AUDIO NOVELTY-BASED SEGMENTATION OF MUSIC CONCERTS

Dalia El Badawy, Patrick Marmaroli and Hervé Lissek

Laboratory of Electromagnetism and Acoustics (LEMA), Swiss Federal Institute of Technology in Lausanne (EPFL) – Station 11, 1015 Lausanne

e-mail: patrick.marmaroli@epfl.ch

The Swiss Federal Institute of Technology in Lausanne (EPFL) is in the process of digitizing an exceptional collection of audio and video recordings of the Montreux Jazz Festival (MJF) concerts. Since 1967, five thousand hours of both audio and video have been recorded with about 60% digitized so far. In order to make these archives easily manageable, ensure the correctness of the supplied metadata, and facilitate copyright management, one of the desired tasks is to know exactly how many songs are present in a given concert, and identify them individually, even in very problematic cases (such as medleys or long improvisational periods). However, due to the sheer amount of recordings to process, it is a quite cumbersome and time consuming task to have a person listen to each concert and identify every song. Consequently, it is essential to automate the process. To that end, this paper describes a strategy for automatically detecting the most important changes in an audio file of concert; for MJF concerts, those changes correspond to song transitions, interludes, or applause. The presented method belongs to the family of audio novelty-based segmentation methods. The general idea is to first divide a whole concert into short frames, each of a few milliseconds length, from which well-chosen audio features are extracted. Then, a similarity matrix is computed which provides information about the similarities between each pair of frames. Next, a kernel is correlated along the diagonal of the similarity matrix to determine the audio novelty scores. Finally, peak detection is used to find significant peaks in the scores which are suggestive of a change. The main advantage of such a method is that no training step is required as opposed to most of the classical segmentation algorithms. Additionally, relatively few audio features are needed which leads to a reduction in the amount of computation and run time. It is expected that such a pre-processing shall speed up the song identification process: instead of having to listen to hours of music, the algorithm will produce markings to indicate where to start listening. The presented method is evaluated using real concert recordings that have been segmented by hand; and its performance is compared to the state-of-the-art.

1. Introduction

At EPFL, the *Montreux Jazz Digital Project* aims at digitizing the archives of the Montreux Jazz Festival (MJF) concerts. A few figures: since 1967, 5000 hours of audio and video are stored on 10000 magnetic tapes. These archives contain approximately 40000 songs. The safeguarding of this heritage has begun in 2010 and will continue until early 2015 when 100% of the archives (1.2 Petabytes) will be processed and stored. In order to improve these archives, make them easily manageable, facilitate copyright management and help with quality control, several applications are desired such as the detection of audio events. In particular, it is required to know exactly which songs were sung in a given concert and even songs in a medley. However, due to the sheer amount of recordings available, it is quite cumbersome and time consuming to listen to each concert and note down the songs; it is therefore required to automate the process.

Detection of song changes using stochastic models is described in [1]; however, it is assumed that there are pauses between the songs which is not always the case for the MJF concerts. Related methods for semantic audio segmentation which deals with finding the constituents of a song like the intro, chorus, and bridge are described in [2], [3], [4], and [5]. In this paper, we describe a methodology to automatically segment MJF audio file concerts in order to aid human listeners and speedup the process of songs identification. Audio novelty scores presented in [4] are used for detection of song changes instead of structures within the same song.

The proposed approach is described in Section 2. Section 3 presents the experiments we carried out and the performance results. Finally, the conclusion and suggestions for future work are presented in Section 4.

2. Concert Segmentation

A concert recording generally contains several acoustic events including the songs, applause, and interludes. The first step in segmenting a concert into those separate events is to represent the audio recording in a format suitable for analysis. This is done by extracting the so-called audio features from the raw audio signal. Then, these audio features which describe the signal are used by the segmentation module to generate a proper segmentation.

2.1 Audio features

Audio feature vectors are computed from small successive frames of size N_f (in samples) with an overlap of N_o (in samples). The number of frames M contained in a signal of length N is given by the formula:

$$M = \left\lfloor \frac{N - N_f}{N_f - N_o} \right\rfloor + 1 \tag{1}$$

where $\lfloor . \rfloor$ stands for the floor function. In order to avoid edge effects when transforming it to the frequency domain, each frame is weighted by a Hann window (defined in [6] p. 397 for instance).

2.2 Segmentation

Once the audio is reduced to feature vectors (one per frame); the segmentation process can begin. The method used here follows [4]. The idea is to calculate the similarity between two successive frames; if they are different, then a change is possibly detected. So first, a distance measure is required.

2.2.1 Distance Measure

A distance measure is used to quantify the similarity (or dissimilarity) between two feature vectors. One such measure is the cosine similarity, which for two feature vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ is calculated as follows:

$$d_{ij} = \frac{< \mathbf{x}_i, \mathbf{x}_j >}{\| \mathbf{x}_i \| \| \mathbf{x}_j \|} \tag{2}$$

where $<.,.>$ denotes the inner product and $\| . \|$ the Euclidean norm. The values range between $[-1,1]$ with 1 for parallel vectors and -1 for antiparallel. It is a suitable measure because it is not affected by the energy levels where if the energy is low in two feature vectors, they can still have a high similarity score [4].

2.2.2 Similarity Matrix

Using the distance measure, the similarity between each pair of frames can be computed and placed in a similarity matrix $\mathbf{S}$ of size $M \times M$ where:

$$\mathbf{S}[i, j] = d_{ij}, \ \forall (i, j) \in \{1, 2, ..., M\}^2 \tag{3}$$

Figure 1 shows an audio excerpt and its corresponding similarity matrix. As indicated on the figure, the excerpt contains the ending and beginning of two different songs separated by applause. The checkerboard patterns where the changes occur are visible in the center of the similarity matrix.

2.2.3 Novelty Score

A novelty score is computed at each frame where the frames with high scores indicate a change and thus the audio should be segmented there. To calculate the score, a “checkerboard kernel” $\mathbf{H}_K$ of size $K \times K, \ \forall K \in \{2, 3, ...M\}$, is used. An example of such a kernel when $K = 4$ is provided below:

$$\mathbf{H}_4 = \begin{pmatrix} 1 & 1 & -1 & -1 \\ 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 \\ -1 & -1 & 1 & 1 \end{pmatrix} \tag{4}$$

Then, the 2D cross-correlation, r, between the kernel and the similarity matrix at each frame is calculated as follows:

$$\mathrm{r}(m) = \mathrm{corr}\left(\mathbf{H}_K, \mathbf{S}_K^m\right) \tag{5}$$

where $\mathbf{S}_K^m$ is the part of the similarity matrix with the same size of the kernel and centered at coordinates (m, m):

$$\mathbf{S}_m^K = \mathbf{S}\left(\left[m - \frac{K}{2} + 1, ..., m + \frac{K}{2}\right], \left[m - \frac{K}{2} + 1, ..., m + \frac{K}{2}\right]\right) \tag{6}$$

Thus, r is high at locations where the checkerboard patterns of the matrix match up with those of the kernel i.e. the positive and negative parts of the kernel multiply their respective positive and negative parts on the matrix.

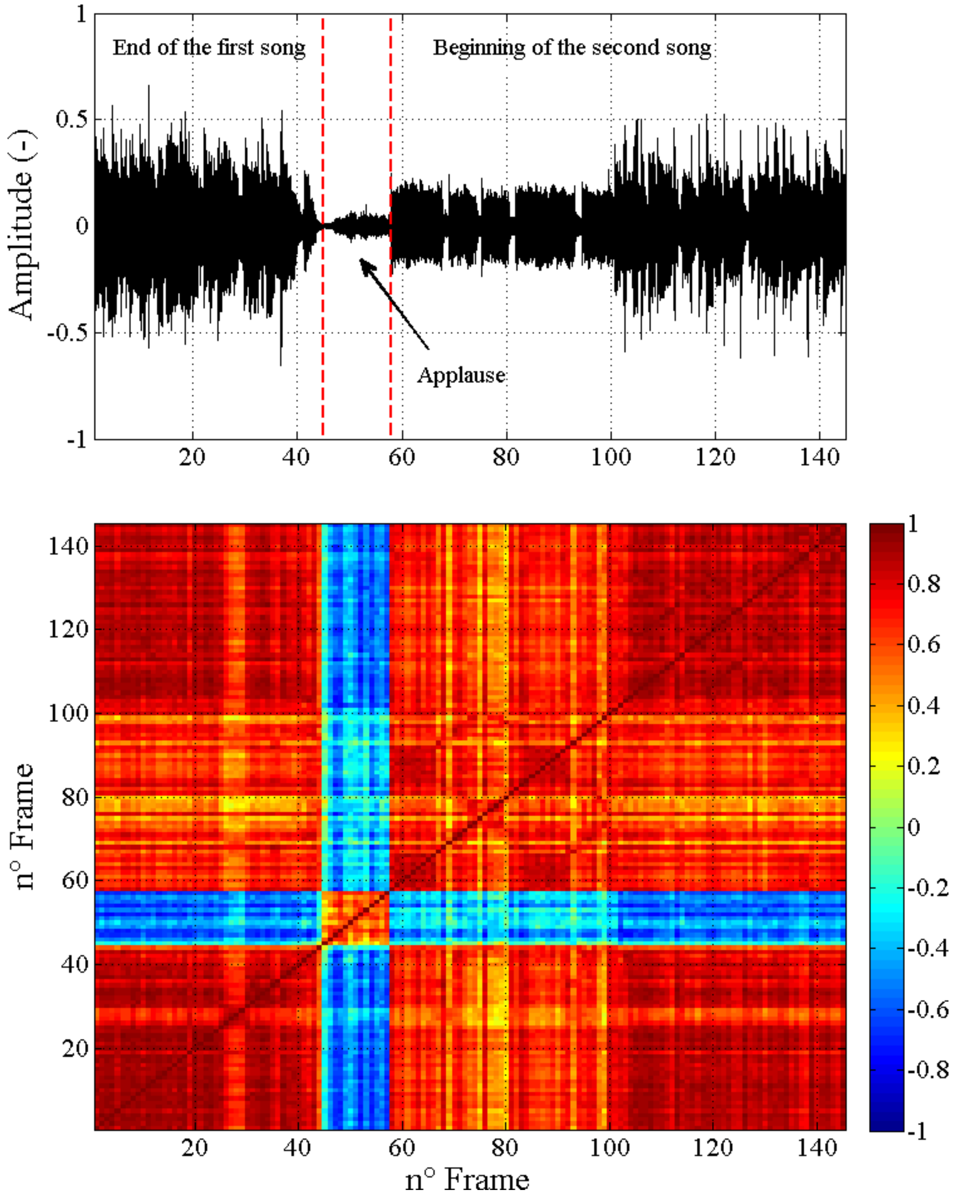

Figure 1: top: waveform of a song change, bottom: corresponding similarity matrix.

Note that K here determines the extent of boundary detection: smaller kernels detect changes on a lower level like individual musical notes. The kernel is correlated only along the diagonal of the similarity matrix to get a score along the time dimension. Then, the scores are normalized to between 0 and 1. Using the generated novelty score at each frame, peak detection is used (with a threshold value) to find the local maxima and thus determine the positions of change.

Only peaks above a threshold are considered as song changes. In our algorithm, the threshold is adjusted automatically to the data. The hypothesis is that when a change occurs, the indicative peak is much higher than the other scores or weaker peaks in the same audio window. Therefore, the threshold Λ is chosen according to the standard deviation of the scores in that window:

$$\Lambda = \mu + \lambda\sigma \qquad (7)$$

where μ, respectively σ, are the mean and standard deviation of the cross-correlation values, $\mathbf{r}=\left[r(1),r(2),...,r(M)\right]$, in the audio file, and λ is a scalar.

2.2.4 Evaluation

In order to evaluate the performance of the segmentation, two measures can be used: recall and precision. Given the ground truth i.e. correct segmentation, *recall* is the ratio of the number correctly detected boundaries to the number of boundaries in the ground truth (it measures how good the system is at finding the required boundaries and equals 1 if all the correct boundaries have been found). Whereas *precision* is the ratio of the number of correctly detected boundaries to all detected boundaries regardless they are right or wrong (it equals 1 if only correct boundaries were found):

$$recall=\frac{\text{number of correct boundaries}}{\text{number of ground truth boundaries}} \qquad precision=\frac{\text{number of correct boundaries}}{\text{number of all boundaries}}$$

3. Experiment

The dataset consists of 50 MJF concerts. All wave files are mono and sampled at 48 kHz. The total duration is of 60 hours 43 minutes 32 seconds. There was a total of 1132 manually placed segments (song, applause, interlude) that served as the ground truth for evaluating the system. A subset of this dataset was used for tuning the parameters (frame size, kernel size, detection threshold, audio features). It consisted of 5 randomly chosen concerts of total duration 6 hours 29 minutes 1 second with 99 manual segments. Audio features that have been compared are the spectrum, MFCC, chroma features, and their combinations; however, the spectrum gave the best results. Finally, the parameters giving us the best results are summarized in Table 1.

Table 1: Parameter values for the experiment.

Frame size (in samples)	N_f	8192 (170 ms)
Overlap (in samples)	N_o	0
Threshold parameter	λ	4.5
Kernel size (in number of frames)	K	64
Audio features	$\mathbf{x}$	Lower spectrum (50 first coefficients) by using the MIRToolbox [7].

Note that in [2] the checkerboard kernel was tapered using a Gaussian function. From our experience, not using a Gaussian weighting provides a slightly better result.

The segmentation results were evaluated with a tolerance of ± 10 seconds around the manual ground truth segmentation. We get a recall of 78.7% which is more than 8% higher than reported in [1]; while the precision is 23.3%. On inspecting some of the manual segmentations, it could be argued that some segments should actually start earlier or later since it is subjective; this affects the recall. However, for the precision, when listening to the extra added segments, it is usually the case that the beat stopped and started; or there is a sudden silence; or even a short impulse sound like a whistle.

4. Conclusion

This study was concerned with song change detection for the Montreux Jazz Festival concerts. The algorithm that has been developed is based on audio novelty detection. Several audio features were tried; the best was found to be the lower frequency spectrum. Other parameters like the frame size and whether to weight the checkerboard kernel were optimized. Finally, the algorithm was tested on 50 MJF concerts and achieved a recall value of 78.7%.

Suggestions for further improvement include overlapping the windows from which frames are extracted since it could be the case that a change between the windows is missed; also, overlapping the frames themselves could be useful. Moreover, for MJF concerts, it is interesting to note that there are 3 main audio events: music, speech, and applause which may occur between songs or even during a song. This extra knowledge can by all means be leveraged to improve the song change detection system. For instance, salient features for speech or applause can be used to detect their respective events which usually occur before or after songs and thus improve the song detection accuracy.

Future work includes matching the detected segments in the end to their respective events. And also, if a database of songs is available, each song can be matched to get its title and other metadata. Furthermore, it is also possible to utilize the video as well where during a song; the camera is most likely focused on the performers then switches to the audience at the end during the applause; so the scene changes can be a useful addition.

REFERENCES

[1] T. Plotz, G.A. Fink, P. Husemann, S. Kanies, K. Lienemann, T. Marschall, M. Martin, L. Schillingmann, M. Steinrucken, and H. Sudek. Automatic detection of song changes in music mixes using stochastic models. In *Proceedings of the International Conference on Pattern Recognition (ICPR)*, volume 3, pages 665–668, 2006.

[2] P. Romano, G. Prandi, A. Sarti, and S. Tubaro. Musical audio semantic segmentation exploiting analysis of prominent spectral energy peaks and multi-feature refinement. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1965–1968, 2009.

[3] M. Levy, M. Sandler, and M. 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)*, volume 5, pages V–V, 2006.

[4] J. Foote. Automatic audio segmentation using a measure of audio novelty. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, volume 1, pages 452–455 vol.1, 2000.

[5] Samer Abdallah, Katy Noland, Mark Sandler, and Mark S. Theory and evaluation of a bayesian music structure extractor. In *Proceedings of the Sixth International Conference on Music Information Retrieval (ISMIR)*, pages 420–425, 2005.

[6] S. Salivahanan, A. Vallavaraj, and C. Gnanapriya, editors. *Digital signal processing*. Tata McGraw-HIll Education, 2007.

[7] Olivier Lartillot and Petri Toiviainen. A Matlab toolbox for musical feature extraction from audio. In *Proceedings of the 10th International Conference on Digital Audio Effects (DAFx)*, pages 237-244, 2007.