Computational Linguistics and Chinese Language Processing
Vol. 12, No. 4, December 2007, pp. 371-390

A System Framework for Integrated Synthesis of Mandarin, Min-Nan, and Hakka Speech

Hung-Yan Gu*, Yan-Zuo Zhou*, and Huang-Liang Liau*

Abstract

In this paper, a framework for integrated synthesis of Mandarin, Min-nan, and Hakka speech is proposed. To show its feasibility, an initial integrated system has been built as well. Through integration, a model only trained with Min-nan sentences is used to generate pitch-contours for all three languages, same rules are used to generate syllable duration and amplitude values, and the same program module implementing the method, TIPW, is used to synthesize the three languages' speech waveforms. Also, in this system, each syllable of a language has just one recorded signal waveform, *i.e*. no chance of unit selection. Under such a restricted situation, the synthetic speech signals still have noticeable naturalness level and signal clarity.

Keywords: Speech Synthesis, Pitch Contour Model, TIPW, Time Axis Warping.

1. Introduction

There are many languages in Taiwan, including Mandarin, Min-nan, Hakka, and others spoken by smaller population groups. Mandarin has been more extensively studied than the other languages because it is the official language. However, the successful construction of a synthesis model or system for Mandarin does not imply that the same modeling method can be directly applied to another language. Developing speech synthesis systems for other languages is strongly desired because Mandarin is not the mother tongue of most people in Taiwan, and all languages except Mandarin face the crisis of disappearance.

If systems developed or speech data collected previously can only be used for Mandarin, then further resources (effort and money) are inevitably needed to study other languages. Such a situation will become more severe if a corpus-based approach [Chou 1999; Chu *et al.* 2003] is adopted. In addition, there will be inconsistency in prosody and timbre among

* Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, 43 Keelung Rd., Sec. 4, Taipei, Taiwan
E-mail: {guhy, M9315058, M9215001}@mail.ntust.edu.tw

[Received October 31, 2006; Revised April 17, 2007; Accepted April 25, 2007]

independently developed speech synthesis systems for different languages. Therefore, a better approach is to construct a more generalized system that can synthesize not only Mandarin but also Min-nan and Hakka speech. Such an approach, if successfully realized, can not only save resources but also obtain much higher consistency among the synthesized speech for different languages. Another advantage is that an improvement, when made to a system component, can immediately benefit all the languages supported.

Mandarin, Min-nan, and Hakka are all syllable prominent languages, and are all tonal languages. Hakka has many accents found in users in Taiwan, of which "four-country" and "sea-land" are the primary ones. If not specified, the sea-land accent is the default accent representing Hakka in this paper. This is because it has more lexical tone and more unique syllables than the four-country accent has, and the authors believe the speech signal of the sea-land accent is more difficult to synthesize. As to the number of different syllables (not distinguishing lexical tones), Mandarin has 405, Min-nan has 833, and Hakka has 783 [Yu 1999]. The languages also vary in the numbers of different lexical tones, being 5, 7, and 7, respectively, for Mandarin, Min-nan, and Hakka. Since the numbers, 405, 833, and 783, are not large, syllable is commonly chosen as the speech unit for synthesis processing. Actually, in this system, each syllable (tone not distinguished) of a language has only one recorded utterance. That is, no extra units are available to do unit selection, and each syllable's waveform must be manipulated to synthesize speech signals with different required prosodic characteristics.

Note that the focus of this study is in the system framework of an integrated speech synthesis system for the three languages. To show the feasibility of the proposed framework, a workable integrated synthesis system is built. This system is just in its initial phase. Therefore, there will be many unsolved problems in the details. Most of these problems belong to text analysis since signal synthesis is the major concern in this research while text analysis is just a minor concern. In general, a speech synthesis system can be divided into three subsystems, *i.e.*, (a) text analysis, (b) prosodic parameter generation, and (c) speech waveform synthesis [Shih *et al.* 1996; Wang 1998]. The framework for the integrated synthesis system is also divided into such subsystems. The main processing flow of this framework is shown in Figure 1. The first two processing blocks are for text analysis, the middle two blocks are for prosodic parameter generation, and the last two blocks are for signal waveform synthesis. The synthesis system built is an integrated system, not a bundle of three independent systems for the three languages. This is because the program modules in the three subsystems are all shared in synthesizing the three languages' speech. For example, the model for pitch-contour parameter generation is shared (or adapted) between Mandarin and Hakka, although it is originally trained with Min-nan sentences. The explanations for why the program modules can be shared are given in the following sections.

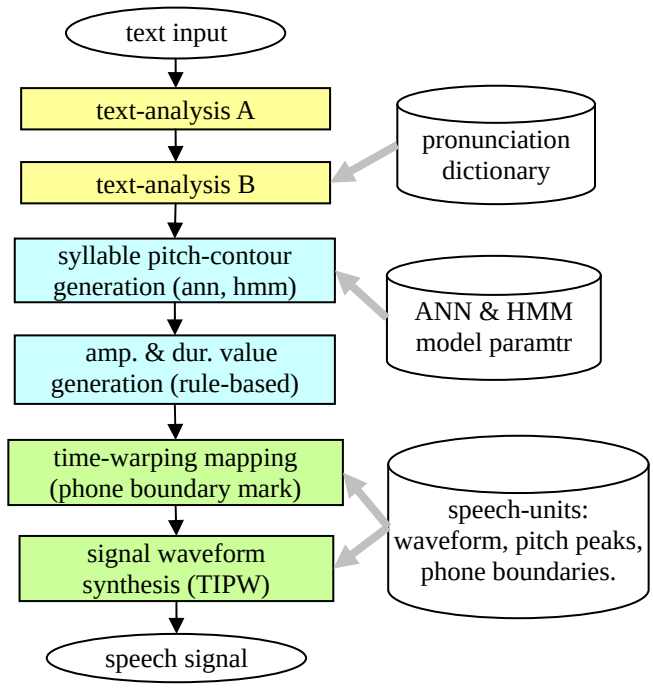

Figure 1. Main processing flow.

In the subsystem of text analysis, the first block in Figure 1, "Text Analysis A", parses the input text to recognize tags and slice the text string into a sequence of Chinese-character or alphanumeric-syllable tokens. Then, each Chinese-character token is tried in the second block (Text Analysis B), to check if it can be looked up in a pronunciation dictionary in order to determine the comprising character's pronunciation syllable. For the subsystem of prosodic parameter generation, the pitch-contour parameters of a syllable are determined by a mixed model of ANN (artificial neural network) [Chen *et al.* 1998; Lee *et al.* 1991] and SPC-HMM (syllable pitch-contour hidden Markov model) [Gu *et al.* 2000] in the third block of Figure 1. As to the parameters, amplitude and duration, their values are determined with a rule-based method [Chiou *et al.* 1991; Shiu 1996] in the fourth block of Figure 1. For the subsystem of signal waveform synthesis, a piece-wise linear time-warping function is first constructed in the fifth block of Figure 1. Then, the method of TIPW (time-proportioned interpolation of pitch waveform) [Gu *et al.* 1998] is used in the sixth block to synthesize speech waveforms. TIPW is an improved variant of PSOLA [Modulines 1990]. To show that the integrated system has noticeable performance in naturalness and signal clarity, the authors have set up a web page, http://guhy.csie.ntust.edu.tw/hmtts/, to demonstrate synthetic speech for the three languages. However, for the purpose of online testing, http://guhy.csie.ntust.edu.tw/hmtts/speak.html is preferable.

2. Text Analysis

In Min-nan and Hakka, there are still many spoken words whose corresponding ideographic words are not known. Therefore, the authors made a decision that, in the input text, Chinese characters may be interleaved with syllables spelled in alphanumeric symbols. For example, "cit-4 tou-5 人" is a Min-nan word whose first two syllables are spelled in alphanumeric symbols. This decision implies that the input text must first be parsed into a sequence of Chinese-character and alphanumeric-syllable tokens. For example, "今天 mai-3 ki-3" is parsed into "今天", "mai-3", and "ki-3". The number at the end of a syllable indicates the lexical tone of the syllable. This parsing processing is executed in the first block of Figure 1.

In addition, the authors have defined several kinds of tags to help carry some necessary controlling information. For example, the tag, "@>2", may be placed between two sentences to command that the sentences behind the tag will be synthesized to Hakka speech until another language-selection tag is encountered. Such a language-selection tag is needed because it is intended that the sentences of an article may be alternatively synthesized to different languages' speech. Another kind of tag is "@>dxxx". This tag may also be placed between two sentences to change the speaking rate of the sentences behind it. The part, "xxx", in the tag represents three decimal digits to specify how many milliseconds on average a syllable will be synthesized to. In addition to the two tags explained, several other kinds of tags are also defined. The details are listed in Table 1. The parsing of such tags is also executed in the first block of Figure 1.

Table 1. Tags and their meanings.

Tag symbol	Explanation
@>x	language selection, x may be 0, 1, or 2. 0: Mandarin, 1: Min-nan, 2: Hakka
@>dxxx	speaking rate, syllable average duration in xxx milliseconds.
@>txxx	average tone height, in xxx Hz.
@>vxxx	vocal track extended (or shrunken) to xxx percents of original length.
<, >	word-constructing tag, *e.g.*, <cit-4 tou-5>
*	breath-break tag

After an input sentence is parsed into a sequence of tokens, the pronunciation syllables for each Chinese character token are determined in the second block of Figure 1. According to the language-selection tag, the corresponding pronunciation dictionaries are consulted to check if the prefix part of a token can be found in the dictionaries. A dictionary consisting of longer words is tried before a dictionary consisting of shorter words. Currently, the authors have collected 55,000, 12,000, and 19,000 multi-syllabic words, respectively, for Mandarin,

Min-nan, and Hakka. Note that input text is usually composed in Mandarin written words. Therefore, use of the dictionary plays a role of word translation. For example, "今天" (today) in Mandarin is translated as "今仔日", in Min-nan, which is pronounced as, "gin-1 a-2 rit-8". Another example, "筷子" (chopstick) in Mandarin is translated to "箸" which is pronounced as, "di-7". These examples also show that the words obtained after translation may have longer or shorter lengths.

After a word is found in a dictionary or a block of syllables bounded with the tags, "<" and ">", is parsed out, one knows the boundaries of a word and its syllabic composition. Then, tone-sandhi rules for the currently selected language can be applied to the compositional syllables of the word. This is executed in the second block of Figure 1. Note that different languages have very different tone-sandhi rules. For example, in a word of Mandarin, if two adjacent syllables are both of the third tone, then the former one must have its tone changed to the second tone. As another example, consider the tone-sandhi rule applied to a word of Min-nan that every syllable of a word except the final one must have its tone changed to its inflected tone.

3. Prosodic Parameter Generation

The prosodic parameters of a syllable include pitch-contour, duration, amplitude, and leading pause. The generation of prosodic parameter values plays a very important role because it determines the level of naturalness of synthesized speech. Therefore, much effort has been devoted to investigate models (or methods) for generating prosodic parameter values [Chen *et al.* 1998; Gu *et al.* 2000; Lee *et al.* 1993; Wu *et al.* 2001; Yu *et al.* 2002].

Among these prosodic parameters, pitch-contour is the most important one for obtaining a higher naturalness level. Therefore, the authors have spent considerable effort in investigating different kinds of models, HMM [Gu *et al.* 2000], ANN, and a mixed model of both [Gu *et al.* 2005b]. In the third block of Figure 1, a mixed model of HMM and ANN is used to generate pitch-contours. Here, model mixing means taking a weighted sum of two pitch-contours generated respectively by HMM and ANN. Note that, in this study, pitch-contour models, HHM and ANN, are both trained with Min-nan spoken sentences. Then, through tone mapping, the Min-nan trained and mixed model is adapted to generate pitch-contours for Hakka and Mandarin. By such a sharing of pitch-contour model, the effort in training other languages' pitch-contour models can be saved.

In contrast to pitch-contour, duration and amplitude are thought to be minor factors for naturalness. Hence, only a rule-based method is used in the fourth block of Figure 1 to generate their values [Chiou *et al.* 1991; Shiu 1996]. The authors program three sets of rules for the three layers, syllable layer, word layer, and breath-group layer. In the syllable layer, a syllable containing different vowel phonemes, /a/, /i/, /u/, /e/, or /o/, is assigned different

amplitude values, 0dB, -4dB, -3dB, -2dB, or -1dB. In the word layer, the first syllable of a word is emphasized 0.5 dB in amplitude. Finally, in the breath-group layer, the first two syllables of a group are emphasized 1dB and 0.5dB respectively. In addition, the last two syllables of the last breath-group of a sentence are deemphasized 0.5dB and 1dB respectively. By interaction of these rules in the three layers, the generated amplitude and duration values appear to have some randomness, and can present a certain level of naturalness.

3.1 Syllable Pitch Contour HMM

A syllable at the beginning of a sentence is usually uttered with higher pitch than one at the end, *i.e.*, the phenomenon of declining. With respect to this phenomenon, the authors imagine that there are three prosodic states corresponding to sentence-initial, sentence-middle, and sentence-final. However, how to assign a sentence's syllables to these states is not explicitly known. Therefore, the authors imagine these prosodic states are hidden and will simulate them by the hidden states of a left-to-right hidden Markov model [Rabiner *et al.* 1993]. Besides the influence of prosodic states, the lexical tones of a syllable and its adjacent syllables also have strong influences. Therefore, the authors take into account the lexical-tones of a syllable and its adjacent syllables, and call such an HMM as syllable pitch-contour HMM (SPC-HMM).

The height and shape of a syllable's pitch-contour are mainly influenced by the lexical tones of the syllable and its immediately adjacent syllables. Therefore, the authors decide to combine the t-th syllable's lexical tone and pitch-contour VQ (vector quantization) code with its left and right adjacent syllables' lexical tones to define the t-th observation symbol, O_t, as

$$O_t = 392 \cdot X_{t-1} + 56 \cdot X_t + 8 \cdot X_{t+1} + V_t, \quad 0 \le X_t \le 6,\ 0 \le V_t \le 7. \tag{1}$$

where X_t is the lexical-tone number of the t-th syllable, and V_t is the pitch-contour VQ code of the t-th syllable in a training sentence. Actually, the number, X_t, is indirectly obtained, *i.e.* lexical-tone number eight is mapped to six beforehand, and then the lexical-tone number is decreased by one. In Equation (1), the number, eight, is multiplied because there are eight codewords in each tone's pitch-contour VQ codebook. The numbers, 56 and 392, may be viewed as 7×8 and 7×7×8, respectively, and 7 is the number of different lexical tones in Min-nan and 8 is the number of code-words in a VQ codebook. When t=1, *i.e.* staying at the first syllable of a training sentence, X_{t-1} is undefined. In this case, the definition of O_t is modified to $7\times7\times7\times8 + 56\ X_t + 8\ X_{t+1} + V_t$. Similarly, the definition of O_t for the last syllable of a sentence must also be modified [Gu *et al.* 2000].

Before VQ encoding, the pitch-contour of each syllable from a training sentences is first time normalized and then pitch-height normalized [Gu *et al.* 2000]. Time normalization means

placing 16 measuring points equally spaced in time. Then, a pitch-contour is represented as a vector of 16 frequency values (in log Hz scale), called a frequency vector. After time normalization, these frequency vectors must be normalized in pitch-height to eliminate the influence of the speaker's mood at the time of recording. Totally, the authors have recorded 643 Min-nan training sentences that are comprised of 3,696 syllables.

Next, consider the generating of pitch-contours by using SPC-HMM. When a sentence is input, it will be analyzed first by the textual analysis components. Hence, its pronunciation-syllable sequence is available. For example, for the short sentence, "我來啊" (I have come), of Min-nan, its corresponding syllable sequence is "qua-1 lai-5 a-7". Then, one can encode the three adjacent syllables' lexical tones partially (because VQ code, V_t, is not known yet) according to Equation (1). Since each lexical tone has 8 codewords in its pitch-contour VQ codebook, each syllable of the sentence has 8 possible encoded observation symbols corresponding to it. For example, for the second syllable "lai-5", its possible encoded observation symbols are 392(1-1)+56(5-1)+8(7-1)+V_t, *i.e.* the value range from 264 to 271 since the value of V_t is not determined yet. Therefore, in the synthesis phase (or testing phase), besides the time (syllable index within a sentence) and state (prosodic-state index) axes, a third axis to index the 8 possible observation-symbol candidates, must be added. Then, the conventional two-dimensional (time and state) DP (dynamic programming) algorithm for speech recognition is extended to a three-dimensional DP algorithm and used to search the most probable path [Gu *et al.* 2000]. The main part of the extended algorithm is shown in Equation (2),

$$\delta_t(n,k) = \left[\max_{n-1 \le i \le n} \max_{0 \le j \le 7} \delta_{t-1}(i,j) \cdot a_{i,n} \right] \cdot b_n(O_t^k), \quad 0 \le n \le 2, \ 0 \le k \le 7, \tag{2}$$

where O_t^k represents the *k*-th possibly encoded observation symbol at time *t*, *n* and *i* are state indices, $a_{i,n}$ is state-transition probability, $b_n(\bullet)$ is symbol-observing probability at state *n*, and $\delta_t(n,k)$ is the largest obtainable probability of a best path that stays at state *n* and selects the *k*-th observation symbol at time *t*. According to the best path found, the state value and *k* value of O_t^k at each time point, *t*, can then be determined. Accordingly, the pitch-contour VQ code, V_t, for the *t*-th syllable of the sentence is set to the value of *k* determined at time *t*.

3.2 Syllable Pitch Contour ANN

The architecture of the artificial neural network used here is shown in Figure 2. It is designed to be a recurrent type ANN in order to have the prosodic state kept internally. The input layer of the ANN has 8 ports to receive contextual parameters. For the hidden and recurrent hidden layers, the numbers of nodes are both set to be 30, according to experiment results. After a syllable's contextual parameters are input and processed, a pitch contour represented as a 16

dimensional frequency vector is output in the output layer. This frequency vector can be interpreted as a sequence of 16 frequency values along a pitch-contour.

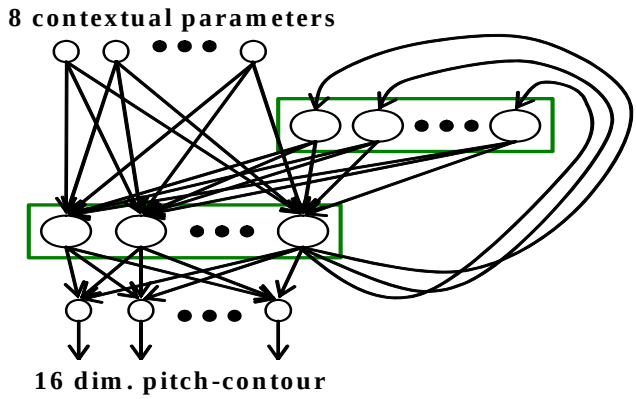

Figure 2. The architecture of the ANN studied here.

Here, the contextual parameters, *i.e*. the inputs to the ANN, are appropriately selected to provide essential contextual information and to lower the quantity of required training sentences. In detail, the contextual parameters are as listed in Table 2. As there are seven lexical tones in Min-nan, 3 bits are enough to represent them. The numbers of different syllable initials and finals are 18 and 61, respectively. Hence, 5 and 6 bits, respectively, are used to represent current syllable's initial and final types. As to the parameter of the previous syllable's final, the authors first group the 61 possible finals into 12 classes, using only 4 bits to represent the 12 final classes. Similarly, the authors first group the 18 possible initials into 6 classes, and use only 3 bits to represent the 6 initial classes for the next syllable's initial. Grouping is made here because the quantity of recorded training sentences is not large enough to let the ANN learn the influences of the detailed combinations of current syllable and previous (or next) syllable. Syllable initial and final classes grouped here are detailed in Table 3 and 4, respectively. The last item in Table 2, the time-progress index, is intended to carry timing information. If the current syllable is the k-th syllable of a sentence of N syllables in length, then the value of time-progress index is set to the floating-point number k/N.

Table 2. Contextual parameters.

Items	Tone of previous syllable	Final class of previous syllable	Tone of current syllable	Initial of current syllable	Final of current syllable	Tone of next syllable	Initial class of next syllable	Time progress index
Bits	3	4	3	5	6	3	3	void

Table 3. Syllable-initial classes. (General Phonetic Symbol System)

Classes	1	2	3	4	5	6
Initials	(null), m, n, l, r, ng, q, v	h, s	b, d, g	z	c	p, t, k

Table 4. Syllable-final classes. (General Phonetic Symbol System)

Classes	Finals	Classes	Finals
1	(null)	7	-i, -u, -ui, -iu
2	-a, -ia, -ua	8	-ing, -eng, -in, -un, -en
3	-o, -io, -ior	9	-ang, -iang, -uang, -ong, -iong, -an, -uan
4	-er, -ier	10	-am, -iam, -im, -om
5	-e, -ue	11	-ah, -eh, -ih, -oh, -uh, -auh, -erh, -iah, -ierh, -ioh, -uah, -ueh
6	-ai, -uai, -au, -iau	12	-ap, -iap, -ip, -op, -at, -et, -it, -uat, -ut, -ak, -iak, -ik, -iok, -ok

3.3 Adaptation of Pitch Contour Model

Here, the pitch-contour model trained with Min-nan sentences is used as the working model. This working model can be adapted in a way to generate pitch contours for a target (Mandarin or Hakka) language's sentences. In detail, a lexical-tone sequence, X_1X_2, …, X_n, extracted from a target language's sentence is first mapped to a lexical tone sequence, Y_1Y_2, …, Y_n, for the working language. Then, the mapped lexical tones are used instead as the input for the working model. The pitch-contours, R_1R_2, …, R_n, generated by the working model are treated as the output of the adapted model for the sequence, X_1X_2, …, X_n.

The reasons why this adaptation method may work are explained in detail in [Gu *et al.* 2005a]. In brief, slight differences in frequency-height or boundary-part shape between two pitch-contour curves need not be worried about for correct recognition of the carried lexical tone. The authors also think it is reasonable to approximate a pitch-contour curve in a target language with a curve-shape class trained in the working language that is of similar shape in the central part. Note that each lexical tone of the working language is usually trained to have several representative curve-shape classes, *e.g.* 8 code-words in each lexical tone's VQ codebook. Hence, for a pitch-contour curve from a target language, one can select from the curve-shape classes trained for the mapped lexical tone to pick out the curve that is most similar. Then, the possible decrease in naturalness due to differences in frequency-height or boundary-part shape can be minimized.

Note that some syllable initials and finals of Mandarin (*e.g.*, /yu/) and Hakka (*e.g.*, /oi, eu/) are not found in Min-nan. With respect to this, it may make one suspect if the adaptation method can indeed work. However, in SPC-HMM, the definition of observation symbol in equation (1) does not include syllable initials and finals. This indicates that pitch-contour is only insignificantly influenced by syllable initials and finals according to previous studies [Gu *et al.* 2000; Gu *et al.* 2005b]. Although the information of syllable initial and final is used in the pitch-contour ANN, the authors still think that the factors of syllable initial and final are insignificant according to previous experiments for evaluating classification methods of Min-nan initials and finals [Gu *et al.* 2005b]. Anyway, to solve the problem of mismatched initials and finals between the two languages, the authors let the program automatically select a similar one (more same letters in spelling) to replace a finial or initial not found in Min-nan. For example, /yu/ is replaced with /u/, and /oi, eu/ are replaced with /ai, au/ respectively. The authors think such replacements are acceptable.

The mappings from Hakka tones to Min-nan tones are listed in Table 5. The mapping from sea-land Hakka to Min-nan, Table 5(a), can be said to be a nice one-to-one mapping because both have the same number of lexical tones, and for each lexical tone of Hakka one can find a lexical tone in Min-nan that has almost same pitch-contour shape. The mapping from four-country Hakka to Min-nan, Table 5(b), is also straightforward. If tone number 7 is removed from Min-nan, then this mapping is still a nice one-to-one mapping.

Table 5. Tone mapping from Hakka to Min-nan.

(a) sea-land Hakka to Min-nan

Hakka tone number	1	2	3	4	5	7	8
Mapped Min-nan tone number	2	5	3	8	1	7	4
Example Chinese characters	衫	短	褲	寬	人	鼻	直

(b) four-country Hakka to Min-nan

Hakka tone number	1	2	3	4	5	8
Mapped Min-nan tone number	5	2	1	4	3	8
Example Chinese characters	夫	虎	富	福	湖	復

The mapping from Mandarin tones to Min-nan tones is listed in Table 6. Three of the Mandarin lexical tones, *i.e.* high-level, rising, and falling, also exist as Min-nan lexical tones. Besides these three tones, the low-level tone of Min-nan and the low-dipping tone of Mandarin are perceived to be almost identical. Therefore, the low-dipping tone of Mandarin

can be mapped to the low-level tone of Min-nan. The neutral tone of Mandarin has a shorter duration than the other tones. This contrast also exists in Min-nan, *i.e.* both abrupt tones have shorter durations. In addition, the low-abrupt tone of Min-nan has a low pitch-height, just as the neutral tone has. Hence, the neutral tone of Mandarin can be mapped to the low-abrupt tone of Min-nan.

Table 6. Tone mapping from Mandarin to Min-nan.

Mandarin tone number	1	2	3	4	5
Mapped Min-nan tone number	1	5	3	2	4
Example Chinese characters	加	油	打	氣	的

Here, to show the abilities of the adapted pitch-contour models, the authors take the Mandarin chunk, "花店的老闆" (boss of a flower shop), as an example. The sequence, X_1X_2, …, X_n, is hence, 1, 4, 5, 2, 3. And after lexical-tone mapping, the sequence, Y_1Y_2, …, Y_n, is obtained as 1, 2, 4, 5, 3. Then the mapped sequence and relevant contextual information are fed into the HMM and ANN models, respectively. The pitch-contours output by the two models are shown in Figure 3. The solid line is generated by the HMM model, the dotted line is generated by the ANN model, and the gray line is a mixture of the former two. Basically, these lines can all be recognized in their carried lexical tones. On the other hand, apparent differences in pitch heights can also be seen in the middle three syllables. Due to this phenomenon, the authors think the mixed pitch contour would be the better choice.

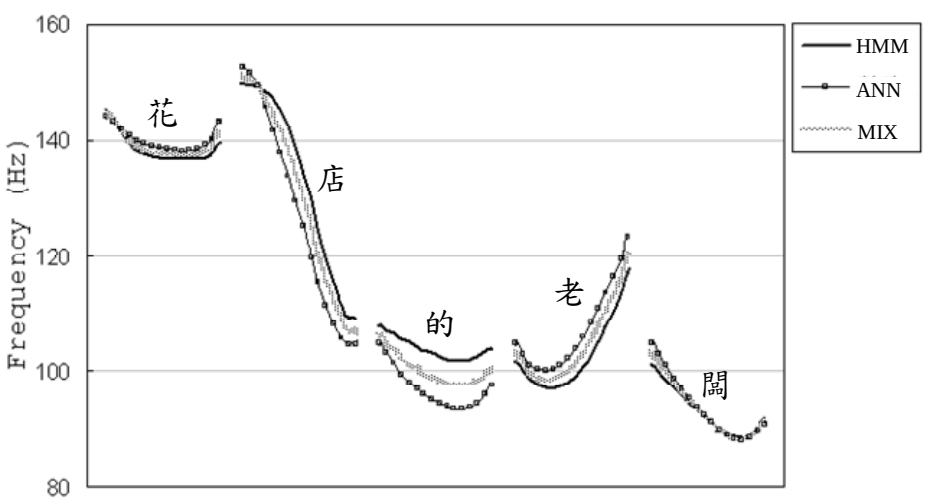

Figure 3. Pitch contours generated by adapted models.

4. Signal Waveform Synthesis

In this system, each syllable of a language has only one utterance recorded in a level tone (high or medium level). Therefore, the original syllable waveform must be manipulated to obtain synthesized waveforms with different prosodic characteristics. The synthesis method used here is TIPW [Gu *et al.* 1998]. TIPW is an improved variant of PSOLA, *i.e*. the effects of chorus and reverberation are largely reduced. Besides, TIPW is capable of adjusting vocal track length through re-sampling [Gu 2001].

Originally, TIPW was developed for synthesizing Mandarin speech. Thus, it does not support the synthesis of signal waveform with suddenly changed amplitude that is often found at the ending portion of an abrupt-tone syllable, *e.g*. /zit8/. Nevertheless, abrupt-tone syllables are very frequently used in Min-nan and Hakka. One method to overcome this difficulty is to treat the end portion of an abrupt-tone syllable as a stop consonant. Then, the same method used in synthesizing a stop consonant at the syllable initial portion can also be adopted to solve this problem.

4.1 A Fluency-Improving Method

In addition, the authors have made another improvement to TIPW. This improvement significantly increases the fluency of the synthesized speech. In an ordinary speech synthesis system, the subsystem of prosodic-parameter generating only determines the duration value, *Es*, of a syllable to be synthesized. The detailed dividing of syllable duration, *Es*, to its comprising phonemes is, however, not controlled by the prosody subsystem. Furthermore, the subsystem of signal waveform synthesis usually extends (or shrinks) the original speech waveform to an intended time length in a linear manner. According to this study, linear extending (or shrinking) of time length is a major cause of a decrease in much of the fluency of synthesized speech.

Consider an example syllable, /man/. Suppose that, in its original recorded waveform, the three phonemes, /m/, /a/, and /n/, occupy *Dm*, *Da*, and *Dn* milliseconds, respectively, and *Ds* = *Dm* + *Da* + *Dn*. A phenomenon that can be observed is that the ratio, (*Dm*+*Dn*)/*Ds*, will become smaller when /man/ is uttered within a sentence instead of being uttered in isolation. Currently, the authors are studying a simple method to simulate this phenomenon. In further research, the authors will study it with a more systematic method. The method used here is as depicted in Figure 4. That is, a piece-wise linear function is used to map the time-axis of a synthetic syllable waveform to the time-axis of its original waveform. In Figure 4, the symbols,

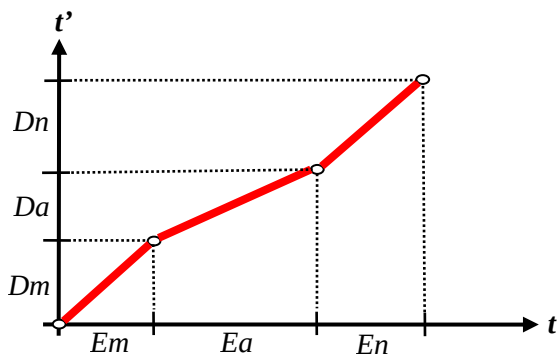

Figure 4. Piece-wise linear mapping function.

Em, *Ea*, and *En* represent the time lengths of the three phonemes, /m/, /a/, and /n/ in the synthesized waveform while *Dm*, *Da*, and *Dn* represent these three phonemes' time lengths in the original waveform. In this system, the values of *Em*, *Ea*, and *En* are determined by the following procedure:

```
r = 0.6;
while ( r >= 0.1 ) {
   Em = (Dm/Ds) * r * Es;
   En = (Dn/Ds) * r * Es;
   Ea = Es - Em - En;
   if (Ea > Es*0.4) break;
   r = r - 0.05;
}
Eb = Em + En;
if (Em > 0 && Em/Eb < 0.35) { Em = 0.35*Eb;  En=Eb-Em; }
if (En > 0 && En/Eb < 0.35) { En = 0.35*Eb;  Em=Eb-En; }
```

If the structure of a syllable is the same as /san/ or /an/, *i.e.* without voiced initial consonant, then the values of *Dm* and *Em* can be set to zero directly. Similarly, if the structure of a syllable is the same as /ma/, *i.e.* without a voiced ending consonant, then the values of *Dn* and *En* can be set to zero directly. Apparently, to apply the procedure given above, the boundary points between adjacent phonemes must be labeled beforehand in order to compute the values of *Dm*, *Da*, and *Dn*, and to construct the mapping function.

4.2 Example Waveforms

4.2.1 TIPW Synthesis Method

Since the synthesis method, TIPW, is not as popular as PSOLA, the authors will illustrate its processing steps with signal waveforms. To obtain a complete view of TIPW, including a detailed explanation of the method, see [Gu *et al.* 1998]. Here, let the two adjacent pitch periods in Figure 5(a) be around the mapped (using the piece-wise linear mapping function) time point, τ_m, in a recorded syllable. The first step of TIPW is to determine the weights, w_1 and w_2, for the left and right pitch periods. The value of w_2 is computed as $(\tau_m - \tau_1) / (\tau_2 - \tau_1)$ where τ_1 and τ_2 are time points of the left and right pitch periods' centers respectively. The value of w_1 is simply $1 - w_2$. By weighting the two pitch periods with w_1 and w_2 respectively, one can obtain the two waveforms shown in Figure 5(b). Here, weighting a signal waveform with a weight, *w*, means that the value of each signal sample in the waveform is multiplied by the weight, *w*.

The second step is to window the pitch waveforms with two Hanning (or cosine) window halves. Here, windowing a signal waveform $x(n)$ with a window function $f(n)$ means that the result sample value at time n is $x(n) \cdot f(n)$, *i.e.* one-to-one multiplying. The two waveforms in Figure 6(a) are obtained by windowing the left pitch period in Figure 5(b) with two symmetric half Hanning windows. Here, the window length, L, is set to the smaller of L_1 and L_m where L_1 is the left pitch period's length and L_m is the length of the period to be synthesized. The detailed formula for the two window functions used in the left and right sides, respectively, of Figure 6(a) are:

$$f_{left}(n) = 0.5 + 0.5\cos\left(\frac{n}{L}\cdot\pi\right), \quad n = 0,\ 1,\ 2,\ ...,\ L-1, \tag{3}$$

$$f_{right}(n) = 0.5 - 0.5\cos\left(\frac{n}{L}\cdot\pi\right), \quad n = 0,\ 1,\ 2,\ ...,\ L-1. \tag{4}$$

After windowing, the signal samples' values will be depressed in proportion to the window function's curve height. This can be seen from Figure 6(a). Similarly, the two waveforms in Figure 6(b) are obtained by windowing the right pitch period in Figure 5(b) with two symmetric half Hanning windows. However, the window length is now set to the smaller of L_2 and L_m. Note that the window length determination rules are important because they can prevent the effects of reverberation and dual-tones.

Then, as the last step, the four waveforms in Figure 6(a) and Figure 6(b) are overlapped and added to obtain a synthesized pitch period whose waveform is shown in Figure 7. Here, "overlapped" means that the four waveforms' locations on the time axis are left or right shifted in order that they have same starting or ending times. In detail, the two waveforms on

the left side of Figures 6(a) and 6(b) are left aligned to have same starting time, 0, in terms of the time axis of Figure 7. In contrast, the two waveforms on the right side of Figures 6(a) and 6(b) are right aligned to have same ending time, L_m-1. "Added", as used here, means that at each time point *n*, the four waveforms' four signal-sample values are added together to become the signal sample at time *n* for the resulted waveform.

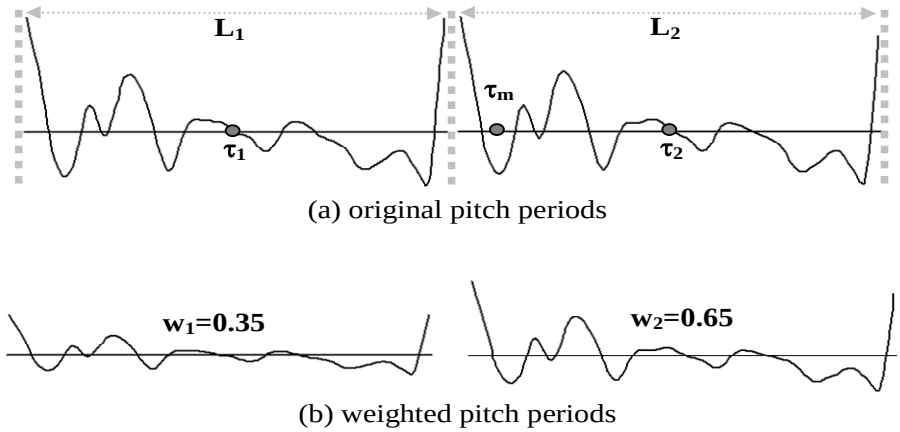

Figure 5. Original and weighted waveforms of two adjacent pitch periods.

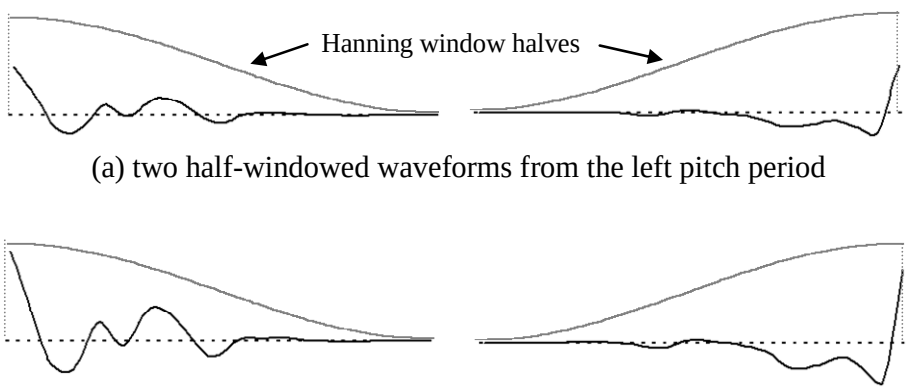

Figure 6. Hanning windowed pitch waveforms.

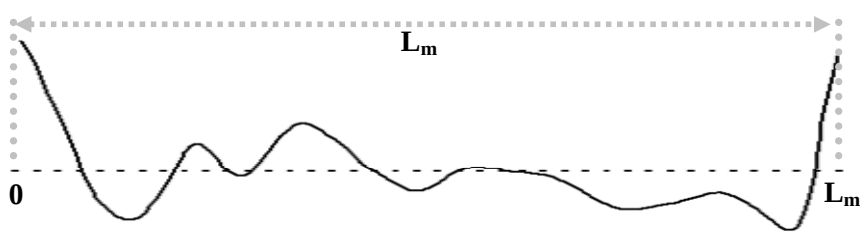

Figure 7. Synthesized pitch waveform.

4.2.2 Piece-wise Linear Mapping

To demonstrate the effect of the piece-wise linear mapping function, take the Mandarin word, "農田" (farmland) as an example and show its signal waveforms obtained from the original recording and the synthesis processing. The waveform in Figure 8 is a direct concatenation of the recorded waveforms of /nong-1/ and /tien-1/ while the waveform in Figure 9 is synthesized by this system. From Figure 8, it can be observed that the /ng/ part in /nong/ and the /n/ part in /tien/ both occupy a large portion of the syllable duration. However, through the remedying of the piece-wise linear mapping function, this phenomenon is largely reduced, and the fluency of the synthesized speech is improved greatly. It can be seen from Figure 9 that the duration ratios of /ng/ to /nong/ and /n/ to /tien/ now apparently become smaller.

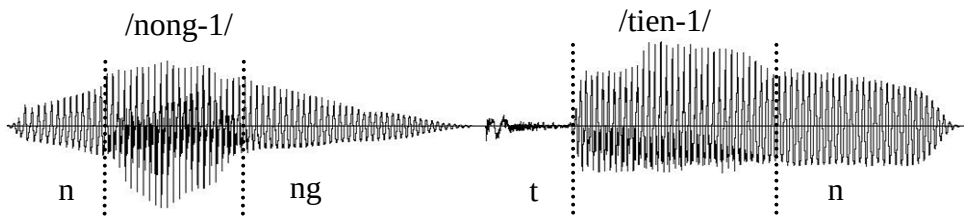

Figure 8. Direct concatenation of the recorded syllables, /nong-1/ and /tien-1/.

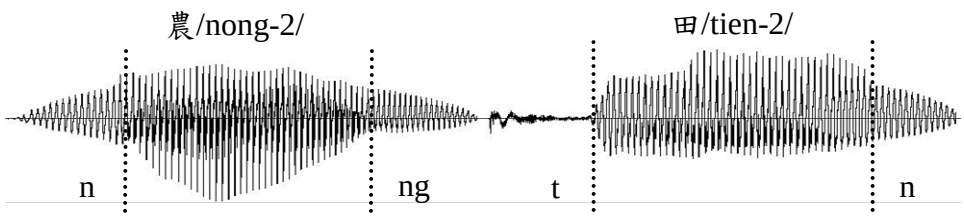

Figure 9. A synthetic waveform for /nong-2 tien-2/.

5. Experiments and Results

After the integrated system is implemented and ready to run, perception tests are conducted. The first issue of concern is the naturalness levels of Mandarin and Hakka pitch-contours generated by the Min-nan trained pitch contour models. Therefore, three short articles written in Mandarin, Hakka and Min-nan, respectively, are fed as inputs into the system. Then, the output speeches are played to each of the persons participating in the tests. Here, ten persons studying in university were invited to evaluate the synthetic speeches. For each person, two pairs of speech files were played, (*Sn*, *Sm*) and (*Sn*, *Sh*). *Sn*, *Sm*, and *Sh* represent synthetic Min-nan, Mandarin, and Hakka speeches, respectively. Then, the person was requested to give a score of -2, -1, 0, 1, or 2 for each pair. Here, 2 and -2 mean "better", 1 and -1 mean "slightly better", and 0 means "almost identical". As to the sign, positive sign means the latter is more

natural, and negative sign means the former is more natural. According to the scores given by the ten persons, the averaged scores are computed to be -0.6 for Mandarin and -0.2 for Hakka. That is, the synthetic Hakka speech was perceived to be almost as natural as the synthetic Min-nan speech. But the synthetic Mandarin speech was perceived to be less natural than the Min-nan speech. This is because the pitch-contours generated for Mandarin were perceived to have a slightly strange accent.

Another issue of concern is the fluency of the synthetic speech. Hence, two synthesis conditions are considered here for synthesizing a short Mandarin article. In the first condition, the mapping function, used in waveform synthesis, between the synthetic syllable's time axis and the recorded syllable's time axis is forced to be linear. In the other condition, the mapping function shown in Figure 4 is adopted. The two synthetic speeches obtained under the two conditions were played to each of the participating persons to compare their fluency. Again, each person was requested to give a score of -2, -1, 0, 1, or 2. The meanings of these numbers are as mentioned above. The average score was computed to be 0.4. That is, the fluency of the synthetic speech under the second condition was better than the one under the first condition.

6. Concluding Remarks

In this paper, the authors intend to promote the idea of synthesizing Mandarin, Min-nan, and Hakka speech with an integrated system. To show it is feasible, a possible system framework and some feasible implementation methods for the system components are proposed. According to the system framework and implementation methods presented, the authors have built an integrated speech synthesis system for the three languages. Then, speech files output from the system were used to perform perception tests. The initial results show that the lexical tone carried in the synthetic Mandarin and Hakka speeches can all be correctly recognized even though the pitch-contour models are trained with Min-nan sentences. As to naturalness level, the synthetic Hakka speech is perceived to be more natural than the synthetic Mandarin speech. How to interpret this phenomenon, though, is left to further studies.

Acknowledgments

This study is supported by National Science Council under the contract number, NSC 94-2218-E-011-007.

Reference

Chen, S. H., S. H. Hwang, and Y. R. Wang, "An RNN-based Prosodic Information Synthesizer for Mandarin Text-to-Speech," *IEEE Trans. on Speech and Audio Processing*, 6(3), 1998, pp. 226-239.

Chiou, H. B., H. C. Wang, and Y. C. Chang, "Synthesis of Mandarin Speech Based on Hybrid Concatenation," *Computer Processing of Chinese and Oriental Languages*, 5(1), 1991, pp. 217-231.

Chou, F. C., Corpus-based Technologies for Chinese Text-to-Speech Synthesis, PhD thesis, National Taiwan University, Taipei, Taiwan, 1999.

Chu, M., H. Peng, Y. Zhao, Z. Niu, and E. Chang, "Microsoft Mulan - a Bilingual TTS System," In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2003, Hong Kong, China, vol. 1, pp. 264-267.

Gu, H. Y., and W. L. Shiu, "A Mandarin-Syllable Signal Synthesis Method with Increased Flexibility in Duration, Tone and Timbre Control," *Proceedings of the National Science Council ROC(A)*, 22(3), 1998, pp. 385-395.

Gu, H. Y., and C. C. Yang, "A Sentence-Pitch-Contour Generation Method Using VQ/HMM for Mandarin Text-to-speech," In *Proceedings of International Symposium on Chinese Spoken Language Processing*, 2000, Beijing, China, pp. 125-128.

Gu, H. Y., "Signal Resampling in Speech Synthesis," In *Proceedings of the 5th World Multi-conference on Systemics, Cybernetics and Informatics*, 2001, Orlando, USA, vol. vi, pp. 521-525.

Gu, H. Y., and H. C. Tsai, "A Pitch-Contour Model Adaptation Method for Integrated Synthesis of Mandarin, Min-Nan, and Hakka Speech," In *Proceedings of the 9th IEEE International Workshop on Cellular Neutral Networks and their Applications*, 2005, Hsin-Chu, Taiwan, pp. 190-193.

Gu, H. Y., and W. Huang, "Min-Nan Sentence Pitch-contour Generation: Mixing and Comparison of Two Kinds of Models," In *Proceedings of Conference on Computational Linguistics and Speech Processing (ROCLING)*, 2005, Tai-Nan, Taiwan, pp. 213-225. (in Chinese)

Lee, L. S., C. Y. Tseng, and C. J. Hsieh, "Improved Tone Concatenation Rules in a Formant-Based Chinese Text-to-Speech System," *IEEE Trans. Speech and Audio Processing*, 1(3), 1993, pp. 287-294.

Lee, S. J., K. C. Kim, H. Y. Jung, and W. Cho, "Application of Fully Recurrent Neural Networks for Speech Recognition," In *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, 1991, Toronto, Canada, pp. 77-80.

Modulines, E., and F. Charpentier, "Pitch-synchronous Waveform Processing Techniques for Text-to-Speech Synthesis Using Diphones," *Speech Communication*, 9(5), 1990, pp. 453-467.

Rabiner, L., and B. H. Juang, *Fundamentals of Speech Recognition*, Prentice Hall, New Jersey, 1993.

Shih, C., and R. Sproat, "Issues in Text-to-Speech Conversion for Mandarin," *International Journal of Computational Linguistics and Chinese Language Processing*, 1(1), 1996, pp. 37-86.

Shiu, W. L., A Mandarin Speech Synthesizer Using Time Proportioned Interpolation of Pitch Waveform, Master Thesis, National Taiwan University of Science and Technology, Taipei, Taiwan, 1996. (in Chinese)

Wang, R. H., "Overview of Chinese Text-to-Speech System," In *Proceedings of International Symposium on Chinese Spoken Language Processing*, 1998, Singapore.

Wu, C. H., and J. H. Chen, "Automatic Generation of Synthesis Units and Prosodic Information for Chinese Concatenative Synthesis," *Speech Communication*, 35(3). 2001, pp. 219-237.

Yu, B. C., Z. C. Syu, and C. N. Wu, *General Phonetic Symbol System for Languages in Taiwan*, Nan-Tien Book Company, Taipei, 1999.

Yu, M. S., N. H. Pan, and M. J. Wu, "A Statistical Model with Hierarchical Structure for Predicting Prosody in a Mandarin Text-to-Speech System," In *Proceedings of International Symposium on Chinese Spoken Language Processing*, 2002, Taipei, Taiwan, pp. 21-24.