Journal of Multidisciplinary Engineering Science Studies (JMESS)
ISSN: 2458-925X
Vol. 8 Issue 7, July - 2022

Building a Korean MIDI-to-Singing Song Synthesis

Hung-Che Shen
Dept. of Digital Multimedia Design
I-Shou University
Kaohsiung, Taiwan
shungch@isu.edu.tw

***Abstract*—This work reports development of a MIDI-to-Singing song synthesis that will produce audio files from MIDI data and arbitrary Romaji lyrics in Korean. The MIDI-to-Singing system relies on the Flinger (Festival singer) framework for singing voice synthesis. Originally, this MIDI-to-Singing system was developed by English. Based on some Korean pronunciation rules, a Korean MIDI-to-Sing synthesis system was developed and derived. For a language transfer like Festival synthesized singing, two major tasks are the modifications of a phoneset and a lexicon. Originally, MIDI-to-Sing song synthesis can create singing voices in many languages, but there is no existing Korean festival diphone voice available right now. We therefore used a voice transformation model in festival to develop Korean MIDI-to-Singing synthesis. An evaluation of a song listening experiment was conducted and the result of this voice conversion showed that the synthesized singing voice successfully migrate from English to Korean with high voice quality.**

Keywords—MIDI-to-Singing; Korean; Festivel; Flinger

I. INTRODUCTION

The goal of this research is to synthesize natural singing Korean song from an English Text-to-Speech voice. With the term MIDI-to-Singing, we mean the production of human-like singing voice based on a given MIDI format music. The MIDI-to-Singing system is an extension from a speech-to-singing synthesis, which converts a speaking voice reading the lyrics of a song to a singing voice given its musical score. Therefore, a different language singing can be easily derived by modifying the original speech features unique to them.

Singing voice synthesis enables computers to “sing” any song. Since 2007, it has become especially popular in Japan because of Yamaha’s VOCALOID singing synthesizer [1]. There can be found a lot of original musical compositions in the video sites such as YouTube or Niko Niko Douga. 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.

Without a score editor environment for end-users, it should be noted that a concatenation-based singing synthesizer was already proposed by Macon et al. [3] in 1992. They finally released open source for this method that is called Flinger [4]. It is written by Mike Macon based on the Festival Speech Synthesis System [5], developed at the University of Edinburgh. Flinger needs the OGIresLPC plugin as the signal processing ``backend'' that was developed at OGI [6]. This takes diphone waveform files, coded as residual-excited LPC parameters, and performs the necessary concatenation, pitch/duration change, and smoothing of the diphone waveforms.

A MIDI-to-Singing system requires the integration of two major components. One is a MIDI sequencer with a lead-sheet view. The other is a singing voice synthesis (SVS). Based on a musical graphical interface such as a lead sheet, the users can compose music/lyrics in a MIDI sequencer.

In Festival, a new language is mainly comprised of constructing a phoneset and building a lexicon. The phoneset is basically a set of syllables and includes the linguistic characteristics of how vowels and consonants are pronounced. A lexicon is a database of words with their respective pronunciations, using the phones from the phoneset. During the past few years, we have been using songs for educational learning based on our previous MIDI-to-Singing system [7]. To extend the applications of singing voice synthesis, a language transfer will be needed. A language transfer refers to speakers applying knowledge from one language to another language. This paper assumes that a MIDI-to-Singing language transfer between English and Korean is feasible. In order to demonstrate Korean MIDI-to-Singing, some songs are available online as shown in Fig. 1.

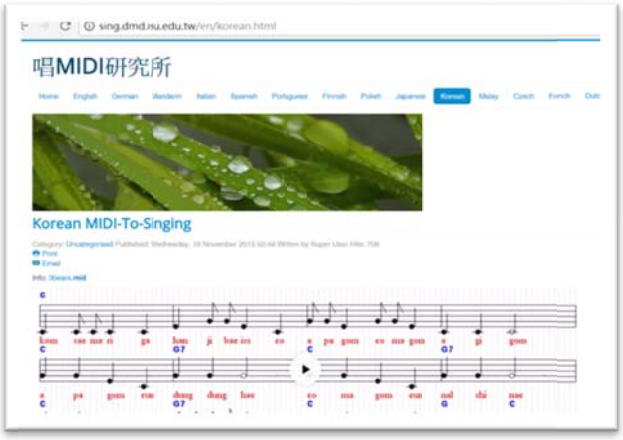

Fig. 1. *Korean MIDI-to-Singing Demo Websites: http://sing.dmd.isu.edu.tw/en/korean.html.*

This paper is organized as follows. In Section 2, we present Korean syllable and word syntax which explains the language transfer in Festival Text-to-Speech. After that, a MIDI-to-Singing song synthesis is described in Section 3. Finally, experimental results are given in Section 4. Section 5 concludes this paper.

II. Derive a Korean Diphone Voice

In text-to-speech applications, grapheme-to-phoneme conversion is an important task. The input text must be converted into phonemic representations that the speech synthesizer can use to generate correct and natural speech. Our research focus on Korean MIDI-to-Singing synthesis, which uses a phonemic alphabetical writing system. Although Korean writing system is phonemic, it is still limited by morphology. For instance, the sentence "못해" ([mo th E], I can't do it) is composed of two morphemes: "못" ([mos], can't) and "해" ([hE], do), not written in its actual pronunciation form "모태" ([mo] + [th E]). Korean written forms do not reflect their actual pronunciation.

Hangul combines two, or more often three, letters into syllabic blocks. When processing strings of these blocks, we first convert them into Roman characters using an online romanization tool [8]. For example, the character 넓, which is composed of four Hangul letters ㄴ(n), ㅓ(eo), ㄹ(r), and ㅂ(b), will be transcribed into "neorb". This romanization is based on the Revised Korean Romanization standard promoted by the South Korean Ministry of Culture. Korean Romanization (Romaji) is the standard way of transliterating Korean into the Latin alphabet. Consonants and vowels are always broken up into the same "blocks" of sound allowing very easy parsing of words.

Korean writing system is phonemic, but still limited by morphology. It causes the graphemes can not reflect their actual pronunciation. Therefore, it is necessary to have a grapheme-to-phoneme conversion system to convert the texts into their phonemic transcription of actual pronunciation. The conversion of written lyrics to the actual Korean pronunciation comprises four stages: word segmentation, phoneme extraction, sound pattern processing, and IPA presentation shown in Fig. 2.

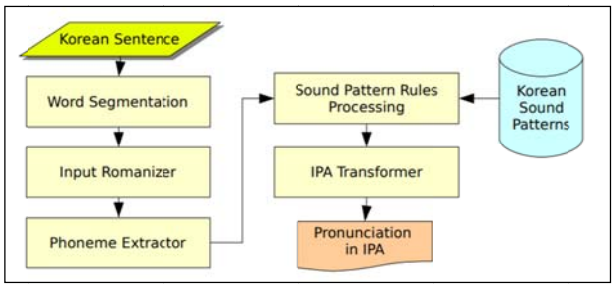

Fig. 2. *Korean written sentence to pronunciation.*

Assuming that a voice set exists, constructing a language in Festival requires modifying a phoneset, building a lexicon, and adding prosody (defining the letter to sound rules). Because a word is usually broken into syllables in singing, when syllables are fed into a singing synthesis system, some considerations required in text-to-speech are ignored such as prosody. Two basic processes need to be addressed only contains:

- Modifying a phoneset,
- Modifying a lexicon,

A. Modifying a phoneset

A phoneme (or phone) is one of the units of sound that distinguish one word from another in a particular language. Syllables are considered the smallest unit of speech, while a phone is only a speech sound. For the Korean language, there exists a 1-1, 1-2 and 1-3 relationship between syllables and phones. In Festival the phones are stored in the phoneset. A phoneset is a set of symbols which may be further defined in terms of features, such as vowel/consonant, place of articulation for consonants, type of vowel etc. This, like everything else in Festival, is a function-based structure in Scheme. The phoneset for a language has been defined in many cases, and it is wise to follow convention when it exists. For Korean, that's probably ok as the Korean phoneset is mostly a subset of the English phoneset. In Festival, the phones' features and their values must be defined with the phoneset. This, like everything else in Festival, is a function-based structure in Scheme. At synthesis time, each Korean phone must be mapped to an equivalent (one or more or modified) US phone. This is done though phoneset members for the Korean as shown in Table I.

Table I. PHONESET MEMBERS FOR THE KOREAN.

Standard vowels	Long vowels	Unvoiced reduced vowels	Consonants
(a + s 3 (i + s 1 (u + s 1 (e + s 2 (o + s 2	(a: + l 3 (i: + l 1 (u: + l 1 (e: + l 2 (o: + l 2	(I + a 1 1 (U + a 1 3 **Geminates** (Qk - g 0 0 (Qs - g 0 0 (Qsh - g 0 0	(k - 0 0 0 (s - 0 0 0 (sh - 0 0 0 (t - 0 0 0 (ch - 0 0 0 (ts - 0 0 0 (n - 0 0 0 (h - 0 0 0 (f - 0 0 0 (m - 0 0 0

The description of Korean phone features is shown in Fig. 3. The descriptions describe the following phone features, for vowels: length (short, long, diphthong, schwa), height (close, close-mid, mid, open-mid, open), frontness (front, central, back), and lip rounding (rounded, unrounded). For consonants: consonant type (stop, affricate, fricative, nasal, lateral, approximant), place of articulation (labial, alveolar, velar, palatal, post-alveolar, labio-dental, dental, glottal), and consonant voicing (voiced, unvoiced).

```
;;;  Phone Features
(;; vowel or consonant
 (vc + -)
 ;; vowel length: short long dipthong schwa geminate (for consonants)
 (vlng s l d a g 0)
 ;; vowel height: high mid low
 (vheight 1 2 3 0)
 ;; vowel frontness: front mid back
 (vfront 1 2 3 0)
 ;; lip rounding
 (vrnd + - 0)
 ;; consonant type: stop fricative affricate nasal lateral approximant
 (ctype s f a n l r 0)
 ;; place of articulation: labial alveolar palatal labio-dental
 ;;                        dental velar glottal
 (cplace l a p b d v g 0)
 ;; consonant voicing
 (cvox + - 0)
 )
(
```

Fig. 3. *Korean phone features.*

However, scanning down the phoneset one will notice an arrangement of syllables. For example, "(a + l 2 2 – 0 0 0)" is the phone from the Radio phoneset included with festival. This phone represents the first 'a' can be the phone of a Korean Romaji word "ai" or "au".

In the phone above, the "a" represents the name of the phone. This name is used in the lexicon to refer to which phones need to be uttered by the speech synthesis system to form words. Next comes a '+' or '-' symbol where the positive affirms that the phone represents a vowel, and the negative indicates the phone represents a consonant. All of the remaining settings can take 0 for a value which indicates that that setting is not applicable. Next in the phone definition there is "l 2 2" which define three linguistic settings for vowels. The first entry corresponds to vowel length can be 's' for short, 'l' for long, 'd' for diphthong, 's' for schwa, or 0 for not applicable. Next, the second setting indicates vowel height and can be an integer from 0 for not applicable, 1 for high, 2 for medium, and 3 for low. Lastly for the vowel section, the third setting corresponds to vowel frontness and can have the values 1 for high, 2 for mid, and 3 for low. For the following setting, in the example above, "-" represents lip rounding. When a human pronounces a phoneme the lip position can affect which phoneme is uttered. This setting can take the values "+" indicating that lip rounding is present, or "-" which indicates the contrary. For the last three settings, the ones for consonants, we start with consonant type which can be 's' for stop, 'f' for fricative, 'a' for affricate, 'n' for nasal, 'l' for lateral, or 'r' for approximant. Next, the second setting provides for the place of articulation and takes the values 'l', 'a', 'p', 'b', 'd', 'v', or 'g'. The place of articulation represents how the mouth or throat says the consonant. Lastly, consonant voicing is represented by the final setting with values '+' for its presence, or '-' for its absence.

B. Modifying a lexicon

A lexicon is basically a large dictionary of words with the corresponding syllables to pronounce the word correctly. The lexicon had to be converted to the phonetic alphabet and to the format that is required by the Festival system. The lexicon format comes in three parts, a compiled lexicon, an addenda, and a rule system for handling unknown words. If any word appears that is not part of the lexicon, the pronunciation will be found by letter-to-sound-rules. Additionally, there exist many websites with examples of correct pronunciation of a subset of Korean words. An addenda of words augment what is in the lexicon, but may not have been compiled and saved into the lexicon itself. This is useful for testing new additions to the lexicon. Some typical example entries of English are ("walkers" n (((w oo) 1) ((k @ z) 0))) for the word "walker" and ("present" v (((p r e) 0) ((z @ n t) 1))) for the word "present".

A pronunciation in Festival requires not just a list of phones but also a syllabic structure. We already discussed about our phones with their features. The lexicon structure that is basically available in Festival takes both a word and a part of speech to find the given pronunciation. An example of an entry in the compiled lexicon is ("honda" nil (((h > n) 1) ((d ^) 0))). Note that this represents the word "Honda" and breaks it into the phones named "Hon," and "Da." Numbers next to the phone names can affect the duration of the phone pronounced. Explicit marking of syllables a stress value is also given (0 or 1). Lexicon entries of Korean are shown in Fig. 4.

```
("a" nil (((A) 0) ((A A) 2)))
("i" nil (((j j) 0) ((i: i:) 2)))
("u" nil (((u) 0) ((u) 2)))
("e" nil (((E) 0) ((E) 2)))
("o" nil (((>) 0) ((oU) 2)))
("sa" nil (((s s) 0) ((A A) 2)))
```

Fig. 4. *Lexicon entries of Korean.*

III. A MIDI-TO-SINGING SONG SYNTHESIS

Festival is a general-purpose concatenative text-tospeech (TTS) system that uses the residual-LPC synthesis technique, and is able to transcribe unrestricted text to speech. Assuming that a voice set exists, constructing a language in Festival requires creating a phoneset, building a lexicon, and defining the letter to sound rules. The voice set is the actual sounds that festival outputs. Since Festival provides some example voice sets, the focus of my research has been on the theoretical construction of the other components which have greater relevance to the Korean language. Building a new voice set using a local Korean speaker can be accomplished using the application FestVox.

A. OGI residual-LPC synthesizer (Festival plug-in)

This OGI residual-LPC synthesizer, which has to be considered as a plug-in for Festival, has been developed at OGI (Oregon Graduate Institute of Science and Technology, Portland, OR), and provides a new signal processing engine, and new voices, not included in the Festival distribution. Specifically, new pitchmark and LPC analysis algorithms, together with some scheme scripts that enable the creation of new voices in the OGIresLPC synthesizer are included. It is freely available for research and educational use. OGI has expanded its range of languages: there are TTS systems for English, Spanish and Welsh.

OGIresLPC is a drop-in module for the Festival TTS system created by CSTR at the University of Edinburgh (http://www.cstr.ed.ac.uk/projects/festival). This version of OGIresLPC has been designed to work with Festival version 1.2.0, released September 1997. It should work with any version 1.2.x newer than this, and can possibly be made to work with other versions of Festival, but this would require some changes to the code and knowledge of Festival internals. It provides waveform synthesis of speech with reasonable quality, but has not been extensively optimized in any way. It is meant to serve as a simple baseline synthesizer in the CSLU Toolkit and for other experiments.

B. Score editor

The Score Editor provides an environment in which the user can input notes, lyrics, and optionally some expressions. The Editor is designed especially for MIDI-to-Singing system. A lead sheet is a form of musical notation that specifies the essential elements of a MIDI song: the melody, lyrics and harmony. The melody is written in modern Western music notation, the lyric is written as text below the staff and the harmony is specified with chord symbols above the staff.

The user can type-in lyrics in normal writing and the Editor automatically converts the lyrics into phonetic symbols by looking into a built-in pronunciation dictionary. If the word consists of two or more syllables, the Editor automatically decomposes it into syllables. The user can easily add vibrato in the Editor. A screenshot of the MIDI-to-Singing process and score editor is shown in Fig. 5(a) and 5(b).

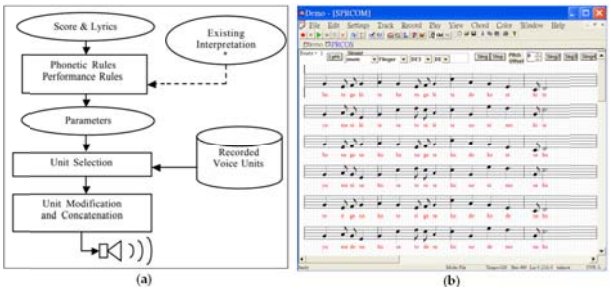

Fig. 5. *(a) Process of a MIDI-to-Singing system. (b) Score editor of MIDI-to-Singing.*

IV. DISCUSSIONS

The task of synthesizing singing has, of course, a lot in common with that of synthesizing speech. For singing, the immense problems of reliably modelling intonation and syllable length are already solved (or rather by-passed) by the composer. In singing, Korean provides relatively a greater ease for the English speaker learning Korean than the English system would to Korean singers. Not taking into account language variation (accent and dialect), there are five vowels and 17 consonant phonemes in Korean while there are 20 vowels and 24 consonants in English language. Therefore, a Korean MIDI-to-Singing system can be accomplished through an English MIDI-to-Singing system.

To evaluate the effectiveness of the proposed Korean MIDI-to-Singing song synthesis, we conducted subjective experiments. Ten Korean songs sung by MIDI-to-Singing system and sung by VOCALINA are used for evaluation. In order to evaluate the enhancement of modification strategies we adopted in singing synthesis, we scheduled a preference test to evaluate our system. The 10 kid MIDI songs are from Mama lisa's website: http://www.mamalisa.com/.

10 pairs of singing utterances were generated by 2 systems and 10 people took part in our experiment to judge which system got the better performance. These pairs of singing voices are played in a randomized order and the listeners gave their scores for each pair. The score is in 3 levels: the first is better, the second is better and they are almost the same. The result is listed in Table 3. As seen from Table II, obviously the system with our proposed strategies got a lower performance than VOCALINA.

VOCALINA (보카리나) is a "text to speech" vocal synthesizer, it was the first music speech synthesis technology (Singing TTS Technology) to be developed focus on the Korean language and is focused on singing. It is designed to be easy to use and produce high quality singing results. It is in the Korean language, and has similar functions to that of the Vocaloid synthesis engine. Using the editor users can alter a vocals height (Pitch), dynamics (Dynamics), Vibe ration (Vibration), Reverb (REV), and Echo (ECO) for better results.

Table II. Preference test score.

Prefer MIDI-to-Singing	Prefer VOCALINA
45%	55%

V. Conclusion

Korean provides relatively a greater ease for the English speaker learning Korean than the English system would to Korean speakers. Not taking into account language variation (accent and dialect), there are five vowels and 17 consonant phonemes in Korean while there are 20 vowels and 24 consonants in English language. Therefore, a Korean MIDI-to-Singing system can be accomplished through an English MIDI-to-Singing system.

This paper outlines the theoretical issues that demonstrate that it is possible for English-to-Korean transform of a singing synthesis project. Although the development of a Korean MIDI-to-Singing system is still in working and far from complete. However, the project is likely to be an iterative process whereby the pronunciation of the system improves over time as the settings of the phones are made more precise, and the number of entries in the lexicon is increased. In the future, speech synthesis could also be fully integrated with singing synthesis. It will be challenging to develop new voice synthesis systems that could seamlessly generate any voice produced by a human or virtual singer/speaker.

Acknowledgment

We would like to thank Dr. Mike Macon for his invaluable contributions especially in developed Flinger to demo a MIDI-to-Singing synthesis. We would also like to thank Oregon Graduate Institute of Science and Technology for having made the OGIresLPC Plug-In for the Festival TTS.

References

[1] H. Kenmochi, H. Ohshita, Proc. of Interspeech, (2007)
[2] “UTAU,” Available online: http://utau2008.web.fc2.com/, (2018)
[3] M. Macon, Jensen-Link, Oliverio, Clements, Geourge, Proceedings of ICASSP 97, 435 (1997)
[4] Flinger. Not available online: http://www.cslu.ogi.edu/tts/flinger URL (2007)
[5] Festival Speech Synthesis, Speech Tools & documentation, http://www.festival.org/.
[6] OGIresLPC PlugIn for Festival. www: http://cslu.cse.ogi.edu/tts/download/index.html (1997)
[7] H. C. Shen, and C. N. Lee, 6th International WOCMAT & New Media Conference, Chungli, Taoyuan, Taiwan, (2010)