HMM-BASED SINGING VOICE SYNTHESIS AND ITS APPLICATION TO JAPANESE AND ENGLISH

Kazuhiro Nakamura, Keiichiro Oura, Yoshihiko Nankaku, and Keiichi Tokuda

Department of Scientific and Engineering Simulation, Nagoya Institute of Technology, Nagoya, Japan

ABSTRACT

The present paper describes Japanese and English singing voice synthesis systems based on hidden Markov models (HMMs). In this approach, the spectrum, excitation, and vibrato of the singing voice are simultaneously modeled by context-dependent HMMs, and waveforms are generated by the HMMs themselves. Japanese singing voice synthesis systems have already been developed and used to create variable musical contents. To extend this system to English, language independent contexts are designed. Furthermore, methods for matching musical notes and pronunciation of English lyrics are presented and evaluated in subjective experiments. Then, Japanese and English singing voice synthesis systems are compared.

***Index Terms*—** English singing voice synthesis, HMM-based speech synthesis, HMM-based singing voice synthesis

1. INTRODUCTION

Singing voice synthesis enables computers to "sing" any song. It has become especially popular in Japan because of Yamaha's VOCALOID singing synthesizer [1]. There is now a growing demand for more flexible systems that can sing songs with various voices as evidenced by the many singer libraries being created and released on the Internet by users for UTAU [2] singing voice synthesis software.

One approach to synthesizing singing voices is to use hidden Markov models (HMMs) [3] [4]. In this approach, the spectrum, excitation, and vibrato of a singing voice are simultaneously modeled, and singing voice parameter trajectories are generated from the HMMs by using a speech parameter generation algorithm [5]. Systems of HMM-based speech synthesis [6] [7] which is the base of HMM-based singing voice synthesis usually have smaller footprints than those of unit-selection synthesis because they store statistics rather than waveforms. This approach makes it possible to model different voice characteristics, speaking styles, and emotions without recording large speech databases. Adaptation [8], interpolation [9], and eigenvoice [10] techniques, for example, have been applied to HMM-based systems, demonstrating that voice characteristics can be modified. As a demonstration of HMM-based singing voice synthesis, our research group publicly released a web service [4] [11], and it has been used by many creators.

If Japanese singing voice synthesis systems were extended to support other languages, people all over the world could also enjoy singing voice synthesis. We are thus working to extend the singing voice synthesis technique to other languages, focusing on English as the first step. In this paper, we present an HMM-based English singing voice synthesis system in addition to Japanese one.

The rest of this paper is organized as follows: Section 2 describes the overview of the HMM-based singing voice synthesis system. Details of the English singing voice synthesis system and comparison to the Japanese one are described in Section 3. Section 4 describes our experimental evaluation and presents the results. The key points are summarized and future work is mentioned in Section 5.

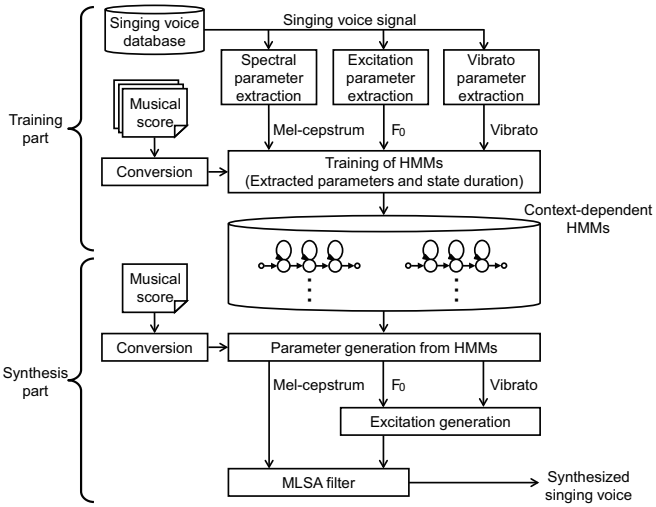

Fig. 1. Overview of HMM-based singing voice synthesis system.

2. HMM-BASED SINGING VOICE SYNTHESIS SYSTEM

The HMM-based singing voice synthesis system is quite similar to the HMM-based text-to-speech synthesis system [6] [7]. However, there are distinct differences. Figure 1 gives an overview of the HMM-based singing voice synthesis system [3] [4]. It consists of training and synthesis parts. In the training part, the spectrum (e.g., mel-cepstral coefficients), excitation, and vibrato are extracted from a singing voice database and then modeled by context-dependent HMMs. Context-dependent models of state durations are also estimated simultaneously [12]. Pitch adaptive training (PAT) [13] is used to generate singing voices in any pitch. An arbitrarily given musical score including the lyrics to be synthesized is first converted into a context-dependent label sequence in the synthesis part. Next, a state sequence corresponding to the song is constructed by concatenating the context-dependent HMMs in accordance with the label sequence. The state durations of the song HMM are then determined with respect to the state duration models. Next, the speech parameters (spectrum, excitation, and vibrato) are generated by an algorithm [5]. Finally, a singing voice is synthesized directly from the generated parameters by using a mel log spectrum approximation (MLSA) filter [14].

The rhythm and tempo of the music are important factors in singing voice synthesis. Therefore, the start timings of the notes and the phoneme durations for each note must be determined from the musical score. However, if the musical score is strictly followed, the synthesized singing voice will be unnatural because of time lags. To overcome this problem, the time lags of individual notes are modeled by Gaussian distributions [3].

978-1-4799-2893-4/14/$31.00 ©2014 IEEE

Table 1. Relationships between Japanese strings and pronunciation.

String	Mora	げ		ん	こ		つ		や		ま		の		た		ぬ		き		さ		ん
Pronunciation	Mora	ge		N	ko		tsu		ya		ma		no		ta		nu		ki		sa		N
	Phoneme	g	**e**	**N**	k	**o**	ts	**u**	y	**a**	m	**a**	n	**o**	t	**a**	n	**u**	k	**i**	s	**a**	**N**

Table 2. Relationships between English strings and pronunciation.

String	Word	rhythm					of	the	classical									music				
	Syllable	rhy		thm			of	the	clas			si		cal				mu			sic	
Pronunciation	Syllable	rih		dhaxm			ahv	dhax	klae			sih		kaxl				myuw			zihk	
	Phoneme	r	**ih**	dh	**ax**	m	**ah**	v	dh	**ax**	k	l	**ae**	s	**ih**	k	**ax**	l	m	y	**uw**	z	**ih**	k

3. ENGLISH SINGING VOICE SYNTHESIS

3.1. Lyrics of English musical scores

Lyrics in Japanese musical scores are generally written in kana characters, which can be converted into labels by using a mora-to-phonemes table. On the other hand, English lyrics are generally written in words, and a word-to-phonemes table is not sufficient for words, like "the" and "lead" for which the pronunciation depends on the context. Thus, morphological analysis is needed to convert the word sequence into syllable and phoneme sequences. A musical phrase that is an uttered part between musical rests is regarded as a sentence and analyzed. A syllable consists of a vowel (syllable nucleus) and consonants around it. Tables 1 and 2 show the relationships between strings and pronunciation in Japanese and English respectively. In these tables, vowels are indicated by boldface.

Contexts for English singing voice synthesis are designed by expanding contexts for Japanese one [4]. English syllables and Japanese moras are allocated to a common level in the context design to standardize contexts of these languages. In addition, a new area is appended to the context design to address language dependent contexts, e.g. stress and accent, which are used only in English. The proposed context design is presented in Table 3. The proposed area is indicated by boldface.

In this paper, the Flite [15] is used for morphological analysis, and the CMU pronouncing dictionary [16] is used as the word dictionary. The phoneme set consists of phonemes in CMU pronouncing dictionary, long silence "sil", silence neighboring uttered parts "pau", and breath "br".

3.2. Syllable allocation methods

The number of syllables for each word is obtained by morphological analysis. However, it is not always equal to the number of corresponding notes. Therefore, a method for allocating syllables to notes is required. Here we propose two methods.

1: Left-to-right allocation

In this method, syllables in a word are allocated to corresponded notes one-by-one from the head note. If the number of syllables is not equal to that of notes, the remaining syllables are allocated to the tail note or each of all remaining notes receives a syllable duplicated from the last syllable.

2: Score-based allocation

In this method, syllables in a word are allocated to corresponded notes based on the number of characters in each note. Each note that has no syllable receives a syllable duplicated from the syllable of previous note. The allocation procedure comprises three steps.

Step 1: Count number of characters corresponding to each note

First, the number of characters corresponding to each note is counted. A character denotes a letter in a lyric string in Table 2. Since many syllables should be allocated to notes that have many vowels (syllable nucleus), we count "a", "e", "i", "o", and "u", which tend to be vowels, as two characters in this paper. Table 2 shows an example. The word "classical" has two "a" and one "i", and they are allocated to three syllables one-by-one as vowels. Similarly, one of the exceptions to "a", "e", "i", "o", and "u" being vowels is "rhythm" in Table 2. Although it contains none of these letters, its pronunciation includes some vowel sounds.

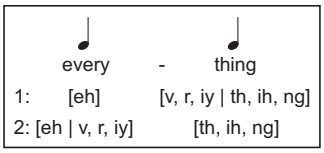

Fig. 2. Two methods for syllable allocation.

Step2: Calculate score for each note

The score w_n of a note n is defined as

$$w_n = \frac{Sc_n}{\sum_{n'=1}^{N} c_{n'}}, \tag{1}$$

where c_n, N and S denote the number of characters corresponding to note n, the number of notes in a word, and the number of syllables obtained by morphological analysis respectively. The summation of all scores is equal to the number of syllables.

Step3: Determine allocation of syllables to notes

Finally, the number k_n of syllables allocated to each note n is determined. The numbers are initialized to 0. The note with the highest score, $\hat{n}$, is selected, and $k_{\hat{n}}$ and $w_{\hat{n}}$ are updated to $k_{\hat{n}} = k_{\hat{n}} + 1$ and $w_{\hat{n}} = w_{\hat{n}} - 1$. The k_n for all n are obtained after S iterations of this procedure. Note that at least one syllable has to be allocated to the head note of a word.

Figure 2 shows an example illustrating these two methods. The word "everything" is converted into three syllables "eh | v, r, iy | th, ih, ng". The symbol "|" represents a syllable boundary. If the word corresponds to two notes, method **1** allocates syllables one-by-one from the head note and allocates all remaining syllables to the tail note. As a result, one syllable "eh" is allocated to the first note, and two syllables "v, r, iy | th, ih, ng" are allocated to the second note. In method **2**, because of $S = 3$, $c_1 = 7$, and $c_2 = 5$, the score for each note is obtained as

$$w_1 = (3 \times 7) / (7 + 5) = 1.75, \tag{2}$$

$$w_2 = (3 \times 5) / (7 + 5) = 1.25. \tag{3}$$

Thus, two syllables, "eh | v, r, iy", are allocated to the first note, and one syllable, "th, ih, ng", is allocated to the second note.

3.3. Syllable duplication methods

If the number of notes is smaller than that of syllables, there are some notes without a syllable. We propose two methods for allocating a syllable to each of these notes by duplicating the syllable of the previous note.

Table 3. Proposed context design. English syllables and Japanese moras are allocated to common level, and new area for language dependent context is appended. The proposed area is indicated by boldface.

Phoneme	Quinphone. (Phoneme within the context of two immediately preceding and succeeding phonemes)
Syllable (Mora)	Number of phonemes in {previous, current, next} syllable.
	Position of {previous, current, next} syllable in note.
	Language dependent context in {previous, current, next} syllable. (English: with or without {accent, stress}, Japanese: undefined)
Note	Musical {tone, key, beat, tempo, and length} of {previous, current, next} note.
	Position of current note in {measure, phrase}.
	With or without a slur between current and {previous, next} note.
	Dynamics to which current note belongs.
	Difference in pitch between current note and {previous, next} note.
	Distance between current note and {next, previous} {accent, staccato}.
	Position of current note in current {crescendo, decrescendo}.
Phrase	Number of {syllables, notes} in {previous, current, next} phrase.
Song	Number of {syllables, notes} / Number of measures.
	Number of phrases.

Table 4. Diphthong duplication rules.

Original	ey	ay	ow	aw	oy
Duplicated	eh, ey	aa, ay	ao, ow	aa, aw	ao, oy

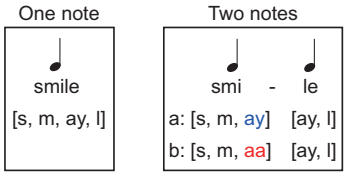

Fig. 3. Two methods for duplicating syllables.

a: Simple duplication

In this method, the nucleus of the syllable allocated to the previous note is simply duplicated, and the syllable is divided.

b: Rule-based duplication

Consecutive diphthongs due to duplication may degrade the continuity of a singing voice, so we defined the duplication rules for diphthongs shown in Table 4.

Figure 3 shows an example illustrating these syllable duplication methods. The word "smile" has one syllable, "s, m, ay, l", and it corresponds to two notes. In method **a**, "ay" is simply duplicated as "s, m, ay" and "ay, l". In method **b**, the "ay" of the first note is converted to "ah" by using a duplication rule.

4. EXPERIMENTS

To evaluate the effectiveness of the proposed methods and compare Japanese and English singing voice synthesis, we conducted subjective experiments. Twenty English songs sung by a female singer who was a bilingual student were used for training English models, and five songs were used for evaluation. For comparison, 17 Japanese songs sung by the same singer were used for training Japanese models, and five songs were used for evaluation. The total length of the voiced parts was adjusted to about 30 minutes for each training data set. Singing voice signals were sampled at a rate of 48 kHz and windowed with a 5-ms shift. The feature vectors were the spectral, excitation, and vibrato feature vectors. The spectrum parameter vector consisted of 49 STRAIGHT [17] mel-cepstral coefficients including the zero-th coefficient. The excitation parameter vector consisted of log F_0. The vibrato parameter vector consisted of fluctuation amplitude and frequency. In addition to these parameters, their deltas and delta-deltas were used.

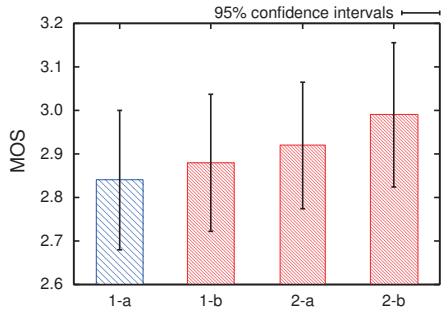

Fig. 4. Effect of syllable allocation and duplication methods.

A seven-state (including the beginning and ending null states), left-to-right, no-skip structure was used for the MSD-HSMM [12] [18]. The phoneme alignment results for the training data obtained by using the deterministic annealing EM (DAEM) [19] algorithm were used as the initial phoneme boundary labels. A decision-tree-based context-clustering technique was separately applied to the distributions for the spectrum, excitation, vibrato, state duration, and time lag. The MDL criterion [20] was used to control the size of the decision trees. The heuristic weight α for the penalty term in Equation (1) in [20] was 3.0. Ten Japanese subjects were asked to evaluate the naturalness of the synthesized singing voices on Mean Opinion Score (MOS) with a scale from 1 (poor) to 5 (good). Each subject was presented 10 randomly selected musical phrases from 30 musical phrases. The average length of the musical phrases was 8.1 seconds. Three experiments were carried out in a sound-proof room.

4.1. Experiment of syllable allocation and duplication

In this experiment, combinations of syllable allocation and duplication methods were compared. The syllable allocation methods were defined as follows.

1: Left-to-right allocation
2: Score-based allocation

The syllable duplication methods were defined as follows.

a: Simple duplication
b: Rule-based duplication

The four possible combinations (**1-a**, **1-b**, **2-a**, and **2-b**) were evaluated in terms of the MOS.

As shown in Fig. 4, combinations **1-b**, **2-a** and **2-b** obtained higher score than combination **1-a**, and combination **2-b** obtained the highest score. This indicates the superiority of the score-based

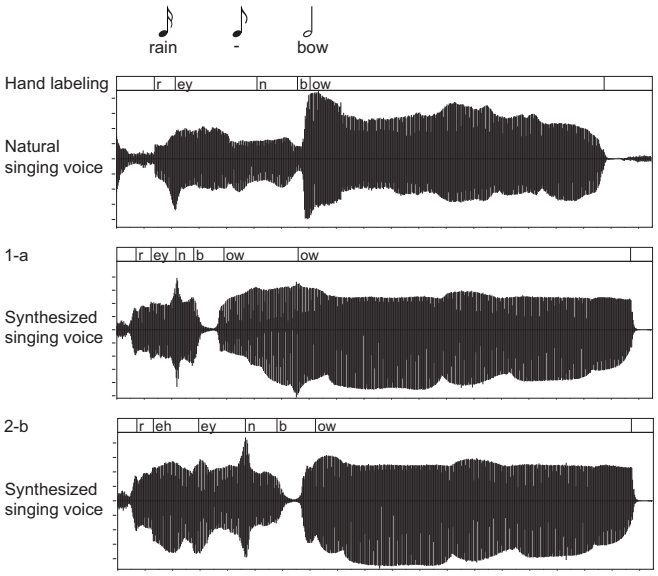

Fig. 5. Comparison of waveforms in terms of differences in syllable allocation and duplication methods for "rainbow". Natural voice is shown in the first waveform, and synthesized waveforms by combination **1-a** and **2-b** are shown in the second and the third waveforms respectively.

syllable allocation method and the rule-based syllable duplication method. Figure 5 shows an example of the differences between a natural singing voice and two synthesized singing voices with combinations **1-a** and **2-b** for "rainbow". The phoneme alignments of the natural singing voice were obtained by hand labeling, and those of the synthesized singing voices were obtained when the singing voices were synthesized. The word "rainbow" consists of two syllables, "r, ey, n" and "b, ow". With combination **1-a**, two syllables were allocated to the head and center notes, and the syllable "b, ow" was duplicated into "b, ow" and "ow". With combination **2-b**, two syllables were allocated to the head and tail notes, and the syllable "r, ey, n" was duplicated into "r, eh" and "ey, n" on the bases of the duplication rule. As a result, combination **2-b** produced a singing voice similar to the natural singing voice and was thus used in the next two experiments.

4.2. Experiment of time lag

In this experiment, the effect of time-lag modeling and where the time-lag should be measured from were evaluated for Japanese and English singing voice synthesis [1]. The following three methods were compared.

- **A**: Without time-lag models
- **B**: With time-lag (from head phoneme) models
- **C**: With time-lag (from syllable nucleus) models

Synthesized voices were played with a click for every quarter note synchronized to the corresponding musical score (only in this experiment).

Figure 6 shows the results of MOS evaluation. Improvement with time-lag modeling was evident for both languages. In Japanese, method **B** obtained a little higher score than method **C**. In English, method **C** obtained higher score than method **B**. A possible explanation for this is that, since two or more consonants can appear in front of the syllable nucleus in English, the phoneme durations before the first vowel may fluctuate widely. Method **C**, which achieved the best score for English, was used in the last experiment.

[1]The obtained results are not comparable in absolute value across languages because these experiments were conducted independently.

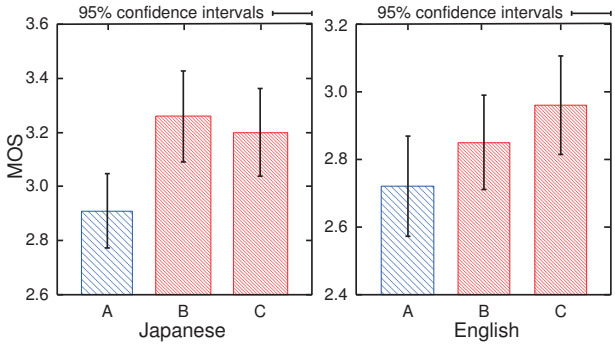

Fig. 6. Effect of time-lag modeling.

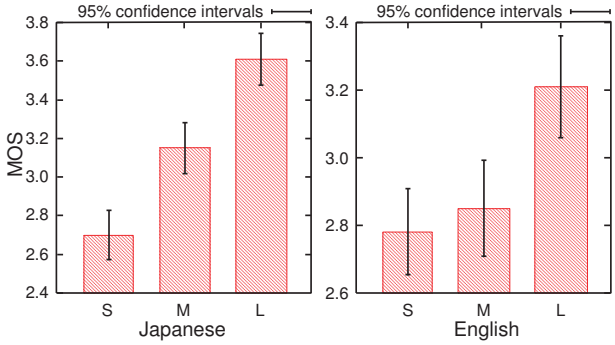

Fig. 7. Effect of amount of training data.

4.3. Experiment of data size

In this experiment, the relationships between training data size and the naturalness of the synthesized voices were compared between Japanese and English singing voice synthesis. There were three sizes for the data (length of voiced part):

- **S**: 8 min. (5 Japanese songs, 5 English songs)
- **M**: 15 min. (9 Japanese songs, 10 English songs)
- **L**: 30 min. (17 Japanese songs, 20 English songs)

As shown in Fig. 7, naturalness improved for both languages with an increasing amount of training data. Moreover, the scores for English varied widely, probably because English is not the native language for subjects.

5. CONCLUSIONS

In this present paper, HMM-based singing voice synthesis and its application to Japanese and English were described. Language independent/dependent contexts were defined for both languages, and syllable allocation and duplication methods for matching English syllables to musical notes were described and evaluated in the subjective experiments. Furthermore, other experiments clarified the effects of time-lag modeling and the relationships between the amount of training data and the naturalness of the synthesized voice in Japanese and English singing voice synthesis. Each of them showed a largely similar trend in both languages. Future work includes subjective evaluation by English native speakers, additional experiments by using other singer voices, and expansion of singing voice synthesis to other languages, e.g., Mandarin.

6. ACKNOWLEDGMENTS

The research leading to these results was partly funded by the Core Research for Evolutionary Science and Technology (CREST) program of the Japan Science and Technology Agency (JST), and the Hori Sciences and Arts Foundation.

7. REFERENCES

[1] H. Kenmochi and H. Ohshita, "VOCALOID – commercial singing synthesizer based on sample concatenation," *in Proc. of Interspeech*, 2007.

[2] "UTAU," http://utau2008.web.fc2.com/.

[3] K. Saino, H. Zen, Y. Nankaku, A. Lee, and K. Tokuda, "An HMM-based singing voice synthesis system," *in Proc. of Interspeech*, pp. 1141–1144, 2006.

[4] K. Oura, A. Mase, T. Yamada, S. Muto, Y. Nankaku, and K. Tokuda, "Recent development of the HMM-based singing voice synthesis system – Sinsy," *in Proc. of Speech Synthesis Workshop*, pp. 211–216, 2010.

[5] K. Tokuda, T. Kobayashi, and S. Imai, "Speech parameter generation from HMM using dynamic features," *in Proc. of ICASSP*, pp. 660–663, 1995.

[6] T. Masuko, K. Tokuda, T. Kobayashi, and S. Imai, "Speech synthesis from HMMs using dynamic features," *in Proc. of ICASSP*, pp. 389–392, 1996.

[7] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis," *in Proc. of Eurospeech*, pp. 2347–2350, 1999.

[8] M. Tamura, T. Masuko, K. Tokuda, and T. Kobayashi, "Adaptation of pitch and spectrum for HMM-based speech synthesis using mllr," *in Proc. of ICASSP*, pp. 805–808, 2001.

[9] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Speaker interpolation in HMM-based speech synthesis system," *in Proc. of EUROSPEECH*, vol. 5, pp. 2523–2526, 1997.

[10] K. Shichiri, A. Sawabe, T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Eigenvoice for HMM-based speech synthesis," *in Proc. of ICSLP*, vol. 1, pp. 1269–1272, 2002.

[11] "Sinsy – HMM-based singing voice synthesis system," http://www.sinsy.jp/.

[12] H. Zen, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "A hidden semi-markov model-based speech synthesis system," *IEICE Trans. Inf. & Sys.*, vol. 90-D, no. 5, pp. 825–834, 2007.

[13] K. Oura, A. Mase, Y. Nankaku, and K. Tokuda, "Pitch adaptive training for HMM-based singing voice synthesis," *in Proc. of ICASSP*, pp. 5377–5380, 2012.

[14] S. Imai, "Cepstral analysis synthesis on the mel frequency scale," *in Proc. of ICASSP*, pp. 93–96, 1983.

[15] "Flite," http://www.festvox.org/flite/.

[16] "CMU pronouncing dictionary," http://www.speech.cs.cmu.edu/cgi-bin/cmudict/.

[17] H. Kawahara, M. K. Ikuyo, and A. Cheneigne, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based f_0 extraction: Possible role of a repetitive structure in sounds," *Speech Communication*, vol. 27, pp. 187–207, 1999.

[18] K. Tokuda, T. Masuko, N. Miyazaki, and T. Kobayashi, "Hidden markov models based on multi-space probability distribution for pitch pattern modeling," *in Proc. of ICASSP*, vol. 1, pp. 229–232, 1999.

[19] N. Ueda and R. Nakano, "Deterministic annealing em algorithm," *Neural Networks*, vol. 11, pp. 271–282, 1998.

[20] K. Shinoda and T. Watanabe, "MDL-based context-dependent subword modeling for speech recognition," *J. Acoust. Soc. Jpn. (E)*, vol. 21, no. 2, pp. 76–86, 2000.