USING DTW BASED UNSUPERVISED SEGMENTATION TO IMPROVE THE VOCAL PART DETECTION IN POP MUSIC

Linxing Xiao, Jie Zhou

Department of Automation, Tsinghua University
Beijing 100084, P.R. China
xiaolx02@mails.thu.edu.cn

Tong Zhang

Hewlett-Packard Laboratories
Palo Alto, CA 94304, USA
tong.zhang@hp.com

ABSTRACT

Vocal part detection, which plays an important role in music information retrieval, is still a tough task so far. Previous works focused on short time features, which cannot capture some essential long term characteristics of singing. In this paper, we propose a Dynamic Time Warping based unsupervised segmentation algorithm to divide a pop song into homogeneous segments, which contain either vocal or pure music sound. This procedure makes it possible to design long term feature or classification schema to improve the accuracy of vocal part detection. We also present a segment level classification schema based on the result of segmentation. It will be shown that the classification accuracy is significantly improved.

Index Terms— Acoustic signal detection, Unsupervised learning, Dynamic programming, Pattern classification

1. INTRODUCTION

As one of the most important characteristics of music, information in the singing voice plays a very important role in automatic content based analysis of song in the field of music information retrieval, such as singer identification, music thumbnailing and lyrics transcription. Thus, the problem of vocal part detection attracts the interests of many researchers.

The solution to this problem lies in both modeling and feature selection. So far, several models have been proposed for vocal part detection. In [1], under the assumption that the adjacent frames in a song are independent, Maddage developed a SVM based algorithm for vocal\non-vocal discrimination. In [2], considering the continuousness of singing, Ellis employed Hidden Markov model (HMM) to model this problem. Furthermore, taking the structure information (intro, verse, and chorus) of pop song into account, TL Nwe [3] introduced a Multi Model HMMs to detect the singing part of pop music.

Lots of works on feature designing have been published as well. In [2], Berenzweig and Ellis used Posterior Probability Features (PPF) obtained from the acoustic classifier of a general-purpose speech recognizer. The performance of Mel-frequency Cepstrum Coefficients (MFCC) was also reported in [2]. In [1], the employed musical audio features are Linear Prediction coefficients (LPC), LPC derived cepstrum (LPCC), MFCC, spectral power (SP), short time energy (STE) and Zero Crossing Rate (ZCR). However, all these published works focused on short-term features, which cannot capture some essential long term characteristics of vocal part, such as the vibrato of singing. And we believe this is the major obstacle which hinders the development of vocal part detection.

Improvements are possible to be made if we can first divide a song into homogeneous segments, which contain either singing or pure music, because long term features can be extracted from each homogeneous segment. In fact, in many applications of audio indexing and information retrieval, blind segmentation methods have been used as a pre-processing step to improve the performance of entire system. In [4, 5], a Bayesian information cretierion based and a cumulative sum algorithm based segmentation algorithm are proposed for the application of broadcasting retrieval.

In this paper, we present a novel audio segmentation approach using the dynamic time warping (DTW) algorithm [6]. The DTW algorithm has been widely used to measure the similarity of two time series in different length. By employing dynamic programming, it finds the optimal alignment of two series which automatically divides the longer series into several segments. Because of the employment of dynamic programming, our segmentation algorithm is computation efficient and able to accurately divide a song into homogeneous segments. We also design a simple segment level classification schema based on the segmentation result. It will be shown that the classification accuracy is significantly improved.

2. DYNAMIC TIME WARPING ALGORITHM

Dynamic time warping (DTW) algorithm was first introduced to measure the similarity of time series of different length. Given two time series, T_1^M=$\{t(1),t(2),...,t(M)\}$ and S_1^N = $\{s(1), s(2),..., s(N)\}$, where T_1^M is shorter than S_1^N, DTW

finds the warping ϕ of the time dimension in S_1^N that minimizes the difference between the two series. And the minimal difference between them is:

$$D(T_1^M, S_1^N) = min_{\phi(k)} \sum_{k=1}^{N} d(t(\phi(k)), s(k)), \quad (1)$$

where $1 \leq \phi(k) \leq M$ and $d(x, y)$ is the Euclidean distance between x, y.

In application, $\phi(k)$ is restricted as an increasing function and in our algorithm, another constrain on $\phi(k)$ is that its slope is no more than 2. Under these constrains, DTW algorithm finds the optimal warping and calculate the minimal difference between two series via a width-first search scheme:

$$\begin{aligned} &D(S_1^{n+1}, T_1^m) = d(s(n+1), t(m)) + \\ &\min\{D(S_1^n, T_1^{m-2}), D(S_1^n, T_1^{m-1}), D(S_1^n, T_1^m)\}, \\ &\\ &D(S_1^N, T_1^M) = d(s(N), t(M)) + \\ &\min\{D(S_1^{N-1}, T_1^{M-2}), D(S_1^{N-1}, T_1^{M-1}), D(S_1^{N-1}, T_1^M)\}. \end{aligned} \quad (2)$$

Fig.1 shows an example of the optimal warping ϕ between two series. When the optimal warping ϕ is found, the segmentation of S is accomplished. And the number of segments of S is equal to the length of T. Therefore the DTW algorithm can be employed to divide a series into several segments with the number of segments pre-determined.

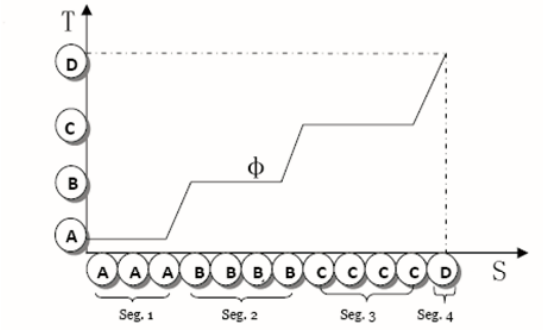


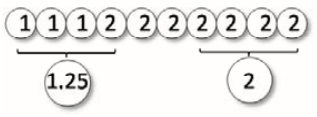
Fig. 1. Example of the optimal warping ϕ between two series .

3. DTW BASED SEGMENTATION

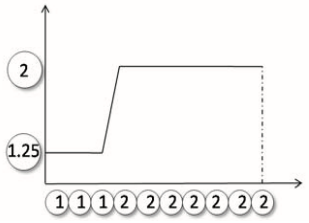
3.1. Single Change Point Detection via DTW

We first apply DTW algorithm to a simplified problem that there is only one change point in the incoming series. The situation of multiple change points in a series will be discussed in the next section.

To apply DTW algorithm, two series are required. Thus, to segment an incoming series S of N observations we first generate a 2-element series, which is called $Template$, such that the first element is the mean of few observations at the beginning of S and the second element is the mean of few observations at the end of S. Then the DTW algorithm is used to find the optimal warping, and meanwhile segment series S into two parts. Denote the hypothesized change point as r^*. Fig. 2 is a simple illustration of single change point detection. The curve in Fig 2 (b) is obtained from (2).



(a) Template generation



(b) Finding the optimal warping

Fig. 2. A toy example of DTW based segmentation.

After the change point is detected, a verification step is needed to evaluate whether this hypothesized change point is real or not. We train two Gaussian models $G_1 = N(\mu_1, \Sigma_1)$ and $G_2 = N(\mu_2, \Sigma_2)$ using observations before and after change point r^*. Then we calculate the verification value:

$$C = \sum_{k=r^*}^{N} l_k, \quad (3)$$

where $l_k = log(P(v(k)|G_2)) - log(P(v(k)|G_1))$ and $v(k)$ is an observation in S.

This verification value is compared to a pre-determined threshold β. If $C >= \beta$, we regard the hypothesized change point as a real one. If $C < \beta$, we think there is no change point in S.

3.2. Multiple Change Points Detection

When there are more than one change points in a series S, since the number of change points is unknown, the length of the $Template$ cannot be determined. One solution is to combine the DTW based detection algorithm with the approach proposed in [4][5], which sequentially detects the change points. It can be described as follows:

Step 1 Initialize the start of series as $f = 0$ and the end as $l = w\_size$

Step 2 Apply single change point detection to find a change in the observations from $s(f)$ to $s(l)$

Step 3 If a change is found at point r, set $f = r$ and $l = r + w\_size$, go to step 2

Step 4 If no change is found, $l = l + w\_incres$

Step 5 If $(l - f) > wmax$, $f = f + w\_incres$ and $l = f + w\_size$, go to $step2$.

It is necessary to bring several crucial factors of our segmentation algorithm for discussion. The first one is what feature should be used to represent a piece of audio. Different features will lead to different segmentation results. For example, if we use a sequence of short time energy (STE) to represent the audio, the energy change points will be detected. Since the major difference between singing and pure music is timbre, we use a timbre related feature MFCC to represent a piece of audio.

The second one is the initial number of observations to be test for a change point, w_size in $Step1$. w_size should not be too large, otherwise there will be more than one change point in these observations. Meanwhile, it should not be too small. In the verification step of Section 3.1, we estimated Gaussian models using observations before and after change point. If w_size is too small, the observations are not sufficient to train good Gaussian models. In our experiment, we empirically set w_size to 6 seconds to find a good trade off.

The final one is the verification threshold β. A large β will result in long segments which might not be homogeneous, because most hypothesized change points will be ignored. On the other hand, a small β will result in short segments whose homogeneity is guaranteed, because most hypothesized change points will be considered real. As a preprocess of vocal part detection, the homogeneity of resulting segments is very important. Thus, in our experiment we choose $-\infty$ as the verification threshold β.

4. SEGMENT LEVEL CLASSIFICATION

In this section, we introduce our classification schema based on the segmentation result. To guarantee the articulation of remainder of this section, we present the procedure of segmentation as follows:

First of all, an audio format song is converted to a sequence of MFCC features, denoted as $S = \{v(1), \ldots, v(N)\}$, where $v(i)$ is a MFCC feature vector. Then, our segmentation algorithm is applied to the MFCC feature sequence. As the result of segmentation, S is denoted as $\{seg_1, seg_2, \ldots, seg_K\}$, where $seg_i = \{v_i(1), \ldots, v_i(n_i)\}$, $n_1 + \ldots + n_K = N$.

The log likelihood ratio of one feature vector $v(i)$:

$$l_i = log(P(v(i)|singing)) - log(P(v(i)|music)) \quad (4)$$

is compared to a pre-determined threshold λ, which is usually 0. When $l_i > \lambda$, $v(i)$ is regarded as singing, and when $l_i \leq \lambda$, $v(i)$ is regarded as pure music. We call this procedure frame level classification.

Under the assumption that a vector in a segment is independent of other vectors, the likelihood of a homogeneous segment seg_i is:

$$P(seg_i|\alpha) = \prod_{k=1}^{n_i} P(v_i(k)|\alpha), \quad (5)$$

where α is $singing$ or $music$.

Thus for a homogeneous segment, the average log likelihood ratio is defined as:

$$\begin{aligned} l_i^s &= \frac{1}{n_i}[log(P(seg_i|singing)) - log(P(seg_i|music))] \quad (6) \\ &= \frac{1}{n_i}\sum_{k=1}^{n_i}(logP(v_i(k)|singing) - logP(v_i(k)|music)) \end{aligned}$$

Similar to (4), comparing the average log likelihood ratio of a segment to a pre-determined threshold λ, we can decide which class a segment belongs to. Fig. 3 illustrates the results of both frame level and segment level classification on the first 60 seconds of a song "Help me make it through the night".

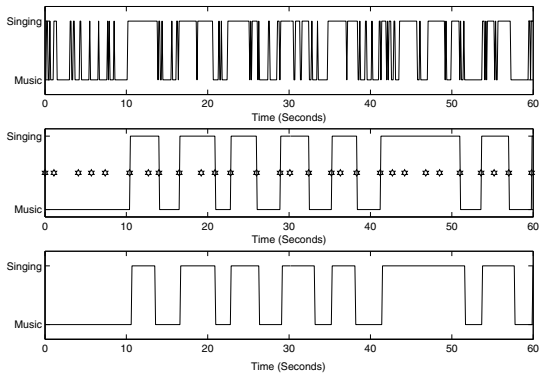


Fig. 3. Results of frame level and segment level classification. Upper pane is the result of frame level classification. Middle pane is the result of segment level classification. Stars represent the detected change points. Bottom pane is the ground truth.

5. EXPERIMENT

We evaluated the proposed algorithm on a collection of 106 pop songs, of which the average duration is 4 minutes. The

data set contains a variety of songs, including soft songs, Rock & Roll, Hip-Hop and other common types of pop songs. The music data is sampled at 22050Hz and windowed to frames of 0.1s without overlap. In every song, the change points between singing and pure music are manually labeled.

First of all, every song is converted to a sequence of MFCC features, which is calculated from a 0.1s audio frame. The parameters of segmentation algorithm are set as follows: The initial number of observations to be test for a change point, w_size, is set to 60. That means we search for a change point every 6 seconds. And the verification threshold β is set to $-\infty$. That means every detected change point is regarded true.

Two metrics are employed to evaluate the performance of segmentation algorithm, the missing probability (MP) and the false alarm probability (FA). They are defined as:

$$\begin{aligned} MP &= 1 - \frac{\#detections}{total\#true\ boundaries} \\ FA &= 1 - \frac{\#detections}{total\#hypo.\ boundaries} \end{aligned} \tag{7}$$

where $\#detections$ is the number of hypothesized boundaries that are within 0.5 second from ground truth boundaries, while $total\#true\ boundaries$ and $total\#hypo.\ boundaries$ are the numbers of ground truth and hypothesized boundaries, respectively.

The aforementioned parameters lead to following result: MP = 4.6%, FP = 88%. Low MP guarantees the homogeneity of each segment. The cost of low MP is high FP, which means a large amount of false change points are detected. Although FP is quite high, the average duration of each homogeneous segment is 3 seconds, and it is long enough to design long term classification schema to improve the classification accuracy. The segmentation procedure costs less than 1 second for a 4-minute song.

Then, we test the performance of segment level classification. In the above described data set, half songs are arbitrarily chosen to form the training set and the rest form the testing set. A 40-center GMMs classifier is trained from training set. The decision function of segment level classification is formula (6). A frame level benchmark algorithm, whose decision function is formula (4), is also implemented for comparison. We vary the decision threshold λ from -10 to 10 to get a ROC curve. This procedure repeats 20 times and Fig. 4 shows the statistics. The improvement of segment level classification is significant.

6. CONCLUSION AND FUTURE WORK

In this paper, a novel unsupervised segmentation method using the dynamic time warping algorithm is proposed. Based on the segmentation result, we present a segment level classification schema. Experiment shows that segment level classification significantly outperforms the frame level classification, even if the segment level classification schema is quite simple. Furthermore, dividing a song into homogeneous segments makes it possible to design long term features, which are able to capture some essential characteristics of singing. Thus, the main objective in future research is to design some long term features to get even better classification results.

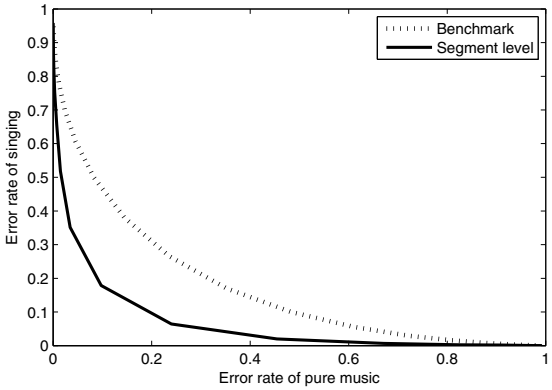


Fig. 4. The average ROC curve of 20 repetitive experiments. Dotted line is the result of frame level classification. Solid line is the result of segment level classification.

7. REFERENCES

[1] N. C. Maddage, C. Xu, and Y. Wang, "An svm-based classifiation approach to musical audio," *International Conference on Music Information Retrieval*, 2003.

[2] A. Berenzweig and D.P.W. Ellis, "Locating singing voice segments within music signals," *Applications of Signal Processing to Audio and Acoustics*, 2001.

[3] TL Nwe and Y. Wang, "Automatic detection of vocal segments in popular songs," *International Conference on Music Information Retrieval*, 2004.

[4] S. S. Chen and P. S. Gopalakrishnan, "Speaker environment and channel change detection and clustering via the bayesian information criterion," *DARPA Speech Recognition Workshop Proc.*, 1998.

[5] M. Omar, U. Chaudhari, and G. Ramaswamy, "Blind change detection for audio segmentation," *Proc. ICASSP*, 2005.

[6] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Transaction on Acoustics, Speech, and Signal Processing*, vol. ASSP-26, no. 1, pp. 43–49, Feb. 1978.