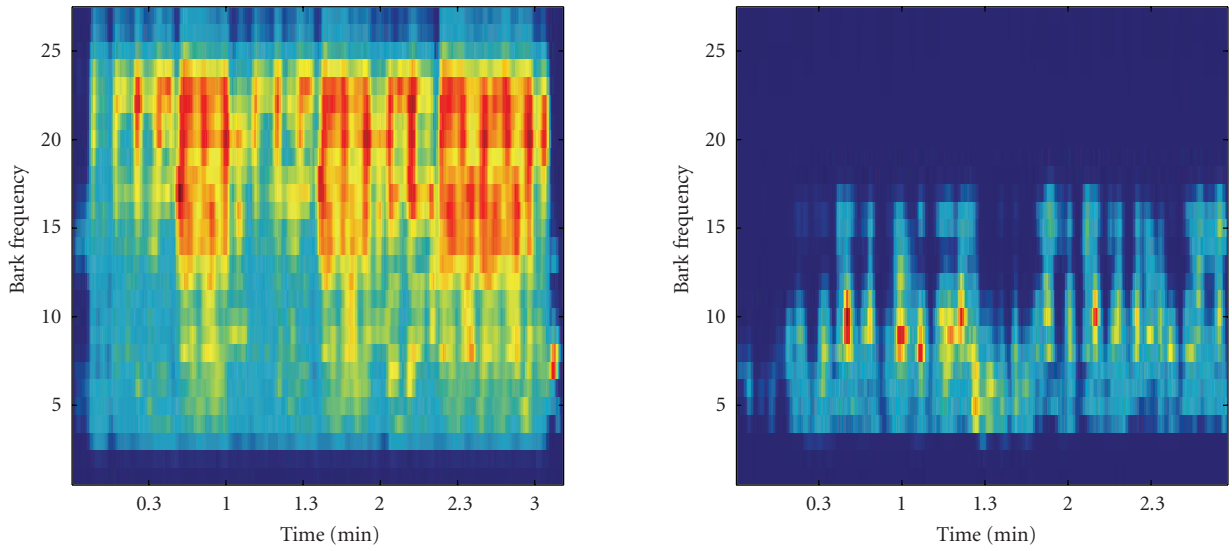

Figure 2: Timbregram: *PLP* calculated using the *GWS* of (a) *Whenever, Wherever* and (b) *All of Me*.

2.3. Timbre

The timbre is understood here as the spectral estimate and done here using the Gaussian average on the perceptual linear Prediction, *PLP* [14]. This involves using the bark [18] scale, together with an amplitude scaling that gives an approximation of the human auditory system. The *PLP* is calculated with a blocksize of approximately 10 milliseconds and with an overlap of 1/2. The *GWS* is calculated from the *PLP* in steps of 1/2 second, and with $\sigma = 100$. This gives a -3 dB width of a little less than one second. A smaller σ would give too scattered information, while a too large value would smooth the *PLP* too much. An example of the *PLP* for the same two songs as above is shown in Figure 2.

The *timbregram* is just as informative as the *rhythmogram*, although it does not give similar information. While the rhythm evolution is illustrated in the *rhythmogram*, it is the evolution of the *timbre* that is shown with the *timbregram*. This includes the insertion of new instruments, such as the trumpet solo in *All of Me* at approximately 1′30″. The voice is most prominent in the *timbregram*. The repeating chorus sections are very visible in *Whenever, Wherever*, mainly because of the repeating singing style in each chorus, while the choruses are less visible in *All of Me*, since it's sung differently each time.

2.4. Harmony

The harmony is calculated on an average spectrum, using the Gaussian average, as is the spectral estimate. In this case, the chroma [3] is used as the measure of harmony. Thus, 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 in the *chromagram*. It 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. By averaging, using the Gaussian average, no specific time localization information is obtained of the individual notes or chords. Instead an estimate of the notes played in the short interval is given as an estimate of the scale used in the interval. A step size of 1/2 second is used, together with a σ value of 200, corresponding to a -3 dB window of approximately 3 seconds. The *chromagram* of the same two songs as above is shown in Figure 3.

It is obvious that the *chromagram* shows yet another aspect of the music. While the *rhythmogram* pinpoints rhythmic similarities, and the *timbregram* indicates the spectral part of the timbre, the *chromagram* gives rather precise information about the chroma of the notes played in the vicinity of the time location. Often, these three aspects of the music change simultaneously at the segment boundary. Sometimes, however, not one of the features can help in, for instance, identifying similar segments. This is the case for the title chorus of *All of me*, where Billie Holiday and the rhythmic section change the key, the rhythm, and the timbre between the first and the second occurrence. Even so, most often, the segment splits are well indicated by any of the features. This is proven in the next section, where first the selfsimilarity of the features are calculated, the segment splits are calculated using a shortest path algorithm with variable segment split cost, and finally these segment splits are matched to manual segment splits of different rhythmic music.

2.5. Visualization

Both the *rhythmogram*, the *timbregram*, and the *chromagram* give pertinent information about the evolution in time of the

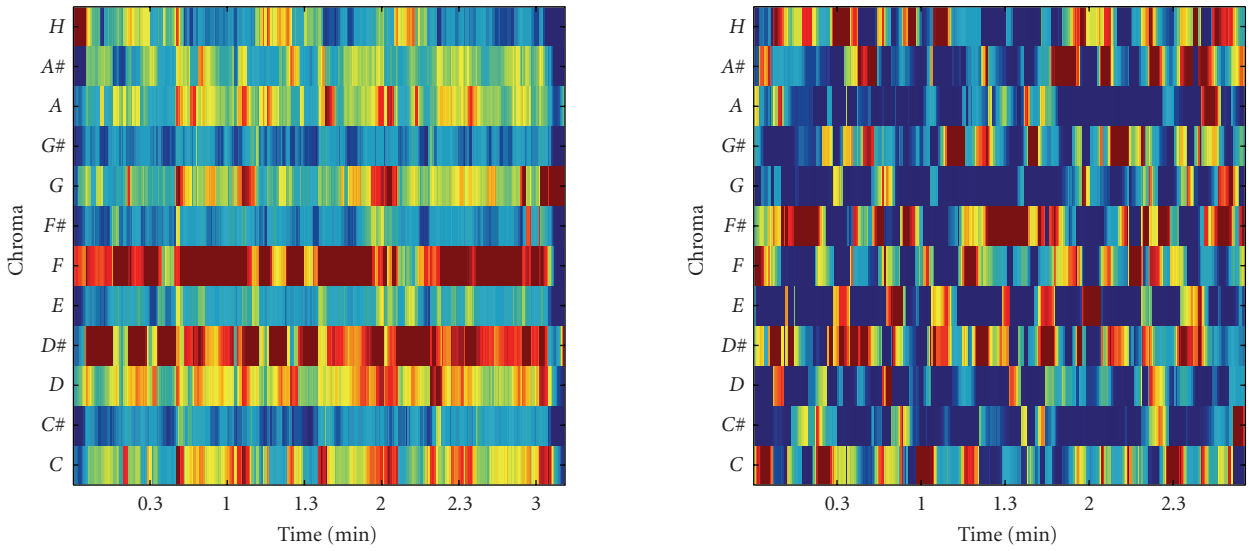

Figure 3: Chromagram: *chroma* calculated using the *GWS* of (a) *Whenever, Wherever* and (b) *All of Me*.

rhythm, timbre, and chroma, as can be seen in Figures 1, 2, and 3. This is believed to be a great help in tasks involving manipulation and analysis of music, for instance for music theorists, DJs, digital *turntablist*, and others involved in the understanding and distribution of music.

3. SELF-SIMILARITY

In order to get a better representation of the similarity of the song, a measure of selfsimilarity is used. This was first used in [19] to give evidence of recurrence in dynamic systems. Self similarity calculation is a means of giving evidence of the similarity and dissimilarity of the features. Several studies have used a measure of selfsimilarity [8] in automatic music analysis. Foote [4] used the dot product on *mfcc* sampled at a 100 Hz rate to visualize the selfsimilarity of different music excerpt. Bartsch and Wakefield [3] used the chroma-based representation to calculate the cross-correlation and identify repeated segments, corresponding to the chorus, for audio thumbnailing. Later Foote [5] introduced a checkerboard kernel correlation as a novelty measure that identifies notes with small time lag, and structure with larger lags with good success. Jensen [7] used smoothed novelty measure to identify structure without the costly calculation of the full checkerboard kernel correlation. In this work, the L_2 norm is used to calculate the distance between two blocks. The self-similarities of *Whenever, Wherever* and *All of Me* calculated for the *rhythmogram*, the *timbregram*, and the *chromagram* are shown in Figure 4.

It is clear that *Whenever, Wherever* contains more similar music (indicated with a dark color) than *All of Me*. It has a distinctly different intro and outro, and three repetitions of the chorus, the third one repeated. While this is visible, in part, in the *rhythmogram*, and quite so in the *timbregram*, it is most prominent in the *chromagram*, where the three repetitions of the chorus stand out. As for the intro and the outro, they are quite similar with regard to rhythm, as can be seen in the *rhythmogram*, rather dissimilar with regard to the timbre, and more dissimilar with respect to the *chromagram*. This is explained by the fact that the intro is played on a guitar, the outro on pan-flute, and although they have similar note durations, the timbres of a pan-flute and a guitar are quite dissimilar, and they do not play the same notes. The situation for *All of Me* is that the rhythm is changing all the time, in short segments with a duration of approximately 10 seconds. The saxophone solo at 1′30 is rather homogenous and similar to the piano intro and some parts of the vocal verse. A large part of the song is more similar with respect to timbre than rhythm or harmony, although most of the song is only similar to itself in short segments of approximately 10 seconds for the timbre, as it is for the *chromagram*.

4. SHORTEST PATH

Although the segments are visible in the self-similarity plots, there is still a need for a method for identifying the segment splits. Such a method was presented in [13]. In order to segment the music, a model for the cost of one segment and the segment split is necessary. When this is obtained, the problem is solved using the shortest path algorithm for the directed acyclic graph. This method provides the optimum solution.

4.1. Cost of one segment

For all features, a sequence $1, 2, \ldots, N$ of N blocks of music is to be divided into a number of segments. $c(i, j)$ is the cost of a segment from block i to block j, where $1 \leq i \leq j \leq N$. This cost of a segment is chosen to be a measure of the selfsimilarity of the segment, such that segments with a high

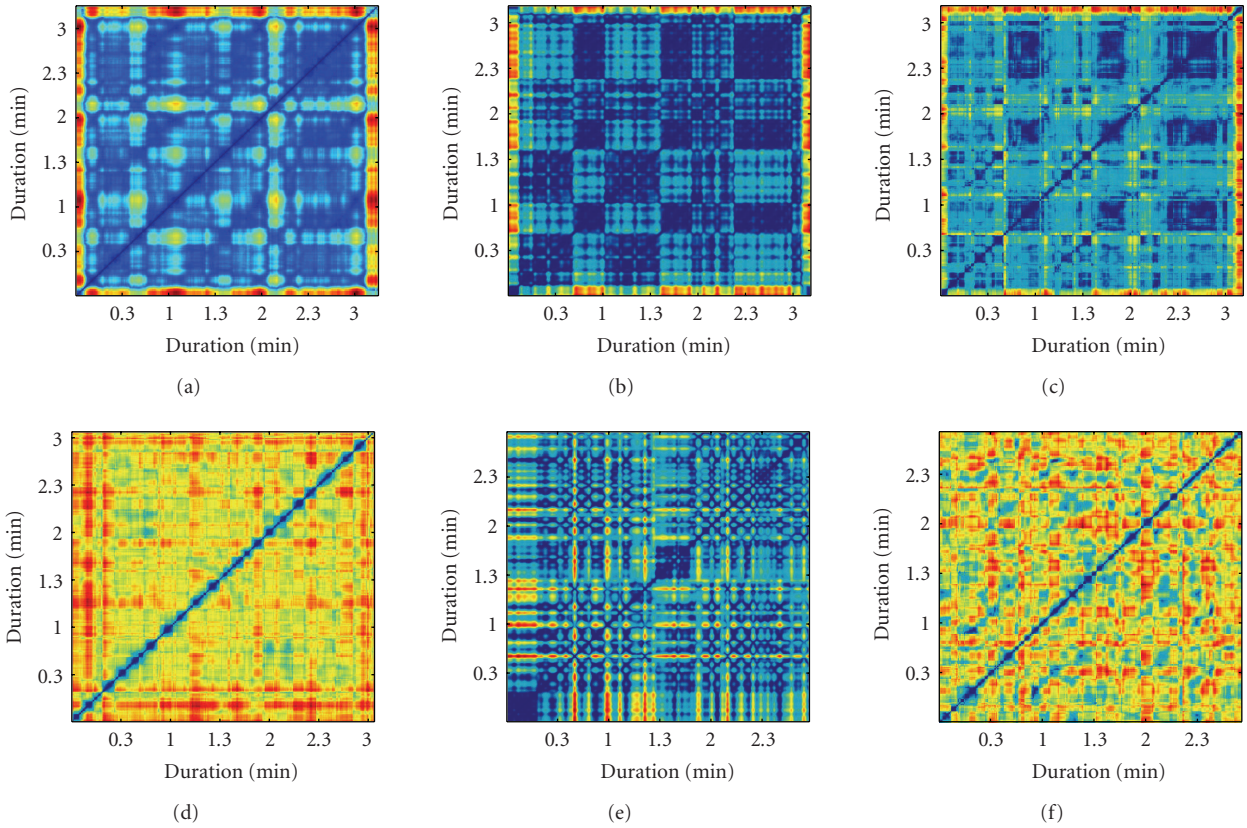

FIGURE 4: L_2 self-similarity for the *rhythmogram* (left), *timbregram* (middle), and *chromagram* (right), of *Whenever, Wherever* (top) and *All of me* (bottom).

degree of selfsimilarity have a low cost,

$$c(i,j) = \left(\frac{1}{j-i+1}\right) \sum_{k=1}^{j} \sum_{l=i}^{k} A_{lk}. \tag{5}$$

This cost function computes the sum of the average self-similarity of each block in the segment to all other blocks in the segment. While a normalization by the square of the segment length $j - i + 1$ would give the true average, this would severely impede the influence of new segments with larger self-similarity in a large segment, since the large values would be normalized by a relatively large segment length.

4.2. Cost of segment split

Let $i_1 j_1, i_2 j_2, \ldots, i_K j_K$ be a segmentation into K segments, where $i_1 = 1$, $i_2 = j_1 + 1$, $i_3 = j_2 + 1, \ldots, j_K = N$. The total cost of this segmentation is the sum of segment costs plus an additional cost, which is a fixed cost for a new segment,

$$E = \sum_{k=1}^{K} \{\alpha + c(i_n, j_n)\}. \tag{6}$$

By increasing α, the number of resulting segments is decreased. The appropriate value of α is found by optimizing the matching of automatic and manual segment splits.

4.3. Shortest path

In order to compute a best possible segmentation, an edge-weighted directed graph $G = (V, E)$ is constructed. The set of nodes is $V = 1, 2, \ldots, N + 1$. For each possible segment ij, where $1 \leq i \leq j \leq N$, an edge $i, j + 1$ exists in E. The weight of the edge $i, j + 1$ is $\alpha + c(i, j)$. A path in G from node 1 to node $N + 1$ corresponds to a complete segmentation, where each edge identifies 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 $O(|V| + |E|) = O(N^2)$, since G is acyclic and has $|E| = O(N^2)$ edges [20]. An illustration of the directed acyclic graph for a short sequence is shown in Figure 5.

4.4. Function of split cost

The segment split cost (α) of the segmentation algorithm is analyzed here. What is interesting is mainly to investigate whether the total cost of a segmentation (6) has a local minimum. Unfortunately, this is not the case. The total cost is very small for small α and it increases with α. This is clear, as a new segmentation (with one less segment) is chosen (for an

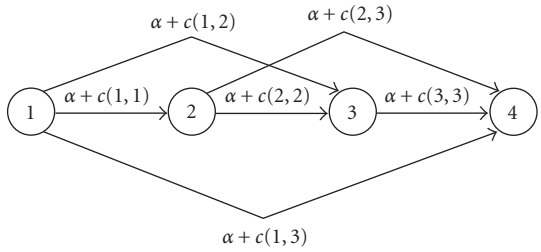

Figure 5: Example of a directed acyclic graph with three segments.

increased α) once the cost of the new segmentation is equal to the original segmentation. The new segmentation cost is now increased with α, until yet another segmentation is chosen at equal cost.

Another interesting parameter is the total number of segments. It is plausible that the segmentation system is to be used in a situation where a given number of segments is wanted. This number decreases with the segment cost, as expected. Experiments with a large number of songs show that the number of segments for a given α is comprised between the half and the double of a median number of segments for most songs.

5. EXPERIMENTS

The segmentation system is now complete. It consists of three different features (the *rhythmogram*, *timbregram*, and *chromagram*), a selfsimilarity measure, and finally the segmentation based on a shortest path algorithm. Two things are interesting in the evaluation of the automatic segmentation system. The first is how the automatic segmentation using the different features actually compare to how humans would segment the music. The second one is whether the different features identify the same segmentation points. In order to test the result, a database on rhythmic music has been collected and manually marked. This database is used here. Three different databases have been segmented manually by three different persons, and segmented automatically using the rhythmic, the timbral, and the harmonic feature. The segmentation points are then matched, and the performance of the segmentation is calculated. No cross-validation has been performed between the subjects.

5.1. Material

Three different databases have been collected. One, consisting of Chinese music, has been segmented using the Chinese numbered notation system [13]. This music consists of 21 randomly selected popular Chinese songs which come from Chinese Mainland, Taiwan, and Hong Kong. They have a variety in tempo, genre, and style, including pop, rock, lyrical, and folk. This music is mainly from 2004. The second database consists of 13 songs, of mainly electronica and techno, from 2004, and the third database consists of 15 songs, with varying style; alternative rock, ethno pop, pop, and techno. This music is from the 1940s to 2005.

5.2. Manual segmentation

In order to compare the automatic segmentation, the databases of music have been manually segmented by three different persons. Each database has been segmented by one person only. While cross-validation of the manual segmentation could prove useful, the added confusion of the experimental results is believed to confuse the situation. The Chinese pop music was segmented with the aid of a notation system and listening, the other two by listening only. The instructions to the subjects were to try to segment the music according to the assumed structure of popular music, consisting of an intro, chorus, and verse, bridge and outro, with repetitions and omissions, and potentially other segments (solos, variations, etc.). The persons performing the segmentation are professional musicians with a background in jazz and rhythmic music. Standard audio editing software was used (Peak and Audacity on Macintosh). For the total database, there is an average of 13 segments per song (first and third quartile are 9 and 17, resp.). The average length of a segment is 20 seconds.

5.3. Matching

The last step in the segmentation is to compare the manual and the automatic segment splits for different values of the new segment cost (α). To do this, the automatic segmentations are calculated for increasing values of α; a low value induces many segments, while a high value gives few segments. The manual and automatic segment split positions are now matched, if they are closer than a threshold. For each value of α, the relative ratios of matched splits to total number of manual splits and to number of automatic splits (recall, R and precision, P, resp.) are found, and the distance to the optimal result is minimized:

$$d(\alpha) = \sqrt{\left(1 - P(\alpha)\right)^2 + \left(1 - R(\alpha)\right)^2}. \tag{7}$$

Since this distance is not common in information retrieval, it is used for matching only here. In the rest of the text, the recall and precision measures, and the weighted sum of these, F_1, are used.

The threshold for identifying a match is important for the matching result. A too short threshold will make the correct, but slightly misplaced segment point unmatched. An analysis of the number of correct matched manual splits shows that it decreases from between 10-11 to approximately 9 when the matching threshold decreases from 5 seconds to 1 second. The number of automatic splits increases significantly, from between 15–17 to 86 (rhythmogram), 27 (timbregram), and 88 (chromagram). The performance of the matching, as a function of the matching threshold is shown in Figure 6. The performance, measured as F_1, increases with the threshold, mainly because the number of automatic splits decreases. While no asymptotic behavior can be detected for threshold values up to 10 seconds, a flattening of the F_1 increase seems to occur at a threshold of between 3-4 seconds. 4 seconds would also permit the subsequent identification of the first beat of the correct measure for tempos up to 60 B/min.

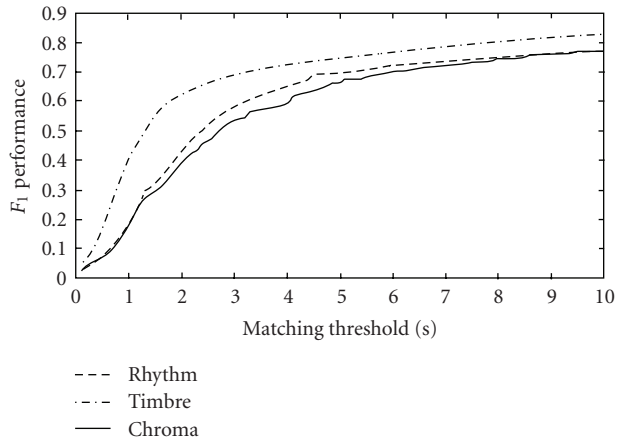

Figure 6: Mean of F_1 performance as a function of the matching threshold for 49 songs.

Table 1: F_1 of the total database for comparison between the segmentation using the *rhythmogram*, *timbregram*, and *chromagram*.

Feature	*Rhythmoghram*	*Timbregram*	*Chromagram*
Rhythmoghram	1.00	—	—
Timbregram	0.60	1.00	—
Chromagram	0.55	0.60	1.00

It is, therefore, used as the matching threshold in the experiments.

5.4. Comparison between features

A priori the rhythmic, timbre, and chroma features should produce approximately the same segmentations. In order to verify this, the distance between the three features has been calculated for all the songs. This has been done for $\alpha = 5.8$, 1.3, and 6.2 for rhythm, timbre, and chroma, respectively. These are the mean values found in the task of optimizing the automatic splits to the manual splits in the next section. The features generally match well. Only a handful of songs has a perfect match. The F_1 performance measure for matching the automatic splits using the three different features are shown in Table 1. An F_1 value of 0.6 corresponds approximately to a recall and performance value of between 50–70%. If the comparison between features are done by selecting an α value that renders a fixed number of splits (for instance the same number as the manual segmentation), the F_1 value increases approximately by 3%.

This still hides some discrepancies, however, as some songs have rather different segmentations for the different features. One such example for the first minute of *The Marriage*[1] is shown in Figure 7. The rhythm only have two segment splits (at 13 and 37 seconds) in the first minute, when the bass-rhythm starts and another when the drums join in. The timbre has one additional split at 21 seconds, start of singing, and another, just before one minute. The chroma have the same splits as the timbre, although the split is earlier, at 24 seconds, seemingly because of the slide guitar changing note.

[1] *The Marriage of Hat and Boots* by August Engkilde presents Electronic Panorama Orchestra (Popscape 2004).

Table 2: F_1 of the three databases for the segmentation using the *rhythmogram*, *timbregram*, and *chromagram*.

Database	*Rhythmoghram*	*Timbregram*	*Chromagram*
Chinese pop	0.70	0.75	0.66
Electronica	0.74	0.77	0.66
Varied	0.68	0.74	0.71
Total	0.70	0.75	0.68
Total with fixed α	0.61	0.67	0.56

5.5. Comparison with manual segmentation

In this section, the match between the automatic and manual segmentations is investigated. For the full database, the rhythm has an average of 10.04 matched splits of 13.39 manual (recall = 75%) and of 17.65 automatic splits (precision = 56.9%). $F_1 = 0.7$. The timbre has an average of 10.73 matched splits, of 13.39 manual (recall = 80.2%), and 15.96 automatic splits (precision = 67.3%). $F_1 = 0.75$. The chroma has an average of 10.12 matched splits, of 13.39 manual (recall = 75.6%), and 20.59 automatic splits (precision = 49.2%). $F_1 = 0.68$. The Chinese pop database has an F_1 values of 0.7, 0.75, and 0.66 for rhythm, timbre, and harmony, the electronica 0.74, 0.77, and 0.66, while the varied database has 0.68, 0.74, and 0.71. These results can be seen in Table 2.

These results have been obtained for an optimal alpha value, found using (7). The mean α values for each feature are 5.8, 1.3, and 6.2, for *rhythm, timbre, and chroma*, respectively. The α values are rather invariant with respect to the song, with a first and third quartile always between ± 50%. The mean α is used to separate training and test. The matching performance for the automatic segmentation using the mean α can be seen in Table 2.

The timbre has a better performance in all cases, and it seems that this is the main attribute used when segmenting music. The rhythm has the next best performance results for the Chinese pop and the electronica, indicating that either the music is more rhythmically based, or that the person performed the manual segmentation based on rhythm, while in the varied database the chroma has the second best performance. All in all, the segmentation identifies most of the manual splits correctly, while keeping the false hits down. The features have comparable results. As the shortest path is the optimum solution, given the error criteria, the performance errors are a result of either bad features, or errors in the manual segmentation.

The automatic segmentation has 65% coincidence between the rhythm and timbre feature, 60% between rhythm and chroma, 63% between timbre and chroma, and 52% coincidence between all three segmentations. While [21] finds

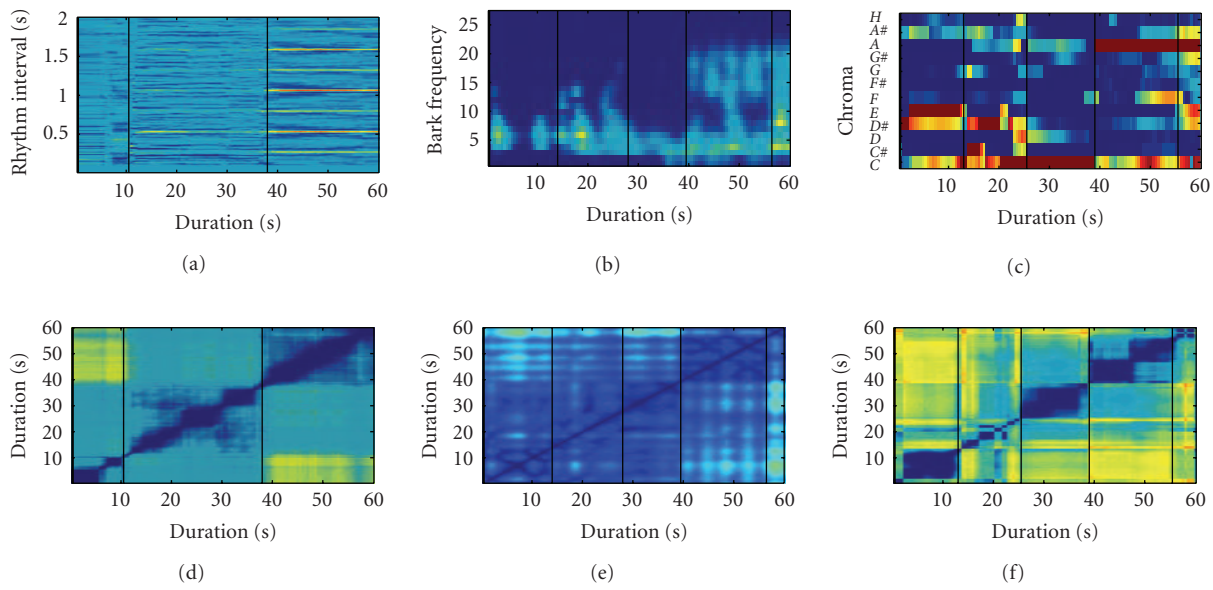

FIGURE 7: Rhythm, timbre and chroma of *The Marriage*. Feature (top) and self-similarity (bottom). The automatic segmentation points are marked with vertical solid lines.

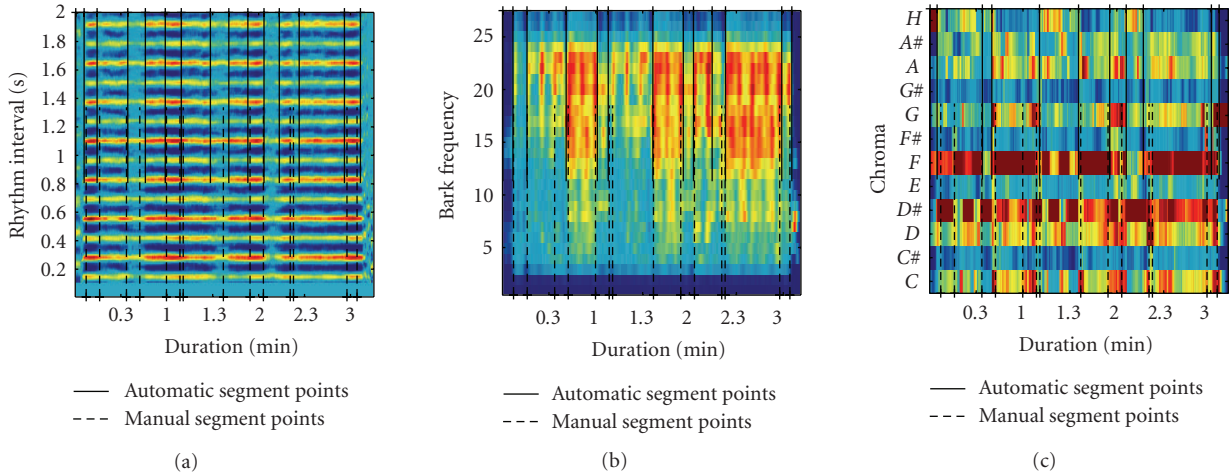

FIGURE 8: (a) Rhythm, (b) timbre, (c) and chroma of *Whenever, Wherever*.

55% correspondence between subjects in a free segmentation task, the results are not easily exploitable because of the short sound files (1 minute). However, since manual segmentation seemingly does not perform better than the matching automatic and manual splits, it is believed that the results presented here are rather good. Indeed, by manual inspection of the automatic and manual segmentation, the automatic segmentation often makes better sense than the manual one, when conflicting.

As an example of the result, the *rhythmogram, timbregram, and chromagram* for *Whenever, Wherever* and *All of You* are shown in Figures 8 and 9, respectively. The manual segmentation is shown in dashed line and the automatic in solid line. The performance for *Whenever, Wherever* is $F_1 = 0.83$, 0.81, and 0.8. Good match on all features. *All of Me* has $F_1 = 0.48$, 0.8, and 0.27. Obviously, in this song, the manual segmentation was made on the timbre only, as it has a significantly better matching score.

6. CONCLUSION

This paper has introduced three features, one associated with rhythm called *rhythmogram*, one associated with timbre called *timbregram*, and one with harmony called *chromagram*. All three features are calculated as an average over time, the *timbregram* and *chromagram* using a novel smoothing based on the Gaussian window. The three features are used to calculate the selfsimilarity. The feature and the selfsimilarity

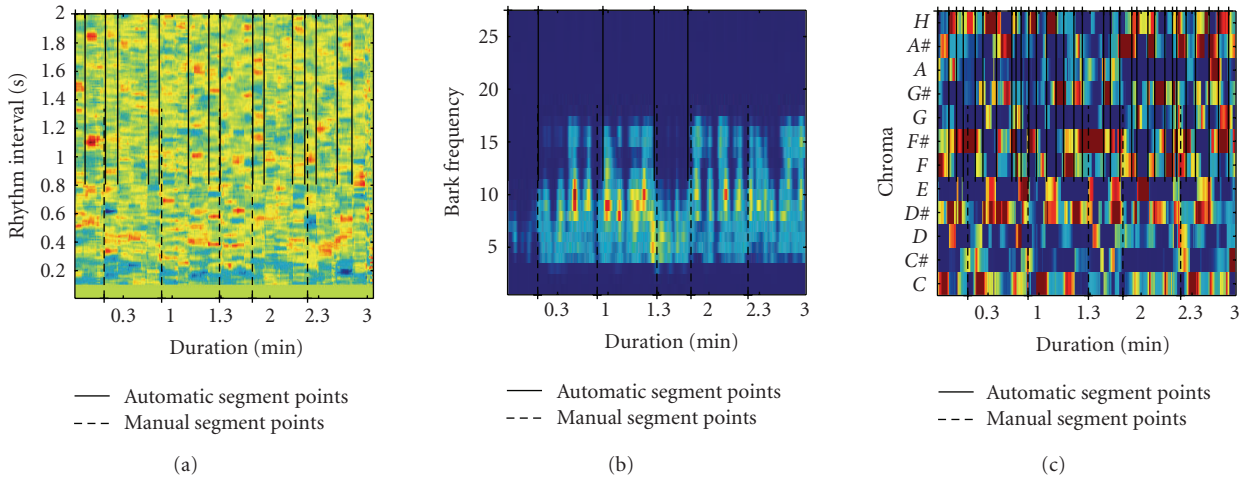

Figure 9: (a) Rhythm, (b) timbre, and (c) chroma of *All of Me*.

are excellent candidates for visualizing the primary attributes of music; rhythm, timbre, and harmony. The songs are segmented using a shortest path algorithm based on a model of the cost of one segment and the segment split. The variable cost of the segment split makes it possible to choose the scale of segmentation, either fine, which creates many segments of short length, or coarse, which creates a few long segment. The rhythm, timbre, and chroma create approximately the same number of segments at the same locations in most of the cases. The matching performances (F_1), when compared to the manual segmentations are 0.7, 0.75, and 0.68 for rhythm, timbre, and chroma, giving indications that the timbre is the main feature for the task of segmenting music manually. This decreases 10% when separating training and test data, but it is always better than how the automatic segmentation compares between features. The automatic segmentation is considered to provide an excellent performance, giving how it is dependent on the music, the person performing the segmentation, or the tools used. The features and the segmentation can be used for audio thumbnailing, making a preview, for use in intelligent music scrolling, or in music recomposition.

REFERENCES

[1] T. H. Andersen, "Mixxx: towards novel dj interfaces," in *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME '03)*, pp. 30–35, Montreal, Quebec, Canada, May 2003.

[2] D. Murphy, "Pattern play," in *Additional Proceedings of the 2nd International Conference on Music and Artificial Intelligence*, A. Smaill, Ed., Edinburgh, Scotland, September 2002.

[3] M. A. Bartsch and G. H. Wakefield, "To catch a chorus: using chroma-based representations for audio thumbnailing," in *Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 15–18, New Paltz, NY, USA, October 2001.

[4] J. Foote, "Visualizing music and audio using self-similarity," in *Proceedings of the 7th ACM International Multimedia Conference & Exhibition*, pp. 77–80, Orlando, Fla, USA, November 1999.

[5] J. Foote, "Automatic audio segmentation using a measure of audio novelty," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME '00)*, vol. 1, pp. 452–455, New York, NY, USA, July-August 2000.

[6] M. Cooper and J. Foote, "Summarizing popular music via structural similarity analysis," in *Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA '03)*, pp. 127–130, New Paltz, NY, USA, October 2003.

[7] K. Jensen, "A causal rhythm grouping," in *Proceedings of 2nd International Symposium on Computer Music Modeling and Retrieval (CMMR '04)*, vol. 3310 of *Lecture Notes in Computer Science*, pp. 83–95, 2005.

[8] G. Peeters and X. Rodet, "Signal-based music structure discovery for music audio summary generation," in *Proceedings of International Computer Music Conference (ICMC '03)*, pp. 15–22, Singapore, Octobre 2003.

[9] R. B. Dannenberg and N. Hu, "Pattern discovery techniques for music audio," *Journal of New Music Research*, vol. 32, no. 2, pp. 153–163, 2003.

[10] M. Goto, "A chorus-section detecting method for musical audio signals," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '03)*, vol. 5, pp. 437–440, Hong Kong, April 2003.

[11] S. Dubnov, G. Assayag, and R. El-Yaniv, "Universal classification applied to musical sequences," in *Proceedings of the International Computer Music Conference (ICMC '98)*, pp. 332–340, Ann Arbor, Mich, USA, October 1998.

[12] T. Jehan, "Hierarchical multi-class self similarities," in *Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA '05)*, pp. 311–314, New Paltz, NY, USA, October 2005.

[13] K. Jensen, J. Xu, and M. Zachariasen, "Rhythm-based segmentation of popular chinese music," in *Proceedings of 6th International Conference on Music Information Retrieval (ISMIR '05)*, pp. 374–380, London, UK, September 2005.

[14] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," *Journal of the Acoustical Society of America*, vol. 87, no. 4, pp. 1738–1752, 1990.

[15] K. Jensen, "Perceptual atomic noise," in *Proceedings of the International Computer Music Conference (ICMC '05)*, pp. 668–671, Barcelona, Spain, September 2005.

[16] N. Collins, "A comparison of sound onset detection algorithms with emphasis on psychoacoustically motivated detection functions," in *Proceedings of AES 118th Convention*, Barcelona, Spain, May 2005.

[17] P. Desain, "A (de)composable theory of rhythm," *Music Perception*, vol. 9, no. 4, pp. 439–454, 1992.

[18] A. Sekey and B. A. Hanson, "Improved 1-bark bandwidth auditory filter," *Journal of the Acoustical Society of America*, vol. 75, no. 6, pp. 1902–1904, 1984.

[19] J. P. Eckmann, S. O. Kamphorst, and D. Ruelle, "Recurrence plots of dynamical systems," *Europhysics Letters*, vol. 4, no. 9, pp. 973–977, 1987.

[20] T. H. Cormen, C. Stein, R. L. Rivest, and C. E. Leiserson, *Introduction to Algorithms*, The MIT Press, Cambridge, UK; McGraw-Hill, New York, NY, USA, 2nd edition, 2001.

[21] G. Tzanetakis and P. Cook, "Multifeature audio segmentation for browsing and annotation," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA '99)*, pp. 103–106, New Paltz, NY, USA, October 1999.

Kristoffer Jensen obtained his Masters degree in 1988 in computer science from the Technical University of Lund, Sweden, and a D.E.A in signal processing in 1989 from the ENSEEIHT, Toulouse, France. His Ph.D. was delivered and defended in 1999 at the Department of Computer Science, University of Copenhagen, Denmark, treating signal processing applied to music with a physical and perceptual point of view. This mainly involved classification and modeling of musical sounds. He has been involved in synthesizers for children, state-of-the-art next generation effect processors, and signal processing in music informatics. His current research topic is signal processing with musical applications, and related fields, including perception, psychoacoustics, physical models, and expression of music. He currently holds a position at the Software and Media Technology Department, Aalborg University Esbjerg as Associate Professor.