# AUDIO THUMBNAILING IN VIDEO SHARING SITES

*Aggelos Pikrakis*

Department of Informatics, University of Piraeus, Greece
80 Karaoli & Dimitriou Str., 18534, Piraeus, Greece, pikrakis@unipi.gr

## ABSTRACT

This paper presents a variant of the Smith and Waterman algorithm that operates adaptively on a continuous feature space of MPEG-7 low level spectral descriptors and is capable of detecting repeating patterns (thumbnails) in audio streams that stem from shared Internet videos. The proposed method is not restricted to specific audio types and does not rely on training data. It has been studied in the context of four frequently encountered categories of audio streams, including TV shows, cover versions of music tracks, history documentaries and animal sounds. The results are encouraging and indicate that the presented scheme provides, in the general case, meaningful thumbnails and exhibits acceptable robustness with respect to audio recording quality.

## 1. INTRODUCTION

The rapid growth of video sharing sites has highlighted the need for efficient summarization methods that can facilitate indexing, retrieval, content tagging and fast browsing mechanisms. In addition, audio is acknowledged to be an important modality in shared multimedia objects and it therefore makes sense to investigate the possibility for *generic* audio summarization methods in video sharing sites.

We focus on *audio thumbnailing* as a special case of summarization that detects instances of a representative extract of an audio stream. *An extract is considered to be representative if it is repeated at least once throughout the audio stream and if it is of sufficient length.* This paper is therefore an attempt to provide a generic thumbnailing scheme, capable of handling various audio types, of varying recording quality. To this end, video-sharing sites is a good source of data. At the core of the proposed scheme lies a variant of the well known Smith and Waterman algorithm (SW) [1] that was originally introduced in the context of molecular sequence analysis.

Audio summarization and thumbnailing have so far been mainly studied in the context of speech and music signals. In the context of speech, several approaches have been proposed that treat summarization as a structural analysis task, where probabilistic models are built on acoustic, prosodic, structural and discourse features [2], [3]. In the case of music signals, emphasis has been given on detecting instances of the chorus of popular music tracks [4]. Music summarization has also been treated as a structural analysis task in [5] and [6]. Such approaches often rely on the fact that the inherent melodic, harmonic (chord progression) and metrical features of a music track manifest themselves as periodicities (at various abstraction levels) that, in turn, can be effectively captured by means of analyzing the self-similarity matrix of selected descriptors (e.g., the chroma vector [4]).

Our method does not compete with the above approaches in the sense that it does not use training data (as is the case with speech summarization methods) nor does it make assumptions on the nature or quality of the signals under study. We have placed emphasis on providing a generic scheme with predictable behavior, capable of generating *"reasonable"* thumbnails for a variety of audio types. Concerning the SW algorithm that lies in the core of our research, the closest relevant work can be traced in the context of *cover song identification*, and *music sequence alignment*, where variants of the SW algorithm have been proposed for discrete problems [7],[8], [9]. In other words, the feature sequence or similarity matrix is first mapped to a discrete domain and therefore, a major concern in such methods is to provide a reliable mapping from the continuous space to the discrete one. The SW variant proposed in this paper operates directly on the feature space of continuous low level multidimensional MPEG-7 descriptors (the MPEG-7 Audio Spectrum Envelop).

The paper is organized as follows: the next Section describes the feature extraction stage, Section 3 presents the proposed variant of the SW algorithm and Section 4 provides details about the experimental setup and the results of the proposed method for various audio types. Finally, conclusions are drawn in Section 5.

## 2. FEATURE EXTRACTION

At a first step, the audio stream is short-term processed by means of a moving window technique. The recommended window length is 250 *ms*, with zero overlap between successive frames. This window length is a trade-off between computational complexity and accuracy at the borders of detected thumbnails. At each frame, the *Audio Spectrum Envelop (ASE)* is extracted [10]. The ASE is a generic, low-level, MPEG-7 audio descriptor (LLD) and was selected for three reasons: (a) it is applicable to all types of audio, (b) it can be tuned to increasing level of refinement and (c) it facilitates reproducibility of results because it is part of a well known standard. The discriminative power of this feature has so far been exploited mainly in the context of audio classification, e.g., [10]. Although the ASE is not an optimal feature selection in any sense, our experimental evidence suggests that it is clearly a strong candidate for inclusion in future research.

The ASE is computed as follows:

- The frequency spectrum is first divided into a set of bands which are logarithmically distributed between 62.5Hz and 16000Hz, i.e., the frequency range covers a 8-octave interval, logarithmically centered at 1000 Hz.

- The spectral resolution, $r$, of each band is $r = 2^j$ octaves, where $j$ is an integer and $-4 \leq j \leq +3$. Small values of $j$ therefore imply better spectral resolution to the expense of increased dimensionality of the feature vector. In our experiments, $j = -2$.

- The edges of each frequency band are computed as: $loF_b = 62.5 * 2^{(b-1)r}$ and $hiF_b = 62.5 * 2^{(b)r}$, where $1 \leq b \leq \frac{8}{r}$.

- After the bands have been determined, the sum of power coefficients in each band yields the respective ASE coefficient.

- In the end, two more coefficients are similarly computed, one for frequencies below 62.5 Hz and one for frequencies above 16KHz.

In our study, due to the origin of audio streams, the sampling frequency, $F_s$, is relatively low ($\approx$ 22050Hz) and therefore, the actual number of bands is less than $\frac{8}{r}$. Specifically, if $j = -2$ then $r = 0.25$ and the number of bands should be $\frac{8}{0.25} + 2 = 34$. However, due to the fact that $F_s = 22050$, the number of bands becomes 31.

Let $X = \{\underline{x}_1, \underline{x}_2, \ldots, \underline{x}_N\}$ be the resulting sequence of ASE vectors, where $N$ is the number of frames. At a final step and as a means to capture the relationship between successive frames, each feature vector is augmented with an approximation of the first order derivative, as it is common with audio processing applications. In our case, the approximation of the derivative, $d\underline{x}_i$, at the $i$-th frame, is computed as $d\underline{x}_i = G \sum_{k=-1}^{k=+1} k \underline{x}_{i+k}$, where $G$ is a scaling factor ($G = 0.375$ after experimentation). Therefore, each feature vector becomes $2 \times 31 = 62$-dimensional.

## 3. THUMBNAIL EXTRACTION

The proposed thumbnailing scheme is inspired by the popular SW algorithm [1], which was originally introduced in the context of molecular sequence analysis. An interesting property of the SW algorithm is that it introduces a similarity function which can also take negative values. In addition, symbol deletions and insertions are penalized (with negative scores). As a result, at the end of the processing stage, the numbers of paths that survive in the search grid is relatively small. Furthermore, the best path, i.e., the path that corresponds to the best subsequence alignment, can be extracted by means of backtracking, starting from the node of the grid with the highest accumulated cost. An iterative procedure may then be applied to extract the second best alignment and so on.

### 3.1 The proposed variant of the SW algorithm

Due to the fact that the SW algorithm was originally introduced in the context of matching sequences of discrete symbols, several modifications need to take place so that it can be directly applied on a multidimensional space of continuous features:

First of all, the similarity $S(i,j)$, between feature vectors $\underline{x}_i$ and $\underline{x}_j$, is now defined as the cosine of their angle

$$S(i,j) = \frac{\sum_{k=1}^{L} \underline{x}_i(k) \underline{x}_j(k)}{\sqrt{\sum_{k=1}^{L} \underline{x}_i^2(k)} \sqrt{\sum_{k=1}^{L} \underline{x}_j^2(k)}} \tag{1}$$

where $L$ is the dimensionality of the feature space. By definition, the proposed similarity measure is bounded in $[-1,1]$, i.e., negative values are also possible. In a way, this is line with the philosophy of the original SW algorithm, where identical symbols yield a similarity value equal to one and different symbols produce a penalty equal to $-\frac{1}{3}$.

At a next step, we construct a $N$x$N$ cost grid, where $N$ is the length of the audio stream. The accumulated cost $H(i,j)$, to reach node $(i,j)$ of the grid, is defined as

$$H(i,j) = \max \begin{cases} H(i-1,j-1) + S(i,j) - T_h, \\ H(i-k,j) + k(1-T_h), & k = 1,\ldots,G, \\ H(i,j-m) + m(1-T_h), & m = 1,\ldots,G, \\ 0 \end{cases} \tag{2}$$

where $i \geq 2$, $i < j$, $T_h$ is a “sensitivity” threshold ($0 < T_h < 1$) and $G$ is a positive integer, indicating the maximum allowed gap (measured in number of feature vectors) during sequence alignment. The value of $T_h$ is computed by means of an iterative procedure, as explained in Section 3.2. Upon initialization ($i = 1$)

$$H(1,j) = max\{S(1,j) - T_h, 0\}, j = 1,\ldots,N$$

After the whole grid has been processed, we locate the node that corresponds to the maximum value of $H$, $(i_{max}, j_{max}) = argmax\{H(i,j); i,j = 1,\ldots,N\}$ and perform backtracking until the fictitious node $(0,0)$ is reached. The resulting best path reveals the two subsequences that yield the strongest alignment. Note that:

**(a)** Equation (2) is only computed if $i < j$, because we are dealing with a self-similarity problem and it therefore suffices to focus on the upper triangle of the $H$ matrix.

**(b)** If for some node, $(i,j)$, the accumulated cost is less than zero, then $H(i,j)$ is set to zero and the predecessor to node $(i,j)$ is the fictitious node $(0,0)$. In other words, $(i,j)$ can only be the first node of an alignment path.

**(c)** Parameter $T_h$ is not part of the original SW and serves as the means to control how many nodes of the grid survive during the computation of the accumulated cost function.

**(d)** The last two branches of the equation deal with deletions (gaps). Each deletion contributes the quantity $(1 - T_h)$ to the gap penalty. The maximum allowable gap was empirically set to 2 seconds (8 nodes given a short-term processing step equal to 0.25 seconds).

### 3.2 Adaptive Computation of $T_h$

The proposed variant depends on the similarity threshold $T_h$. In order to remove the need for manually setting $T_h$, the following iterative procedure is adopted:

*Initialization*: $T_h$ is set to a sufficiently low value, e.g., a value in the range $[0.2, 0.7]$.

*t-th iteration, $t >= 1$*: The algorithm of Section 3.1 is executed. Let $A_1(t), B_1(t), A_2(t), B_2(t)$ be the frame indices at the endpoints of the extracted best path, i.e., backtracking started at node $(B_1(t), B_2(t))$ and ended at node $(A_1(t), A_2(t))$. For the sake of simplicity of notation, let $[A_1(t), B_1(t)]$ stand for the audio segment (feature sequence) starting at feature vector with index $A_1(t)$ and ending at feature vector with index $B_1(t)$. As a result, the best path corresponds to the alignment of segments $[A_1(t), B_1(t)]$ and $[A_2(t), B_2(t)]$. $T_h$ is then increased by a predefined value, e.g., 0.01 and the current step is repeated unless the termination criterion holds.

*Termination*: Both $(B_1(t) - A_1(t) + 1)$ and $(B_2(t) - A_2(t) + 1)$ are shorter than a predefined minimum thumbnail length (measured in number of frames) or $H(i,j) = 0, \forall i,j = 1,\ldots,N$.

After the iterative scheme has terminated, we distinguish two cases:

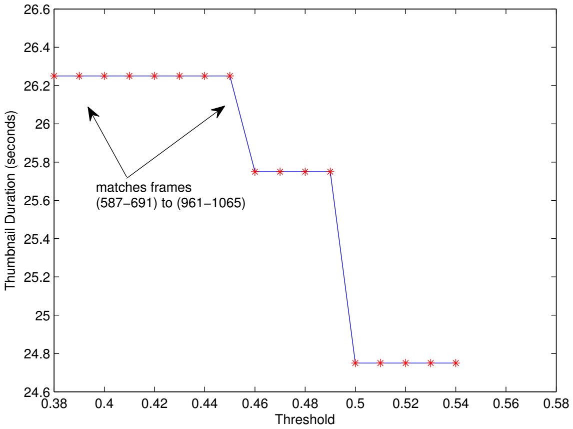

Figure 1: Threshold adaptation for an audio clip of a TV show. $T_h$ was initialized to 0.2 and it was required that the minimum thumbnail length is 10 seconds. The first pair of non-overlapping thumbnails occurs at $T_h = 0.38$ and survives until $T_h = 0.45$. This is also the most frequent pair (winner). The scheme terminates at $Th = 0.54$ because it is no longer possible to detect thumbnails of the desired length.

**(a)** We seek the most frequent pair of non-overlapping segments. This pair corresponds to two instances (repetition) of the detected thumbnail.

**(b)** If (a) does not hold, i.e., if all iterations have yielded overlapping segments, then the most frequent pair of overlapping segments is selected, after filtering out all pairs for which the overlap is larger than half the length of the longest of the two segments.

The rationale behind this approach is that successive iterations correspond to neighboring values of $T_h$ and therefore, *the best path between successive iterations is expected to remain unchanged, provided that it corresponds to a strong match*, i.e, to two instances of a clearly perceived thumbnail. This is presented in Figure 1 for an audio clip of a TV show. Obviously, it is desirable that the two instances of the thumbnail correspond to non-overlapping segments, as it is for example the case with the chorus of a music track in popular music or the sound of applause occurring at clearly separated parts of a recording. If this is not the case, overlap is tolerated provided that it is not intense, otherwise no thumbnail is extracted.

## 4. EXPERIMENTS

The proposed method is generic, in the sense that it makes no assumption on the audio type being processed or the recording conditions. In this section, we are making an attempt to examine the behavior of the method for different audio types stemming from YouTube© video clips. Video sharing sites are a good source, because of the large amounts data that are directly accessible. A number of video clips were downloaded from the YouTube© video sharing site and the audio stream of each clip was extracted and decoded. The resulting audio streams were then clustered into four categories that are described below.

### 4.1 Corpus Description

**(1)** *TV Shows*: 38 video clips originating from the "David Letterman" TV show. These are mixed audio streams, in the sense that they contain speech by more than one speaker, prolonged applause, music and (sometimes) isolated audio effects. The goal of this corpus is to test the method in the context of mixed audio streams, where speech and music are the two dominant modalities.

**(2)** *Music Clips*: 106 cover versions of music track "Breathless" of the Irish band "The Corrs" (female vocalist). A definition of a cover version is that it can be any performance of the original music track ([7]), by the original band or other musicians, including live stage performances and amateurs recorded at their home environment. Cover versions differ in terms of instrumentation, tempo, harmonicity and recording quality. In some recordings, especially when mobile devices have been used, the audio can be hardly perceptible. Therefore, this first set of audio tracks aims at answering the question whether thumbnail extraction is possible for decreasing audio quality and performance variation.

**(3)** *History Documentaries*: 33 extracts from "History Channel" documentaries of the "Ancient Warriors" series. Speech is the dominant modality, with music and battle sounds in the background. This category serves to study the behavior of the algorithm in the context of audio streams where narrative speech prevails and the rest of the sounds function as sound carpet.

**(4)** *Animal Sounds*: 42 amateur recordings, mainly in open spaces (e.g., zoos), that usually consist of a human voice speaking close to the camera (describing the scene or interacting with the animals) and animal sounds relatively close to the recording device. Crowd noise and environmental sounds can be frequently heard in the background.

Table 1 summarizes audio duration related statistics.

| **Category** | **# Clips** | **Min Dur. (sec)** | **Max Dur. (min)** | **Aver. Dur. (min)** |
|---|---|---|---|---|
| TV Shows | 38 | 25.2 | 10.1 | 3.8 |
| Music Clips | 106 | 30 | 9.5 | 3.5 |
| Documentaries | 33 | 240 | 10 | 8.8 |
| Animals | 42 | 25.8 | 7.9 | 2.2 |
| Total | 219 | Total ≈ 15 hours | | |

Table 1: Duration statistics per audio category.

### 4.2 Audio Thumbnailing Results

We now report the performance of the method for each audio category.

#### 4.2.1 *Category # 1*

Given that the audio streams in this category are mixed, we use the proposed method to answer the following questions:
- "Is it possible to extract non-overlapping (NO) thumbnails that are at least $M$ seconds long?"
- "What is the nature of the extracted thumbnails?" Table 2 presents the answers to both questions for $M = 10$, $M = 5$ and initial similarity threshold equal to 0.7. It can be seen that the answer to the first question is positive in 17 out of 38 cases (≈ 45% of the audio clips). The length of the extracted

patterns is in the range of $[10.25, 64.25]$ seconds. In 5 cases the thumbnail consists of prolonged applause and laughter ($10-12$ seconds long), in one case it is the sound logo of the TV show and in all remaining audio clips it coincides with the chorus of a music performance or even a longer section that, for example, consists of a pre-chorus plus chorus pair. It is possible to remove one of the 21 failures by extracting two overlapping thumbnails (20 seconds long), that consist of prolonged applause, laughter and short speech extracts. The remaining failures are due to the fact that the audio stream is a succession of speech, shorts bursts of applause and very short music intervals that do not follow a repetitive pattern of the desired pattern length.

| | **Non-overlapping thumbnails for a TV show** | |
|---|---|---|
| | *# Clips* | *Characterization* |
| M=10 | 5 | Laughter+applause ($10s-12s$) |
| M=10 | 1 | Sound Logo of TV Show |
| M=10 | 11 | Music pattern ($10.25s-64.25s$) |
| M=5 | 16 | Laughter+applause ($< 10s$ ) |
| M=5 | 4 | Music pattern ($< 10s$ ) |
| M=5 | 11 | Music pattern ($10.25s-64.25s$) |

Table 2: Thumbnails: David Letterman TV show.

If the requested thumbnail length decreases to $M=5$ seconds, the number of failures decreases to 7, but there now exist 20 patterns less than 10 seconds long. An explanation is that if short thumbnails are permitted, then the diagonals of the self-similarity matrix that are close to the main diagonal, are now more likely to yield the dominant repetition, i.e., the best match generated by the algorithm. All the detected short thumbnails now correspond to prolonged laughter and applause, with the exception of 4 that consist of short music extracts of highly repetitive music.

### 4.2.2 *Category # 2*

The music set under study consists of cover versions of a typical pop music track with constant beat and regular metric structure. The chorus (CH) of the CD version is $\approx 23.5$ s long and is encountered three times during the track. Furthermore, each instance of the chorus is preceded by a pre-chorus (PCH) section that is $\approx 14$ s long. To begin with, we treat as a valid thumbnail any detected instance of the chorus or any instance of the PCH section followed by the CH part. To this end, a pattern is accepted as a representative extract, if it contains or overlaps significantly (at least 80%) with any instance of the chorus or any instance of the PCH+CH section.

Figure 2 summarizes the thumbnailing results. The figure indicates that the chorus is extracted as a thumbnail in $\approx 40\%$ of the cases (circle marked with CH in figure), when the similarity threshold is initialized to 0.2 and the minimum thumbnail duration is 15 s. It is worth noting that in almost 30% of the clips, the detected pattern is approximately one minute long (second marked circle). This type of pattern models a longer structure of the recording, that starts with a violin theme, proceeds with verse and continues with a pre-chorus section followed by a complete chorus. One more remark is that some of the detected patterns correspond to repetition but have shorter duration (approcimately 15s). Such patterns usually coincide with a characteristic violin theme in the music track and it makes sense to accept them as valid thumbnails too, if the thumbnail definition given above is relaxed. Finally, in 6 cases the algorithm failed to extract any thumbnail at all. Those case refer to recordings of very poor quality from handheld devices during live performances.

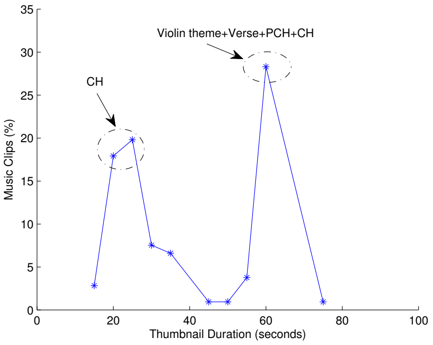

Figure 2: Thumbnails: Music Clips (no overlap). The circles mark results with which a standard interpretation can be associated.

### 4.2.3 *Category # 3*

Our study has revealed that meaningful thumbnails that are at least 10 seconds long, can still be extracted in 3 out of 33 cases, when the initial value of the similarity threshold is equal to 0.7. In all three cases, the detected pattern is a short theme of music (less than 15 s long). In four more cases, it was possible to extract overlapping thumbnails (15-20 seconds long) consisting of music themes mixed with sound effects.

If, on the other hand, the desired minimum pattern length, $M$, is reduced to 5 seconds, short thumbnails are successfully returned for $9+1=10$ audio clips (one consisting of overlapping (O) patterns). In all cases, music prevails, i.e., the patterns consist of short intense/melancholic/epic music themes, that serve to emphasize/dramatize narration.

Another interesting remark is that if the initial value of the similarity threshold is set to 0.4 then *the results change radically*. Specifically, meaningful patterns are now extracted in $19+2+3+4+1=29$ cases, including $4+1=5$ pairs of overlapping patterns. A study of results reveals that the thumbnails now contain sentences of the narrator's speech as well. For example, in a documentary describing the Aztec Warriors, each instance of the thumbnail refers to the description of a different stage of battle preparation. The narrator's speech follows a similar pattern of rhythm in both cases and carries similar sentiment. Furthermore, in the background, music and sound effects can be heard in both detected instances. Overall, it can be concluded that *music thumbnails are formed at higher levels of similarity* compared to *thumbnails that contain narrative speech*. Table 3 summarizes the results of this category.

### 4.2.4 *Category # 4*

In the case of animal sounds, as the similarity threshold increases (for fixed minimum thumbnail length $10s$), the possibility to capture a short repetitive sound (e.g., lion roar) decreases. On the other hand, for low levels of similarity, the proposed algorithm behaves like a scene detector; longer

|  | **History Channel Documentaries** |  |
|---|---|---|
|  | **Initial Similarity Threshold = 0.7** |  |
|  | *# Clips* | *Characterization* |
| M=10 | 3 | Music Theme ($10s - 15s$) (NO) |
| M=10 | 4 | Music Theme ($15s - 20s$) (O) |
| M=5 | 9 | Music Theme ($5s - 15s$) (NO) |
| M=5 | 1 | Music Theme ($10.5s$) (O) |
|  | **Initial Similarity Threshold = 0.4** |  |
| M=10 | 19 | Music-Narration with background sound effects (NO) |
| M=10 | 2 | Epic Music Theme (NO) |
| M=10 | 3 | Music Theme+Sound Effects (NO) |
| M=10 | 4 | Music Theme+Sound Effects (O) |
| M=10 | 1 | Music-Narration with background sound effects (O) |

Table 3: Thumbnails from History Channel Documentaries.

| **Animals - Initial Similarity Threshold = 0.7** |  |
|---|---|
| *# Clips* | *Characterization* |
| 2 | Dog Barking ($11s - 12s$) |
| 2 | Nightningale Song ($17s - 24s$) |
| 3 | Music theme ($10s - 30s$) |
| 1 | Water Splashing by dolphin ($12s$) |

Table 4: Thumbnails: Animal sounds, high initial threshold.

| **Animals - Initial Similarity Threshold = 0.2** |  |
|---|---|
| *# Clips* | *Characterization* |
| 1 | Narration over Forest sounds ($14s$) |
| 2 | Dog Barking ($11s - 17s$) |
| 1 | Narration about dogs ($22s$) |
| 1 | Crowd over distant lion sounds ($11s$) |
| 1 | Narration over Hornbill sounds ($18s$) |
| 1 | Hornbill sounds+isolated human sounds ($10s$) |
| 1 | Distant Radio broadcast from a room ($15s$) |
| 1 | Canary+house noise ($26s$) |
| 3 | Added Music theme ($10s - 30s$) |
| 1 | Excited crowd near dolphin pool ($12s$) |
| 1 | Road traffic ($13s$) |
| 2 | Intense Lion Roaring ($10s - 13s$) |
| 1 | Wildlife near river ($13s$) |
| 6 | Nightningale Song ($12 - 24s$) |
| 1 | Rooster over distant birds ($20s$) |
| 1 | Sheep sounds ($14s$) |
| 1 | Dolphin Splashing+human voice ($30s$) |

Table 5: Thumbnails: Animal sounds, low initial threshold.

non-overlapping audio patterns are extracted, that consist of a succession of events, including the animal sound. The thumbnail is therefore more likely to contain a tapestry of environmental sounds in the background, as it is for example the case with jungle sounds that are briefly interrupted by a more dominant event of short duration (e.g., a lion's roar). Tables 4 and 5 present a summary of the thumbnailing results for this category, for two different initial values of the similarity threshold (0.7 and 0.2 respectively).

## 5. CONCLUSIONS

This paper made an attempt to investigate a generic audio thumbnailing scheme in the context of various audio streams stemming from shared Internet videos. Our study revealed that it is possible to extract meaningful repeating patterns from various audio types of varying quality that were recorded under various conditions. It was observed that different thumbnails are extracted depending on the allowable similarity and duration constraints. In the general case, music generates thumbnails at high similarity levels, whereas narrative speech, animal sounds, etc., form repeating patterns at significantly lower similarity levels. It is in our future priorities to study and extend the proposed scheme in a multimodal summarization environment.

## REFERENCES

[1] T.F. Smith and M.S. Waterman, "Identification of common molecular subsequences," *J. Mol. Biol.*, vol. 147, pp. 195–197, 1981.

[2] J.J. Zhang, R.H.Y. Chan, and P. Fung, "Extractive speech summarization using shallow rhetorical structure modeling," *Audio, Speech, and Language Processing, IEEE Trans. on*, vol. 18, no. 6, pp. 1147 –1157, Aug. 2010.

[3] S. Furui, "Recent advances in automatic speech summarization," in *Spoken Language Technology Workshop, 2006. IEEE*, Dec. 2006, pp. 16 –21.

[4] M.A. Bartsch and G.H. Wakefield, "Audio thumbnailing of popular music using chroma-based representations," *Multimedia, IEEE Trans. on*, vol. 7, no. 1, pp. 96 – 104, Feb. 2005.

[5] M. Levy and M. Sandler, "Structural segmentation of musical audio by constrained clustering," *Audio, Speech, and Language Processing, IEEE Trans. on*, vol. 16, no. 2, pp. 318 –326, Feb. 2008.

[6] J. Paulus and A. Klapuri, "Music structure analysis using a probabilistic fitness measure and a greedy search algorithm," *Audio, Speech, and Language Processing, IEEE Trans. on*, vol. 17, no. 6, pp. 1159 –1170, Aug. 2009.

[7] J. Serra, E. Gomez, P. Herrera, and X. Serra, "Chroma binary similarity and local alignment applied to cover song identification," *Audio, Speech, and Language Processing, IEEE Trans. on*, vol. 16, no. 6, pp. 1138 –1151, Aug. 2008.

[8] A.M. Stark and M.D. Plumbley, "Performance following: Tracking a performance without a score," in *IEEE ICASSP 2010*, March 2010, pp. 2482 –2485.

[9] Pierre Hanna, Thomas Rocher, and Matthias Robine, "A robust retrieval system of polyphonic music based on chord progression similarity," in *32nd ACM SIGIR*, New York, NY, USA, 2009, SIGIR '09, pp. 768–769, ACM.

[10] H.G. Kim, N. Moreau, and T. Sikora, *MPEG-7 Audio and Beyond: Audio Content Indexing and Retrieval*, Wiley, 2006.