TOWARDS EFFICIENT AUDIO THUMBNAILING

Nanzhu Jiang and Meinard Müller

International Audio Laboratories Erlangen
{nanzhu.jiang,meinard.mueller}@audiolabs-erlangen.de

ABSTRACT

Audio thumbnailing, which aims at finding the most representative audio segment of a music recording, is an important task in music information retrieval. In this paper, we show how the computational efficiency of a recently proposed state-of-the-art thumbnailing approach can be improved significantly. The basic idea of the previous approach is to compute for each possible segment a fitness value that expresses repetitiveness and then to define the thumbnail as the fitness-maximizing segment. As a first acceleration strategy, we propose an efficient multi-level sampling strategy to reduce the number of segments the fitness has to be computed for. Second, we obtain further accelerations by suitably adjusting the resolution used in the fitness computation depending on the level of the segment. As a third contribution, we exploit an intrinsic property of the fitness computation that allows us to estimate the fitness for certain segments without any further computation. Our experimental results show that combining these three strategies leads to accelerations by a factor of 20 to 200 depending on the duration of the song while keeping the overall accuracy for the thumbnail estimation.

Index Terms— Audio structure analysis, audio thumbnailing, efficiency

1. INTRODUCTION

The automatic extraction of structural information from audio recordings is a central research topic in the field of music information retrieval [1, 2]. A prominent subproblem is referred to as *audio thumbnailing*, where the objective is to automatically determine the most representative section of a given music recording [3, 4, 5, 6, 7, 8]. Such a representative section may serve as some kind of "preview" which gives a listener a first impression of the song or piece of music. Based on such previews, the user should be enabled to quickly decide if he or she likes to listen to the song or to move on to the next recording. Thus, audio thumbnails are an important browsing and navigation aid for finding interesting pieces in large music collections.

Often sections such as the chorus section or the main theme of a song are good candidates for being suitable audio thumbnails. Such parts are typically repeated several times throughout the recording. Therefore, to determine a thumbnail automatically, most procedures try to identify a section that has on the one hand side a certain minimal duration and on the other side many (approximate) repetitions. In this paper, we build on such a procedure described in [8], where a fitness measure that captures repetitiveness as well as coverage is computed for each possible segment of a given audio recording.

This work has been supported by the German Research Foundation (DFG MU 2682/5-1). The International Audio Laboratories Erlangen are a joint institution of the Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU) and Fraunhofer Institut für Integrierte Schaltungen IIS.

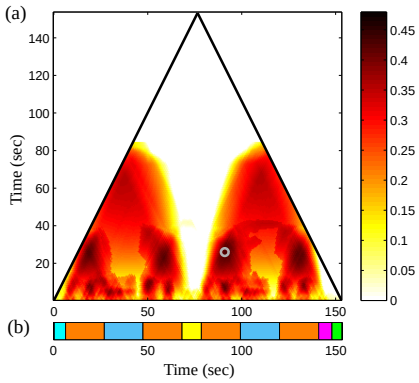

Fig. 1. Thumbnailing procedure for Beatles song "Act Naturally". **(a)** Fitness scape plot with thumbnail segment indicated by the circle point. **(b)** Ground-truth structure annotation.

Then, similar to [5, 9], the audio thumbnail is defined to be the segment having maximal fitness. Furthermore, representing each audio segment by means of its center and length, the fitness values of all segments can be visualized by a *scape plot*, which reveals the repetitive structure of the entire music recording in a hierarchical and compact way [10, 11, 12]. An example is shown in Figure 1, which shows a scape plot for the song "Act Naturally" by the Beatles. The fitness maximizing point, indicated by the circle, corresponds to the verse section, which appears four times in the song.

Even though the procedure described in [8] yields good thumbnailing results representing the state-of-the-art, it has the drawback of being computationally expensive. First, the fitness is computed for all possible segments, the number of which is quadratic in the duration of the song. Second, in computing the fitness of a single segment, the segment is brought into relation to other repeating segments, a process that again requires a quadratic running time. Altogether, this yields a complexity that is proportional to the fourth power in the duration of the song.

As main contribution of this paper, we introduce three different strategies that lead to significant accelerations of the original procedure. As a first strategy, we introduce a hierarchical multi-level approach, where the fitness is first computed on a coarse grid of scape plot points (the points corresponding to audio segments). Then, suitable neighborhoods of only those grid points that have the highest fitness are selected and iteratively refined, which significantly reduces the overall number of fitness computations. The second strategy is to accelerate the actual fitness computation by adjusting the resolution used in deriving the mutual repetition relations of the segments, where the resolution is coupled to the level of the previously described grid sampling approach. As the third strategy, we exploit the mutual relations that are detected in the fitness computation. These

relations express repetitiveness within certain segment families and allow us to estimate the fitness for all these segments in one step without any further computation. Our experiments on two different datasets (Beatles Songs, Chopin Mazurkas) show that each of these strategies lead to significant accelerations that are independent from each other. Using a combined approach, we obtain accelerations by a factor of roughly 20 to 200 (depending on the duration of the song) while keeping the overall accuracy of the thumbnailing procedure.

The remainder of this paper is organized as follows. We describe the three acceleration strategies in Section 2, Section 3, and Section 4, respectively. Then, in Section 5, we report on our systematic experiments and draw some conclusions.

2. ACCELERATION BY MULTI-LEVEL SAMPLING

Before we describe the first acceleration strategy, we need to introduce some notation. Following [8], we assume that the given music recording is represented by a feature sequence with a sampled time axis indexed by $[1 : N] = \{1, 2, \dots N\}$. (In our experiments we use a feature resolution of 2 Hz.) A segment is then understood to be a subset $\alpha = [s : t] \subseteq [1 : N]$ specified by its starting point s and its end point t with $|\alpha| := t - s + 1$ denoting its length. In [8], a fitness measure is used that assigns to each segment α a fitness value $\varphi(\alpha) \in \mathbb{R}$. At the moment, the definition of the fitness measure is not important, and we will have a closer look at it in Section 3. All we need to know at this stage is that the fitness value $\varphi(\alpha)$ expresses the repetitiveness of a segment α and that the thumbnail is defined as the fitness-maximizing segment $\alpha^* := \operatorname{argmax}_{\alpha}(\varphi(\alpha))$.

The fitness values can be visualized by means of a triangular scape plot [10, 11, 12]. Each point of the scape plot corresponds to a segment $\alpha = [s : t]$, where the horizontal coordinate encodes the center $c(\alpha) := (s + t)/2$ of the segment and the vertical coordinate its length $|\alpha|$. The fitness value $\varphi(\alpha)$ is then visualized in some color-coded form, see Figure 1. The fitness scape plot represents the repetitive structure of the music recording in some hierarchical way, see [8] for details.

The first of our acceleration strategies is rather straightforward. Instead of computing the fitness for all possible segments (at the given resolution of the feature sequence indexed by $[1 : N]$), we apply an iterative multi-level approach, see Figure 2 for an overview. To this end, we consider a *regular grid* of points in the two-dimensional scape plot representation, where neighboring grid points are d samples apart either in horizontal or in vertical direction, see Figure 2a. The fitness of these grid points are then computed, see Figure 2b. The parameter $d \in \mathbb{N}$ determines the density of the grid with $d = 1$ yielding the scape plot in full resolution. In the first step of the multi-level approach, we use a parameter $d = d_1 > 1$ and compute the fitness only for those points that lie on the resulting scape plot grid.

One crucial observation is that all points that lie in a neighborhood of a scape plot point of high fitness also typically have large fitness values. Therefore, it is reasonable to assume that the thumbnail segment lies in the neighborhood of one of the grid points of high fitness. This observation justifies the next steps of our procedure. Fixing a parameter $M \in \mathbb{N}$, we select among all grid points the M points that have the largest fitness values (or less points if the grid contains less than M points). These M points are also referred to as *anchor points*. Using a parameter $d_2 \in \mathbb{N}$ with $1 \leq d_2 < d_1$ (in our experiments we use $d_2 = d_1/2$ assuming that d_1 is a power of two), we consider all points on the refined grid (based on d_2) that are direct neighbors of one of the M anchor points. For all the resulting additional points, we then compute the fitness, see Figure 2c.

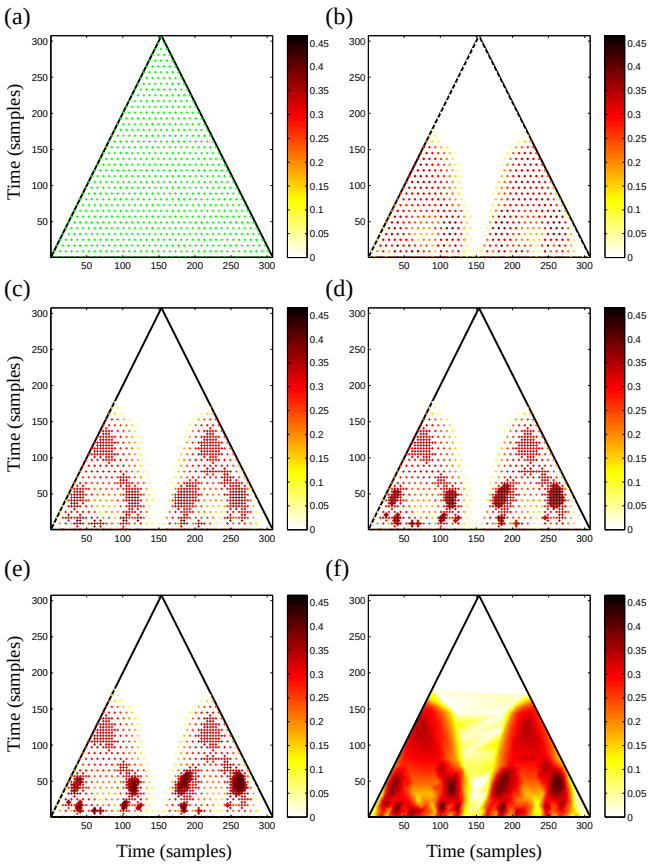

Fig. 2. Illustration of the multi-level grid sampling approach for the Beatles song "Act Naturally," see Figure 1. The used parameters are $I = 4$ with $(d_1, d_2, d_3, d_4) = (8, 4, 2, 1)$ and $M = 100$. **(a)** Grid of the first step (using $d_1 = 8$). **(b)** Fitness computed for the grid points in (a). **(c)** Refinement after second step (using $d_2 = 4$). **(d)** Refinement after third step (using $d_3 = 2$). **(e)** Refinement after forth step (using $d_4 = 1$). **(f)** Scape plot obtained from (e) by interpolation, compare with Figure 1a.

This last step can be iterated by selecting again the M points of highest fitness (among all previously considered points in the first two steps), using a finer grid based on some $d_3 \in \mathbb{N}$ with $1 \leq d_3 < d_2$, and again considering the neighbors, see Figure 2d. This process is repeated until one reaches the finest resolution. Finally, we define the scape plot point α_1^* to be the one of maximal fitness over all grid points considered in the entire procedure.

We say that our procedure has been *successful* if α_1^* coincides or, at least, is close to the actual thumbnail α^*. Here, as we will discuss later, we mean by "close" that the segments α^* and α_1^* induce the same repetitive structure. Besides finding the original thumbnail, also the visualization of the scape plot may be of interest. To this end, we generate a visualization on the finest possible resolution using simple interpolation techniques applied to all grid points considered in the multi-level approach, see Figure 2f.

To conclude the description of the first acceleration strategy, note that the parameters should be chosen in such a way that the procedure is successful, the running time is reduced as much as possible, and the visual impression of the interpolated scape plot is close to the original one. In our experiments, as we will present in Section 5, the specific setting turned out not to be crucial within a wide range of parameters leading to similar results. In particular, using

$I = 4$ with $(d_1, d_2, d_3, d_4) = (8, 4, 2, 1)$ and $M \in [10 : 100]$ has turned out to be a reasonable choice.

3. ACCELERATION BY MULTI-RESOLUTION FITNESS COMPUTATION

In the first strategy, we have reduced the number of segments the fitness measure has to be evaluated for. We now describe a second acceleration strategy which speeds up the actual fitness computation. To this end, we first need to summarize how the fitness measure is defined, see [8] for details. In the computation of the fitness measure, an enhanced self-similarity matrix (SSM) is computed on the basis of chroma features extracted from the music recording, see Figure 3a. Then for each segment α, an *optimal path family* that simultaneously reveals the relations between α and all other similar segments is computed. By projecting such an optimal path family to the vertical axis, one obtains an induced segment family, where each element of this family defines a segment similar to α. As an illustration, Figure 3b shows such an optimal path family for the segment $\alpha = [158 : 209]$ (horizontal axis) as well as the induced segments $\alpha_1, \alpha_2, \alpha_3$, and α_4 (vertical axis). Note that these four segments are exactly the four repeating verse sections of the song.

The computation of an optimal path family over a given segment α can be done using dynamic programming in $O(|\alpha| \cdot N)$ operations, see [8]. The algorithm is similar to the one used for dynamic time warping, see, e. g., [13, Chapter 4]. Obviously, the running time can be reduced when reducing the resolution in the underlying SSM. For example, in theory, reducing the resolution by a factor of two yields a speed up of the dynamic programming step by a factor of four. However, reducing the resolution too much may also lead to a deterioration of the similarity matrix, where important structural properties may get lost [14] and may lead to inaccuracies in the fitness computation. As a result, certain relations to be captured by the optimal path family may be missed as illustrated by Figure 3d. Therefore, applying this strategy needs to be done with care.

In a pilot experiment, we accelerated the fitness computation by simply reducing the SSM resolution from 2 Hz to 1 Hz. This led to a substantial reduction in running time with only a small decrease in the overall accuracy of the thumbnail estimation. Next, we further reduced the SSM resolution to 0.5 Hz. While further speeding up the computation, this resolution resulted in a severe deterioration of the thumbnail estimation, in particular in the case of thumbnails of short duration. In other words, 0.5 Hz is too low a resolution to reveal the desired structures. Therefore, we apply the strategy of reducing the SSM resolution in a level-dependent way not going beyond a 1 Hz resolution. Using $(d_1, d_2, d_3, d_4) = (8, 4, 2, 1)$ as explained in Section 2, we use the finest resolution of 2 Hz only for the last step ($d_4 = 1$). For all previous steps we use a reduced SSM resolution of 1 Hz. As for practical computations, as we will discuss in Section 5 in more detail, the resolution reduction becomes particularly important for the first step, where the fitness measure is evaluated for all points on the coarse scape plot grid. Here using a 1 Hz instead of a 2 Hz resolution yields in our experiments a speed up of roughly a factor of four (or even more) without any significant loss in the overall accuracy.

4. ACCELERATION BY FITNESS REUSE

We describe a third acceleration strategy, where we exploit the intrinsic properties of the fitness computation. Recall that in the computation of the fitness value $\varphi(\alpha)$ of a segment α, an optimal path family over α is determined, and the segments induced by this path family are the (approximate) repetitions of α (see Figure 3b). Now, the crucial observation is that each of the induced segments (being similar to α) also has more or less the same repetition relations as the segment α. As a result, the fitness of each of the induced segments is also close to the one of α. For example, in the case of $\alpha = [158 : 209]$ shown in Figure 3g, we obtain $\varphi(\alpha) \approx \varphi(\alpha_i)$, where α_i, $i \in [1 : 4]$, denote the four induced segments as defined in Section 3.

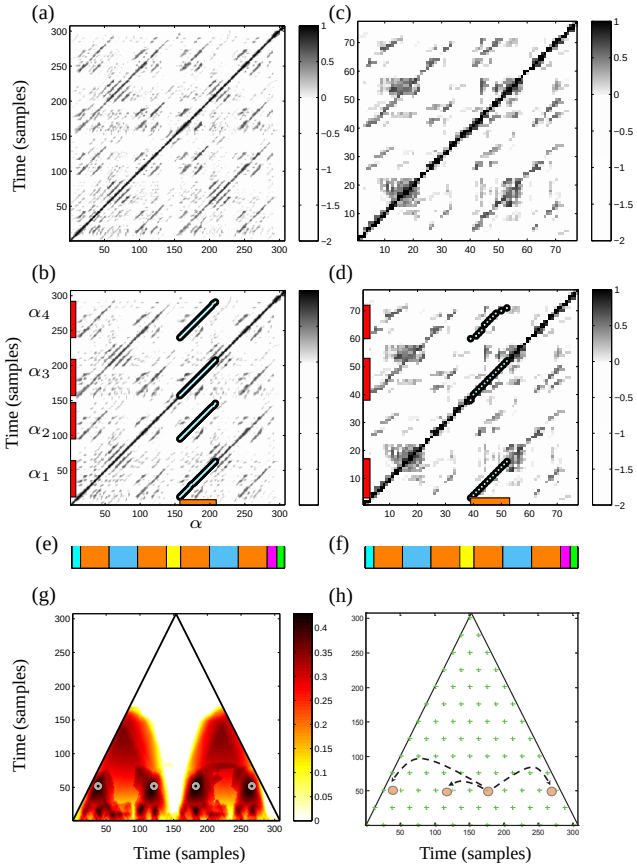

Fig. 3. Illustration of the fitness computation and possible risks of the reduction of the SSM resolution. **(a)** SSM with 2 Hz resolution. **(b)** Optimal path family for the segment $\alpha = [158 : 209]$ using the SSM from (a). **(c)** SSM with 0.5 Hz resolution. **(d)** Optimal path family for the segment $\alpha = [158 : 209]$ using the SSM from (c). **(e)/(f)** Ground-truth segmentation. **(g)** Fitness scape plot obtained from (a). The segment $\alpha = [158 : 209]$ and all induced segments are indicated by circles. **(h)** Exploiting path relations for fitness estimation of the induced segments.

Based on this observation, we proceed as follows. When computing the fitness $\varphi(\alpha)$ for a segment α in the overall procedure, we reuse the value $\varphi(\alpha)$ as estimate for the fitness of all segments induced by α, see Figure 3h. This information is stored in a suitable data structure. In this way, when we need to compute the fitness of another segment β at a later stage, we first check if there is already a segment β' in its suitable neighborhood, whose fitness value is available (either computed or estimated at a previous stage). If yes, we skip the fitness computation of β. Instead of β, we then use β' and its known fitness value for the subsequent steps. In our experiments, the above mentioned neighborhood is chosen to be two seconds.

5. EXPERIMENTS AND CONCLUSIONS

We now describe our systematic experiments and investigate the effect of our acceleration strategies. Note that it is not in the scope of this paper to discuss the specific parameter settings of and to evaluate the actual thumbnailing procedure—this has been done in [8]. Instead our goal is to illustrate to which extent the original procedure can be accelerated while obtaining the same thumbnail accuracy and visual impression of the scape plot as described in [8].

We have conducted our experiments on the basis of two datasets that have also been used in [8]. The first dataset, denoted by BEATLES, consists of recordings from the 12 studio albums by "The Beatles" [15][1]. The second dataset, denoted by MAZURKA, consists of the complete recordings of the 49 Mazurkas composed by Frédéric Chopin in three different versions[2] played by the pianists Rubinstein (1966), Cohen (1997), and Ezaki (2006), respectively. For both datasets, we derived thumbnail annotations from existing structure annotations, where a thumbnail annotation consists of an entire family of repeating segments with each segment serving equally well as a thumbnail.

In the original thumbnail procedure [8], which we denoted by OR, an SSM resolution of 2 Hz is used. As for the multi-level acceleration procedure (ML) from Section 2, we use $I = 4$ with $(d_1, d_2, d_3, d_4) = (8, 4, 2, 1)$ and $M = 100$. In the multi-resolution fitness approach (MR), we use the setting as described in Section 3. Finally, for the fitness reuse strategy (FR), we use a neighborhood of two seconds as described in Section 4. Note that all acceleration procedures can be used in a combined fashion. As said before, the specific settings are not crucial at this point and are chosen in a more conservative way to yield similar thumbnail results as the original procedure (and a similar visual impression of the interpolated scape plot to the original one, see Figure 3f). To demonstrate this, we consider three measures. The first two measure EvalSame and EvalSameTol indicate if the accelerated procedure yields exactly the same or nearly the same (with a tolerance of two seconds) thumbnail segment as the original approach. Note that these measure are not really suitable to measure the success since there is an entire family of valid thumbnail segments. Therefore, as a third measure, we use the same thumbnail F-measure EvalThumbF as described in [8] to show if the thumbnail obtained by accelerated procedures has the same quality as the thumbnail computed by the original approach.

Table 1 summarizes the experimental results. The algorithms have been implemented in Matlab R2012 (using C/C++ for the dynamic programming component), and tests were run on a computer with Intel Core i5-3470, 3.20 GHz CPU, 8 GByte RAM, under 64-bit Windows 7. In the following, we assume that the SSM on the finest level as used in [8] has been pre-computed and is given to all procedures as input. Then the times shown in Table 1 indicate the average running times per song given in seconds to derive the scape plot and the thumbnail from the SSM. For example, in the original procedure OR, it took in average 61.67 s to compute the thumbnail for the songs of BEATLES resulting in an overall thumbnail F-measure of EvalThumbF $= 0.77$. Applying the multi-level procedure ML results in an average running time of 2.49 s per song, which is a speed up of a factor of 24.7. The individual running times for the four levels

(a) BEATLES

	OR	ML	ML+MR	ML+FR	ML+MR+FR
TimeOverall	61.67	2.49	0.69	1.41	0.55
SpeedUp	-	24.7	88.8	43.6	112.5
TimeLevel1	-	2.01	0.35	0.94	0.21
TimeLevel2	-	0.13	0.06	0.13	0.06
TimeLevel3	-	0.13	0.06	0.12	0.06
TimeLevel4	-	0.13	0.13	0.10	0.11
EvalSame	-	0.50	0.31	0.70	0.40
EvalSameTol	-	0.95	0.83	0.92	0.82
EvalThumbF	0.77	0.77	0.75	0.76	0.76

(b) MAZURKA

	OR	ML	ML+MR	ML+FR	ML+MR+FR
TimeOverall	143.67	5.68	1.12	2.88	0.77
SpeedUp	-	25.3	128.7	50.0	187.1
TimeLevel1	-	5.10	0.71	2.34	0.38
TimeLevel2	-	0.16	0.06	0.15	0.06
TimeLevel3	-	0.15	0.07	0.14	0.06
TimeLevel4	-	0.15	0.15	0.10	0.12
EvalSame	-	0.60	0.37	0.69	0.35
EvalSameTol	-	0.88	0.78	0.88	0.73
EvalThumbF	0.71	0.71	0.73	0.71	0.71

Table 1. Experimental results for the running time behavior and the accuracy of various acceleration strategies for audio thumbnailing, see text for explanation. The times indicate average running times per song given in seconds.

are indicated in the next four rows of Table 1a. The full grid computation at the first level (using $d_1 = 8$) is TimeLevel1 $= 2.01$ s and takes much more time then the subsequent refinement steps. As for the accuracy, the value EvalSame $= 0.50$ shows that the acceleration procedure yields exactly the same thumbnail as OR in only half of the cases. However, as indicated by EvalSame $= 0.95$, in most cases one only has a small shift in the computed segments. In the other cases, the acceleration procedure may yield a different thumbnail segment, which is in the same segment family. This is shown by the fact the that thumbnail F-measure for ML (and also for the other procedures) is basically the same as for OR. From an application point of view, such a segment is equally suited as thumbnail.

Now, let us have a look at the other acceleration procedures and their combinations. Using MR on top of ML increases the overall running time by an additional factor of roughly four. In particular, this speed up mainly results from the usage of a coarser resolution at the full grid computation at the first level. Similarly, FR on top of ML increases the overall running time by an additional factor of roughly two. As a main result of this paper, our experiments show that one obtains by far the largest speed up when combining all three strategies without a substantial loss in the thumbnail accuracy. For example, in case of BEATLES, the combined approach needs an average running time of 0.55 s per song compared to 61.67 s of the original procedure, which is a speed up of a factor of 112.5. As shown in Table 1b, similar findings hold for the independent dataset MAZURKA. The reason for the slightly higher running times is that MAZURKA tends to comprise longer recordings than BEATLES. Altogether, this also demonstrates that our methods scale well to other types of music beyond popular music.

In conclusion, we have introduced three conceptually different acceleration strategies that lead to substantial speed-ups for a recent state-of-the-art thumbnailing procedure. These accelerations are important steps towards computing a thumbnail on-the-fly, which paths the way to applications in real-time services.

[1] Since for five of these songs 'HappinessIsAWarmGun', 'HerMajesty', 'Revolution9', 'TheEnd', and 'YouNeverGiveMeYourMoney' no clear repetitions are present in the annotations, these songs are left out resulting in 175 recordings (instead of the original 180 songs) used in our experiments.

[2] The resulting 147 files are a subset of the dataset of the Mazurka Project mazurka.org.uk.

6. REFERENCES

[1] Roger B. Dannenberg and Masataka Goto, "Music structure analysis from acoustic signals," in *Handbook of Signal Processing in Acoustics*, David Havelock, Sonoko Kuwano, and Michael Vorländer, Eds., vol. 1, pp. 305–331. Springer, New York, NY, USA, 2008.

[2] Jouni Paulus, Meinard Müller, and Anssi P. Klapuri, "Audio-based music structure analysis," in *Proceedings of the 11th International Conference on Music Information Retrieval (ISMIR)*, Utrecht, The Netherlands, 2010, pp. 625–636.

[3] Mark A. Bartsch and Gregory H. Wakefield, "Audio thumbnailing of popular music using chroma-based representations," *IEEE Transactions on Multimedia*, vol. 7, no. 1, pp. 96–104, 2005.

[4] Wei Chai and Barry Vercoe, "Music thumbnailing via structural analysis," in *Proceedings of the ACM International Conference on Multimedia*, Berkeley, CA, USA, 2003, pp. 223–226.

[5] Matthew Cooper and Jonathan Foote, "Automatic music summarization via similarity analysis," in *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, Paris, France, 2002, pp. 81–85.

[6] Masataka Goto, "A chorus section detection method for musical audio signals and its application to a music listening station," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 5, pp. 1783–1794, 2006.

[7] Mark Levy, Mark Sandler, and Michael A. Casey, "Extraction of high-level musical structure from audio data and its application to thumbnail generation," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Toulouse, France, 2006, pp. 13–16.

[8] Meinard Müller, Nanzhu Jiang, and Peter Grosche, "A robust fitness measure for capturing repetitions in music recordings with applications to audio thumbnailing," *IEEE Transactions on Audio, Speech & Language Processing*, vol. 21, no. 3, pp. 531–543, 2013.

[9] Geoffrey Peeters, "Sequence representation of music structure using higher-order similarity matrix and maximum-likelihood approach," in *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, Vienna, Austria, 2007, pp. 35–40.

[10] Meinard Müller and Nanzhu Jiang, "A scape plot representation for visualizing repetitive structures of music recordings," in *Proceedings of the 13th International Conference on Music Information Retrieval (ISMIR)*, Porto, Portugal, 2012, pp. 97–102.

[11] Craig Stuart Sapp, "Harmonic visualizations of tonal music," in *Proceedings of the International Computer Music Conference (ICMC)*, La Habana, Cuba, 2001, pp. 423–430.

[12] Craig Stuart Sapp, "Visual hierarchical key analysis," *ACM Computers in Entertainment*, vol. 3, no. 4, pp. 1–19, 2005.

[13] Meinard Müller, *Information Retrieval for Music and Motion*, Springer Verlag, 2007.

[14] Meinard Müller and Frank Kurth, "Enhancing similarity matrices for music audio analysis," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Toulouse, France, 2006, pp. 437–440.

[15] Matthias Mauch, Chris Cannam, Matthew E.P. Davies, Simon Dixon, Christopher Harte, Sefki Kolozali, Dan Tidhar, and Mark Sandler, "OMRAS2 metadata project 2009," in *Late Breaking Demo of the International Conference on Music Information Retrieval (ISMIR)*, Kobe, Japan, 2009.