AUDIO SEGMENTATION BASED ON MULTI-SCALE AUDIO CLASSIFICATION*

Yibin Zhang and *Jie Zhou*
Department of Automation, Tsinghua University, Beijing 100084, China
zyb00@mails.tsinghua.edu.cn, jzhou@tsinghua.edu.cn

ABSTRACT

Content-based audio segmentation plays an important role in multimedia applications. In order to segment accurately and on-line, most conventional algorithms are based on small-scale feature classification and always result in a high false alarm rate. Our experimental results show that large-scale audio can be more easily classified than small ones. According to this fact, we present a novel multi-scale framework for audio segmentation. First, a rough segmentation step based on large-scale classification is taken to ensure the integrality of the content of segments, which can avoid the consecutive audio belonging to the same kind being segmented into different pieces. Then a subtle segmentation step is taken to further locate the segmentation points for the boundary areas computed by the rough segmentation step. Experimental results show that a low false alarm rate can be achieved while preserving a low missing rate.

1. INTRODUCTION

Audio segmentation can provide useful information for both audio content understanding and video content analysis [1-6]. This paper will also address on this topic.

Among existing works, [1-4] are mainly aiming on speech/music discrimination and they cannot meet the needs of real applications. In [5], Zhang and Kuo proposed an approach to segment and classify audio data into speech, music, song, environmental sound, speech with music background, environmental sound with music background, and silence based on four short-time features. In it, the audio stream was segmented by locating abrupt changes in these features of two adjacent 1-s windows and then the segments were classified by a heuristic rule-based procedure. A classification rate was reported but no segmentation evaluation such as false alarm rate or missing rate was given. In [6], the authors proposed a method to classify and segment an audio stream into speech, music, environment sound, and silence. The first step is to distinguish speech and non-speech. The second step further divides non-speech class into music, environment sounds, and silence with a rule-based classifier. 1-second audio clip is taken as the basic classification unit in their algorithm. Similar with [5], only classification rate was reported in it.

For a segmentation problem, it is important to know the missing rate and false alarm rate of segmentation point detection. But unfortunately very few literatures deal with them. Actually, since the above algorithms are mainly based on small-scale classification, the false alarm rate of detecting segmentation points is always high (see Section 3). To overcome it, some complicated criterion is needed to smooth the classified results in order to obtain a good segmentation [5,6].

In real audio streams, such as broadcast and TV programs, the change frequency of audio types is usually rather low. That means the duration of the same type is relatively long. The classification result based on larger scale will be much better than that based on small ones. Therefore, the probability of segmentation error will be greatly reduced and the integrality of the content of segments is also ensured. Based on it, we present a novel approach of audio segmentation in this paper, which can be seen as a multi-scale framework. A rough segmentation step based on large-scale classification ensures the integrality of the content of segments, which can avoid the consecutive audio belonging to the same type being segmented into different pieces. After that, a subtle segmentation step will locate the segmentation points in the boundary areas computed by the rough segmentation step. Our algorithm is tested on a five-class audio segmentation problem, including piano, symphony, Beijing opera, popular song, and speech.

2. FEATURE EXTRACTION

To describe small-scale audio features, five short-time features (i.e. short-time energy (SE), zero-crossing rate (ZCR), mel frequency (MF) [7], spectral centroid (SC), and bandwidth (BW) [8]) can be used. They can be extracted from an audio frame. All audio data we used is

* This work is partially supported by Natural Science Foundation of China under grant 60205002 and 60332010, and Natural Science Foundation of Beijing.

sampled at a sampling rate of 11.025k/s. We split an audio signal into many frames and each frame contains 512 sample points (about 46 ms). The neighboring frames have an overlap of 112 sample points (about 10ms) in order to smoothen the feature values.

The middle-scale feature (spectrum flux (SF) [6]) is defined as the average variation value of spectrum between the adjacent two frames in a 1.5-s window. There is a 0.5-s overlap between neighboring windows.

Three large-scale features (low short-time energy ratio (LSTER) [6], high zero-crossing rate ratio (HZCRR) [6], and harmonious degree (HD) [7]) are derived from three short-time features, SE, ZCR and MF, respectively. They can be calculated from audio segments directly.

3. AUDIO CLASSIFICATION ON DIFFERENT SCALES

Audio segmentation is usually based on audio classification. A satisfying segmentation result is always produced by a good classification. In this section, we'll first discuss the audio classification problem under different scales.

We took an experiment of five-class audio classification. A database of total 747 audio signals (99 piano, 177 symphony, 147 Beijing opera, 156 popular songs and 168 speech records) is used in the experiments, which is much larger than the previous works. The length of each signal is about several minutes.

Four scale levels (huge-scale, large-scale, small-scale, and tiny-scale) are considered in our experiment. The whole database is split into two parts evenly. One is used for the training, and the other is used for the testing. In the huge-scale experiment, we extract a feature vector with fifteen dimensions from each audio signal. The feature vector is composed of three large-scale features, means and standard deviations of the rest features. Please note that, for the mel-frequency, we only need to calculate the mean and standard deviation of all the non-zero values. In the large-scale experiment, we collect 1100 6-seconds segments for each class and split them into training set and test set evenly. From each segment, we extract a feature vector just like the huge-scale experiment. In the small-scale experiment (similar with the usage in [5,6]), we collect 2200 1-seconds segments for each class and split them into training set and test set evenly. From each segment, we extract a 13-dimension feature vector, the features are the same as the ones used in large-scale experiment, except the mean and standard deviation of spectrum flux. In the tiny-scale experiment, we collect 2100 frames for each class and also split them into training set and test set evenly. From each frame, we can only extract five short-time features and directly use them to compose the feature vector.

In our experiment, we tried different classifiers, including Back Propagation Neural Network Classifier (BPNNC), K-NNC, MMDC, and SVMC [7]. Among them, BPNNC worked best. The experiment results are presented in Table 1.

Table 1. Audio classification results using BPNNC under different scale levels

huge-scale	large-scale	small-scale	tiny-scale
98.4%	91.2%	78.9%	66.1%

From the experiment results, we can see the trend that the longer the audio segment is, the better the classification result is. This is because that longer audio segment contains more information and its type property is more steady. It reveals that a larger scale classification instead of small-scale will result in a more robust segmentation. But in order to get an accurate segmentation point, the scale for classification cannot be too large.

4. ROUGH SEGMENTATION

In the rough segmentation step, we'll choose a 6-seconds window as the segmentation scale and the step is 3-seconds. Rough segmentation is achieved by classifying each 6-seconds window into an audio class. In our experiment, we assign each audio class a type label, 1 to 5, to represent piano, symphony, Beijing opera, popular song, and speech, respectively. Classifier and feature set are just the same as the ones used in the "large-scale" audio classification experiment described in Section 3.

Meanwhile, considering that the audio stream is always continuous, it is highly impossible to change the audio types too suddenly or too frequently. Under this assumption, we apply two simple smoothing rules upon the result of rough segmentation step.

Suppose s[i], i = 1,…,5, represents a consecutive type label sequence in the rough segmentation result. Then the first rule is set as

$$\textit{if}\,(s[1]=s[2]\,\&\,s[4]=s[5]\,\&\,s[2]\neq s[3]\neq s[4]),\quad \textit{then let}\ s[3]=s[2]. \tag{1}$$

This rule has three meanings: (1) Only consecutive type label units with the same type is believable. (2) If a type label unit is different from its adjacent four units while its adjacent units with the same audio type, it is considered as misclassification. For example, we should rectify the sequence "11211" into "11111". (3) If a type label unit is different from its adjacent four units while the anterior two units are also different from the posterior ones, it can be rectified as the previous or the succeeding audio type. In our approach, we'll uniformly rectify the middle unit according to its previous audio type. For example, we should rectify the sequence "11233" into "11133".

When i > 3, the second rule can be used as

$$if\ (s[1]=s[2]\ \&\ s[3]\neq\ldots\neq s[i]\ \&\ s[i+1]=s[i+2]),\quad (2)$$
$$then\ \text{reclassify}\ s[j],\ j=3,\ldots,i.$$

That means s[3] to s[i] are all unbelievable, then we'll merge s[3] to s[i] into a larger segment and reclassify it as a whole. For example, if "4412355" is a subsequence of the rough segmentation result, we'll reclassify the segment represented by "123".

Our rough segmentation algorithm is tested on one-hour audio stream, which is quite similar with the literature programs in radio or TV. There are 58 real segmentation points in this audio stream, and 56 of them can be successfully found. That means the accuracy ratio reaches over 96%. There are 28 false alarms and the false alarm ratio is lower than 0.5/minute.

An example of rough segmentation is given, in which the total length of the test audio stream is five minutes. There are five class audio data in it and the length of each class is one minute. The connection relationship from left to right is piano, symphony, Beijing opera, popular songs, and speech. The waveform of the test audio stream and its rough segmentation result are showed in Figure 1. There are four real segmentation points, and all of them have been found, the accuracy ratio achieves 100%, simultaneity, four false alarms have occurred.

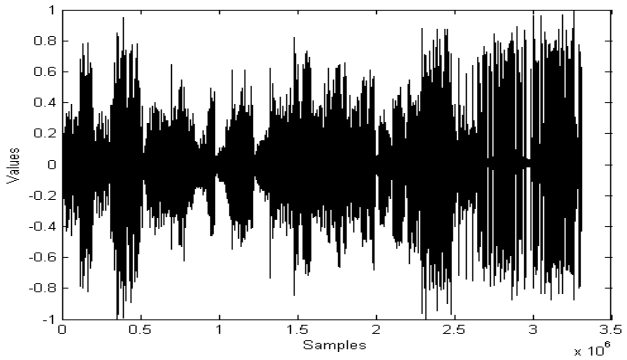
Through the rough segmentation step, we ensure the integrality of the contents of segments. In the next section, we'll accurately locate the segmentation points in the boundary areas computed from the rough segmentation result.

5. SUBTLE SEGMENTATION

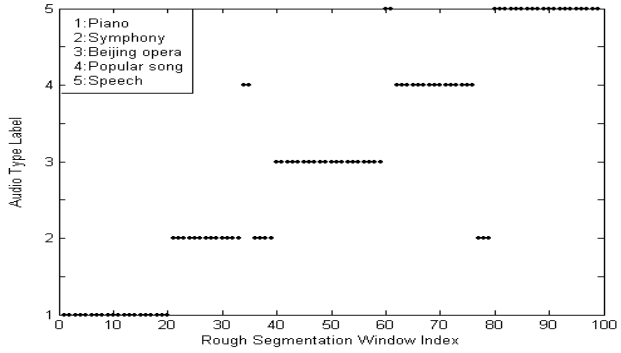
First, we'll scan the type label sequence and find out all of the pairs that have two adjacent but different units. Each pair represents an audio interval of 9-seconds. An appropriate segmentation point should exist in the interval. For instance, the pair, (1,4), represents such a 9-s audio interval that its foreside belongs to piano and its rearward belongs to popular song. The aim of the subtle segmentation is to locate this segmentation point in this interval.

In this small bound, the problem can also be transformed to a corresponding two-class classification question. Suppose P_* as the real segmentation point in a certain interval X=[P_i, P_j], whose foreside belongs to class I and rearward belongs to class J. We'll find P_* in [P_i, P_j]. For any point, P, it separates the interval X into two parts, $X_{i,p}$ and $X_{j,p}$. We define $Lv_{i,p}$ as the degree of $X_{i,p}$ belonging to class I, $Lv_{j,p}$ as the degree of $X_{j,p}$ belonging to class J. So, the evaluation function of P can be defined as:

$$LV_p = 0.5\left[\frac{Lv_{i,p}-MIN_i}{MAX_i-MIN_i}+\frac{Lv_{j,p}-MIN_j}{MAX_j-MIN_j}\right], \quad (3)$$



(a)



(b)

Figure 1. (a) Waveform of the test audio stream
(b) Final rough segmentation result

where, $MIN_i=\min_p(Lv_{i,p})$, $MAX_i=\max_p(Lv_{i,p})$;
$MIN_j=\min_p(Lv_{j,p})$, $MAX_j=\max_p(Lv_{j,p})$.

Let's analyze the trend of the evaluation function while P moves from P_i to P_j. When P is on the left of P_*, the segment $X_{i,p}$ really belongs to class I, so the value of $Lv_{i,p}$ will be larger and the value of $Lv_{j,p}$ is oppositely smaller. When P is on the right of P_*, the thing will be reverse. When P is just at the position of P_*, the segments $X_{i,p}$ and $X_{j,p}$ are all completely belonged to their classes, so the evaluation function will reach its maximum. Hence, we confirm the final segmentation point of interval X, P_*, which satisfies the equation:

$$LV_{p_*}=\max_p(LV_p), \quad (4)$$

Then, how to define $Lv_{i,p}$ and $Lv_{j,p}$? When the candidate point, P, separates the interval X into two parts, $X_{i,p}$ and $X_{j,p}$, we extract a feature vector from $X_{i,p}$ and $X_{j,p}$, respectively. The features are the same as the ones used in the "small-scale" audio classification experiment. Then we use a series of two-class classifiers, $BPNNC_{i,j}$, to get the values of $Lv_{i,p}$ and $Lv_{j,p}$. The two output values of classifier $BPNNC_{i,j}$ can be regarded as the values of $Lv_{i,p}$ and $Lv_{j,p}$, respectively.

We use the database of the "large-scale" audio classification experiment described in section 3 again. For

any two-class combination, its data set is split into two parts evenly. One is used for the training and the other is used for the testing. Ten classifiers have been trained and their capability is showed in Table 3.

Table 3. The capability of classifiers used for subtle segmentation, figures are got from test sets.

	2	3	4	5
1	93.5%	98.5%	99.5%	98.2%
2	---	95.8%	94.1%	97.5%
3	---	---	98.9%	96.9%
4	---	---	---	98.2%

In order to further evaluate the algorithm of subtle segmentation, we design the following experiment. We produce more than 200 9-seconds audio segments, including every possible situation. In them, the midpoints of segments are the real segmentation points. The experiment result shows that most segments whose segmentation-point errors are limited in $\pm$ 0.5-s.

An example is showed in Figure 2. In this test audio segment, the foreside belongs to piano and the rearward belongs to symphony. We can see that it is very hard to locate the segmentation point directly in the waveform, but the experiment result is very well. It is perfectly consistent with the theoretical analysis.

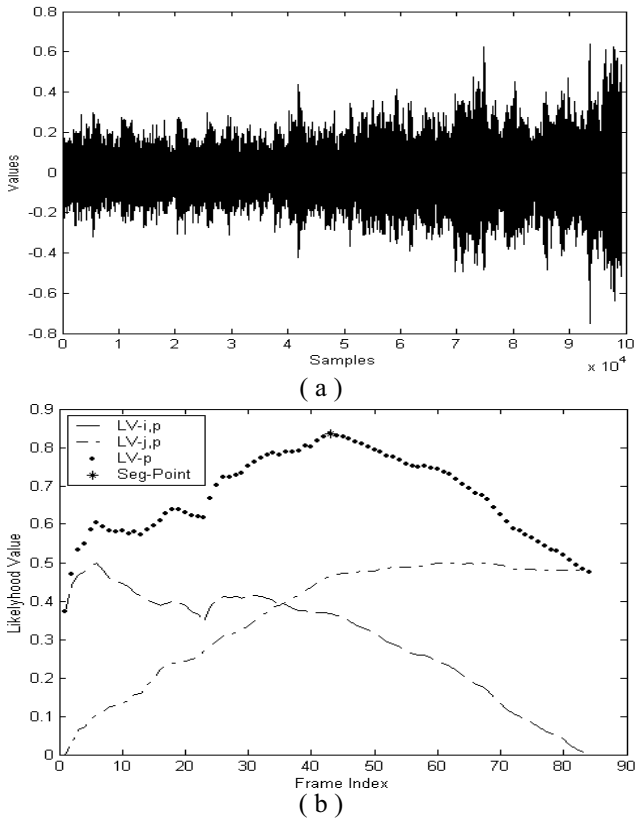


Figure 2. (a) Waveform of the test audio segment, (b) Subtle segmentation result of the test audio segment.

6. CONCLUSIONS

In this paper, we have presented our study on audio segmentation for applications in audio/video content analysis. We have proposed a novel audio segmentation scheme, which is a multi-scale framework. The rough segmentation step ensures the integrality of the content of segments. It avoids the consecutive audio belonging to the same kind being segmented into pieces, so a low false alarm rate can be achieved. The subtle segmentation step can further accurately locate the segmentation points in the boundary areas computed by the rough segmentation step.

In the future, our audio segmentation scheme will be upgraded to discriminate more audio classes and the performance of our segmentation algorithm will be improved. We'll also focus on developing an effective scheme to apply audio content analysis to assist video content analysis and indexing.

7. REFERENCES

[1] M.J. Carey, E.S. Parris and H. Lloyd-Thomas, "A Comparison of Features for Speech, Music Discrimination", in *IEEE Proc. ICASSP,* Phoenix, AZ, Vol.1, pp. 149-152, Mar 1999.

[2] K. El-Maleh, M. Klein, G. Petrucci and P. Kabal, "Speech/Music Discrimination for Multimedia Applications", in *IEEE Proc. ICASSP,* Istanbul, Turkey, Vol.4, pp. 2445-2448, Jun 2000.

[3] W. Chou and L. Gu, "Robust Singing Detection in Speech/Music Discriminator Design", in *IEEE Proc. ICASSP,* Salt Lake City, USA, Vol.2, pp. 865-868, 2001.

[4] J. Ajmera, I.A. Mccowan and H. Bourlard, "Robust HMM-Based Speech/Music Segmentation", in *IEEE Proc. ICASSP,* Orlando, USA, Vol.1, pp. 297-300, May 2002.

[5] T. Zhang and C.J. Kuo, "Audio Content Analysis for Online Audiovisual Data Segmentation and Classification," *IEEE Trans. Speech and Audio Processing*, Vol.9, No.4, pp. 441-457, May 2000.

[6] L. Lu, H.J. Zhang and H. Jiang, "Content Analysis for Audio Classification and Segmentation", *IEEE Trans. Speech and Audio Processing*, Vol.10, Issue.7, pp. 504-516, Oct 2002.

[7] Y.B. Zhang and J. Zhou, "A Study on Content-Based Music Classification", in *IEEE Proc. Seventh International Symposium on Signal Processing and Its Applications,* Paris, France, Vol.2, pp. 113 –116, July 2003.

[8] D.G. Li, I.K. Sethi, N. Dimitrova and T. Mcgee, "Classification of General Audio Data for Content-Based Retrieval", *Pattern Recognition Letters,* Vol.22, Issue.5, pp. 533-544, Apr 2001.