Word Level Automatic Alignment of Music and Lyrics Using Vocal Synthesis

NAMUNU C. MADDAGE, Royal Melbourne Institute of Technology (RMIT)
KHE CHAI SIM, HAIZHOU LI, Institute for Infocomm Research (I^2R)

We propose a signal-based approach instead of the commonly used model-based approach, to automatically align vocal music with text lyrics at the word level. In this approach, we use a text-to-speech system to synthesize the singing voice according to the lyrics. In this way, aligning the music signal with the corresponding text lyrics becomes the alignment of two audio signals. This study uses the results of music information modeling and singing voice synthesis. In music information modeling, we study different music representation strategies for music segmentation, music region indexing and region content descriptions; in singing voice synthesis, we generate singing voice by making use of music knowledge to approximate the target vocal line in terms of tempo. The experimental results on a 20-song database show 26.3% and 36.1% word level alignment error rates at eighth note and sixteenth note alignment tolerances respectively. The proposed approach presents an alternative and effective solution to music-lyrics alignment which may require less training dataset.

Categories and Subject Descriptors: H.5.5 [**Information Interfaces and Presentation**]: Sound and Music Computing

General Terms: Algorithms, Experimentation, Performance

Additional Key Words and Phrases: Music-lyrics alignment, rhythm analysis, music indexing, music information modeling, singing voice synthesis

ACM Reference Format:

Maddage, N. C., Sim, K. C., and Li, H. 2010. Word level automatic alignment of music and lyrics using vocal synthesis. ACM Trans. Multimedia Comput. Commun. Appl. 6, 3, Article 19 (August 2010), 16 pages.
DOI = 10.1145/1823746.1823753 http://doi.acm.org/10.1145/1823746.1823753

1. INTRODUCTION

The time alignment between singing voice and its text lyrics provides essential indexing information for music information retrieval. It has been a topic of interest to automate the music-lyrics alignment process in the music community. A reasonable solution to the problem may lead to many highly desirable applications such as vocal-lyrics synchronization in Karaoke music production industry.

Suppose that we have a piece of music that comes with a singing voice and its lyric, but the singing voice and the words in the lyric are not aligned to each other in time. A music-lyric alignment task is to

This research was conducted when the N. C. Maddage was a research staffer at the Institute for Infocomm Research (I^2R).
Authors' addresses: N. C. Maddage, School of Electrical and Computer Engineering, Royal Melbourne Institute of Technology (RMIT) University, Swanston Street, Melbourne, 3000, Australia; email: namunu.maddage@rmit.edu.au; K. C. Sim, H. Li, Institute for Infocomm Research (I^2R), South Tower Connexis, 1 Fusionopolis Way, Singapore 138632; email: {kcsim.hli}@i2r.a-star.edu.sg.
Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org.
© 2010 ACM 1551-6857/2010/08-ART19 $10.00
DOI 10.1145/1823746.1823753 http://doi.acm.org/10.1145/1823746.1823753

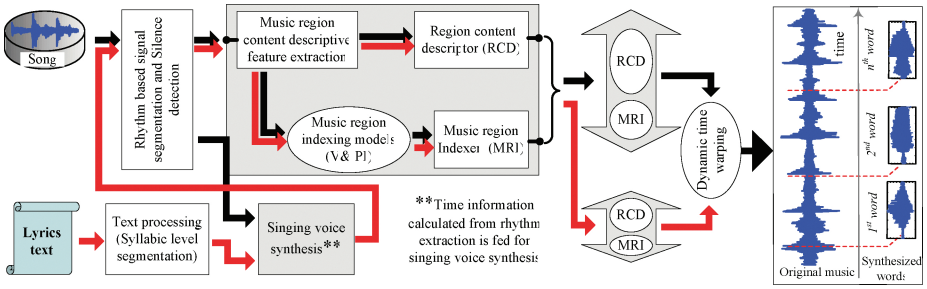

Fig. 1. Alignment process of music and synthesized audio lyrics.

automatically synchronize every word in the lyric with its vocal. Current state-of-the-art music-lyric alignment frameworks are inspired by speech recognition research, where acoustic models are forced to align the singing vocals. Typically, hidden Markov models (HMMs) are used to model the phoneme units. This can be considered as the model-based alignment approach. However, a singing voice, also known as the vocal in music, is different from speech in terms of production and perception. Two major challenges must be solved to use in the model-based approach for music-lyrics alignment.

(1) Natural speech has a relatively constant speaking rate and a narrow bandwidth of pitch variation. On the other hand, singing vocals comprise a sequence of phonemes with a larger variation in pitch values and durations to form the desired melody and music tempo. Consequently, the vocal signals may have very different spectral characteristics from the speech signals. Therefore, the acoustic mismatch between the speech models (phoneme models) and the vocal signals may degrade the music-lyric alignment quality.

(2) In practice, a singing voice is accompanied by strong background music, especially in popular songs. We consider background music as noise in the music-lyric alignment task. Background music differs from song to song, and it introduces another level of mismatch between a speech model and the singing voice. Background music suppression/removal can be an effective solution to this problem.

This article proposes a novel framework for music-lyric alignment at word level using the synthesized vocal signals. This framework is illustrated in Figure 1, and we assume the starting of a phrase in a song is known. Thus, our objective is to align the text words with the phrase which is in the form of audio signal. According to the framework, the lyrics are first converted into vocal signals using a text-to-speech (TTS) system. In this way, text and audio alignment task effectively transforms into an alignment problem between two audio signals (the original and synthesized vocals). Next, a rhythm-based segmentation technique is applied to the music signal to obtain frames (regions) whose size corresponds to the smallest note length in the song. Frame-based acoustic features are then extracted to construct a music region indexer (MRI), which specifies whether music a frame belongs to a pure instrument (PI) region or a vocal (V) region. We compare both soft and hard indexing schemes for MRI. We also construct a region content descriptor (RCD) using the same acoustic features to characterize the content within the music frame.

In the text lyrics processing, the words are first segmented into syllable sequences. Then we synthesize the segmented syllables using a text-to-speech (TTS) system according to time information computed from the rhythm-based signal segmentation process. The synthesized vocal is then processed in the same way as described above to obtain the MRI and RCD. Based on the frame level similarities computed between music signals and synthesized vocals computed using both MRIs and RCDs, we align the text lyrics with the original music.

This article describes the algorithms and presents results of an experiment to test them. It is organized as follows. Section 2 discusses prior work related to music-lyric alignment. Section 3 discusses rhythm-based signal segmentation and music information modeling. Section 4 discusses the of text-to-vocal rendering process. The music-lyric alignment and experimental results are explained in Section 5 and Section 6 respectively. Finally we conclude our study in Section 7.

2. RELATED WORK

Since the early 1980s, there have been research efforts to align a music signal to its corresponding musical score. Dannenberg [1984] proposed an algorithm that aligns live performance (solo instrumental line) with accompaniment (score). Music notes in the performance were identified using a pitch detection algorithm and then aligned with the score using dynamic programming. In the 1990s, speech recognition [Inoue et al. 1994] and pitch detection techniques [Grubb and Dannenberg 1997] were employed for vocal lyrics alignment [Korst and Geleijnse 2006]. These early studies focused on challenges in aligning lyrics with vocals without background music.

Recently, complex systems have been proposed to align lyrics with popular songs containing heterogeneous source mixtures. For example, Wang et al. [2004] assumed that the song has a 4/4 meter and the I-V_1-C_1-V_2-C_2-B-O structure, where I, V, C, B, and O represent the instrumental intro, verse, chorus, bridge, and outro respectively. First, low level feature extraction and information modeling such as vocal/nonvocal region detection and harmony contour detection are conducted with a quarter note time resolution. Section-level and line-level music lyric alignment experiments were performed. Iskandar et al. [2006] extended this work to word level alignment assuming that the line level music–lyrics alignment information is available. HMM-based acoustic models were trained to represent the phonetic units. These models were then aligned with the corresponding music piece using dynamic programming. The alignment error rate of the baseline reported in Iskandar et al. [2006] was about 50% for 3 songs. With additional local and global constraints, the error rate was reduced to 43.5%. Fujihara et al. [2006] were the first to apply harmony line suppression and nonvocal section removal to acoustic model based music-lyric alignment, as discussed in [Chen et al. 2006; Inoue et al. 1994; Korst and Geleijnse 2006]. In Korst and Geleijnse [2006], lyrics alignment was performed on pure vocal music that does not contain background music. The methods reported in [Chen et al. 2006; Fujihara et al. 2006; Inoue et al. 1994; Korst and Geleijnse 2006] are known as the model-based approach, where the text lyrics are represented as acoustic models and they are aligned to music using Viterbi forced-alignment.

Another direction of research follows the signal-based approach where the idea is to synchronize two signals. Hu et al. [2003], discussed a matching method between audio and the corresponding MIDI music. First, text based MIDI music was rendered to audio and the chroma features [Deutsch 1988; Sheh and Ellis 2003] were extracted from both the synthesized and original music. Then, these feature sequences were aligned using dynamic time warping (DTW). In the lyrics and Cantonese pop song alignment system [Wong et al. 2006], the vocal line was first enhanced by suppressing the bass clef notes. Then the syllables and their pitches were detected. Finally, features were extracted from both lyrics text and vocals and they are matched using the DTW algorithm [Sakeo and Chiba 1978]. In lyrics retrieval and alignment research, Korst and Geleijnse [2006] proposed an effective way to align similar text phrases in the different lyrics files posted for the same song in websites. Furini and Alboresi [2004] synchronized the compressed domain music and lyrics using the embedded alignment information in the music. However, this alignment process is considered manual alignment.

In the proposed signal-based approach, we attempt to synthesize vocals to be closer to the original vocals, using the time information extracted from the music. Most model-based approaches use HMM-based acoustic models to align the music to the lyrics. One limitation of the HMM models is the

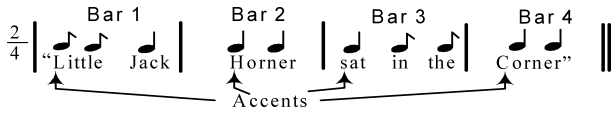

Fig. 2. Rhythmic group of words.

assumption of constant statistics within the states, which then become a piecewise-constant trajectory model. Furthermore, the model-based approaches have yet to find an effective way to incorporate time information into the alignment process. On the other hand, by synthesizing the vocals, a TTS system particularly one that uses unit concatenation is able to generate vocal signals with richer trajectory information, and incorporate time information. Thus, we expect more accurate performance in the alignment process.

3. MUSIC INFORMATION MODELING

Our objective is to align the vocal line with the lyrics at the word or syllable level. As depicted in Figure 1, the music is first segmented into frames at a rate proportional to the beat interval. Each frame is represented by low level features called the region content descriptor (RCD). These features are also used by the music region indexer (MRI). The remainder of this section describes how the RCD and MRI are computed.

3.1 Music Phrase Construction

In music, a phrase is a section of music that is relatively self contained and coherent over a medium time scale. In a music phrase, words or syllables fall on the beats [Royal Schools of Music 1949]. Beats are the steady throbs to which one could clap along with a song. Stronger beats are known as accents. Figure 2, shows the time alignment between the music notes and the words of a music phrase. Since accents are positioned over the important syllables, the time signature of this musical phrase is 2/4, that is, two quarter notes per bar.

Approximately 90% of sound generated during singing is voiced [Kim 2003]. These voiced sounds are held longer to align with the duration of a music note. In Figure 2, the time duration of the word "Jack" is equal to a quarter note and the length of a quarter note is defined according to the tempo of the song measured as the number of Beats per Minute (BPM). Rest notes, which constitute the silence (S) regions, are inserted between words to align the music phrases with its rhythm.

3.2 Frame Level Music Segmentation

Signal segmentation is an important step for information modeling, where the information within a segment is ideally considered stationary. From the musicology point of view, velocity of information flowing through a piece of music is inversely proportional to the inter-beat interval. A musical note can be considered as the smallest measuring unit in music [Jourdain 1997; Royal Schools of Music 1949], and the information within a note can be considered stationary. Thus, note-level segmentation is more suitable than fixed-length segmentation (FIX), which is typically used in speech processing [Rabiner and Juang 1993]. Since the interbeat interval is equal to an integer multiple of the smallest note duration, it is desirable to segment music into frames of this size. We refer to these frames as beat space frames.

We assume that the songs have constant tempo and 4/4 meter. To compute the duration of the smallest note, we first detect the beats and onsets in the music using a similar approach discussed in Duxburg et al. [2002]. Then, dynamic programming is used to find the equally spaced onset patterns to obtain the smallest note length. This gives us the frame size for music segmentation. We call

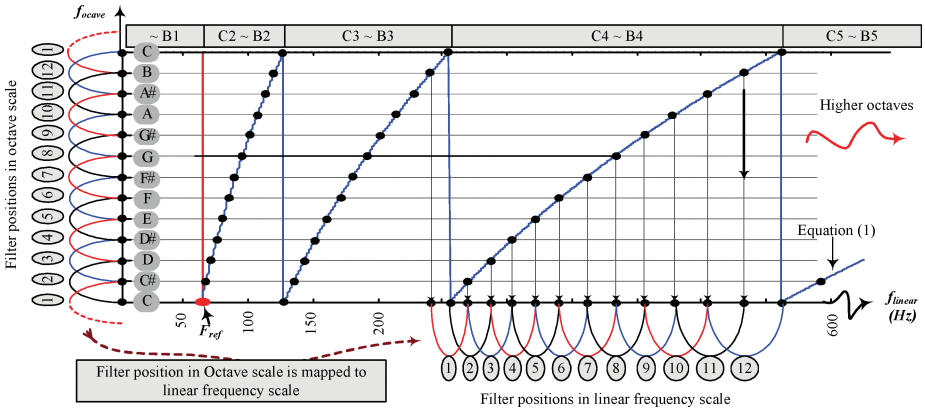

Fig. 3. Transformation of octave scale filter positions to linear frequency scale.

this segmentation the beat space/rhythm-based signal segmentation [Maddage et al. 2006]. Similar beat spaced/rhythm-based signal segmentation approaches have been discussed in [Ellis and Poliner 2006; Wang et al. 2004].

3.3 Music Region Content Modeling

In general, a music signal can be classified into five regions: pure instrumental (PI), pure vocal (PV), instrumental mixed vocal (IMV) and silence (S). Since PV regions are rare in popular music, we class both PV and IMV regions as vocal (V) regions. As depicted in Figure 1, after beat space segmentation, we identify frames of silence regions (S) using a short-time signal energy threshold algorithm [Maddage et al. 2006]. Acoustic features are extracted from nonsilence frames to index both the region identity (V or PI) and content, using the music region indexer (MRI) and the region content descriptor (RCD) respectively. We examine the Octave Scale Cepstral Coefficient (OSCC) and Log Frequency Cepstral Coefficient (LFCC) features for music region information modeling. In the following sections, computation of the OSCC and LFCC features, and construction of RCD and MRI, are discussed.

3.3.1 *Octave Scale Cepstral Coefficient (OSCC).* Octave scales that arrange the audible frequencies in octaves have been used for early music content analysis research [Brown and Puckette 1992; Maddage et al. 2006; Nwe and Wang 2004; Stevens et al. 1937]. As shown in Figure 3, OSCCs are computed using a filter bank, where overlapping filters equally divide an octave.

Equation (1), explains the relationship between linear frequency scale (f_{linear}) and octave frequency scale (f_{octave}), where Fs, N, F_{Ref}, and C are sampling frequency, number of FFT points, reference frequency and number of equal divisions in an octave respectively.

$$f_{octave} = \left[C * \log_2 \left(\frac{Fs * f_{linear}}{N * F_{ref}} \right) \bmod C \right]. \quad (1)$$

In our experiments, we set $C = 12$, $F_{ref} = 64$ Hz in Equation (1) so that 12 overlapping rectangular filters are positioned in each octave from C2B2 to C9B9 octave (64 $\sim$ 16,384) Hz. Thus, filters are a semitone apart [Sheh and Ellis 2003]. A response, $Y(b)$ of the b^{th} filter is computed according to Equation (2), where $S(.)$ is the frequency spectrum in decibel (dB), computed from the signal frame. $H_b(.)$ is the b^{th} rectangular filter where m_b and n_b are lower and upper frequency boundaries of the

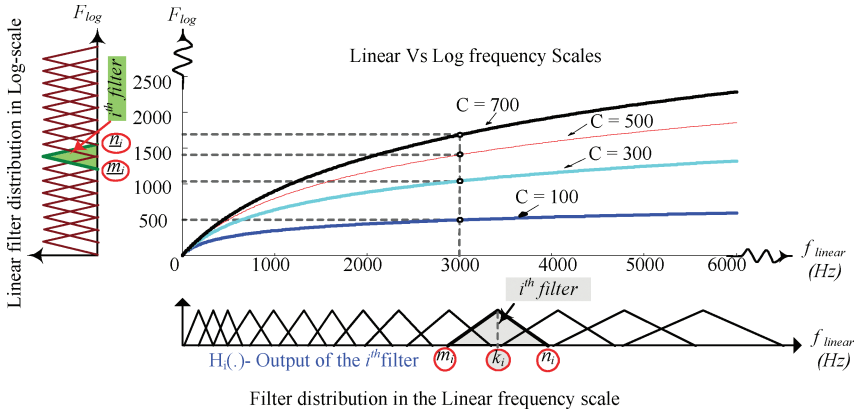

Fig. 4. Transformation of log scale filter positions to linear frequency scale.

filter.

$$Y(b) = \sum_{a=m_b}^{n_b} S(a) H_b(a) \tag{2}$$

Equation (3) describes the computation of the β^{th} cepstral coefficient where k_b, N and Fn are the center frequency of the b^{th} filter, number of frequency sampling points, and the number of filters respectively. In our case we use 12 filters per octave.

$$C(\beta) = \frac{2}{\beta} \sum_{b=1}^{Fn} Y(b) \cos\left(k_b \frac{2\pi}{N} \beta\right) \tag{3}$$

3.3.2 *Log Frequency Cepstral Coefficient* (*LFCC*). Even though computational steps are similar, filter positioning for LFCC and OSCC features are directly associated with Log10 and Log2 scales respectively. The relationship between log frequency scale F_{log} and linear frequency scale f_{linear} is explained in Equation (4), where C is the scaling parameter [John et al. 1999].

$$F_{\log} = \frac{C \log_{10}\left(1 + F_{linear}/C\right)}{\log_{10}(2)} \tag{4}$$

As shown in Figure 4, the filter distribution in f_{linear} can be changed when C is varied. We calculate LFCCs using Equation (2) and Equation (3). In our experiments, we vary C to effectively position the filters in order to maximize the region content sensitivity. C = 400 is found to be the optimum value in our task.

3.4 Music Region Indexer (MRI)

Vocal lines carry more descriptive information about the song than other regions. Previous techniques [Bartsch and Wakefield 2004; Berenzweig and Ellis 2001; Tsai et al. 2004] for music region modeling incorporated speech processing techniques, such as fixed-length signal segmentation (FIX) and Mel Frequency Cepstral Coefficient (MFCC) feature in the backend systems. However, recent studies suggested incorporation of music knowledge, such as beat space signal segmentation [Ellis and Poliner 2006; Maddage et al. 2006; Nwe and Wang 2004] and octave scale information characterization [Brown and Puckette 1992; Maddage et al. 2006] improves music information modeling.

A song can be modeled as a sequence of interwoven PI and V events. Our earlier experiments indicated OSCCs are more suitable than MFCCs for vocal and instrumental region detection [Maddage et al. 2006]. Thus using HTK toolbox [Young et al. 2006] we train two Gaussian mixture models (GMMs) using the corpus in [Maddage et al. 2006], to model each music region/event where $r_1 =$ PI and $r_2 =$ V. It was found by experimentation, that 20 OSCCs and 64 Gaussians in each GMM, maximizes the PI and V detection accuracy. At runtime, a music frame o_n is assigned an event $\hat{I}$, as either V or PI, based on Equation (5).

$$\hat{I} = \arg\max_{i=1,2} p(o_n \mid r_i) \tag{5}$$

Based on the two likelihood responses from the PI and V models $p(o_n \mid r_i),\ i = 1, 2$, we study two indexing terms [Maddage et al. 2006], namely hard-indexing and soft-indexing, to describe music frame related to music regions. A hard-indexing vector is a binary score where the highest model response is noted as one and the rest of the model responses are noted as zeros. A soft-indexing vector is constructed using model responses that are likelihood probabilities.

3.5 Region Content Descriptor (RCD)

An RCD that describes the contents in V and PI regions is constructed using low-level feature coefficients extracted from the music frames. Since our objective is to align the synthesized vocals with the corresponding frames in the vocal regions, we examined the OSCC and LFCC coefficients discussed above, for their effectiveness in describing contents in the vocal regions. However, due to strong instrumental lines and the noise in the music, the sensitivity of low-level features decreases. Thus, combining MRI and RCD, the region identity (V or PI) of the frames and the content characteristics within the frame respectively, strengthens the alignment between synthesized and original vocal frames.

4. VOCAL SYSNTHESIS USING TEXT-TO-SPEECH SYSTEM

Speech synthesis, also known as text-to-speech (TTS), is the conversion of normal language text into speech. In this work, the Festival TTS engine was used to synthesize the vocals [Taylor et al. 1998]. The synthesis of natural speech and singing vocals are different in terms of the phoneme duration and pitch contour. In Section 4.1, a concatenative TTS system is described, followed by a description of singing voice synthesis in Section 4.2.

4.1 Text-to-Speech (TTS) System

There are many techniques used for TTS. In this paper, a concatenation-based system was used. This system requires an inventory of speech units typically at the phoneme level called a voice database, from which the speech units are selected to form the desired pronunciation of the target speech. Thus, the voice database should be large enough to cover all possible sounds or pronunciations for a particular language. A simple database may use phonemes to represent the speech units. An English phoneme dictionary constructed by Carnegie Mellon University has 40 phonemes including silence. Using a diphone (phoneme pair) as a speech unit improves the quality of the synthesized speech [Hamon et al. 1989]. Figure 5 depicts the stages in a basic concatenation-based TTS system.

Text analysis is first performed on the input text to normalize special terms (e.g., numbers, dates, addresses) into spoken forms.

Prosody analysis is performed to identify the speaking rate and intonation of the input text. This stage is particularly important for natural speech synthesis. For singing voice synthesis, the prosody information corresponds to music score. Thus, syllables can be synthesized based on the note duration and fundamental frequency F0 of the note. We selected quarter note duration and pitch of the C4 note in the syllable synthesis process.

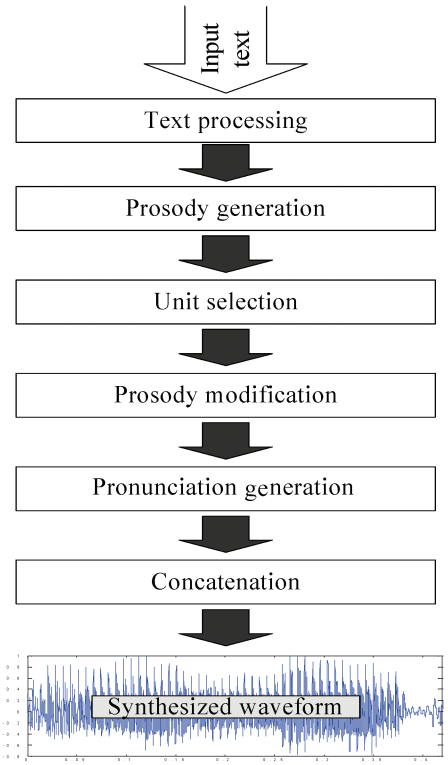

Fig. 5. Basic stages involved in a concatenate speech synthesis system.

Pronunciation generation converts the input text into a sequence of speech units according to its pronunciation. This conversion is typically achieved by looking up a phonetic dictionary or using a grapheme-to-phoneme (G2P) technique to generate the pronunciations if the words are not defined in the dictionary.

Unit selection selects the appropriate sequence of speech units from the database to represent the input text. To achieve high quality synthesis, a large database is required and a unit selection scheme that minimizes the matching and joining costs to choose the best unit for the given contexts.

Prosody modifications, such as time and frequency scaling [Bartsch and Wakefield 2004], are applied to the speech units to achieve the desired prosody. This stage plays an important role in synthesizing the singing voice and is described further in Section 4.2.

Concatenation is the final stage where the sequence of prosody-modified speech units are joined together to form the final speech output. Overlap-add methods are typically applied to avoid "clicks" at the concatenation boundaries. The Time Domain Pitch-synchronous Overlap-add algorithm [Makhoul 1975] is popular as it allows efficient real-time prosody modifications to the synthesized voice. Pitch synchronous concatenation ensures phase coherency between speech units.

4.2 Singing Voice Synthesis

Vocal singing can be synthesized by using a TTS system by modifying the time-scale and pitch-scale in the speech units according to music note duration and fundamental frequency of the note, respectively. Time-scale modification refers to changing of the speed of the waveform while keeping the other

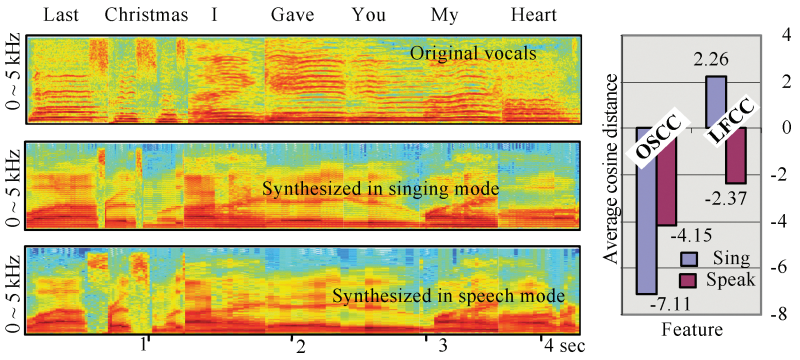

Fig. 6. Comparison of spectrograms of the original vocal clip "Last Christmas I gave you my heart" (top), synthesized vocals in singing mode (middle) and speech mode (bottom).

prosodic characteristics constant. Time-scaling is used to "stretch" the waveform to match the beat or tempo of the song. On the other hand, pitch-scale modification changes the pitch of the waveform while retaining the other prosodic attributes. Pitch-scaling is used to generate speech units at a particular singing note.

In this article, vocals are synthesized using the Festival synthesis toolkit [Taylor et al. 1998]. We assume constant tempo and 4/4 meter. Duration of the syllable is set to a quarter note duration computed using the smallest note detected duration in section 3.2. Incorporating not only time information but also harmonic information makes synthesized vocals closer to the original vocals. However, in this article we do not extract harmony contours. Thus, pitches of the synthesized syllables are set to the frequency of the middle C, that is, C4. An American male voice database was used to generate the singing voices for the experiments. Figure 6 shows the spectrograms of a song from the original vocal clip as well as the synthesized vocal line using both speech and singing modes. To find the similarity of the synthesized vocals and the original clip, we first extracted 10 OSCCs and 10 LFCCs from the 16^{th} note length frames the tempo of the clip is 82BPM and meter assumed is 4/4. Then the cosine distance between synthesized and original vocal frames are computed according to Equation (8). Note that the distance indicates the similarity between two frames. As shown in the bar chart in Figure 6, average cosine distance is much higher with LFCC features and with singing mode vocal synthesis. This implies that LFCC features are more suitable for describing the content in music regions that is, RCD (Section 3.5).

5. MUSIC & SYNTHESIZED SINGING VOICE ALIGNMENT

For the music and synthesized vocal alignment, we use DTW [Sakeo and Chiba 1978], which is a pattern matching algorithm with a nonlinear time-normalization effect. In this algorithm, the timing differences between two signals are eliminated by warping the time axis of one so that the maximum coincidence is attained with the other. With the spectral features discussed in Section 3.3, singing voice A and synthesized voice B can be expressed by two sequences of feature vectors:

$$A = a_1, a_2, \ldots a_i, \ldots a_I \qquad (6)$$
$$B = b_1, b_2, \ldots b_j, \ldots b_J$$

The timing alignment between the signals can be depicted by a sequence of points $c = (i, j)$,

$$F = c(1), c(2), \ldots c(k), \ldots c(K), \qquad (7)$$

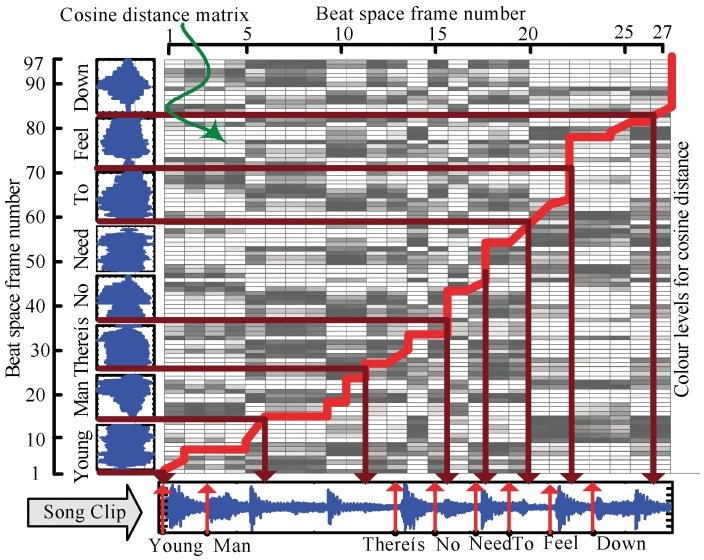

Fig. 7. Phrase alignment between singing vocals with respective synthesized vocals. Sixteenth note length beat space frame is 123.052ms.

where $c(k) = (i(k), j(k))$. This sequence can be considered to represent a function that approximately realizes a mapping from the time axis of pattern A onto that of pattern B as indicated by F in Equation (7). To estimate the mapping function F, we define a distance metric between the two feature vectors.

$$d(c) = d(i, j) = \frac{a_i \bullet b_j}{\|a_i\| \bullet \|b_j\|} \tag{8}$$

Then the weighted summation of the distances on the warping function F becomes

$$D(A, B) = \left(\sum_{k=1}^{K} d\left[c\left(k\right)\right] \bullet w\left(k\right) \right) \Big/ \sum_{k=1}^{K} w(k), \tag{9}$$

where $w(k)$ is a nonnegative weighting coefficient, which is introduced as a local constraint to control the flexibility of the warping function. The denominator $\sum_{k=1}^{K} w(k)$in Equation (9) is used to normalize the effect of K. Now the time sequence alignment becomes an optimization problem that minimizes the objective function of Equation (9). Taking note that the time-scaling (see Section 4.2) is effective in generating singing voice of desired tempo, we adopt a simplified path and symmetric form weighting coefficient as the restrictions on the warping function [Sakeo and Chiba 1978]. The warping function F, in Equation (7), provides a frame level time alignment between pattern A and B, with a resolution of a frame length. We use beat space frames.

Figure 7 illustrates an alignment of a song clip and its synthesized lyrics. In this example, we considered only the RCD of the acoustic event modeled by LFCC features. As explained in Section 3.5, we are not only interested in incorporating RCD but also the MRI of the acoustic event. For synthesized voice, we apply the same hard indexing method to label the frames because we know that they are all pure vocals. Thus, we label the synthesized voice with 1 and 0 for the V and PI models. Incorporating both MRI and RCD information, the distance $d(i, j)$ in Equation (8) is rewritten by Equation (10).

$$d\left(i, j\right) = d(i, j)_{RCD} \times d(i, j)_{MRI}, \tag{10}$$

Table I. Song Database Used for Music-Lyrics Alignment

No	Song Name	Artist	Year	No	Song Name	Artist	Year
1	YMCA	Village People	1978	11	One Love	Blue	2002
2	Super Trouper	ABBA	1980	12	Sleeping Child	MLTR	1993
3	When You Say Nothing at All	Ronan Keating	2000	13	That's Why (You Go Away)	MLTR	1995
4	Every Breath You Take	The Police	1983	14	Hotel California	Eagles	1976
5	I Just Called to Say I Love You	Stevie Wonder	1984	15	Nothing's Gonna Change My Love for You	Glen Medeiros	1989
6	Lady in Red	Chris DeBurgh	1986	16	Words	Boyzone	1996
7	25 Minutes	MLTR	1993	17	I Want to Know What Love Is	Foreigner	1984
8	How Many Hours	MLTR	1995	18	(I Just) Died in Your Arms Tonight	Cutting Crew	1986
9	Paint My Love	MLTR	1996	19	Unbreak My Heart	Toni Braxton	1996
10	The Actor	MLTR	1991	20	Save Tonight	Eagle-Eye Cherry	1998

where $d(i, j)_{RCD}$ and $d(i, j)_{MRI}$ are the similarity measures of as in Equation (8), for RCD and MRI, respectively. We use LFCC features to form the RCD vectors; a two-dimensional output scores as in Equation (5) to form the MRI vectors. For example, the vector $[0.6, 0.4]^{\mathrm{T}}$ indicates that the frame has a probability of 0.6 being a vocal. For the synthesized voice, only the vocal portions are generated. Therefore, the MRI feature for the synthesized voice is always $[1, 0]^{\mathrm{T}}$. Hence, $d(i, j)_{MRD}$ in Equation (10) becomes the vocal probability and the total distance measure is the RCD distance weighted by the vocal probability.

6. EXPERIMENTS

This section describes the experiments conducted to test the music-lyrics alignment algorithm we have described. We used the same database and evaluation setup as those described by Iskandar et al. [2006]. We also assume that the position of the music phrase in the text lyrics file is known. This database consists of 20 English pop songs with one channel sampled at 11kHz and 16 bits per sample. The songs in the database are listed in Table I. The words of the lyrics are marked with the start time based on the listening tests. The database contains 6580 words. In this work, the lyrics are synthesized into vocal signals using quarter note duration per syllable and constant pitch at the C4 frequency.

First, alignment performance using synthesized vocals in speech and singing modes is compared. The preliminary distance similarity analysis discussed in Section 4.2 indicates that the singing mode in the Festival synthesis toolkit [Taylor et al. 1998] is more suitable for vocal synthesis compared to the speech mode in it. Furthermore, by using the first 3 songs in Table I, we compare the alignment error rates between the speech and singing modes. We only considered RCD for this alignment experiment. Distance similarity experiment discussed in section 4.2 also indicated that the similarity matching between synthesized and original music region contents is more improved when LFCC feature is used instead of OSCC feature. Thus, we constructed the RDC using LFCC feature. It was empirically found that the best alignment results were obtained using the following settings: $C = 400$ in the Equation (3), the number of filters, $Fn = 15$, and the number of coefficients, $\beta = 1$ for computing the LFCC feature. During the vocal synthesis, each syllable unit was synthesized with single-beat duration (one quarter note length). The pitch was set at the frequency of the middle C for the singing mode. Sixteenth and eighth note frame sizes were empirically selected for feature extraction in original song and synthesized words respectively.

Table II. Alignment Error Rate When Vocals are Synthesized Using Singing Mode and Speech Mode

Exp No	Experiments	Alignment error rate (%)		
		T = l/4	T = l/8	T = l/16
1	RCD only (synthesized vocals using singing mode)	18.5	30.7	44.2
2	RCD only (synthesized vocals using speech mode)	22.2	38.58	49.7

Table III. Word-Level Alignment Error Rate with Respect to Tolerance T (in bars)

Experiments				Alignment error rate (%)		
Exp No	Feature	Frame size for original song	Frame size for synthesized words	T = l/4	T = l/8	T = l/16
1	LFCC, C = 400, β = l, Fn = 15	16^{th} note	Quarter note	16.3	29.8	39.1
2		16^{th} note	8^{th} note	15.8	27.9	38.8
3		16^{th} note	16^{*} note	16.6	30.1	41.7
4		16^{th} note	32^{nd} note	18.4	32.6	45.5
5		30ms	30ms	20.7	40.4	52.2
6	OSCC	16^{th} note	16^{*} note	19.9	35.4	46.0

Table II reports the lowest alignment error rates with different tolerances, when vocals are synthesized using both singing and speech modes. Tolerance levels T = 1/4, 1/8 and 1/16 denote the quarter, eighth and sixteenth note duration alignment tolerances respectively. The results in Table II reveal that the singing mode consistently outperforms the speech mode in vocal synthesis, with the former achieving 5% lower alignment error rate with sixteenth note duration alignment tolerance (T = 1/6). Therefore, the singing mode vocal synthesis is used in subsequent experiments.

Next, we compare the performance of using LFCC and OSCC features as RCD coefficients using full database. For the OSCC features, it was found that the coefficients computed from a filter bank positioned on the C1B1 to C7B7 octaves with 12 filters in each octave, gives the best alignment results among other settings investigated. On the other hand, the optimum parameters for the LFCC feature was empirically determined as C = 400 (cf. Equation (4)), $Fn = 15$ (number of filters), and $\beta = 1$ (number of coefficients).

The alignment results are summarized in Table III. Exp No-1 to Exp No-5 use the LFCC features as RCD while Exp No-6 uses the OSCC features. By comparing Exp No-3 and Exp No-6, it was found that the LFCC features outperformed the OSCC features as RCD. Furthermore, by examining Exp No-1 to Exp No-4, the optimum frame size for the original and the synthesized vocals were found to be at the 16th and 8th note length. As a contrast experiment, Exp No-5, which uses a fixed frame size of 30ms, gave the worst performance. From these results, we conclude that beat space frames are more suitable for vocal signals. This result further supports the claim in section 3.2 that the information within the beat space frames are more stationary than within fixed length frames. Therefore, the RCD configuration in Exp No-2, which gave the best alignment performance, will be used in subsequent experiments.

Next, we combine MRI with RCD for the alignment process. As explained in Section 3.4, MRI indicates whether the given frame belongs to the vocal region (V) or to the pure instrumental region (PI). We found that for MRI, the soft indexing scheme outperforms the hard indexing scheme. This result is expected because the hard indexing scheme may remove correct vocal frames from the alignment process due to misclassifying the vocal regions as pure instrument. On the other hand, the soft indexing scheme incorporates both PI and V model responses and does not remove the PI frames in the alignment process. Thus, the soft indexing scheme reduces the effects of V-PI classification errors and enhances the vocal region characteristics in RCDs, which reduces the alignment error rate.

Table IV. Alignment Tests with and Without MRI Information

Exp No	Experiments	Alignment error rate (%)		
		T = l/4	T = l/8	T = l/16
1	RCD only	15.8	27.9	38.8
2	RCD + MRI	11.6	26.3	36.1

Table V. Alignment Error Rate Comparison

Exp No	Experiments	Test on first 3 songs Allignment error rate (%)		
		T = l/4	T = l/8	T = l/16
Proposed signal based approach				
1	RCD only	18.5	30.7	44.2
2	RCD + MRI	15.8	26.3	40.7
Model based approach (model adoptation using 20 songs)				
3	Discreatize to l/16th bar frame	19.7	31.1	46
4	Global and local constrains(L+G)	18.7	28.8	43.5

Table VI. Error Rates in Model-Based Alignment Approach when the Adaptation Dataset Increased by 50%

Exp No	Experiments–Model based approach Model adoptation using 30 songs	Test on 3 songs Alignment error rate (%)		
		T = l/4	T = l/8	T = l/16
1	Discreatize to 1/16th bar frame	18.8	30.5	45.7

Table IV shows the average alignment error rates for experiments on 20 songs using RCD with and without MRI. The RCD+MRI (with soft indexing scheme) in Exp No-2 gave consistent improvements compared to using only the RCD in Exp No-1. An absolute rate reduction of 2.7% with T = 1/16 resolution was observed.

We also compared results of the proposed signal-based approach with the model-based approach as discussed by Iskandar et al. [2006]. To obtain comparable numbers, we used the same 3 test songs and the results are given in Table V. Both Exp No-1 in the signal-based approach and Exp No-3 in the model-based approach do not use the music region identity information. Thus results of these two experiments are directly comparable. There is approximately 2% (with T = 1/16) absolute alignment accuracy improvement with the signal-based approach compared to model-based approach. We then compare the best performing model-based approach (Exp No-4), which incorporates the music structure information, with the best performing signal-based approach (Exp No-2) which incorporates the music region identity. It is noticed that the signal-based approach outperforms the model-based approach by around 3% with T = 1/16.

Next we evaluated performance of the model based approach when the adaptation dataset is increased. The model-based approach, as reported in Exp No-3 and Exp No-4 in Table V, used a set of 20 songs for model adaptation. These songs are different from the three songs used for testing. We increased the model adaptation dataset by 50% to 30 songs and repeated the experiments with the same 3 test songs. The results are reported in. Notice that the alignment error rate is reduced by 0.3% (T = 1/16) with the additional data. With respect to the signal based approach result of Exp No 1 in Table VI, this alignment error reduction in the model based approach is around 16% relative alignment accuracy improvement with the 50% increase of model adaptation training set. Therefore, the model-based approach may work better with a large training set, while the signal-based approach performs well with a smaller training set.

7. DISCUSSIONS AND CONCLUSIONS

In this article, we have proposed a signal-based approach, instead of the commonly used model-based approach, for text lyrics alignment with corresponding music. We synthesized a singing voice to represent the lyrics and performed the alignment between the synthesized vocal and the original music. To conduct the music-lyrics alignment, we need to process both the music and the text streams. For music, we first analyze the rhythm and segmented the music into smallest note size frames. We then construct the music region indexer (MRI) and the region content descriptor (RCD) to characterize the music frame. The *Log Frequency Cepstral Coefficient* features extracted from music frames are used as the RCD. In addition, the *Octave Scale Cepstral Coefficient* features were used by the MRI to index the Vocal (V) and pure instrumental (PI) regions. Both soft and hard indexing were examined. The hard indexing scheme discards the PI frames in the alignment stage while the soft indexing scheme incorporates both V and PI model responses in the alignment stage. Soft indexing scheme was found to yield superior performance. Furthermore, it was found that incorporating the MRI with RCD leads to further improvement in alignment accuracy.

For the text lyrics, the syllables of the words are synthesized with the duration of the quarter note, which is calculated based on the computed duration of the smallest note length. A concatenation-based text-to-speech system was used to synthesize the lyrics in singing mode using the Festival synthesis toolkit [Taylor et al. 1998]. We assume 4/4 meter and a constant tempo in the song. The synthesized vocal signals are segment into 8th note size frames and the MRI and RCD are constructed. The dynamic time warping algorithm is used for the frame level alignment between the synthesized vocal line and the music. The word level alignment error rates are reported with quarter note (T = 1/4), eighth note (T = 1/8) and sixteenth note (T = 1/16) duration alignment tolerances.

It was found that incorporating the MRI with the RDC gives a 2.7% alignment error reduction over that using RCD alone. The proposed signal-based approach gives an average alignment error rate of 36.1% at T = 1/16. Furthermore, the proposed signal-based approach outperformed the model-based method [Iskandar et al. 2006] by 3% (T = 1/16). By increasing the model adaptation training set by 50%, the model-based approach obtains an overall 16% error reduction with respect to the results of the signal-based approach. This result indicates that the model-based approach may perform better with a very large adaptation dataset whereas the signal-based approach is more robust for smaller training dataset.

We have synthesized the singing vocals at a constant pitch (middle C) and duration (1 beat per syllable). The DTW algorithm is able to cope with the time scale mismatch. To overcome the mismatch in the pitch values, we have selected one song and normalized the syllabic level pitch values of the vocal in the original song to middle C to match the synthesized vocals. Pitch scaling was achieved using the frequency-scale modification and PSOLA [Makhoul 1975]. The preliminary result shows that tone normalization yielded a promising gain of 3% absolute improvement for one song. For future work, we will carry out experiments systematically, including tone normalization using automatic pitch detection techniques. Furthermore, the synthesis of the vocals can be improved if we know more information about the original vocal line in the music. By adapting the synthesizer to match the singing style and the singer characteristics we are able to generate the synthesized vocals that are closer to the original vocals. The other direction is the music information modeling. At this stage we do not extract harmony contour of the music that could potentially lead to a better synthesis of the singing voices.

ACKNOWLEDGMENTS

We thank Mr. Denny Iskandar for sharing the database and algorithm. We also thank the two reviewers for their valuable comments and suggestions regarding the research direction. We appreciate very much Professor Lawrence Rowe for helping us improve the quality of the article.

REFERENCES

ROYAL SCHOOLS OF MUSIC. 1949. *Rudiments and Theory of Music*. The associated board of the Royal Schools of Music. London.

BARTSCH, M. A. AND WAKEFIELD, G. H. 2004. Singing voice identification using spectral envelop estimation. *IEEE Trans. Audio Speech Lang. Proc. 12*, 2, 100–109.

BERENZWEIG, A. L. AND ELLIS, D. P.W. 2001. Location singing voice segments within music signals. In *Proceedings of IEEE Workshop on Applications of Signal processing to Audio and Acoustics (WASPAA)*.

BROWN, J. C. AND PUCKETTE, M. S. 1992. An efficient algorithm for the calculation of a constant Q transform. *J. Acoustic Soc. Amer. 92*, 5, 1933—1941

CHEN. K., GAO, S., ZHU, Y. AND SUN, Q. 2006. Popular song and lyrics synchronization and its application to music information retrieval. In *Proceeding of Multimedia Networking and Computing*.

DANNENBERG, R. B. 1984. An on-line algorithm for real-time accompaniment. In *Proceedings of the International Computer Music Conference*. 193–198.

DEUTSCH, D. 1988. The perceived height of octave-related complexes. *J. Acous. Soc. Amer. 80*, 5, 1346–1353.

DUXBURG, C., SANDLER, M. AND DAVIES, M. 2002. A hybrid approach to musical note onset detection. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*.

ELLIS, D. P. W. AND POLINER, G. E. 2006. Identifying 'cover songs' with chroma features and dynamic programming beat tracking. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*.

FLETCHER, H. 1931. Some physical characteristics of speech and music. *J. Acousti. Soc. Amer. 3*, 2, 1–26.

FUJIHARA, H., GOTO, M., OGATA, J., KOMATANI, K., OGATA, T. AND OKUNO, H. G. 2006. Automatic synchronization between lyrics and music CD recordings based on viterbi alignment of segregated vocal signals. In *Proceedings of the IEEE International Symposium on Multimedia (ISM)*.

FURINI, M. AND ALBORESI, L. 2004. Audio-text synchronization inside MP3 files: A new approach and its implementation. In *Proceedings of the IEEE Consumer Communications & Networking*.

GRUBB, L. AND DANNENBERG, R. B. 1997. A stochastic method of tracking a vocal performer. In *Proceedings of the International Computer Music Conference (ICMC)*. 301–308.

HAMON, C. MOULINE, E., AND CHARPENTIER, F. 1989. A diphone synthesis system based on time-domain prosodic modifications of speech. In *Proceedings of the IEEE International Conference on Acoustics Speech Signal Processing (ICASSP)*. 238–241.

HU, N., DANNENBERG, R.B. AND TZANETAKIS, G. 2003. Polyphonic audio matching and alignment for music retrieval. In *Proceedings of the IEEE Workshop on Application of Signal Processing to Audio and Acoustics*.

INOUE, W., HASHIMOTO, S. AND OHTERU, S. 1994. Adaptive karaoke system-human singing accompaniment based on speech recognition. In *Proceedings of the International Computer Music Conference (ICMC)*. 70–77.

ISKANDAR, D., WANG, Y., KAN, M. Y., AND LI., H. 2006. Syllabic level automatic synchronization of music signals and text lyrics. In *Proceedings of the ACM Multimedia Conference (ACM MM)*. 659–662.

JOHN, R. D., JOHN, H. L., AND JOHN, G. P. 1999. *Discrete-Time Processing of Speech Signals*. IEEE Press.

JOURDAIN, R. 1997. *Music, The Brain, and Ecstasy: How Music Captures Our Imagination*. HarperCollins.

KATAYOSE, H., KANOMORI, T., KAMEI, K., NAGASHIMA, Y., SATO, K., INOKUCHI, S., AND SIMURA, S. 1993. Virtual performer. In *Proceedings of the International Computer Music Conference (ICMC),* 138–145.

KIM, Y. E. 2003. Singing voice analysis/synthesis. PhD. Thesis, Massachusetts institute of Technology.

KORST, J., AND GELEIJNSE, G. 2006. Efficient lyrics retrieval and alignment. In *Proceedings of Philips Symposium on Intelligent Algorithms*.

LOSCOS, A., CANO, P., AND BONADA, J. 1999. Low-delay singing voice alignment to text. In *Proceedings of the International Computer Music Conference (ICMC*).

MADDAGE, N. C., LI, H., AND KANKANHALLI, M. S. 2006. Music structure based vector space retrieval. In *Proceedings of ACM Special Interest Group on Information Retrieval (ACM SIGIR) Conference*, 67–74.

MAKHOUL, J. 1975. Linear Prediction: A tutorial review. *Proc. IEEE*, *63*, 4, 561–580.

NWE, T. L. AND WANG, Y. 2004. Automatic detection of vocal segments in popular songs. In *Proceedings of the 5th International Symposium/Conference of Music Information Retrieval (ISMIR)*.

RABINER, L. R. AND JUANG, B. H. 1993. *Fundamentals of Speech Recognition*. Prentice-Hall.

SAKEO, H. AND CHIBA, S. 1978. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Trans. Audio Speech, Lang. Proce. 26*, 1, 43–49.

SHEH, A. AND ELLIS, D. P. W. 2003. Chord segmentation and recognition using EM-trained hidden Markov models. In *Proceedings of the International Conference on Music Information (ISMIR)*.

STEVENS, S. S., VOLKMANN, J. AND NEWMAN, E. B. 1937. A scale for the measurement of the psychological magnitude pitch. *J. Acoust. Soc. Amer. 8*, 3, 185–190.

TAYLOR, P. A., BLACK, A. W., AND CALEY, R. J. 1998. The architecture of the festival speech synthesis system. In *Proceedings of the 3rd International Workshop on Speech Synthesis*.

TSAI, W. H., WANG, H. M., RODGERS, D., CHENG, S. S. AND YU, H. M. 2004. Blind clustering of popular music recordings based on singer voice characteristics. In *Proceedings of the International Symposium of Music Information Retrieval (ISMIR)*.

WANG, Y., KAN, M. Y., NWE, T. L., SHENOY, A. AND YIN, J. 2004. LyricAlly: Automatic synchronization of acoustic musical signals and textual lyrics. In *Proceedings of the ACM Multimedia Conference*. 212–219.

WONG, C. H., SZETO, W. M. AND WONG, K. H. 2006. Automatic lyrics alignment for Cantonese popular music. In *Multimedia Systems*.

YOUNG, S., EVERMANN, G., GALES, M., HAIN, T., KERSHAW, D., LIU, X., MOORE, G., ODELL, J., OLLASON, D., POVEY, D., VALTCHEV, V., AND WOODLAND, P. 2006. *The HTK Book* Version 3.4. Department of Engineering, University of Cambridge.

Received February 2008; revised November 2008, June 2009; accepted June 2009