Journal of Computer Science and Cybernetics, V.31, N.1 (2015), 1–16
DOI: 10.15625/1813-9663/31/1/5064

IMPROVING THE NATURALNESS OF CONCATENATIVE VIETNAMESE SPEECH SYNTHESIS UNDER LIMITED DATA CONDITIONS

PHUNG TRUNG NGHIA[1], LUONG CHI MAI[2] AND MASATO AKAGI[3]

[1] *Thai Nguyen University of Information and Communication Technology*;
[2] *Institute of Information Technology, Vietnam Academy of Science and Technology*;
[3] *Japan Advanced Institute of Science and Technology.*
Email: ptnghia@ictu.edu.vn

Abstract. Building a large speech corpus is a costly and time-consuming task. Therefore, how to build high-quality speech synthesis under limited data conditions is an important issue, specifically for under-resourced language like Vietnamese. As the most natural-sounding speech synthesis is currently concatenative speech synthesis (CSS), it is the target speech synthesis under study in this research. All possible units of a specific phonetic unit set are required for CSS. This requirement might be easy for verbal languages, in which the number of all units of a specific phonetic unit set such as phoneme is relatively small. However, the numbers of all tonal phonetic units are significant in tonal languages, and it is difficult to design a small corpus covering all possible tonal phonetic units. Additionally, as all context-dependent phonetic units are required to ensure the naturalness of corpus-based CSS, it needs a large database with a size up to dozens of gigabytes for concatenation. Therefore, the motivation for this work is to improve the naturalness of CSS under limited data conditions, and both of these two mentioned problems are solved. First, the authors attempt to reduce the number of tonal units required for the CSS of tonal languages by using a method of tone transformation and second to reduce mismatch-context errors in concatenation regions to make the CSS available if matching-context units could not be found from the database. Temporal Decomposition (TD), which is an interpolation method decomposing a spectral or prosodic sequence into its sparse event targets and corresponding temporal event functions, is used for both tasks. Previous studies have revealed that TD can efficiently be used for spectral transformation. Therefore, a TD-based transformation of fundamental frequency (F0) contours, which represents the lexical tones in tonal languages, is proposed. The concept of TD is also close to that of co-articulation of speech, which is related to the contextual effect in CSS. Therefore, TD is also used to model, select, and modify co-articulated transition regions to reduce the mismatch-context errors. The experimental results obtained from a small Vietnamese corpus demonstrated that the proposed lexical tone transformation is able to transform lexical tones, and the proposed method of reducing the mismatch-context errors in the CSS of the general language is efficient. As a result, the two proposed methods are useful to improve the naturalness of Vietnamese CSS under limited data conditions.

Keywords. Concatenative speech synthesis, temporal decomposition, co-articulation, tone transformation, limited data, Vietnamese speech

1. INTRODUCTION

Building a large-scale speech corpus is a costly task requiring a long time and a great deal of effort by engineers, acousticians, and linguists. Therefore, to build high-quality speech synthesis with limited data is an important and practical issue, specifically for under-resourced languages with which only a few small speech corpora are usable.

CSS is based on the concatenation of segments of recorded speech [1, 2] and state-of-the-art CSS is corpus-based that requires a large database to select the matching units for every concatenation. As corpus-based CSS, which is usually referred to as the unit selection, is currently the most natural-sounding speech synthesis [2], it is chosen as the target speech synthesis discussed in this paper.

Speech is the result of sequential linking of phonetic units such as phonemes, which are the minimal distinctive units. Therefore, a speech synthesizer needs a database that covers all phonetic units in a specific unit set to synthesize any input text content. In CSS, this database is used to concatenate for synthesizing. The need of covering all possible units leads to a requirement of significant amount of data to build a CSS. Since the number of all units of a specific phonetic unit set is limited in verbal languages, this drawback is not serious for these languages. On the contrary, the numbers of all tonal units significantly increases in tonal language like Vietnamese, and it is difficult to design a small corpus that covers all possible tonal phonetic units. As a result, reducing the number of tonal units for CSS of tonal languages is an important issue studied in this research to improve the usability of CSS under limited data conditions.

The boundaries between adjacent phonetic units such as phonemes are usually blurred, resulting in the lying of essential information in the sound transitions. This phenomenon of the mutual influence of adjacent phones, which are the acoustic realization of phonemes, is called co-articulation. Due to the efforts of co-articulation in speech synthesis, not only all context-independent phonetic units but also all context-dependent phonetic units are necessary to synthesize natural speech. Therefore, state-of-the-art CSS systems require large-scale speech corpora with sizes up to dozens of gigabytes to synthesize natural speech [3]. On the contrary, mismatch-context error occurs frequently under limited data conditions. Therefore, reducing mismatch-context error in CSS is one serious problem also studied in this research to solve for constructing high-quality CSS under limited data conditions.

The motivation for this work is to improve the naturalness of Vietnamese CSS under limited data conditions. Therefore, we solve both problems aforementioned are solved. A method of tone transformation is proposed to reduce the number of tonal units required for Vietnamese CSS. Other methods of reducing mismatch-context errors in concatenation regions are also proposed to make Vietnamese CSS available even if the matching-context units are not found from the database. Although there are many researches on Vietnamese speech synthesis using large corpus [4, 5, 6], the problem of Vietnamese speech synthesis under limited data conditions mentioned in this paper have not been considered.

2. PROPOSED TONE TRANSFORMATION FOR CSS OF TONAL LANGUAGES

2.1. Using Tone Transformations in CSS of Tonal Languages

Changing the tone for each pronunciation in tonal languages provides a set of tonal units, referred to as a same-phonation set in this paper. An example of a same-phonation set for monophone /a/ in Vietnamese is (a, à, á, a?, ạ, ã). The spectral envelope features for all units in a same-phonation set are almost the same because they are related to similar vocal tract parameters produced by similar pronunciation behaviors. Therefore, tone transformation can be applied to the CSS of tonal languages by combining the transformed F0 contours of tonal units such as (à, á, a?, ạ, ã) with the original spectral envelope of a representative unit in a same-phonation set such as *a* to produce synthetic sounds of these tonal units with a source/filter vocoder. A neutral unit with neutral tone, which is a tone with a flat F0 contour that is usually found in tonal languages [7], can be used as the easiest representative unit of a same-phonation set [8]. In some voice transformation systems [9], the spectral envelope features are also preserved and only F0 contours are transformed. Therefore, F0 contour of a lexical tone of a unit can be transformed to those of other units in a same-phonation set with the manner similar to that in voice transformation systems. As a result, in this paper, the proposed F0 contour transformation method for converting lexical tones is based on the general framework of voice transformation systems.

Assuming that all tonal units are converted instead of the original ones being stored and denote the theoretical percentage of data reduction as $\mathrm{r_f}$. Then, $\mathrm{r_f}$ can be approximately computed as given in Eq. (1),

$$\mathrm{r_f} = (1 - N_n/N_t) \times 100\% \tag{1}$$

where N_n is the number of neutral units and N_t is the number of tonal units.

There are a total of approximately 7000 meaningful tonal syllables and 1200 neutral syllables in Vietnamese [10]. Thus, $r_t \approx 83\%$ with Vietnamese CSS if the tones are transformed for all tonal syllables. As a result, transforming the F0 contour of lexical tones reduces a significant amount of the tonal units required for the CSS of tonal languages.

2.2. Proposed MRTD-GMM for Transforming F0 Contours of Lexical Tones

The state-of-the-art F0 transformation in voice transformation is based on the Gaussian-Mixture-Model (GMM) [9]. However, although the conventional GMM-based voice transformations have many advantages such as the use of a few target data, they suffer from several drawbacks, including insufficiently precise GMM models and parameters, insufficiently smooth converted parameters between frames, and over-smooth converted frames [11]. A framework for spectral sequence transformation combined by GMM and Modified-Restricted-Temporal-Decomposition (MRTD) [12], named MRTD-GMM [11], is proposed to overcome these drawbacks of conventional GMM-based voice transformation with significant improvements. The results on transformation of spectral sequences obtained by B. Nguyen and Akagi [11] demonstrate that converting only static event targets and preserving dynamic event functions could efficiently improve the estimates of GMM parameters as well as efficiently eliminate the frame-to-frame discontinuities compared with conventional GMM voice transformations, resulting in natural and smooth transformed speech. However, MRTD-GMM still suffers from two main drawbacks when being applied to prosodic features such as the F0 contour. Because dynamic features of F0 are important, both static and dynamic features of F0 need to be transformed. Normally, transforming the dynamic features with TD requires the transformation of dynamic event functions, which is not usable in original MRTD-GMM.

There are two options of transforming dynamic features with TD [13], one is transforming the dynamic event functions and the other is transforming the deltas of static event targets. As the dynamic event functions presents the relations between sparse event targets and static frames, transforming them means transforming dynamic features in all frames. This is sophisticated and may not suitable to transform the lexical tones because F0 contours of a source neutral unit and a target tonal unit are usually distinct in their approximations rather than in their details [7]. On the contrary, transforming the deltas of event targets is easy, suitable for statistical training, and also suitable to transform the lexical tones because only the dynamics between sparse event targets are transformed. It has been found that low-dimensional vectors are not suitable for modeling with GMM because they might cause GMM over-fitting. Therefore, using the delta features of F0 to extend the dimensions of F0 vectors can also improve the accuracy wherein GMM parameters are estimated [9].

Assuming that there are M F0 targets for the aligned source and target speech, where $\{f0^{x_i}$ and $f0^{y_i^t}\}$ correspond to the static F0 targets for the source F0 contour x and target F0 contour y^t with tone t^{th}. Here, $i = 1, 2, \cdots, M$, and $t = 2, 3, \cdots, \Im$. The $\Im$ is the number of tones and $\Im = 6$ in Vietnamese. The two-dimensional (2-D) source and target F0 target vectors $f0^X$ and $f0^{Y^t}$ are represented as given in Eqs. (2) , (3), and (4).

$$f0^X = [{f0^{X_1}}^T, ..., {f0^{X_i}}^T, ..., {f0^{X_M}}^T], \tag{2}$$

$$f0^{Y^t} = [{f0^{Y_1^t}}^T, ..., {f0^{Y_i^t}}^T, ..., {f0^{Y_M^t}}^T] \tag{3}$$

where

$$f0^{X_i} = [f0^{x_i}, \Delta f0^{x_i}]^T, \qquad f0^{Y_i^t} = [f0^{y_i^t}, \Delta f0^{y_i^t}]^T \tag{4}$$

The joint source-target vector of F0 targets z is computed as in Eq. (5).

$$z = [{f0^X}^T, {f0^{Y^t}}^T]^T \tag{5}$$

The distribution of z is modeled by GMM λ, caculated as presented in Eq. (6).

$$\mathrm{p}(z|\lambda) = \sum_{q=1}^{Q} \alpha_q \mathrm{N}(z; \mu_q, \Sigma_q), \tag{6}$$

where Q is the number of Gaussian components, $\mathrm{N}(z; \mu_q, \Sigma_q)$ denotes the distribution with mean μ_q and covariance matrix Σ_q, and α_q is the prior probability of z generated by component q. The parameters $(\alpha_q, \mu_q, \Sigma_q)$ are estimated using EM algorithm and the transformed F0 contour $\hat{y}^t$ with target tone t^{th} is determined by maximizing the likelihood following Toda et al. [14].

2.3. Proposed NNS-based Alignment for Transforming F0 Contour of Lexical Tones

The parallel phoneme-based target alignment and training inside MRTD-GMM require large database covering all phonemes to train all phoneme-based GMMs. Therefore, it is difficult to accomplish with limited amounts of training data, especially when some tonal phonemes only occur in a few samples. The non-parallel method of alignment using nearest neighbor search (NNS) [9] can be used with limited amounts of training data. However, Wu et al.'s method of alignment [9] searches the closest neighbors in the whole data space, which may reduce the accuracy of alignment.

Wu et al.'s NNS-based alignment [9] is modified in this research, and is integrated with the modified MRTD-GMM for F0 transformation by clustering available phonetic units based on their articulatory similarities. The easies mode is using each phoneme for each cluster. Each cluster produces a phonetic-dependent subspace for searching in the modified NNS-based alignment. Thus, the source and target units for each aligned source-target pair are selected from corresponding subspaces to which the source/target units belong.

When the F0 contour of lexical tones is transformed, the spectral envelope parameters for all units in each same-phonation set are almost the same because they are related to similar vocal tract parameters produced by similar pronunciation behaviors. Thus, spectral envelope feature LSF is used for the alignment instead of directly using F0. Then, the F0 targets in the positions of the aligned LSF target pairs are used as the inputs of the phonetic-dependent GMM models for training.

Assume that the source LSF target vector computed from neutral units is $\{lsf_m\}$ and $m = 1, 2, \cdots, M$, where M is the number of event targets of these neutral units. When training for target tone t^{th}, and $t = 2, 3, \cdots, \Im$, the set of all tonal units with tone t^{th} is $\hat{ws}^t$. The $\hat{ss}^{t,m}$ is a tonal subspace of $\hat{ws}^t$ containing all units belonging to the phonetic unit cluster that lsf_m belongs to. The target vector for alignment is computed as given in Eq. (7).

$$\tilde{lsf}_m = \text{NNS}(lsf_m, \hat{ss}^{t,m}), \hat{ss}^{t,m} \in \hat{ws}_t. \tag{7}$$

The NNS function here returns the closest neighbors found in target space. The aligned LSF target pairs are therefore $\{lsf_m$, and $\tilde{lsf}_m\}$. The positions of the aligned LSF target pairs are needed for F0 transformation rather than their values. The positions of aligned pairs are $\{m, \text{p}(\tilde{lsf}_m)\}$ in this case, where $\text{p}(\tilde{lsf}_m)$ is the position of $\tilde{lsf}_m$.

Target-source alignment is also used. If it is assumed that the target LSF target vector computed from tonal units with tone t^{th}, is $\{\tilde{lsf}^t_n\}$, the source vector for alignment is computed as presented in Eq. (8).

$$lsf^t_n = \text{NNS}(\tilde{lsf}^t_n, \hat{ss}^{1,n}) \tag{8}$$

where $\hat{ss}^{1,n} \in \hat{ws}^1, n = 1, 2, \cdots, N$, N is the number of event targets of these tonal units, $\hat{ws}^1$ is the set of all neutral units, and $\hat{ss}^{1,n}$ is a neutral subspace of $\hat{ws}^1$ containing all neutral units belonging to the phonetic unit cluster that lsf^t_n belongs to. The position of aligned pairs is $\{\text{p}(lsf^t_n), n\}$ where $\text{p}(lsf^t_n)$ is the position of lsf^t_n.

Combining both source-target and target-source alignments, GMM transformation function F is trained from the aligned pairs of F0 vectors: $\{f0^X(m), f0^{Y^t}(\text{p}(\tilde{lsf}_m))\}$ and $\{f0^X(\text{p}(lsf^t_n)), f0^{Y^t}(n)\}$. Here, $f0^X$ and $f0^{Y^t}$ correspond to the F0 target vectors combined from static F0 targets and their deltas of source neutral units and target tonal units with tone t^{th} which are the same as those in Eqs. (2) and (3).

2.4. Implementation and evaluations

2.4.1. Data preparation

Vietnamese is a tonal monosyllabic language [10] that has six distinct tones. Each tone has a distinct F0 contour shape [4, 7]. More detail on Vietnamese language can be found in [10].

The small Vietnamese corpus DEMEN567, which is also called TTSCorpus [15], is used in this paper. DEMEN567 includes 567 utterances with a total time duration of less than one hour. The size of DEMEN567 in 16bit PCM format is approximately 70 MB and the sampling frequency is 11025 Hz.

The original DEMEN567 corpus is extracted into a syllable-based dataset of 1000 tonal syllables, covering all six Vietnamese tones, to train the tone transformations. A group of neutral syllables is used as the source while five other tonal syllable groups are used as targets for the F0 contour transformations. The numbers of syllables in each group differ between the tones. For evaluations, ten tonal syllables of mono-syllable words are evaluated for each tone. Thus, a total of 50 syllables are used for these evaluations.

2.4.2. Experimental setup

The frame sizes are set to 20 ms and the update intervals to 1 ms for two transformations for F0 contours of lexical tones, which are the proposed method and the GMM-based method of Wu [9]. The orders of LSF are 32 for the alignments. The numbers of GMM mixtures are 4. When using TD analysis/synthesis, each phoneme is represented by five F0 event targets. STRAIGHT version 4 [16] is used to synthesize the transformed tonal syllables in both methods.

2.4.3. Evaluation results

Subjective tests on intelligibility and naturalness were conducted with five subjects who were native Vietnamese speakers with normal hearing. The intelligibility scores were measured by using word error rates (WER) while mean opinion scores (MOS) was used to evaluate the naturalness of tone transformations.

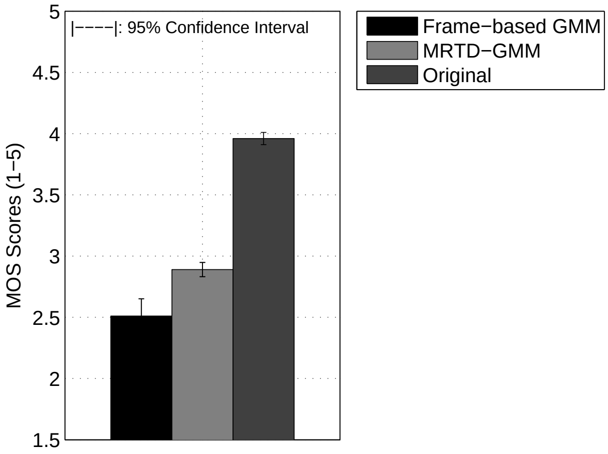

Figure 1: MOS scores for tone transformations: Wu's method and proposed method, calculated for all tones.

Tonal syllables of single-syllable words transformed by the state-of-the-art F0 transformation using frame-based GMM [9], and by the proposed F0 transformation, and original tonal syllables were used for evaluations. The results on naturalness are in Figure 1 and on intelligibility are in Figure 2, which indicate that the proposed MRTD-GMM F0 transformation significantly outperformed the state-of-the-art F0 transformations [9] in terms of both naturalness and intelligibility. The results on intelligibility also reveal that the proposed tone transformation is efficient for three Vietnamese tones rising, broken, and falling, while its performance is reduced with two tones curve and drop. The reason that why the proposed method not useful for some tones is the lack of a method of transforming the power. It has been known that although F0 contours can represent Vietnamese tones, the power contours also affect to some Vietnamese tones [4].

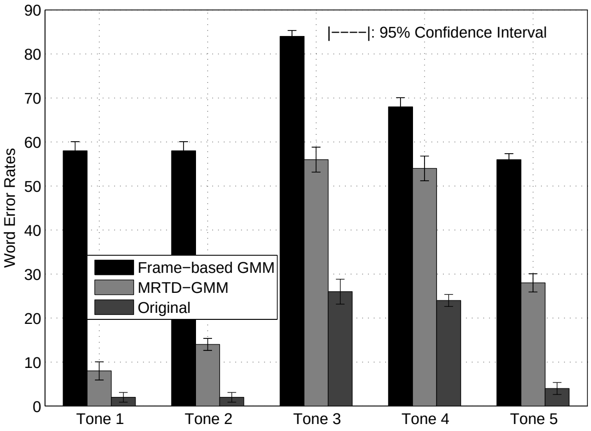

Figure 2: WER scores for tone transformations: Wu's method and proposed method, calculated for each separate Vietnamese tone, where Tone 1 is Rising, Tone 2 Broken, Tone 3 Curve, Tone 4 Drop, Tone 5 Falling.

3. PROPOSED METHOD FOR REDUCING MISMATCH-CONTEXT ERROR IN VIETNAMESE CSS UNDER LIMITED DATA CONDITIONS

3.1. Modeling co-articulated transition region between phonemes in CSS

This section presents the proposed model of the co-articulated transition region between two adjacent phonemes, using a framework for the proposed speech modification method presented later in subsection 3.3.

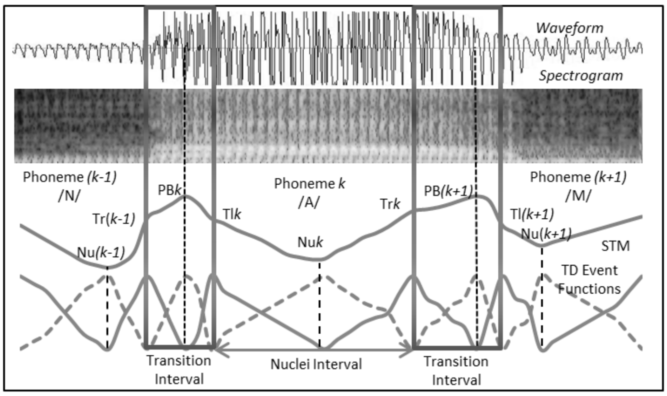

Figure 3: Modeling contextual effects using TD, STM and folded STM (FSTM): PBs are phoneme boundary points extracted from label data, N_us are nuclei points, T_rs are onsets and T_ls are offsets of joint transition regions.

3.1.1. General model

The basic supposition in the proposed model is the assumption that each phoneme can be divided into one nucleus interval and two co-articulated transition intervals at two sides. The proposed model attempts to determine the positions and the durations of these intervals. The existence of the stationary and quasi-stationary intervals inside vowels, semi-vowels and vowel-like consonants has been confirmed [17, 18]. General Locus theory [19] suggests that there is also a nucleus interval inside a non-vowel-like (or non-stationary) consonant, referred to as the narrow region around the ideal articulatory target of consonant. These nuclei intervals are referred to as pseudo-stationary intervals in this paper due to similarities between their behaviors and those of stationary and quasi-stationary intervals under the effect of co-articulation. All of the stationary, quasi-stationary and pseudo-stationary intervals of phonemes, called nuclei intervals for short, in the proposed model are supposed to be context-less-sensitive and can be preserved for concatenation within different contexts.

The spectral transition measure (STM) [20] and MRTD [12] are respectively used to determine the positions and the durations of the nuclei and transition intervals of phonemes, to interpolate speech parameters and to modify the joint transition intervals.

The context-sensitive co-articulated transition region between adjacent phonemes in proposed model is described by the TD event targets and the overlapping TD event functions restricted by the two event targets located at the onset and offset of the co-articulated transition region shown in Figure 3.

3.1.2. Estimating co-articulated transition region and TD event locations

Previous research on the co-articulation of speech has revealed that the transition movements caused by co-articulation can be observed by analyzing the transitions of formant frequencies [19]. STM [20], which is one representation of the first-order derivative of the spectral sequence, has also been used to detect the spectral transition rates of speech. STM can be used with any spectral parameter. The STM of LSF is used in this paper to estimate the nuclei and co-articulated transition intervals due to the close relations between LSF and formant frequency.

The STM at time t, $\mathrm{STM}(t)$ is defined in Eqs. (9) and (10), where time t refers to the location of the frame in the time domain.

$$\mathrm{STM}(t) = \frac{\partial LSF}{\partial t} = (\sum_{i=1}^{P} a_i^2)/P \tag{9}$$

where

$$a_i = (\sum_{n=-n0}^{n0} LSF_i(n).n)/(\sum_{n=-n0}^{n0} n^2) \tag{10}$$

Here $LSF_i(n)$ is LSF coefficient $i^{th}, (1 < i < P)$, in frame n^{th} inside a window whose center is time t, and $-n_0 < n < n_0$. The regression coefficient a_i, corresponds to the linear variation in the spectral envelope pattern in a unit time. Consequently, $\mathrm{STM}(t)$, which is the mean-square value of $a_i, i = 1..P$, corresponds to the variation in the spectral envelope smoothed by polynomial fitting.

The nucleus interval for interpolation with TD is represented by one central event target with the location determined based on criteria that maximize the stability of spectral transition [12], which is referred to as the location where STM is minimized. The three-step algorithm used to estimate this central event is:

Step 1. Initialize window size n0 = 1.

Step 2. Detect local minima of STM. If there is 1 minima, return, else move to step. 3

Step 3. - If there is more than 1 minima, increase n0 = n0 + 1, and return to step. 2

- If there are no minima, the location of the central event target is determined as the central location of the phoneme.

When moving from a stable nucleus interval to a dynamic transition interval (and in the inverse case), the rate at which speech is changing is maximum at the onset and offset of the dynamic interval. Therefore, the onset and offset of the co-articulated transition region are estimated based on criteria that maximize the dynamics of spectral transition, referred to as locations where FSTM, which is one representation of the second-order derivative of spectral sequence $\partial^2 LSF/\partial t^2$, is maximized.

$$\mathrm{FSTM}(t) = \frac{\partial^2 LSF}{\partial t^2} = (\sum_{m=-m0}^{m0} \mathrm{STM}(m).m)/\sum_{m=-m0}^{m0} m^2 \tag{11}$$

The three-step algorithm to estimate the two events located at the onset and offset of the joint co-articulated transition region is:

Step 1. Initialize window size m0 = 1.

Step 2. Detect local maxima of FSTM(t). If there is 1 maxima, return, else move to step. 3

Step. 3.

- If there is more than 1 maxima, increase m0 = m0 + 1, and return to step. 2

- If there are no maxima, the locations of outermost event targets are determined as the central locations of the left and right half-phonemes.

A total of three true event targets, including one central event target where STM is minimum and two event targets where FSTM is maximum, are used to interpolate each phoneme with TD. Two pseudo-targets at phoneme boundaries are also used to represent, select, and modify the co-articulated transition region, as explained in Sub-sections 3.1.3., 3.2., and 3.3.

3.1.3. Representing co-articulated transition regions with pseudo-targets

A simple and unique parameter representing a co-articulated transition region is necessary to easily modify the joint co-articulated transition regions. In this research, TD [13] is used to present co-articulated transition regions. It has been known that TD [13] decomposes a time sequence of speech parameters $y(n)$ into K dynamic event functions ϕ_k and K static event targets a_k and $k = 1..K$, as given in Eq. (12). Here, $\hat{y}(n)$ is the approximation of $y(n)$. As there are K event targets in the total of N frames and $K << N$, then TD is a sparse representation of speech. The event functions are interpolation functions representing the temporal transition movements between sparse event targets.

$$\hat{y}(n) = \sum_{k=1}^{K} a_k \phi_k(n), \quad 1 \leq n \leq N \tag{12}$$

Equation (12) can be written in matrix notation as Eq. (13), where P is the dimension of the speech parameter. The original TD [13] is proposed for the spectral linear prediction (LP) parameter with order P. However, TD can be used for both the spectral and prosodic parameters of speech [12].

$$\hat{Y}_{P\times N} = A_{P\times K}\Phi_{K\times N} \tag{13}$$

After event functions are estimated, event targets of both the original TD and MRTD are re-estimated as given in Eq. (14), where ^{T} is matrix transpose transformation.

$$A = Y\Phi^T(\Phi\Phi^T)^{-1} \tag{14}$$

Following Eqs. (12) and (13), the acoustical parameters for the joint transition region between two units L (left), which is restricted in locations from $\mathrm{n_L}(K-1)$ to $\mathrm{n_L}(K)$, and R (right), which is restricted in locations from $\mathrm{n_R}(1)$ to $\mathrm{n_R}(2)$, of one concatenation are represented in Eqs. (15) and (16), and are described in Figure 4. Here, $\mathrm{n_L}(i)$ and $\mathrm{n_R}(j)$ return the locations (frame indexes) of the targets i^{th} and j^{th} of the left and right units, respectively. Note that the indexes of y are the locations of frames, those of a are the locations of the sparse event targets, and the first-ordered indexes of ϕ are the locations of event targets and the second-ordered indexes of ϕ is the locations of frames.

The pseudo-event-targets are the two outermost events: event K^{th} of left unit L and event 1^{th} of right unit R. Therefore, the pseudo-target-vectors of left unit L and right unit R are $a_L(K)$ and $a_R(1)$.

$$\begin{aligned} y_L(\mathrm{n_L}(K-1):\mathrm{n_L}(K)) &= a_L(K-1,K) \\ &\times\phi_L(K-1:K,\mathrm{n_L}(K-1):\mathrm{n_L}(K)) \end{aligned} \tag{15}$$

$$y_R(\mathrm{n_R}(1):\mathrm{n_R}(2)) = a_R(1,2)\times\phi_R(1:2,\mathrm{n_R}(1):\mathrm{n_R}(2)) \tag{16}$$

Following the determination of event functions of MRTD [12], at the locations of event targets, the event function in the current interval approximates to one and other event functions approximate to zeroes. Therefore, the re-estimated target vectors, followed Eq. 14, are just slightly different from the frame-based vectors at the same locations. However, while modifying frame-based vectors just affects to these frames, when two pseudo-targets $a_L(K)$, $a_R(1)$ are modified, all frames in the transition parts of left unit L and right unit R will be gradually modified respectively, derived from Eqs. (15) and (16). Therefore, while pseudo-targets and frame-based vectors are equivalent when computing concatenation costs for CSS presented in Sub-section 3.2., only pseudo-targets can be used as unique paramaters for the modification task, presented in Sub-section 3.3..

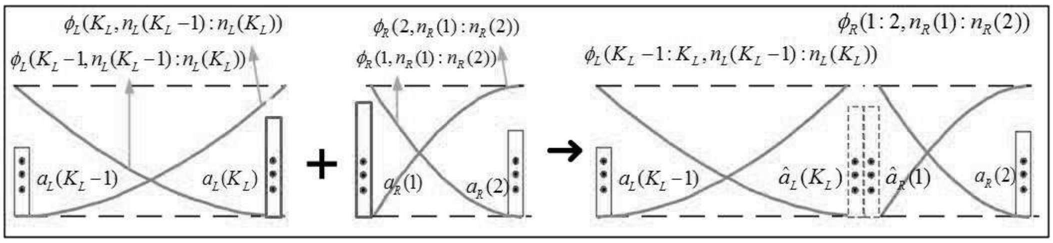
$\phi_L(K_L, n_L(K_L-1):n_L(K_L))$
$\phi_L(K_L-1, n_L(K_L-1):n_L(K_L))$
$a_L(K_L-1)$
$a_L(K_L)$
+
$\phi_R(2, n_R(1):n_R(2))$
$\phi_R(1, n_R(1):n_R(2))$
$a_R(1)$
$a_R(2)$
→
$\phi_R(1:2, n_R(1):n_R(2))$
$\phi_L(K_L-1:K_L, n_L(K_L-1):n_L(K_L))$
$a_L(K_L-1)$
$\hat{a}_L(K_L)$
$\hat{a}_R(1)$
$a_R(2)$

Figure 4: Modify joint transition regions of left unit L (left panel) and right unit R (central panel) for concatenation of unit $L+R$ (right panel): ϕ_L and ϕ_R represented by the curves are event functions of units L and R; a_L and a_R represented by the solid bars are event targets of units L and R; the two pseudo-targets $a_L(K_L)$ and $a_R(1)$ are averagely modified to $\hat{a}_L(K_L)$ and $\hat{a}_R(1)$ represented by the dot bars.

3.2. Proposed phoneme-based selection cost with pseudo-targets

In conventional unit selection CSS, the most important part is the methods of selecting units that uses both target cost and concatenation cost [1]. The concatenation cost is to find the best match units which join together most smoothly. The target cost is to find units in the database which best match their target predictions. It is clear that the actual target speech units are unknown in synthesis stage. Thus, the target predictions are usually determined as the centroids of the clusters of phonetic units existed in the database [21]. The conventional methods of unit clustering, such as CART method [21], need sufficient number of candidates for each unit and significant differences between candidates to ensure the accuracy of the clustering algorithm [21]. However, under limited data conditions, the number of candidates for each unit is usually small, resulting in the inaccuracy of the unit clustering and the target prediction. Therefore, determining an efficient target cost under limited data conditions is difficult and target cost is not considered in this work. Although the lack of target cost for unit selection may reduce quality of synthesized speech, especially in the timing and segmental duration, this research focuses on reducing the mismatch-context error for CSS and the use of only concatenation cost can be still sufficient to observe the efficiency of the proposed method.

Conventional concatenation cost includes both the cost of spectral features and prosody features such as F0 and power [1]. Hence, this research also uses concatenation costs that are computed by the distances between two pseudo-targets for LSF, F0, and PL of two joint phonemes (or two joint boundary phonemes of non-uniform units). Equation (17) describes the summed concatenation cost with a set of three acoustical parameters of LSF, F0, and PL.

$$\mathrm{C} = \omega_{LSF}\mathrm{c}_{LSF} + \omega_{F0}\mathrm{c}_{F0} + \omega_{PL}\mathrm{c}_{PL} \tag{17}$$

The component costs $\mathrm{c}_{LSF}, \mathrm{c}_{F0}$ and c_{PL} are computed in Eqs. (18), (19) and (20). The $\omega_{LSF}, \omega_{F0}$ and ω_{PL} are weighted factors that can be chosen from experiments.

$$\mathrm{c}_{LSF} = \frac{|a_{L\_LSF}(K) - a_{R\_LSF}(1)|}{\pi} \tag{18}$$

$$\mathrm{c}_{F0} = \frac{|log(a_{L\_F0}(K)) - log(a_{R\_F0}(1)|)}{max(log(F0))} \tag{19}$$

$$\mathrm{c}_{PL} = \frac{|a_{L\_PL}(K) - a_{R\_PL}(1)|}{max(PL)} \tag{20}$$

3.3. Proposed phoneme-based method of modifying co-articulated transition regions

The two units selected from the selection process need to be modified to smooth out the discontinuity and reduce the mismatch-context error.

Since modifying two pseudo-targets $a_L(K)$ and $a_R(1)$, all frames in the transition regions of the two phonemes L and R as shown in Sub-section 3.1.3. can be modified. The modification of the co-articulated transition region here is the average one of two pseudo-targets for each acoustical feature given in Eqs. (21), (22).

$$\Delta_{X_i} = \frac{a_{R\_X_i}(1) - a_{L\_X_i}(K)}{2} \tag{21}$$

$$a_{L\_X_i(K)} = a_{L\_X_i(K)} + \frac{\Delta_{X_i}}{2}, a_{R\_X_i(1)} = a_{R\_X_i(1)} - \frac{\Delta_{X_i}}{2} \tag{22}$$

Here, $i = 1..3$, X_1 is LSF, X_2 is F0 and X_3 is PL. Since two pseudo-targets $a_L(K)$, $a_R(1)$ are modified, all frames in the transition parts of left unit L and right unit R are gradually modified respectively. The proposed modification method can approximate the recovery of the smooth transition between adjacent phones that occurred in their original contexts.

Component thresholds $\delta_{F0}, \delta_{LSF}$ and δ_{PL} for the decision to modify the targets of F0, LSF, and PL are determined by experiments to avoid the modification of the joint concatenated phonemes or units that are already smooth.

3.4. Implementations and Evaluations

3.4.1. Data preparation

To evaluate the proposed CSS, a "limited data condition" for CSS is simulated. None of definitions of "limited data conditions" for CSS have been proposed. The speech corpus for state-of-the-art CSS is usually gigabytes in size. A dataset in this paper is considered to be under "limited data conditions" the threshold when the monophone coverage reaches approximately 100 %. This requirement means that although all phonemes exist, their frequencies of occurrence are limited. Therefore, the possibility of selecting matching-context units for concatenation is small, and the role of modification tasks is more important. A "limited data condition" with a dataset of 300 utterances extracted from DEMEN567, simulated by taking this requirement into account. The tonal Vietnamese phoneme coverage is nearly 100 %. Although some monophones are still missing, most of widely-used Vietnamese monophones appear in this dataset. The size of this dataset in PCM 16bit format is approximately 30MB and its duration is approximately 20 minutes. This dataset is used for concatenation in the proposed TD-based CSS and also used to concatenate the two non-uniform unit selection CSS for Vietnamese in comparison with the proposed CSS. The first CSS does not have spectral smoothing, which is referred to as CSS A, and the second has spectral smoothing, which is referred to as CSS B in this paper.

Semantically unpredictable sentences (SUS) have been used as a standard measure to evaluate the intelligibility of speech synthesis, but there are no designs on Vietnamese SUS lists at present. Therefore, 20 testing sentences were chosen to evaluate intelligibility with four restricted rules (rules 1–4) that prevented the subjects from easily predicting meanings, and two restricted rules (rules 5–6) that ensured the evaluation were reliable. The six rules were: (1) the Vietnamese words in the testing sentences were all low frequency, (2) only sentences composed of monosyllabic words were used to prevent subjects from predicting the meanings of compound words from their constituent parts, (3) repeating the words between testing sentences was avoided to prevent subjects from remembering words they had heard previously, (4) sentences with fewer semantic relations were selected to prevent subjects from predicting the meanings of sentences, (5) sentences covering all Vietnamese tones that minimized the repetition of tonal phonemes were selected, and (6) only short sentences were selected to avoid the difficulty for subjects to remember syllables that they had heard in each testing sentence.

These 20 testing sentences were chosen from a set of sentences that were not used to concatenate the proposed CSS and two conventional unit selections of CSS A and B.

Another testing dataset of 20 short sentences were used to evaluate naturalness, but not for concatenating three CSSs.

Since prosody trajectories are not controlled in the three CSSs, the authors focus on improving the quality of synthesized speech in terms of the local smoothness and the short-term naturalness in synthesized speech, which are the smoothness and naturalness that can be observed in short sentences with a few words. Therefore, both testing sentences for evaluating the intelligibility and the naturalness are short sentences with a few words.

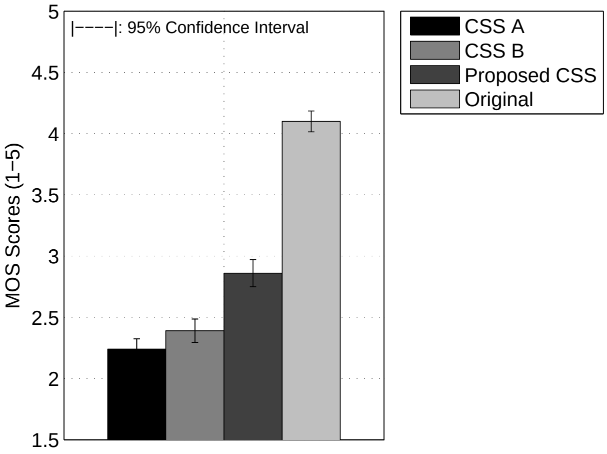

Figure 5: MOS scores (CSS A is without linear smoothing and CSS B is with linear smoothing)

3.4.2. Experimental setup

There are three CSSs used for comparisons, including two conventional ones: CSS A without smoothing methods and CSS B with a conventional smoothing method, and the proposed CSS.

As analysed in sub-section 3.2., it is difficult to determine an efficient target cost under limited data conditions and the proposed CSS uses only concatenation cost. Therefore, CSS A and CSS B also used only concatenation cost for selecting units to ensure fair comparisons. Same as in the proposed CSS, three acoustical parameters of LSF, F0, and PL are used. The concatenation cost for CSS A and CSS B approximate to that in the proposed CSS as described in Sub-section 3.1.3..

Although there are some sophisticated smoothing methods in CSS [22, 23], the linear spectral smoothing by Paliwal [24] is still one of the most popular and efficient methods for smoothing CSS [25]. Therefore, linear spectral smoothing with LSF interpolation is adopted to build CSS B in this research, in which two new frames at both sides of each concatenation are interpolated from the boundary frames (anchor frames) and were inserted as in [25].

The frame lengths are 20 ms and the frame steps 5 ms in there CSSs. The orders of LSF in three CSSs 24. The w_{LSF}, w_{F0} and w_{PL} correspond to 0.8, 0.05 and 0.15 in Eq. (18), these weighting coefficients are also adopted to the compute concatenation costs of CSS A and CSS B. The component

Table 1: Word Error Rates for CSSs(%)

	CSS A (without linear smoothing)	CSS B (with linear smoothing)	Proposed CSS	Original
Mean	10.33	10.83	5.5	0
95% confidence	0.88	0.80	0.36	0

thresholds for the decision on modification in the proposed TD-based CSS are $\delta_{LSF} = 0.01$, $\delta_{F0} = 0.1$ and $\delta_{PL} = 0.01$, $Max(F0) = 800$ and $Max(PL) = 0.1$ in Eqs. (19), (20).

STRAIGHT version 4 [16] is also used as a vocoder to generate the output waveform of three CSSs.

Since both the text-searching algorithms are same in for all three CSSs, the methods of unit selection in three CSSs are similar, and as STRAIGHT is used for all CSSs with the same manner, the differences on the performance of three CSSs can be mostly caused by the modification methods inside these CSSs.

3.4.3. Evaluation results

Subjective tests on intelligibility and naturalness were conducted with five subjects who were native Vietnamese speakers with normal hearing. The intelligibility scores were measured by using word error rates (WER) while mean opinion scores (MOS) was used to evaluate the naturalness of CSSs.

The results obtained from evaluating intelligibility are summarized in Table. 1 and they indicate that the WERs of the proposed TD-based CSS are minimal, and vastly superior to both CSS A and B.

The results from evaluating naturalness are presented in Fig. 5, where the proposed TD-based CSS is also superior to both CSS A and B.

The results from the intelligibility and naturalness evaluations confirmed that the proposed CSS efficiently reduces the mismatch-context errors in concatenations, and the proposed CSS runs efficiently under limited data conditions.

The results also show that CSS B just slightly outperforms CSS A. Thus, the linear spectral smoothing is not efficient under limited data conditions. On the contrary, the modification method inside the proposed CSS shows its efficiency in terms of both intelligibility and naturalness.

All sentences used to evaluate the three CSSs are very short sentences. Therefore, the performances of the three CSSs on segmental duration and timing is not clearly observed. In further experiments not presented in this paper, the authors find out that extending the sentence length for evaluation would increase the un-naturalness due to the mismatched segmental duration and timing. Therefore, the proposed CSS should be improved by supplementing target cost for unit selection and controlling prosodic trajectories, which are important for segmental duration and timing of synthesized speech. This is one of the authors' future works.

4. CONCLUSIONS

A method of reducing the number of stored units and a method of reducing mismatch-context errors in Vietnamese CSS are proposed. The experimental results with Vietnamese datasets revealed that the proposed lexical tone transformation is efficient for the CSS of tonal Vietnamese languages, while the proposed CSS convincingly outperforms two conventional unit selection CSSs with and without speech modifications in terms of naturalness and intelligibility. Consequently, the proposed speech modification and transformation methods presented in this paper appear to be capable of resolving the problems with Vietnamese CSS under limited data conditions. The naturalness of speech synthesized by the proposed methods under limited data conditions is significantly improved compared with using conventional methods.

REFERENCES

[1] Hunt A. Black and W. Alan, "Unit selection in a concatenative speech synthesis system using a large speech database," *Proc. ICASSP-96*, vol. 1, 1996, pp. 373–376.

[2] T. Shoham, D. Malah, and S. Shechtman, "Quality preserving compression of a concatenative text-to-speech acoustic database," *IEEE Trans. on Audio, Speech, and Lang. Proc.,* vol. 20, no. 3, pp. 1056–1068, 2012.

[3] J. Kominek and A. Black, "CMU ARCTIC databases for speech synthesis CMU Language Technologies Institute," *Tech Report CMU-LTI-03-177,* 2003.

[4] T. T. Do and T. Takara, "Vietnamese Text-To-Speech system with precise tone generation," *Acoustical Science and Technology,* vol. 25, no. 5, pp. 347-353, 2004.

[5] T. T. Vu, M. C. Luong and S. Nakamura, "An HMM-based Vietnamese speech synthesis system, Speech Database and Assessments," *Proc. COCOSDA-2009*, pp. 116–121.

[6] T. V. Do, D. D. Tran, and T. T. Nguyen, "Non-uniform unit selection in Vietnamese speech synthesis," *Proc. SoICT '11,* 2011, pp. 165-171 .

[7] VL. Nguyen and A. Edmondson, "Tones and voice quality in modern northern Vietnamese: Instrumental case studies," *Mon-Khmer Studies*, vol. 28, pp. 1-18, 1998.

[8] Hansjörg Mixdorf, Dung Tien Nguyen, Mai Chi Luong, Huy Hoang Ngo, Bang Kim Vu, Toward integrating the Fujisaki model into Vietnamese TTS, Proceeding of the International Conference on Spoken Language Processing, Korea, October 2004, pp. 177-180.

[9] Z. Wu, T. Kinnunen, E.S Chng, H. Li, "Text-Independent F0 Transformation with Non-Parallel Data for Voice Conversion," *Interspeech 2010,* pp. 1732-1735, 2010.

[10] H. Phe, *Chinh ta Tieng Viet (Vietnamese Grammar)*, Da Nang Publisher, pp. 9–15, 2003.

[11] P.B. Nguyen and M. Akagi, "Phoneme-based spectral voice conversion using temporal decomposition and Gaussian mixture mode," *Proc. ICCE-08,* 2008, pp. 224-229 .

[12] P. C. Nguyen, T. Ochi, and M. Akagi, "Modified restricted temporal decomposition and its application to low rate speech coding," *IEICE Trans. Inf. and Syst.*, vol. 86, no. 3, pp. 397-405, 2003.

[13] B. S. Atal, "Efficient coding of LPC parameters by temporal decomposition," *Proc. ICASSP-83*, 1983, pp. 81–84.

[14] T. Toda, A. Black, K. Tokuda, "Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," *IEEE Trans. Audio Speech Language Process,* vol. 15, iss. 8, pp. 2222-2235, 2007.

[15] L. C. Mai and D. N. Duc, "Design of Vietnamese speech corpus and current status," *Proc. ISCSLP-06*, 2006, pp. 748–758 .

[16] H. Kawahara, "STRAIGHT, Exploration of the other aspect of VOCODER: Perceptually isomorphic decomposition of speech sounds," *Acoust. Sci & Tech.*, vol. 27, no. 6, 349–353, 2006.

[17] Kent R.D and R. Charles, "The acoustic analysis of speech," *San Diego: Singular Publishing Group,* ISBN 1-879105-43-8, 1992.

[18] H. W. Strube, R. Wilhelms, "Synthesis of unrestricted German speech from interpolated log-area-ratio coded transitions," *Speech Communication*, pp. 93-102, 1982.

[19] P. Delattre, "Co-articulation and the Locus theory," *Studia Linguistica*, vol. 23, no. 1, pp. 1–26, 1969.

[20] S. Furui, "On the role of spectral transition for speech perception," *J. Acoust. Soc. Am.*, vo.. 80, no. 4, pp. 1016–1025, 1986.

[21] A. W. Black and P. Taylor, "Automatically Clustering Similar Units For Unit Selection In Speech Synthesis," *Eurospeech97*, 1997, pp. 601–604.

[22] J. Wouters, and M. W. Macon, "Control of spectral dynamics in concatenative speech synthesis," *IEEE Trans. on Audio, Speech, and Lang. Proc.,* vol. 9, no. 1, pp. 30–38, 2001.

[23] A. Kain, Q. Miao, and J. van Santen, "Spectral control in concatenative speech synthesis," *Proc. ISCA Workshop on Speech Synthesis,* 2007, pp. 11–16.

[24] K. K. Paliwal, "Interpolation properties of linear prediction parametric representations," *Proc. Eurospeech: ESCA,* 1995, pp. 1029–1032.

[25] T. Dutoit and H. Leich, "On the ability of various speech models to smooth segment discontinuities in the context of text-to-speech synthesis by concatenation," *Proc. EUSIPCO,* vol. 1, 1994, pp. 8–12 .

Received on October 07 - 2014
Revised on March 07 - 2015