

AN INTEGRATED APPROACH TO MUSIC BOUNDARY DETECTION

Min-Yian Su, Yi-Hsuan Yang, Yu-Ching Lin, Homer Chen

National Taiwan University

sui751004@gmail.com, affige@gmail.com, vagante@gmail.com, homer@cc.ee.ntu.edu.tw

ABSTRACT

Music boundary detection is a fundamental step of music analysis and summarization. Existing works use either unsupervised or supervised methodologies to detect boundary. In this paper, we propose an integrated approach that takes advantage of both methodologies. In particular, a graph-theoretic approach is proposed to fuse the results of an unsupervised model and a supervised one by the knowledge of the typical length of a music section. To further improve accuracy, a number of novel mid-level features are developed and incorporated to the boundary detection framework. Evaluation result on the RWC dataset shows the effectiveness of the proposed approach.

1. INTRODUCTION

Popular songs usually comprise several music sections such as intro, verse, chorus, bridge and outro. A music boundary is the time point where a section transits to another. Identifying such boundaries is important because it allows us to divide a song into semantically meaningful sections. This information can also be applied to music summarization [1] and thumbnailing [2] to facilitate music browsing and structure-aware playback [3]. Boundary detection also serves as a front-end processor for music content analysis since it provides a local description of each section rather than a global but coarse representation of the whole song [5].

Although there is a rich literature in music theory about music structure analysis for symbolic music (e.g. [20]), music boundary detection for music signals is still a challenging task because precise pitch detection in polyphonic music is not yet achievable. Under this condition, most work on music boundary detection utilizes the similarity between short-term (e.g., 23ms) audio frames within a song to identify the repetitive parts and divide a song into a number of sections [1–3, 6–8]. A more recent work formulates boundary detection as a clustering problem and considers that the audio frames of each cluster belong to the same music section [9].

The accuracy of this *unsupervised* approach, however, may be limited because only the information of a song itself is exploited. For example, identifying repetitive parts cannot correctly identify the boundary between two adjacent music sections that always occur successively in

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page.

© 2009 International Society for Music Information Retrieval

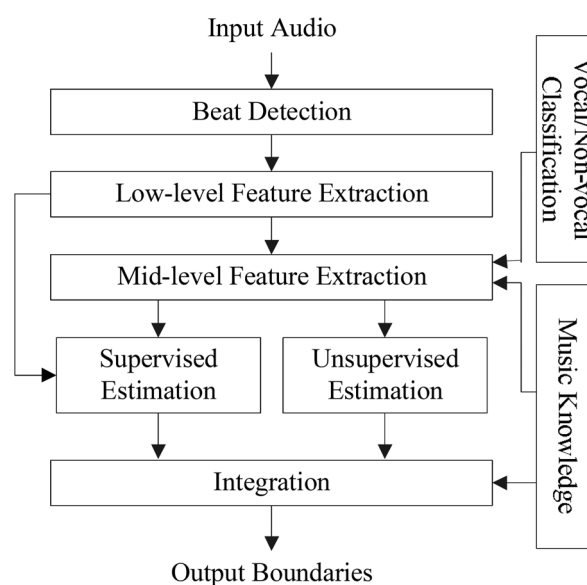


Figure 1. A schematic diagram of the proposed music boundary detection system.

a song. On the other hand, clustering-based methods tend to produce over-segmented results if the acoustic property of the frames in a music section varies greatly. Using histograms to gather statistic of spectral characteristics of neighboring audio frames [9] does not necessarily solve the problem because the histograms of two adjacent frames are usually similar, making boundary detection even more difficult.

To address the aforementioned drawbacks, Turnbull *et al* formulate music boundary detection as a supervised problem and train a binary classifier to classify whether a time point is a boundary or not [10]. In this way, we can mine more information from a large number of training songs and identify features that are relevant to boundary detection.

However, because a supervised system is pre-trained by using the training data and fixed afterwards, it is not as adaptive to test songs as its unsupervised counterpart. The detection accuracy may significantly degrade when the characteristics of the training data and a test song are considerably different. For instance, if the system detects boundary according to the energy level in a certain frequency range, the system may not work for a song whose energy in that frequency range maintains high throughout the song.

Based on the above observations, we propose to take advantage of both methodologies by aggregating the results of an unsupervised model and a supervised one. In this way, we can exploit the discriminative information provided by the training data and the song-specific

information of a test song at the same time. Moreover, to better capture the discriminative characteristics of a boundary, we further propose a number of novel *mid-level* features, including novelty score, dissonance level and vocal occurrence. Comparing to low-level features such as the spectral properties, these mid-level features carry more semantic meaning that improves music boundary detection.

A schematic diagram of the proposed system is shown in Fig. 1. An input song is partitioned by the beat onsets and represented by a set of low-level and mid-level features. The probability of each beat onset of being a boundary is then computed by both supervised and unsupervised methods with the features extracted from the subsequent beat interval. We then model the beat onsets as the vertices of a directed graph, with the vertex weights determined by the probability of being a boundary and the edge weights determined based on the music knowledge of the typical length of a music section [7, 11]. Finally, we formulate music boundary detection as a shortest path problem and identify the true boundaries by the Viterbi algorithm [18].

The paper is organized as follows. Section 2 describes the feature representation of music, including low-level and mid-level features. Section 3 elaborates on the system framework and the adopted supervised and unsupervised approaches. Experimental result is presented in Section 4. Section 5 concludes the paper.

2. MUSICAL REPRESENTATION

Before feature extraction, each song is converted to a standard format (mono channel and 22,050 Hz sampling rate) and partitioned into several beat intervals by the beat onset detection algorithm BeatRoot [12]. We adopt beat interval instead of frame as the basic time unit because the characteristics of a song are more likely to be consistent within a beat interval and because a music boundary tends to occur at a beat onset [7].

2.1 Low-level Features

For low-level local features, we use 40-dim Mel-scale cepstral coefficients (MFCCs), 24-dim chromagram, and 52-dim fluctuation patterns (FPs) [19] to represent the timbre, harmony, and rhythm aspects of music. We extract MFCCs and chromagram with a 40ms and non-overlapping sliding window and aggregate the frame-level features within each beat interval by taking the mean and the standard deviation. FPs are computed directly for each beat interval. These features have been found useful for music boundary detection [10]. Note these features only capture the local property of music.

2.2 Mid-level Features

Below we describe three mid-level features: novelty score, dissonance level, and vocal occurrence. While the first one is originally proposed by Cooper *et al* in [4], it has been used in an unsupervised setting rather than as a mid-level feature in a supervised one. On the other hand, though the latter two features have been studied in the context of music theory [21], few attempts have been

made to incorporate them to the task of music boundary detection for raw audio signals.

2.2.1 Novelty Score

The novelty score is computed by two steps [4]. First, a similarity matrix is constructed by measuring the similarity of the low-level feature vectors of every two beats in a song. In this matrix, the two segments beside the boundary produce two adjacent square regions of high within-segment similarity along the main diagonal and two rectangular regions of low between-segment similarity off the main diagonal. As a result, each boundary produces a *checkerboard* pattern in the matrix and the beat interval that boundary occurs is the crux of this checkerboard. To identify these patterns, we correlate a Gaussian-tapered checkerboard kernel along the main diagonal of the similarity matrix to compute the so-called novelty scores, which measures both the dissimilarity between two different adjacent segments beside each potential boundary as well as the similarity within these segments. We define the term *segment* here to represent a set of consecutive beat intervals and the term *section* as a segment which is semantically meaningful (such as verse, chorus or bridge).¹

In this work, we compute three novelty scores based on the three low-level features. Because the novelty scores of adjacent beats tend to be similar,² we also divide the novelty score of a certain beat interval by the sum of the novelty scores of neighboring beat intervals and use the normalized score as additional feature, resulting in a total of 6 features for each beat interval.

2.2.2 Dissonance Level

It is known in musicology that the relaxation or release of tension plays an important role in the transition of music sections. Because changes in tension often occur when dissonance giving way to consonance [13], we develop a novel feature based on the dissonance level of music. We first define the *dissonant intervals* according to the relationship between the pitches of two notes that cause tension (e.g., Tritone and Minor Second [14]), and then compute the dissonance level as the weighted sum of the corresponding dissonant intervals from the unwrapped chromagram of a beat,

$$y_t = \frac{\sum_{q \in D} \sum_m k_q c_m c_{m+q}}{\sum_m c_m}, \quad (1)$$

where y_t denotes the dissonance level of a beat t , q denotes the interval that has q semitones between the two notes, D is the set of dissonant intervals, c_m is the m th bin of the chromagram, and k_q is a constant corresponding to q , which is empirically set according to the ratio of frequencies of the two pitches in q . The denominator is a normalization term.

¹ While a segment can be of arbitrary length, the length of a section often follows a typical pattern, see Section 3.3.

² The novelty scores of adjacent beats are similar because the submatrices of the similarity matrix of these beats overlap a lot.

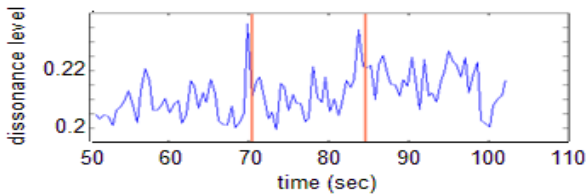


Figure 2. The dissonance level of a part of Billie Jean by Michael Jackson. The two red lines label a transition from verse to bridge and a transition from bridge to chorus. These boundaries occur right after high dissonance levels.

We compute the dissonance level for each beat interval and obtain a sequence of dissonance levels. We compute the derivative from the resulting sequence as the dissonant features to capture the changes in tension,

$$\Delta y_t = \frac{\sum_{m=-p}^p m y_{t-m}}{\sum_{m=-p}^p m^2}, \quad (2)$$

where p denotes the window size. In this work, we set p to 1 and 2 and generate a two dimensional dissonance level feature. Fig. 2 illustrates the relationship between music boundary and dissonance level; clearly the music boundaries occur right after peaks of dissonance level (the rise and relax of tension).

2.2.3 Vocal Occurrence

In pop/rock songs, the time points that a vocalist sings often correspond to the music boundaries. For example, if a beat onset falls in the middle of a segment with pure instrument and another segment with singing voice, it is very likely a music boundary. Furthermore, because a music section is comprised of several music phrases,³ a transition of music sections must also be a transition of music phrases. Therefore, if a beat onset falls in a short instrumental interval between two vocal music phrases, it is more likely to be a music boundary.

In light of the above observation, we train a vocal/non-vocal classifier by support vector machine (SVM) [15], with MFCC as the feature representation, to estimate the probability of the vocal occurrence for each beat interval. If the sum of these probabilities from the beat intervals in a segment exceeds a threshold, we regard the segment as a *vocal segment*. More specifically, the vocal occurrence feature of a certain beat interval is computed as follows. For a beat interval, if both of its neighboring segments are non-vocal, the vocal occurrence is set to 0; if only one of the neighboring segments is non-vocal, the vocal occurrence is set to 1. When both neighboring segments are vocal, we set the vocal occurrence according to the following formula:

$$z_t = \frac{1 - v_t}{2w + 1} \sum_{j=t-w, j \neq t}^{t+w} v_j, \quad (3)$$

where z_t is the vocal occurrence feature of beat interval t ,

³ Several music phrases constitute a music section.

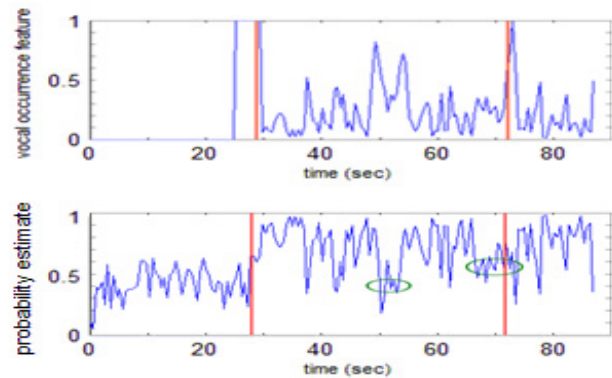


Figure 3. Top: the possibility of vocal estimated by SVM for a part of Billie Jean by Michael Jackson. The two red lines label a transition from intro to verse and a transition from verse to bridge. The green circles label two obvious transition points of music phrases. Bottom: corresponding vocal occurrence feature.

v_t is the probability estimate of beat interval t generated by the vocal/non-vocal SVM classifier, and w is the window size that represents the length of the segment. We vary the value of w and generate a multi-dimensional feature vector. In this work we set the value of w to 8 and 12. An illustrative example is shown in Fig. 3. The first red line labels a transition from a non-vocal section (intro) to a vocal section (verse). The green circles label two obvious transition points of music phrases, while the latter one is in fact a transition point of music sections. We can see the corresponding vocal occurrence feature is highly correlated to music boundaries. A pitfall of this feature is that it may regard every phrase boundary as a section boundary and result in over segmentation. The use of other features may offset this mistake.

Representing the acoustic properties of music by these low-level and mid-level features, we then employ the system described below to detect boundaries.

3. SYSTEM DESCRIPTION

In this section, we first introduce the supervised and unsupervised approaches adopted in our system. Both approaches estimate the possibility of each beat onset of being a music boundary. Second, we describe how we integrate these two estimations with the music knowledge of typical section length.

3.1 Supervised Estimation

We train a SVM classifier with polynomial kernel and probability estimates to obtain the possibility of a beat onset being a music boundary. The label for a beat interval is marked 1 if a boundary occurs at that beat onset and 0 otherwise. Besides mid-level features, we also use the low-level features to train the classifier because low-level features also contain some relevant information. For example, a drum-fill is usually played when a music section ends; this characteristic can be detected by FP. For a test song, the SVM model

computes the probability of the occurrence of a boundary at every beat onset. We utilize this probability as the output of the supervised approach.

3.2 Unsupervised Estimation

As for the unsupervised part, we construct three similarity matrices based on the kinds of low-level features and detect the peaks of the mean of the novelty scores from these matrices. We then use these peaks to divide the test song into a number of segments [4]. The low-level features of a segment are integrated to one vector by taking the mean and the standard deviation and a distance matrix among the segments is constructed by computing the pairwise distance between these vectors. The normalized cut algorithm [16] is then performed on the distance matrix to group these segments into acoustic similar clusters. At each beat interval, we further count the cluster indices of neighboring beat intervals within a predefined window size and establish two histograms: one for the beat intervals preceding to the beat onset, and the other for the subsequent beat intervals. The Euclidean distance of the resulting histograms can represent the possibility of a music boundary occurs at the designated beat onset, and the ratio of this possibility value of a beat onset to the sum of the possibility values of its neighboring ones is regarded as the estimation of the unsupervised approach.

3.3 Integration

Because music sections tend to have some typical length (e.g., 8 or 16 bars) [7, 11], it should be beneficial to incorporate this knowledge to the music boundary detection framework. As Fig. 4 illustrates, we construct a directed graph $G = (V, E)$ to integrate the estimates of supervised and unsupervised models and to take advantages of this music knowledge. In this graph, a vertex represents a beat onset, with the weight of it determined by the weighted sum of the estimates of supervised and unsupervised models

$$w_{v_i} = p_{u_i} + k_1 p_{s_i}, \quad (4)$$

where w_{v_i} denotes the weight of a vertex i , p_{u_i} and p_{s_i} are the probability estimates produced by an unsupervised model and a supervised one respectively, and k_1 is a parameter balancing the effect of the two models. The music knowledge of section length is incorporated as follows. If there exists the possibility that vertices v_i and v_j are two successive music boundaries, we form an edge between these two vertices. The weight of the edge is determined by the music knowledge of the length of a music section. We gather the statistics from training data to obtain the probability of two beats with specific temporal distance being music boundaries. That is, the weight of e_{ij} equals to the weight of e_{mn} if $j-i$ equals to $n-m$. To achieve this goal, a histogram is constructed by simply counting the number of beats of each music section from the training data.

Therefore, a path in this constructed graph can be regarded as a set of music boundaries. We further define the weight of a path B as the sum of the weights of its constituent edges and vertices,

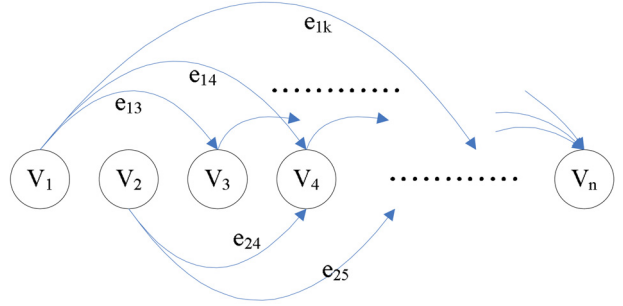


Figure 4. The directed graph G of a song, which has n beat onsets (vertices) and $k-1$ possible section lengths (possible jumps). The vertex weights are determined by the probability of being a boundary and the edge weights are determined based on the music knowledge of the typical length of a music section [7, 11]. We assume that every music section contains at least one beat interval.

$$w_B = \sum_{v \in B} w_v + k_2 \sum_{e \in B} w_e, \quad (5)$$

where w_v and w_e are the weights of a vertex and an edge in B , and k_2 is a constant to balance the effects of vertices and edges. We regard w_B as the probability of the associated beat onsets being correct music boundaries.

Because the path with maximum w_B consists of vertices that are most likely the music boundaries, we formulate the problem as a shortest path problem and employ the Viterbi algorithm [18] to solve it,

$$B^* = \arg \max_B w_B, \quad (6)$$

where B^* denotes the optimal solution. In practice, we only apply Viterbi to a feasible number of paths to reduce the complexity.

4. EXPERIMENT

4.1 Experimental Setup

We conduct an empirical evaluation on the RWC music dataset [17], which contains 100 pieces of song that are originally produced for experiment; most of the pieces (80%) are recorded according to 1990s Japanese chart music, while the rest resemble the 1980s American chart music. RWC dataset provides clear annotations of music boundaries and is adopted in many literatures in music boundary detection [6, 10].

We evaluate the performance in terms of precision (the proportion of true boundaries among the detected ones), recall (the proportion of true boundaries in the ground truth that are detected by the system), and f-score (the harmonic average of precision and recall). A detected boundary is considered correct if it falls within 1.5 seconds of the ground-truth, which is stricter than the one used in prior work [9] and should be reasonable for real-world applications.

For the unsupervised methods, we process each of the 100 songs independently and take the average result. For the supervised methods, we evaluate the system with

Approach	Method	Precision	Recall	F-score
Supervised only	N +D+V+L	0.2461	0.2932	0.2641
Unsupervised only	Cluster-based (normalized cut) [9]	0.2770	0.5166	0.3517
	Histogram-based	0.3068	0.3428	0.3124
Directly sum		0.3274	0.3470	0.3385
Integrated with section length	Viterbi algorithm [18]	0.3800	0.4452	0.4094

Table 2. Evaluation result of different musical boundary detection methods.

Feature	# feature	Precision	Recall	F-score
MFCC	40	0.1910	0.2574	0.2142
chromagram	24	0.1665	0.2131	0.1842
fluct. pattern	52	0.1906	0.2190	0.2019
local (L)	116	0.1982	0.2629	0.2206
difference [10]	6	0.1602	0.2519	0.1885
novelty (N)	6	0.2427	0.2770	0.2549
dissonance (D)	2	0.2109	0.2505	0.2198
vocal (V)	2	0.2128	0.2687	0.2240
N+L	122	0.2354	0.2900	0.2594
N+D+V	10	0.2322	0.2909	0.2592
N +D+V+L	126	0.2461	0.2932	0.2641

Table 1. Evaluation result of different features used in supervised musical boundary detection methods.

stratified five-fold cross validation: 20 random songs are held out as test data and the rest are used for training. The evaluation is iterated five times to get the average result.

4.2 Results

We first evaluate the supervised approach with different feature representations, including low-level and mid-level features. To compare the performance against previous work, we also implement the difference feature and its derivative proposed in [10]. The difference feature is computed by sliding a window along the audio signal and comparing the statistic of low-level features in the first half of the window with the ones in the second half. A beat onset is detected as a boundary if its probability estimate assigned by SVM exceeds a threshold. Instead of using a fixed threshold, we adaptively set the threshold of each song to be the mean plus one standard deviation of the probability estimates of the song.

The evaluation result is shown in Table 1. The three low-level features bring about similar accuracy, with FPs slightly worse than the other two, implying that the characteristics of music boundaries are represented more in timbre and rhythm. The direct concatenation of the three low-level features, which are denoted as local (L) in the table, further improves the f-score to 0.2206.

We then compare four mid-level features, including the difference feature proposed in [10]. It can be found that, with much lower feature dimension, the use of mid-level features achieves similar or superior performance to that attained by low-level features. The novelty score, in particular, achieve an f-score of 0.2549 that significantly outperform all other low-level or mid-level features. We can also find that the difference feature does not perform

well, which possibly due to the disregard of the similarity of the beats in each segment.

The combination of mid-level and low-level features only brings about slight improvement, which somewhat implies that most of the information carried by low-level features has already been well represented by the mid-level features. The combination of novelty score (N), dissonance level (D), vocal occurrence (V), and local features (L) achieves the highest f-score of 0.2641.

We then compare the two unsupervised methods described in Section 3.2. For the cluster-based method, we simply mark the boundary of two consecutive segments that are associated with different clusters as a music boundary without smoothing. The result is shown in Table 2. As expected, the clustering-based approach exhibits a remarkably high recall but a relatively low precision. For the histogram-based method, we consider the segments whose probability estimates exceed a threshold as boundaries. The threshold value is set in the same way as in the supervised methods. The performance of the histogram-based method is slightly worse than the clustering-based one, showing that gathering statistics of neighboring frames does not improve the precision of boundary detection. Moreover, it can be noted that in our evaluation the unsupervised approaches generally outperform the supervised counterparts, showing that the ability of the unsupervised approach to be adaptive to each test song is essential in boundary detection.

Finally, we evaluate the performance of integrating the result of unsupervised and supervised methodologies. For comparison, we further implement a baseline method that simply sums up the supervised and unsupervised estimates with the same weight as the one in proposed graph-theoretical fusion method without exploiting the music knowledge of section length.

The result is also shown in Table 2. It can be found that simply taking the average has achieved a higher f-score than any of the supervised-only or unsupervised one, showing that the two methodologies are indeed complementary and the fusion of them is plausible. The proposed graph-theoretical fusion further improves the f-score to 0.4094, which greatly outperform the taking average baseline, especially in recall. This result shows the integration of the two methodologies and the incorporation of music knowledge are essential to music boundary detection.

A sample segmentation result is displayed in Fig. 5. In this example, all the boundaries can be correctly detected by the proposed system. Nevertheless, there is an over segmentation problem because the characteristics of the segments of the same music section may be incoherent.

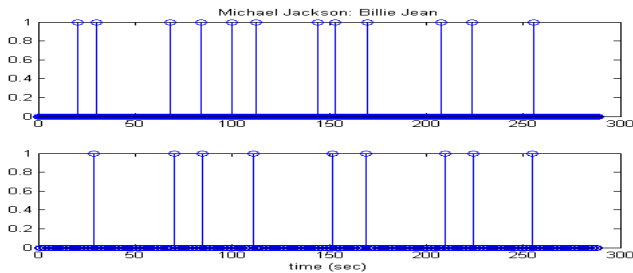


Figure 5. The segmentation result of Billie Jean by Michael Jackson. Top: the boundaries detected by the proposed system. Bottom: the manual annotation.

To resolve this problem, we are working on incorporating more music knowledge and mid-level features.

5. CONCLUSION

In this paper, we have presented an integrated system that combines the information from supervised approach, unsupervised approaches, and music knowledge. We formulate music boundary detection as a shortest path problem and employ the Viterbi algorithm to solve it. We also propose a number of novel mid-level features to better capture the discriminative characteristics of music boundaries. Experiments conducted on the RWC dataset show significant improvement over the state-of-the-art supervised-only and unsupervised-only methods.

6. ACKNOWLEDGEMENT

This work was supported by the National Science Council of Taiwan under the contract number NSC 97-2221-E-002-111-MY3. The authors would like to thank the anonymous reviewers for valuable comments that greatly improved the quality of this paper.

7. REFERENCES

- [1] W. Chai, "Semantic segmentation and summarization of music," in *IEEE Signal Processing Magazine*, Vol. 23, No. 2, pp. 124–132, 2006.
- [2] M. Levy, M. Sandler, and M. Casey, "Extraction of high-level musical structure from audio data and its application to thumbnail generation," in *Proc. ICASSP*, pp. 1433–1436, 2006.
- [3] M. Goto, "A chorus-section detection method for musical audio signals and its application to a music listening station," in *IEEE Trans. Audio, Speech and Language Processing*, Vol.14, No.5, pp. 1783–1784, 2006.
- [4] M. Cooper and J. Foote, "Summarizing popular music via structural similarity analysis," in *Proc. IEEE Workshop Applications of Signal Processing to Audio and Acoustics*, pp. 127–130, 2003.
- [5] M. Casey, R. Veltkamp, M. Goto, M. Leman, C. Rhodes, and M. Slaney, "Content-based music information retrieval: current directions and future challenges," in *Proceedings of the IEEE*, Vol. 96, No. 4, pp. 668–696, 2008.
- [6] J. Paulus and A. Klapuri, "Music structure analysis using a probabilistic fitness measure and a greedy search algorithm," in *IEEE Trans. Audio, Speech and Language Processing*, Vol. 17, No. 6, pp. 1159–1170, 2009.
- [7] N. C. Maddage, C. Xu, M. S. Kankanhalli, and X. Shao, "Content-based music structure analysis with applications to music semantics understanding," in *Proc. ACM Multimedia*, pp. 112–119, 2004.
- [8] L. Lu, M. Wang, and H. Zhang, "Repeating pattern discovery and structure analysis from acoustic music data," in *Proc. ACM SIGMM Int. Workshop on Multimedia Information Retrieval*, 2004.
- [9] M. Levy and M. Sandler, "Structural segmentation of musical audio by constrained clustering," in *IEEE Trans. Audio, Speech and Language Processing*, Vol. 16, No. 2, pp. 318–326, 2008.
- [10] D. Turnbull, G. Lanckriet, E. Pampalk, and M. Goto, "A supervised approach for detecting boundaries in music using difference features and boosting," in *Proc. ISMIR 2007*.
- [11] C. Rhodes et al, "A Markov-chain Monte-Carlo approach to musical audio segmentation," in *Proc. ICASSP*, 2006.
- [12] S. Dixon, "Evaluation of the audio beat tracking system BeatRoot," in *Journal of New Music Research*, Vol, 36, No. 1, pp. 39–50, 2007.
- [13] DeLone et al, *Aspects of Twentieth-Century Music*. Englewood Cliffs, New Jersey: Prentice-Hall, 1975
- [14] Wyatt and Keith, *Harmony & Theory*. Hal Leonard Corporation, pp. 77, 1998.
- [15] N. Maddage et al, "An SVM-based classification approach to musical audio," in *Proc. ISMIR*, 2003.
- [16] Ji. Shi and J. Malik, "Normalized cuts and image segmentation," in *Proc. CVPR*, pp. 731–737, 1997.
- [17] M. Goto, "AIST annotation for RWC music database," in *Proc. ISMIR*, 2006.
- [18] G. D. Fomey, "The Viterbi algorithm," in *Proceedings of the IEEE* Vol. 61, No. 3, pp.268–278, 1973.
- [19] E. Pampalk, "Computational models of music similarity and their application in music information retrieval," in PhD thesis, Vienna University of Technology, 2006.
- [20] F. Lerdahl and R. Jackendoff, "An overview of hierarchical structure in music," in *Machine Models of Music*, pp. 289–312, 1993.
- [21] D. Pressnitzer, S. McAdams, "Two phase effects in roughness perception," in *The Journal of the Acoustical Society of America*, Vol. 105, No. 5, pp.2773–2782, 1999.