GENERATING EXPRESSIVE SUMMARIES FOR SPEECH AND MUSICAL AUDIO USING SELF-SIMILARITY CLUES

Mustafa Sert, Buyurman Baykal and Adnan Yazıcı

Başkent University, Department of Computer Engineering
06530 Ankara, TURKEY
msert@baskent.edu.tr
Middle East Technical University, Department of Electrical and Electronics Engineering
06531 Ankara, TURKEY
Middle East Technical University, Department of Computer Engineering
06531 Ankara, TURKEY

ABSTRACT

We present a novel algorithm for structural analysis of audio to detect repetitive patterns that are suitable for content-based audio information retrieval systems, since repetitive patterns can provide valuable information about the content of audio, such as a chorus or a concept. The *Audio Spectrum Flatness* (ASF) feature of the MPEG-7 standard, although not having been considered as much as other feature types, has been utilized and evaluated as the underlying feature set. Expressive summaries are chosen as the longest patterns by the *k-means* clustering algorithm. Proposed approach is evaluated on a test bed consisting of popular song and speech clips based on the ASF feature. The well known *Mel Frequency Cepstral Coefficients* (MFCCs) are also considered in the experiments for the evaluation of features. Experiments show that, all the repetitive patterns and their locations are obtained with the accuracy of 93% and 78% for music and speech, respectively.

1. INTRODUCTION

Feature extraction plays an important role in content-based retrieval of multimedia information since most applications require the ability to search content at both low-level (e.g., color, audio energy) and semantic level (e.g., objects, events). In general, most of the feature extraction methods deal with audio segmentation. These studies can be examined in three categories. One of them is a segmentation method which involves with the classification of audio data segments into predefined classes (e.g., music, speech, and environmental sound) [1]. The other category is to detect the abrupt changes in feature values of audio [2]. The last one is another segmentation method which consists of relative silences or pauses in an audio [3]. Our segmentation method relies on the second category and advances it by realizing a structural similarity analysis technique.

In the literature, some recent research has been realized on detection of repetitive patterns and summarization of musical clips based on structural analysis. Chai and Vercoe [4] presented a method to automatically analyze the repetitive pattern of music from acoustic signals. They have presented an algorithm that generates structural information in the form of “AABABA” by indicating the beginning and ending locations of each section. Cooper and Foote [5] presented analytic methods to find repetitive structure in musical audio. They proposed a representation, formally known as *similarity matrix*, to analyze the structure of audio [6, 7]. However, none of the previous methods addressed the problem of detecting all the chorus sections in a song. One attempt based on this motivation is explored by Goto [8]. Goto presented a method that detects all the chorus sections and their locations by using the *acoustic* features. However, the method needs some prior knowledge about the spectral characteristics of chorus sections in order to improve the accuracy. Speech data, on the other hand, has not been explored as music data, as well.

In this paper, we are concerned with developing a method for detecting repetitive structures in music and speech clips that are suitable for content-based audio information retrieval systems. Specifically, we describe a novel approach based on the ASF descriptor of the MPEG-7 standard [9] and evaluate the descriptor for detecting repetitive patterns in music and speech data.

The rest of the paper is organized as follows. After the introduction, Section 2 briefly describes the extraction process of the ASF descriptor. Structural similarity analysis based on ASF is described in Section 3. Section 4 presents an algorithm to extract expressive summaries. Experimental results and evaluation of the ASF and MFCC features are discussed in Section 5. Finally, our concluding remarks and future directions are presented in Section 6.

2. EXTRACTION OF THE ASF DESCRIPTOR

In order to extract this feature, a $30ms$ hamming window function ($\omega_i = (0.5 - 0.46cos(\frac{2\pi i}{N}))$, where N is the number of samples in the window such that $1 \leq i \leq N$) is applied to corresponding analysis segment and transformed into the frequency domain. The recommended approach of the standard is to compute the ASF from a power spectrum based on the frequency range between $62.5Hz$ and $16kHz$ with the resolution of $1/4$ *octaves*. The ASF feature uses logarithmic scale to obtain the number of sub-bands, and defined as

$$\sharp bands = \frac{\log_2 (hiFreq/loFreq)}{octaveResolution} \ . \qquad (1)$$

where *hiFreq* and *loFreq* is the upper- and lower-frequency limits of the analysis frame, and *octaveResolution* denotes the logarithmic measure of the band resolution. The flatness measure is then computed as the ratio of geometric mean to arithmetic mean of spectral power for each sub-band and defined as follows:

1-4244-0367-7/06/$20.00 ©2006 IEEE

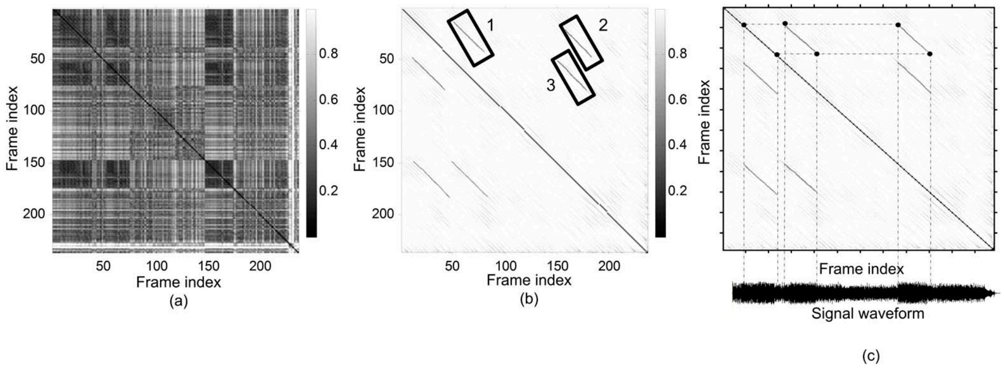

Fig. 1. Similarity matrix computed using ASF feature from the song "Tahitian Moon" by Perry Farell. (a) Initial similarity matrix, (b) the similarity matrix computed after the postprocessing, (c) projections of the pattern locations

$$ASF = \frac{\sqrt[N]{\prod_i^N P_i}}{\left[\left(\sum_i^N P_i\right)/N\right]} \ . \tag{2}$$

where *P* is the power spectrum of each sub-band, *N* is the length of sub-band, and ASF represents the *audio spectrum flatness* coefficients of the whole signal. The resulting description of ASF is a matrix with the dimension of $n \times m$. The length of *n* depends on audio length *t* with the relation of $t/hopSize$, and *m* represents the number of logarithmic sub-bands. Spectral flatness descriptions of each analysis frame ($30ms$ duration) are stored in the rows of ASF matrix and represented as a time series of feature vectors (V_i):

$$ASF = \begin{bmatrix} V_{1,1} & V_{1,2} & ... & V_{1,j} \\ V_{2,1} & . & . & . \\ V_{i,1} & . & . & V_{i,j} \end{bmatrix} \tag{3}$$

where $1 \leq i \leq n$, $1 \leq j \leq m$, and $n, m \in Z^+$.

3. STRUCTURAL SIMILARITY ANALYSIS

The proposed approach is based on the work in [10], and it is extended to improve the recognition accuracy by altering the underlying feature set, utilizing the *k-means* algorithm to cluster the detected chorus sections, and performing additional post processing techniques. These steps are explained in the following subsections.

3.1. Constructing the Similarity Matrix

In order to detect the repetitive patterns within a music or speech, our analysis begins with the calculation of similarities between the feature vectors of ASF (4). If the result of comparisons between feature vectors lead to small distances, then it is represented with dark pixels in the matrix, otherwise it is represented with brighter pixels.

Let V_i and V_j be the feature vectors of frames *i* and *j* in the feature matrix. Then the similarity is defined by the *Euclidean* norm as follows:

$$\sigma(V_i, V_j) = \sqrt{\sum_{k=1}^{m} (V_{i_k} - V_{j_k})^2} \qquad (i, j, m \in Z^+). \tag{4}$$

where $\sigma(V_i, V_j)$ denotes the similarity between frame *i* and *j*, while *m* represents the vector lengths. The resulting similarity matrix is a diagonally symmetric matrix with the dimension of $n \times n$. Fig. 1.a presents an example of the similarity matrix using a 2-D gray image plot in which each pixel represents the distance value as a gray level.

3.2. Post Processing

As depicted in Fig. 1.a, this form of the similarity matrix does not reveal the repetitive patterns, since many discrete values of the feature vectors are small and very close to each others even if they have significant differences between them. Therefore, we realize two consecutive approaches to reveal the differences between the feature vectors. The first one is to normalize the values of similarity matrix σ between 0 and 1 to obtain a uniform distribution (5) and the second one is to strengthen the diagonal stripes by performing a diagonal summation process over the matrix (6).

$$\overline{\sigma}_{x,y} = \frac{\sigma_{x,y} - \min(\sigma)}{\max(\sigma) - \min(\sigma)} \ . \tag{5}$$

$$\widehat{\sigma}_{x,y} = \sum_{i=1}^{k} \overline{\sigma}_{x+i,y+i} \ . \tag{6}$$

where $\widehat{\sigma}$ represents the resulting similarity matrix and *k* represents the strengthen order. Selection of the strengthen order is important, since it depends to the length of the sought pattern. Our experiments shown that strengthen order between 5 and 12 is suitable for detecting repetitive patterns in music and speech. The result of filtering is not included in Fig. 1 for brevity.

In order to prevent vertical and horizontal lines disturbing diagonal lines, we remove the horizontal and vertical lines from the similarity matrix. To this end, we implement an image processing technique by applying a kernel of size $b \times b$. In our experiments we use a kernel of size 11×11. The main diagonal elements of this kernel are filled by 10, while other elements are all -1. If this kernel is moved around the similarity matrix $\widehat{\sigma}$, it would respond more strongly to lines in the -45^o direction. As a result of this process, we obtain the matrix $\breve{\sigma}$ (Fig. 1.b).

4. PATTERN EXTRACTION

Above, similar regions within the matrix are becoming visible as diagonal stripes. The next step towards the matrix is to interpret and

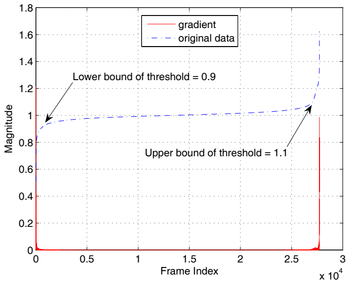

Fig. 2. Computation of the thresholds for the song 'Tahitian Moon" by *Perry Farell*

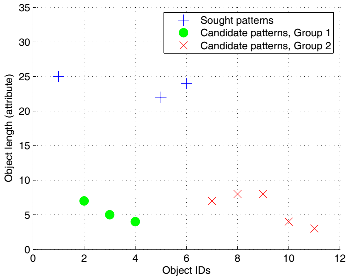

Fig. 3. Clustering of patterns for the song "Tahitian Moon" by *Perry Farell* ($k = 3$)

cluster the visible patterns.

4.1. Interpretation of Patterns

Fig. 1.b depicts three repeated patterns that have been extracted for the song "Tahitian Moon" by *Perry Farell*. The diagonal line with label 1 represents the first repetition of the chorus section. The line with label 2 corresponds to the second repetition of the chorus section. The first occurrence of the chorus section is invisible due to the main diagonal. The line with label 3 represents the repetition of the chorus section 2. Thus, the chorus section with number 3 is not required in the pattern extraction process. The location of first chorus section is obtained by projecting the diagonal line with label 1 to the main diagonal and then to the time line. The locations of other chorus sections are determined in a similar fashion. The projections of detected patterns are depicted in Fig. 1.c.

4.2. Clustering of Patterns

In some cases, one can encounter with distinct repetitive patterns within a given audio. To the best of our knowledge, the longest pattern is perceived as the *most salient* in the similarity matrix. Hence, we choose the longest pattern as the summary to enable the expressive power of the summaries. This approach not only yields expressive summaries, it also eliminates the stop-words in speech, as well.

In general, given a similarity matrix, three steps are needed for salient object extraction. Firstly, we compute the lower and upper bounds of threshold ($lThr$ and $uThr$) by the direction of maximum rate of change of $\mathbf{S}$, where $\mathbf{S}$ is the sorted vector of the similarity matrix in ascending order. Since the similarity matrix is symmetric to its main diagonal, we only sort the upper triangular of the matrix. The algorithm obtains the direction of maximum rate of change by computing first order derivative of $\mathbf{S}$ and denoted by ∇S. Afterwards, we assign $lThr$ and $uThr$ to the first and second minimum values of ∇S, respectively. This process is depicted for the song "Tahitian Moon" by Perry Farell in Fig. 2. Consequently, a thresholding is performed over the final similarity matrix by utilizing the computed thresholds. Let F be the final similarity matrix, then the thresholding is defined as follows:

$$H(x,y) = \begin{cases} 1 & \text{if } uThr \geq F(x,y) \geq lThr \\ 0 & \text{otherwise} \end{cases}$$

where pixel (x, y) denotes a gray-level value and H represents the similarity matrix after the thresholding.

The next step after thresholding is to compute the lengths of all candidate patterns from the similarity matrix, H. We compute the lengths of each pattern by using the method that has been described in Fig. 1.c. The result of this step is a matrix and denoted by $\mathbf{A}$. The first column of $\mathbf{A}$ represents the object (each diagonal line) IDs, and the second column represents the object lengths as the attributes.

Since a similarity matrix may contain many diagonal lines in different sizes, and our aim is to extract the most representative ones, we need a method to group the similar (in terms of length) lines into the same cluster. Therefore, we make use of the *k-means* clustering algorithm [11] for this purpose. We utilize the algorithm as follows: After defining the number of clusters k, we determine the initial centroid coordinate of matrix $\mathbf{A}$, sequentially. For instance, if the number of cluster is three, then the first three elements of the matrix $\mathbf{A}$ is selected as the coordinate of the centroids. The grouping is then performed by minimizing the sum of squares of distances between the data and the corresponding cluster centroid. Knowing the members of each group, then we compute the new centroid of each group based on the new memberships. This process is continued in the same manner and terminated when there is no object that moves to a group. Our experiments show that $k = 3$ is suitable for extracting chorus sections and key concepts from music and speech, respectively. The result of clustering for the song "Tahitian Moon" by Perry Farell is depicted in Fig. 3.

5. EXPERIMENTAL RESULTS AND EVALUATION

The proposed approach was evaluated on a test set of popular pop/rock songs and speech clips. The MFCC feature, which is well known by the ASR community, has also been considered in the experiments for two reasons. One of them is to provide a comparison between the features in detecting repetitive patterns, and the other one is to compare the results of our approach with similar studies [5, 12], since many of them utilized the MFCC-based features. In total, we have approximately 45 minutes of songs and 32 minutes of speech clips in our corpus. All the song and music clips are sampled at the rate of $44.1KHz$, stereo channel, and encoded by $16-bit$ per sample.

For *musical audio*, the output of the proposed approach is compared with the number of chorus sections obtained by manually annotating each audio clip. The results for musical audio are listed in Table 1. In order to evaluate and compare the accuracy of our approach, we have included the first seven songs from the study in [5] to our courpus (Table 1). The method correctly extracted and located 28 of 30 chorus sections within the songs. This results $28/30 = 93\%$ recall rate within the songs. In addition, the proposed method deals with one error for the song "Magical Mystery

Table 1. Accuracy results for popular songs. *M* denotes the $\sharp$ of manually annotated chorus sections. *ASF*, *MFCC*, and [5] columns represent the $\sharp$ of automatically detected chorus sections based on ASF, MFCC, and 80-bin spectrogram features, respectively.

SONG	*$\sharp$ of Chorus Section*			
Title / Artist	*M*	*ASF*	*MFCC*	[5]
Wild Honey / U2	3	3	2	3
Mystery Tour / The Beatles	3	2	2	3
Tahitian Moon / Perry Farell	3	3	3	3
Lucy in the Sky / The Beatles	4	4	4	3
Zephyr Song / Chili Peppers	3	3	3	2
Hash Pipe / Weezer	3	3	3	2
Optimistic / Radiohead	2	2	2	2
Sub total	21	20	19	18
Sub Recall Rate		95%	91%	86%
Fast Car / Tracy Chapman	3	2	2	-
Blue Hotel / Chris Isaak	3	3	3	-
Always / Jon Bon Jovi	3	3	3	-
Total	30	28	27	-
Overall Recall Rate		93%	91%	-

Table 2. Accuracy results for speech data. The results are presented in the form of A/M. The labels *A* and *M* represent the $\sharp$ of automatic and manual annotations for a given speech.

	ASF		*MFCC*	
Speech Text	*male*	*female*	*male*	*female*
News (key concept: *South Asia Quake*)	18/20	14/20	19/20	17/20
Lecture Note (key concept: *OOA/D*)	13/15	11/15	14/15	12/15
Lecture Note (key concept: *Java language*)	20/25	17/25	23/25	19/25
Overall Recall Rate	$93/120 = 78\%$		$104/120 = 87\%$	

Tour" by the Beatles compared to work in [5], which produces three errors (Table 1). Compared to MFCC, which results a recall rate of 91%, there is no clear difference in respect of ASF, that is, the ASF performs slightly better job with the recall rate of 93%.

For *speech data*, we prepared a scenario that deals with synthetic data due to the lack of sufficient annotated speech clips. In order to generate synthetic data that are suitable for our system, we prepared three speech texts that include some words repeating themselves in regular frequencies. The first one is obtained from a news portal concerned with *South Asia Quake*, and the other two are selected from computer science lecture notes. These texts are then recorded by talking to a microphone. In order to take into account the potential acoustics variations, this procedure is carried out for five male and female speakers. The evaluation for speech data is performed in a similar fashion as music data. The results for speech data are listed in Table 2. The method correctly extracted and located 93 of 120 chorus sections within the songs. This results $93/120 = 78\%$ recall rate within the songs. Compared to music, proposed approach yields poorer accuracy rate due to the recording characteristics and tone variations. Particularly, the recognition rate of the MFCC features (87%) is slightly better than the recognition rate of the MPEG-7 ASF feature (78%), since some male and female speeches are better recognized by the MFCC compared to MPEG-7 ASF.

6. CONCLUSION

In this study, a novel approach is proposed, and ASF descriptor of the MPEG-7 standard is evaluated for detecting repetitive patterns in music and speech data. Our experiments show that, the proposed approach together with ASF feature is adequate for efficient detection of repetitive patterns in music and speech. On the other hand, MFCC feature yields speech recognition rate superior to ASF and thus can be preferred instead of ASF. In addition, proposed method does not require a priori knowledge and thus can be considered as an unsupervised method.

The results of this study can be used in wide range of applications that require rapid browsing of audio archives, such as accessing to desired song in a hand-held MP3 player, locating to, deleting, and playing only specific sections of an audio, compact description of TV/Radio archives, and so forth. Video highlight identification applications and audio information retrieval systems can benefit from the results of this study by using the generated audio summaries.

Our future works lies in two directions. The first consists of exploring the possible performance improvements of the proposed method. The second direction is to benefit from the obtained results in similarity queries for audio content-based retrieval systems.

7. REFERENCES

[1] L. Lu, H. Jiang, and H. Zhang, "A robust audio classification and segmentation method," in *Proc. of the 9th ACM Intl. Conf. on Multimedia*, 2001, pp. 203–211.

[2] G. Tzanetakis and P. Cook, "Multifeature audio segmentation for browsing and annotation," in *Proc. of IEEE WASPAA*, 1999.

[3] S. Pfeiffer, "Pause concepts for audio segmentation at different semantic levels," in *Proc. of the 9th ACM Intl. Conf. on Multimedia*, 2001, pp. 187–193.

[4] W. Chai and B. Vercoe, "Structural analysis of musical signals for indexing and thumbnailing," in *Proc. of the 3rd ACM/IEEE-CS joint conference on Digital libraries*, 2003, pp. 27–34.

[5] M. Cooper and J. Foote, "Summarizing popular music via structural similarity analysis," in *Proc. of the IEEE WASPAA*, New York, NY, USA, 2003, pp. 127–130.

[6] J. Foote, "Visualizing music and audio using self-similarity," in *Proc. of the 7th ACM Intl. Conf. on Multimedia (Part 1)*, New York, NY, USA, 1999, pp. 77–80.

[7] M.D. Cooper and J. Foote, "Summarizing video using non-negative similarity matrix factorization.," in *IEEE Workshop on Multimedia Signal Processing*, 2002, pp. 25–28.

[8] M. Goto, "A chorus-section detecting method for musical audio signals," in *Proc. of the IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing*, 2003, pp. 437–440.

[9] MPEG-7, "Mpeg-7 multimedia content description interface - part 4: Audio iso/iec jtc 1/sc 29/wg 11," Tech. Rep., 2000.

[10] J. Wellhausen and H. Crysandt, "Temporal audio segmentation using mpeg-7 descriptors," in *Proc. of the Storage and Retrieval for Media Databases*, 2003.

[11] S. Lloyd, "Least squares quantization in pcm," *IEEE Trans. on Inf. Theory*, vol. 28, no. 2, pp. 129–137, March 1982.

[12] L. Lu, L. Wenyin, and H.J. Zhang, "Audio textures: Theory and applications," *IEEE Transactions of Speech and Audio Processing*, vol. 12, no. 2, pp. 156–167, March 2004.