Phonetic Segmentation using Multiple Speech Features

IOSIF MPORAS, TODOR GANCHEV[1], NIKOS FAKOTAKIS

Wire Communications Laboratory, Department of Electrical and Computer Engineering
University of Patras, Rion-Patras 26500, Greece

{imporas, fakotakis}@upatras.gr, [1]tganchev@ieee.org

Abstract. In this paper we propose a method for improving the performance of the segmentation of speech waveforms to phonetic segments. The proposed method is based on the well known Viterbi time-alignment algorithm and utilizes the phonetic boundary predictions from multiple speech parameterization techniques. Specifically, we utilize the best, with respect to boundary type, phone transition position prediction as initial point to start Viterbi time-alignment, for the prediction of the successor phonetic boundary. The method was evaluated on the TIMIT database, with the exploitation of several, well known in the area of speech processing, Fourier-based and wavelet-based speech parameterization algorithms. The results for the tolerance of 20 milliseconds indicated an improvement of the absolute segmentation accuracy by approximately 0.70%, when compared to the baseline speech segmentation scheme.

Keywords: speech segmentation, Viterbi algorithm, hidden Markov models

1. Introduction

The development of speech technology over the past years strongly relies on the availability of large speech corpora, containing information about the word and phonetic context as well as the corresponding time-alignment. Specifically, the most successfully used approach in speech synthesis, the unit-selection technique, is based on the concatenation of pre-recorded phonetic units. The quality of the synthetic speech directly depends on the precision of the available time-alignment of the corpus. While in speech synthesis the availability of high quality phonetic time-alignment is indispensable, in speech recognition phone boundaries are not needed. However, Hidden Markov Models (HMMs) are better initialized from bootstrap data, with available time-alignment (Malfrere et al., 2003).

Presently, the most precise way for time-aligning a speech waveform against the corresponding phonetic sequence is manually, by expert phoneticians. Despite its superiority, manual segmentation is a tedious and time-consuming task, which makes it a prohibitive choice for large speech corpora. Furthermore, the use of human annotators introduces subjectivity in the position of the phone transitions (van Hemert, 1991; Pellom and Hansen, 1998). These reasons enforced the development of techniques for the automatic segmentation of speech waveforms to speech units, with most popular the phonetic segments. Automatic segmentation techniques can roughly be divided into two major categories, implicit and explicit segmentation (van Hemert, 1991). In explicit case, or text-dependent, each speech waveform is time-aligned against a known phonetic sequence, using a set of phone models or reference patterns. In implicit or text-independent case, there is no prior knowledge of the corresponding phonetic sequence, so the number of detected phone transitions is not always equal to the real number of transitions.

Several approaches have been proposed for the task of speech segmentation. The most frequently used approach is based on HMM phone models (Ljolje and Riley, 1991; Brugnara et al., 1993; Ljolje et al., 1997; Pellom and Hansen, 1998; Mporas et al., 2008). This approach is inspired from the speech and phone recognition tasks, and became popular because of its well known structure. Specifically, in this method each speech waveform is initially decomposed into a sequence of feature vectors, using a speech parameterization technique. Afterwards, a set of HMM phone models (phone recognizer) is utilized to extract the corresponding phonetic sequence as well as the positions of the phonetic boundaries. When segmenting speech corpora the word transcription of each speech waveform is usually known and can be converted to a phonetic label sequence through a letter-to-sound converter. In this linguistically constrained case, where the present work falls in, the phone recognizer is confined at the detection of the

phonetic transition positions. Specifically, each phone label sequence is force-aligned against the corresponding feature vector sequence and the phone model set, through the Viterbi algorithm (Viterbi, 1967). The block diagram of the HMM-based segmentation for the linguistically constrained (explicit) case is shown in Fig. 1.

FIGURE 1

The training of the HMM phone models can be performed directly from the speech data that are going to be segmented (target data), with flat-initialization and parameter refinement via the Baum-Welch algorithm (Baum et al., 1970). Phone models can be trained from other speech corpora and further be used with/without adaptation on the target data. If manually segmented speech data are available (bootstrap data), they can be used for phone model training via Viterbi algorithm.

HMM-based segmentation has successfully been combined with post-processing techniques to refine the predicted phone boundaries (Sethy and Narayanan, 2002; Kim and Conkie, 2002; Toledano et al., 2003; Matousek et al., 2003; Wang et al., 2004; Adell et al., 2005; Lin and Jang, 2007; Lo and Wang, 2007).

Other methods have been proposed in the literature. Some of them include detection of variations/similarities in spectral (Svendsen and Soong, 1987; Dalsgaard et al., 1991; van Hemert, 1991; Grayden and Scordilis, 1994; Petek et al., 1996; Aversano et al., 2001) or prosodic (Adami and Hermansky, 2003) parameters of speech, template matching using dynamic programming and/or synthetic speech (Bajwa et al., 1996; Paulo and Oliveira, 2003; Malfrere et al., 2003) and discriminative learning segmentation (Keshet et al., 2007).

Methods for the fusion of the segmentation outputs from different approaches and/or systems have been proposed (Kominek and Black, 2004; Park and Kim, 2006; Park and Kim, 2007; Jarifi et al., 2008).

Various speech parameterizations have been utilized in the phonetic segmentation task, with the Mel Frequency Cepstral Coefficients (MFCC) (Davis and Mermelstein, 1980) among the most widely used (Brugnara et al., 1993; Pellom and Hansen, 1998), especially in the HMM-based approach. Other speech features such as Perceptual Linear Prediction (PLP) (Hermansky, 1990) utilized in (Hosom, 2002), Line Spectral Frequencies (LSF) (Itakura, 1975) used in (Paulo and Oliveira, 2003; Lin and Jang, 2007), Linear Predictive Coding (LPC) utilized in (van Hemert, 1991; Malfrere et al., 2003), short-time energy (Brugnara et al., 1993; Paulo and Oliveira, 2003; Malfrere et al., 2003), formants (Paulo and Oliveira, 2003) and wavelet-based (Ziolko et al., 2006) have also been studied.

The results reported in the above mentioned studies indicate that several speech features can be used equally successfully for the phonetic segmentation task. Moreover, it has been shown that specific speech features present significantly better ability to detect certain types of phone transitions comparing with other speech parameterization techniques (Pauws et al., 1996; Paulo and Oliveira, 2003). Here, by the term *phone transition type* we refer to the transition between the left context phonetic class of a boundary to the right context class, e.g. vowels, affricates, fricatives, nasals, glides, stops and silence. In addition, the segmentation precision on specific phone boundary types varies among different segmentation methods (Jarifi et al., 2008). Finally, specific boundary types present specialized misalignments between the real and detected boundaries (Matousek et al., 2003; Kawai and Toda, 2004; Lin et al., 2005).

Taking into account the aforementioned remarks and motivated by equivalent conclusions coming out of previous research in the area of phone recognition (Lee and Hon, 1989; Sarikaya and Hansen, 2000) we propose a method for the utilization of the best speech feature set per phone boundary type. Specifically, investigating the appropriateness of a number of dissimilar speech parameterizations on the speech segmentation task, we observed that for different phone class transitions, specific speech parameterization techniques have supremacy over other techniques. Based on this observation, we propose an efficient Viterbi-based segmentation scheme for the utilization of multiple speech features. This scheme exploits the best feature per phone transition type in order to time-align each boundary, and thus improve the overall speech segmentation accuracy. Since in the present work we do not investigate the phonetic recognition ability but we focus on the detection of phonetic boundaries we follow the explicit segmentation approach (time-alignment).

The remaining of this article is outlined as follows. In Section 2 we offer a detailed description of the proposed Viterbi segmentation with multiple speech features, which implements speech segmentation using multiple features and boundary specific corrections. In Section 3, we briefly outline a number of, successfully used in the past, speech features that were evaluated here. In Sections 4 and 5 we explain the experimental setup followed in this work and report the phonetic segmentation results. Finally, in Section 6 we conclude this paper.

2. Viterbi Segmentation with Multiple Speech Features

In terms of articulation, a speech waveform can roughly be divided into three areas, namely the target positions (phones) and the time spent moving forward and away from the target positions (co-articulation). As reported in the literature (Sarikaya and Hansen, 2000; Paulo and Oliveira, 2003),

different speech parameterization techniques are not able to capture equally successfully the same co-articulation phenomena, due to the different trade-offs they offer between the achievable frequency and time domain resolutions. These differences result to different misalignments between real and predicted phone transition positions in the task of speech segmentation, for the same segmentation method. To exploit these differences we introduce the use of multiple phone boundary predictions produced by different parameterization techniques.

Specifically, the proposed Viterbi Segmentation with Multiple Speech Features (VSMSF) method each time employs the best phonetic boundary prediction, among all utilized speech features, as initial point for the prediction of the next boundary. Thus, the errors introduced to Viterbi algorithm by inaccurate predecessor boundary predictions, and spread to successor boundary predictions, are reduced. This is mainly due to the use of less inaccurate position predictions, when compared to the predictions of each speech parameterization separately.

The general architecture of the proposed method for phonetic segmentation is illustrated in Fig. 2. As the figure shows, word labels are converted to phonetic label sequence L through a letter-to-sound converter, and different feature extraction engines *FE* extract observation vector sequences O^i from the speech waveform. For each boundary type a function f defines the best feature to predict its position. Subsequently, the Viterbi algorithm force-aligns the selected observation vector sequence against the phonetic labels *L*, starting from the predicted position t_s of the predecessor boundary.

FIGURE 2

We formalize our technique as follows. Let us consider *R* different parameterization techniques that decompose a speech waveform to *R* distinct observation sequences O^r of short-time feature vectors. Assuming uniform length of the time window and uniform frame rate for all speech features we can define

$$O^r = \left\{ o_1^r, o_2^r, ..., o_t^r, ..., o_T^r \right\},\ 1 \le r \le R,\ 1 \le t \le T \tag{1}$$

where *T* is the number of observations. During Viterbi time-alignment the observation sequence *O* is force-aligned against a sequence L of P phone labels

$$L = \left\{ l_1, l_2, ..., l_p, ... l_P \right\},\ 1 \le p \le P \tag{2}$$

and an HMM-based phone recognizer. Specifically, each phone is modeled by a left-to-right HMM of S states, with the first and last state non-emitting. A network consisting of the HMM phone models of the *L*

sequence is constructed by merging the end non-emitting state of l_k with the starting non-emitting state of l_{k+1}. Subsequently, the probability $\phi_j^r(t)$ of observing vectors o_1^r through o_T^r and finishing in HMM state j at time t is computed recursively as

$$\phi_j^r(t) = \max_i \left\{ \phi_i^r(t-1) a_{i,j}^r \right\} b_j^r(o_t^r), \ 1 \le j \le N \tag{3}$$

where N is total number of states in the HMM network, i.e. $N = P(S-1)+1$. The initial conditions of the recursive computation of $\phi_j^r(t)$ are

$$\phi_1^r(1) = 1, \ \phi_j^r(1) = a_{1,j}^r b_j^r(o_1^r) \tag{4}$$

and the final conditions are

$$\phi_N^r(T) = \max_i \left\{ \phi_i^r(T) a_{i,N}^r \right\} \tag{5}$$

Here $a_{1,j}^r$ represents the probability of transitioning from state i to state j, $b_j^r(o_1^r)$ represents the probability of observing o_1^r given the HMM state j, with respect to parameterization technique r. The maximum likelihood state sequence is extracted by tracking the state i that maximizes equation (3) for each t. The subsequent observation vectors that are aligned with the end and starting states respectively of two subsequent HMM phone models indicate the predicted phone boundaries B_r.

In VSMSF we define a mapping function f that associates each phone transition type $b(l_k, l_{k+1})$ with one parameterization technique, i.e.

$$f : b(l_k, l_{k+1}) \to r(l_k, l_{k+1}), \ 1 \le r(l_k, l_{k+1}) \le R \tag{6}$$

where $r(l_k, l_{k+1})$ indicates the parameterization technique r that the mapping function f orders, for the boundary type of the transition from phone l_k to phone l_{k+1}. Here, the mapping criterion was set as the parameterization technique that maximizes segmentation accuracy on a bootstrap training set for each specific boundary type. After estimating the mapping function we apply the Viterbi algorithm utilizing only the observation sequence of the output of f, $r(l_1, l_2)$, with respect to the first boundary type $b(l_1, l_2)$. The predicted position serves as initial time stamp to apply Viterbi algorithm, for the computation of the next boundary $b(l_2, l_3)$, utilizing feature $r(l_2, l_3)$. Similarly, for the prediction of boundary k the Viterbi algorithm is applied from the previously predicted phonetic boundary k–1 position up to the end of the speech waveform, o_T^r, utilizing exclusively feature $r(l_k, l_{k+1})$. This corresponds to recursive computation of $\phi_j^r(t)$ with initial conditions

$$\phi_{(k-1)(S-1)+1}^{r(l_k, l_{k+1})}\left(t_s(l_k, l_{k+1})\right) = 1 \tag{7}$$

$$\phi_j^{r(l_k,l_{k+1})}\left(t_s(l_k,l_{k+1})\right) = a_{(k-1)(S-1)+1,j}^{r(l_k,l_{k+1})} b_j^{r(l_k,l_{k+1})}(o_{(k-1)(S-1)+1}^{r(l_k,l_{k+1})}), \quad (k-1)(S-1)+1 \le j \le N \tag{8}$$

where t_s is the predicted position of the phonetic transition from phone l_k to phone l_{k+1}, i.e.

$$t_s(l_k,l_{k+1}) = \phi_{(k-1)(S-1)+1}^{r(l_{k-1},l_k)\,-1}\left(\max_i\left\{\phi_{(k-1)(S-1)+1}^{r(l_{k-1},l_k)}(t-1)a_{i,(k-1)(S-1)+1}^{r(l_{k-1},l_k)}\right\} b_{(k-1)(S-1)+1}^{r(l_{k-1},l_k)}(o_t^{r(l_{k-1},l_k)})\right), \quad 1 < t_s(l_k,l_{k+1}) < T \tag{9}$$

where ϕ^{-1} is the inverse function of ϕ. Once the final boundary t_P has been computed the segmentation process terminates.

In every step of the VSMSF method the length of the observation sequence to be segmented is restricted between the observation vector that corresponds to the previously detected phonetic boundary and the last observation vector. Thus, since for each boundary the most precise prediction is selected, with respect to a bootstrap training set, the resulting alignment between the HMM states and the successor observation vectors is more precise.

3. Speech Parameterization Techniques

The speech parameterization techniques considered in this work have been reported (Lee and Hon, 1989; Hermansky, 1990; Sarikaya and Hansen, 2000) to offer competitive performance on the speech recognition and phone recognition tasks. This motivated us to evaluate these well established speech features on the speech segmentation problem together with some recently proposed and thus less studied speech parameterizations. Specifically, here we consider two major categories of speech parameterization techniques:

(i) Discrete Fourier Transform (DFT)-based, among which are speech features like the MFCC, LFCC, Human Factor Cepstral Coefficients (HFCC-E) (Skowronski and Harris, 2004), PLP, and

(ii) Discrete Wavelet Packet Transform (DWPT)-based, among which are the Wavelet Packet Features (WPF) (Farooq and Datta, 2001), Subband Based Cepstral Coefficients (SBC) (Sarikaya and Hansen, 2000) and the Mixed Wavelet Packet Advanced Combinational Encoder (MWP-ACE) speech features (Nogueira et al., 2006).

Utilizing the standardized speech processing procedure (ETSI, 2007) we adapted the aforementioned speech parameterization techniques to a common experimental setup. In this manner, we resolved a number of set-up dependent disparities (e.g. sampling frequency, frequency bandwidth of speech signal, etc) that the original studies, proposing these speech parameterizations, considered.

Specifically, assuming speech signal sampled at 16 kHz, we adapted all speech parameterization techniques of interest to a common frequency bandwidth. Moreover, a uniform pre-processing, consisting

of pre-emphasis with factor a=0.97, frame blocking and windowing, of the speech signal was carried out. Afterwards the speech frames obtained so far were subject to DFT or DWPT, depending on the specific parameterization scheme. A feature-specific filter-bank was applied on the resulting coefficients, and the filter-bank outputs were logarithmically compressed. Finally, a set of cepstral coefficients were computed by applying Discrete Cosine Transform (DCT) on the logarithmically compressed filter-bank outputs. The only exception of this common processing scheme was the PLP cepstral coefficients estimation, which does not require speech pre-emphasis in the time domain but performs different post-processing of the filter-bank output. Specifically, the filter-bank output is subject to equal loudness pre-emphasis and intensity-to-loudness compression, followed by inverse DFT, autoregressive analysis and LPC to cepstral coefficient conversion. In all speech parameterization schemes we computed only the first thirteen cepstral coefficients.

A comprehensive description of the different speech parameterizations can be found in the aforementioned references. We followed the methodology as introduced by the corresponding authors but unified the frequency range of all filter-banks to the needs of our experimentations as follows:

3.1. DFT-based speech features

Mel-Frequency Cepstral Coefficients (*MFCC*): The MFCC implementation of Slaney (Slaney, 1998) utilized a filter-bank of forty equal-area filters, which covers the frequency range [133, 6855] Hz. This bandwidth was considered binding for the filter-banks implemented in the other speech parameterization schemes. The first thirteen filters in the filter-bank are with linearly spaced centre frequencies in the range [200, 1000] Hz, and the next twenty-seven have their centres logarithmically spaced in the range [1071, 6400] Hz, with log factor 1.0711703.

Linear Frequency Cepstral Coefficients (*LFCC*): The LFCC parameterization (Davis and Mermelstein, 1980) was adapted by implementing a filter-bank of forty equal-width equal-height filters, each one with pass-band of 164 Hz. This resulted in filter-bank that covers the frequency range [133, 6857] Hz, which is the closest feasible approximation of the desired frequency range.

Human Factor Cepstral Coefficients (*HFCC-E*): The HFCC filter-bank (Skowronski and Harris, 2004) that was designed with twenty-nine filters covering bandwidth [0, 6250] Hz was adapted here to the desired frequency range. Specifically, keeping the original spacing between the filters we discarded the two filters with lowest centre frequencies and added a new one at the high-frequency end of the filter-

bank. This resulted in filter-bank that covers the frequency range [125, 6844] Hz with twenty-eight filters. The filter-bank was designed for E-factor equal to one.

Perceptual Linear Prediction (*PLP*): The eighteen-filter Bark-spaced filter-bank utilized in the PLP (Hermansky, 1990) covering the frequency range [0, 5000] Hz was adapted by discarding the lowest-frequency filter and adding two new high-frequency filters, with Bark-spacing. This led to a filter-bank of nineteen filters that cover the frequency range [100, 6400] Hz, which is the closest feasible implementation.

3.2. DWPT-based speech features

Wavelet-Packet Features (*WPF*): The twenty-four frequency subbands (Farooq and Datta, 2001) approximating the Mel-scale in the frequency range [0, 8000] Hz were reduced to twenty-two subbands, frequency range [125, 7000] Hz, by eliminating the lowest and highest frequency subbands. The WPF utilize wavelet packet decomposition (WPD) based on the Daubechies wavelet of order 12.

Subband-Based Cepstral parameters (*SBC*): In the SBC (Sarikaya and Hansen, 2000) the authors used twenty-four Mel-spaced subbands to cover the frequency range [0, 4000] Hz. We adjusted this frequency division to the desired frequency range by discarding the two lowest subbands and adding at the end six new subbands of 500 Hz each. This resulted in Mel-scale frequency warping with twenty-eight subbands that cover the frequency range [125, 7000] Hz. The SBC utilize WPD based on the Daubechies wavelet of order 32.

Mixed Wavelet Packet Advanced Combinational Encoder (*MWP-ACE*): The MWP-ACE speech features (Nogueira et al., 2006) utilize twenty frequency subbands to cover the frequency range [0, 8000] Hz. In our implementation, we discarded the lowest and highest subbands, which resulted in a total of eighteen subbands that cover the frequency range [125, 7000] Hz. The MWP-ACE features utilize WPD based on the Symlets family with the Symlets wavelet of order 6 on the first level, Symlets 5 on the second, etc.

All changes in the filter-bank design described in Sections 3.1 and 3.2 meant to preserve the properties of speech parameterization, but were needed in order to unify the different sampling frequency and frequency bandwidth used in the different studies. We deem this did not affect the fundamental properties of the resultant speech descriptors, nor it harmed the diversity of signal representations that they offer.

4. Experimental Setup

The method presented in Section 2 aims at obtaining higher segmentation accuracy by combining the advantages that multiple speech features offer, when compared to any of the parameterization techniques separately. For every parameterization technique we utilize an HMM-based speech segmentation engine.

4.1 Speech Segmentation Engine

In the present work we followed a common experimental setup with (Brugnara et al., 1993). Specifically, we utilized a 6-state left-to-right HMM, without skipping transitions and start and end non-emitting states, to train one model for each phone. It has been shown (Toledano et al., 2003) that context-independent phone models present higher segmentation accuracy than context-dependent, since the latter tend to lose the alignment with the boundaries during training. Thus, here context-independent HMM models are trained. Every HMM state was modeled by 1 up to 6 linear combinations of continuous Gaussian densities, and diagonal covariance matrix. For the construction of the HMM phone models we employed the HTK (Young et al., 2006) toolkit.

Each speech waveform was frame blocked every 5 milliseconds, using a 16 millisecond window. Here we do not use the standard 20 millisecond window length, after the restriction of the wavelet-based features to be computed on number of samples equal to a power of two. In the case of DFT based speech parameterizations we employed Hamming window. Because of the compact support of wavelets, rectangular window was considered for the DWPT-based speech parameterizations. The HMM models were trained with feature vectors consisting of twenty-six parameters – the thirteen static speech features together with their first derivatives.

4.2 Evaluation Database

TIMIT (Garofolo, 1988) is the most widely used corpus for phone segmentation, and has been established for this task (Brugnara et al., 1993; Wightman and Talkin, 1997; Pellom and Hansen, 1998; Aversano et al., 2001; Keshet et al., 2007). In brief, it consists of microphone quality recordings of 630 American-English speakers, with sampling frequency 16 kHz and resolution 16 bits.

We followed the standard train/test subset division of the database, i.e. the train subset was utilized for the training of the HMM phone models, and the segmentation accuracy was measured on both the train and test subsets. The SA sentences were excluded from the evaluation. The established for American-English 48 phone set, proposed by (Lee and Hon, 1989), was adopted here. Adjacent

occurrences of the same phone were merged to one single occurrence as in (Brugnara et al., 1993; Pellom and Hansen, 1998).

Considering the phonetic clustering of the phone set of TIMIT, we followed the classes defined in the database documentation. Specifically, the phonetic classes listed in are: affricates (*AFF*), fricatives (*FRI*), nasals (*NAS*), semivowels and glides (*GLI*), stops (*STO*), vowels (*VOW*) and silence (*SIL*).

5. Experimental Results

In order to validate the practical significance of the proposed VSMSF approach, we compare it to the baseline system, which utilizes a single speech feature set. In the following Section 5.1, we firstly investigate the performance of the speech parameterization techniques described in Section 3, on the phone segmentation task. These results are utilized to estimate the performance of these speech features per phonetic transition type, and serve as the basis for implementing and evaluating the VSMSF method in Section 5.2.

In the present evaluation the segmentation accuracy is measured in terms of the most commonly used figure of merit, which is the percentage of predicted boundaries within a tolerance of t milliseconds from the manually annotated boundary labels (Brugnara et al., 1993; Pellom and Hansen, 1998; Keshet et al., 2007; Park and Kim, 2007; Jarifi et al., 2008).

5.1 Evaluation of the Speech Parameterization Techniques

As a first step we computed the segmentation accuracy for each parameterization technique separately. The results for 1, 2, 4 and 6 Gaussian mixtures and tolerances 5, 10, 15, 20, 25 and 30 milliseconds are reported in Table 1. The performance was measured both on the *Train* and *Test* subsets of TIMIT.

TABLE 1

As can be seen in Table 1, HMM states modeled by a single Gaussian distribution offer the highest segmentation accuracy for all speech features of interest. The splitting to more mixtures of Gaussians reduces the segmentation precision. One explanation of this tendency to produce better results with fewer Gaussians was given in (Toledano et al., 2003), where the authors explained it by the inherent variance of the spectrum in the vicinity of a phonetic transition, which could make a simpler model more adequate. The superiority of HMM modeled with fewer Gaussians is more intense for small tolerances, while for intermediate and large tolerances this tendency is weakened or even inverted, as in (Toledano et al., 2003) for tolerance equal to 50 milliseconds. Another explanation could be the amount of data in the training

subset of TIMIT, which might be insufficient to efficiently train (48*x*4=192) multiple mixture Gaussian distributions. In the rest of the present work's experimentations we utilize HMMs with states modeled by a single Gaussian distribution, since they achieve higher phonetic segmentation accuracy.

In Table 1 the maximum segmentation accuracy, for the best case of a single Gaussian, across the examined tolerances is indicated in bold. For each pair of parameterization techniques and for the case of a single Gaussian, paired *t*-test was performed to examine the statistical significance between the performances of the evaluated features. *T*-test has also been utilized for the task of phonetic segmentation in (Park and Kim, 2007). Segmentation performances which correspond to statistically identical, in terms of segmentation accuracy, pairs of features are indicated with similarly colored cells.

As the experimental results indicate the best segmentation performance was achieved by HFCC-E, followed by the PLP and MFCC features, across all examined tolerances. For small tolerances, i.e. $t<20$ milliseconds, wavelet-based features present significantly worse segmentation precision than the 3 best features, mentioned above. For large tolerances, i.e. $t>20$ milliseconds, the difference in segmentation accuracy between the evaluated parameterization techniques is obliterated.

These results indicate that, in average, the HFCC-E, PLP and MFCC parameterization techniques provide phonetic transition predictions more closely to the real boundaries than the wavelet-based ones. When the tolerance is large enough to include most of the boundary prediction positions the features do not present significant differences.

Fig. 3 shows the number of predicted boundaries with respect to their distance from the real ones, for each parameterization technique. As can be seen from the histogram the HFCC-E, MFCC and PLP features predict significantly more phone transitions around a short area (±10 milliseconds) of the real boundaries, comparing to the rest parameterization algorithms. When the tolerance increases over 20 milliseconds the number of predicted boundaries balances across the features.

FIGURE 3

The above analysis provides only an overall idea about the behavior of these speech parameterization techniques on the task of speech segmentation, since the results are averaged among all categories of phonetic transition types. In order to examine in depth the behavior of these speech features, we computed the segmentation accuracy separately for every phonetic class transition type. The results across different tolerances for every phonetic class transition of the train subset of TIMIT are presented in Table 2.

TABLE 2

As can be seen in Table 2 the different phonetic class transition types are captured better by different speech parameterization techniques, and not always by the best, in terms of average, HFCC-E. This fact can be explained by the different characteristics of these phonetic classes, e.g. continuant/non-continuant, periodic/non-periodic, short/long duration (Deller et al., 1993). At the neighborhood of a phone boundary these phone class specific characteristics are transited from one target articulation area to another. Thus, different speech parameterization techniques offer advantage in different phonetic transitions, due to their inherent differences.

Broadly speaking, there are two main differences between the DWPT- and DFT-based speech parameterization techniques, depending on: (i) the time-frequency resolution their transform can offer for a given time-window and (ii) the choice of basis function. For instance the DWPT offers a better localization of the signal components in time, while for the same time-window the DFT offers a better frequency resolution at high-frequencies. Here due to the use of filter-banks with similar design in all DWPT- and DFT-based speech features this effect is weakened. As mentioned before, the other source of difference between the DWPT- and DFT-based methods is the choice of the basis function. Specifically, the basis function is fixed to cosine in the DFT-based speech features and differs among the various DWPT-based techniques. The different basis functions might offer advantage for the transition between specific phonetic classes but this advantage might be lost when the performance is averaged for all phone transitions.

5.2 Experimental Results using Multiple Speech Features

The VSMSF algorithm described in Section 2, exploits the ability of the various speech parameterization methods to successfully capture specific phonetic transitions better than other speech parameterizations. The segmentation accuracy, with respect to each boundary type, of the evaluated parameterization techniques on the train subset of TIMIT was used to define the mapping function f described in equation 6. In Table 3 we report the segmentation accuracy achieved for the VSMSF method across the evaluated tolerances. For the purpose of direct comparison with the best performing feature, the segmentation accuracy of HFCC-E is duplicated here. The last row of Table 3 shows the absolute difference in segmentation accuracy between VSMSF and HFCC-E. The tabulated results indicate that the VSMSF method improves segmentation accuracy for all evaluated tolerances. This improvement is owed to the enforcement of the Viterbi algorithm to be applied on shorter intervals, which start from a more precise

initial point when predicting each boundary. Thus, given that each HMM models a minimum number of subsequent observation vectors, here 4 vectors, the alignment of each HMM starting from more precise predecessor boundary positions introduces less errors to successor boundary predictions.

TABLE 3

In order to examine the statistical significance between the VSMSF algorithm and the baseline alignment with HFCC-E, with respect to the segmentation accuracy for each tolerance, we performed the paired *t*-test. The *t*-values as well as the *p*-values are shown in Table 4. As can be seen in the table, the VSMSF results are statistically different from the ones of HFCC-E, for tolerance values smaller than 30 milliseconds. The two methods are statistically identical for tolerances larger than 30 milliseconds, since in this case most of the predicted boundaries fall inside this wide range (±30 milliseconds) allowed for misalignments in both methods.

TABLE 4

6. Conclusion

In this work we proposed a modification of the Viterbi algorithm that utilizes multiple parameterization techniques for phonetic segmentation. In the proposed algorithm we utilize the best, with respect to boundary type, phone transition position prediction as initial point to start Viterbi time-alignment, for the prediction of the successor phonetic boundary.

The experimental results show significant improvement in the segmentation accuracy using the proposed method, when compared to the best performing speech parameterization technique separately. Furthermore, for 20 milliseconds tolerance, which is considered as an acceptable limit for producing good quality synthetic speech (Matousek et al., 2003; Wang et al., 2004), the improvement of the segmentation accuracy reaches 0.7%, in terms of absolute performance.

7. References

Adami, A.G., Hermansky, H. (2003). Segmentation of speech for speaker and language recognition. In *Proceedings of the 8th European Conference on Speech Communication and Technology* (*EUROSPEECH 2003*), pp. 841–844.

Adell, J., Bonafonte, A., Gomez, J.A., Castro, M.J. (2005). Comparative study of automatic phone segmentation methods for TTS. In *Proceedings of the 2005 IEEE International Conference on Acoustics, Speech, and Signal Processing* (*ICASSP 2005*), pp. 309–312.

Aversano, G., Esposito, A., Esposito, A., Marinaro, M. (2001). A new text-independent method for phoneme segmentation. In *Proceedings of the 44th IEEE Midwest Symposium on Circuits and Systems*, vol. 2, pp. 516–519.

Bajwa R.S., Owens R.M., Kelliher, T.P. (1996). Simultaneous speech segmentation and phoneme recognition using dynamic programming. In *Proceedings of the 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing* (*ICASSP 1996*), vol. 6, pp. 3213–3216.

Baum, L.E., Petrie, T., Soules, G., Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Annals of Mathematical Statistics*. 41(1), 164–171.

Brugnara, F., Falavigna, D., Omologo, M. (1993). Automatic segmentation and labeling of speech based on hidden Markov models. *Speech Communication*. 12, 357–370.

Dalsgaard, P., Andersen, O., Barry, W. (1991). Multi-lingual label alignment using acoustic-phonetic features derived by neural-network technique. In *Proceedings of the 1991 IEEE International Conference on Acoustics, Speech, and Signal Processing* (*ICASSP 1991*), vol. 1, pp. 197–200.

Davis, S.B., Mermelstein, P. (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech and Signal Processing*. 28(4), 357–366.

Deller, J., Hansen, J., Proakis, J. (1993). *Discrete-time processing of speech signals*. Macmillan Publishing. New York.

ETSI (2007). *ETSI ES 202 050, V1.1.5* (*2007-1*). ETSI Standard: Speech Processing, Transmission and Quality Aspects (STQ); Distributed speech recognition; Extended advanced front-end feature extraction algorithm; Compression algorithms; Back-end speech reconstruction algorithm, January, 2007, Section 5.3, pp.21–24.

Farooq, O., Datta, S. (2001). Mel filter-like admissible wavelet packet structure for speech recognition. *IEEE Signal Processing Letters*. 8(7), 196–198.

Garofolo, J. (1988). *Getting started with the DARPA-TIMIT CD-ROM: An acoustic phonetic continuous speech database*. National Institute of Standards and Technology (NIST). Gaithersburgh, MD, USA.

Grayden, D.B., Scordilis, M.S. (1994). Phonemic segmentation of fluent speech. In *Proceedings of the 1994 IEEE International Conference on Acoustics, Speech, and Signal Processing* (*ICASSP 1994*), vol. 1, pp. 73–76.

Hermansky, H. (1990). Perceptual linear predictive (PLP) analysis for speech. *Journal of the Acoustical Society of America*. 87(4), 1738–1752.

Hosom J.-P. (2002). Automatic phoneme alignment based on acoustic-phonetic modeling. In *Proceedings of the 7th International Conference on Spoken Language Processing* (*ICSLP 2002*), pp. 357–360.

Itakura, F. (1975). Line spectrum representation of linear predictive coefficients. *Journal of the Acoustical Society of America*. 57, Suppl., no. 1, p. S35.

Jarifi, S., Pastor, D., Rosec, O. (2008). A fusion approach for automatic speech segmentation of large corpora with application to speech synthesis. *Speech Communication*. 50, 67–80.

Kawai, H., Toda, T. (2004). An evaluation of automatic phone segmentation for concatenative speech synthesis. In *Proceedings of the 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing* (*ICASSP 2004*), vol. 1, pp. 667–680.

Keshet, J., Shalev-Shwartz, S., Singer, Y., Chazan, D. (2007). A large margin algorithm for speech-to-phoneme and music-to-score alignment. *IEEE Transactions on Audio, Speech, and Language Processing*. 15(8), 2373–2382.

Kim, Y.-J., Conkie, A. (2002). Automatic segmentation combining an HMM-based approach and spectral boundary correction. In *Proceedings of the 7th International Conference on Spoken Language Processing* (*ICSLP 2002*), pp. 145–148.

Kominek, J., Black, A. (2004). A family-of-models approach to HMM-based segmentation for unit selection speech synthesis. In *Proceedings of the 8th International Conference on Spoken Language Processing* (*ICSLP 2004*), pp. 1385–1388.

Lee, K.-F., Hon, H.-W. (1989). Speaker-independent phone recognition using hidden Markov models. *IEEE Transactions on Acoustics, Speech and Signal Processing*. 37 (11), 1641–1648.

Lin, C.-Y., Chen, K.-T., Roger Jang, J.-S. (2005). A hybrid approach to automatic segmentation and labeling for Mandarin Chinese speech corpus. In *Proceedings of the 9th European Conference on Speech Communication and Technology* (*EUROSPEECH 2005*), pp. 1553–1556.

Lin, C.-Y., Jang, R.J.-S. (2007). Automatic phonetic segmentation by score predictive model for the corpora of mandarin singing voices. *IEEE Transactions on Audio, Speech, and Language Processing*. 15(7), 2151–2159.

Ljolje, A., Hirschberg J., van Santen, J.P.H. (1997). Automatic speech segmentation for concatenative inventory selection. In van Santen, J.P.H., Sproat, R.W., Olive, J.P., Hirschberg, J. (Eds.): *Progress in Speech Synthesis*, Springer, pp. 304–311.

Ljolje, A., Riley, M.D. (1991). Automatic segmentation and labeling of speech. In *Proceedings of the 1991 IEEE International Conference on Acoustics, Speech, and Signal Processing* (*ICASSP 1991*), vol. 1, pp. 473–476.

Lo, H.-Y., Wang, H.-M. (2007). Phonetic boundary refinement using support vector machine. In *Proceedings of the 2007 IEEE International Conference on Acoustics, Speech, and Signal Processing* (*ICASSP 2007*), vol. 4, pp. 933–936.

Malfrere, F., Deroo, O., Dutoit T., Ris, C. (2003). Phonetic alignment: speech synthesis-based vs. Viterbi-based. *Speech Communication*. 40, 503–515.

Matousek, J., Tihelka, D., Psutka, J. (2003). Automatic segmentation for Czech concatenative speech synthesis using statistical approach with boundary-specific correction. In *Proceedings of the 8th*

European Conference on Speech Communication and Technology (*EUROSPEECH 2003*), pp. 301–304.

Mporas, I., Ganchev, T., Fakotakis, N. (2008). A hybrid architecture for automatic segmentation of speech waveforms. In *Proceedings of the 2008 IEEE International Conference on Acoustics, Speech, and Signal Processing* (*ICASSP 2008*), pp. 4457–4460.

Nogueira, W., Giese, A., Edler, B., Büchner, A. (2006). Wavelet packet filter-bank for speech processing strategies in cochlear implants. *In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing* (*ICASSP 2006*), vol. 5, pp. 121–124.

Park, S.S., Kim, N.S. (2006). Automatic speech segmentation based on boundary-type candidate selection. *IEEE Signal Processing Letters*. 13(10), 640–643.

Park, S.S., Kim, N.S. (2007). On using multiple models for automatic speech segmentation. *IEEE Transactions on Audio, Speech, and Language Processing*. 15(8), 2202–2212.

Paulo, S., Oliveira, L.C. (2003). DTW-based phonetic alignment using multiple acoustic features. In *Proceedings of the 8th European Conference on Speech Communication and Technology* (*EUROSPEECH 2003*), pp. 309–312.

Pauws, S., Kamp, Y., Willems, L. (1996). A hierarchical method of automatic speech segmentation for synthesis applications. *Speech Communication*. 19, 207–220.

Pellom, B.L., Hansen, J.H.L. (1998). Automatic segmentation of speech recorded in unknown noisy channel characteristics. *Speech Communication*. 25, 97–116.

Petek, B., Andersen, O., Dalsgaard, P. (1996). On the robust automatic segmentation of spontaneous speech. In *Proceedings of the 4th International Conference on Spoken Language Processing* (*ICSLP 1996*), vol. 2, pp. 913–916.

Sarikaya, R., Hansen, J.H.L. (2000). High resolution speech feature parameterization for monophone-based stressed speech recognition. *IEEE Signal Processing Letters*. 7(7), 182–185.

Sethy, A., Narayanan, S. (2002). Refined speech segmentation for concatenative speech synthesis. In *Proceedings of the 7th International Conference on Spoken Language Processing* (*ICSLP 2002*), pp. 149–152.

Skowronski, M.D., Harris, J.G. (2004). Exploiting independent filter bandwidth of human factor cepstral coefficients in automatic speech recognition. *Journal of the Acoustical Society of America.* 116(3), 1774–1780.

Slaney, M. (1998). *Auditory toolbox, Version 2*. Technical Report #1998-010. Interval Research Corporation.

Svendsen, T., Soong, F.K. (1987). On the automatic segmentation of speech signals. In *Proceedings of the 1987 IEEE International Conference on Acoustics, Speech, and Signal Processing* (*ICASSP 1987*), pp. 77–80.

Toledano, D.T., Gomez, L.A.H., Grande, L.V. (2003). Automatic phonetic segmentation. *IEEE Transactions on Speech and Audio Processing*. 11(6), 617–625.

van Hemert, J.P. (1991). Automatic segmentation of speech. *IEEE Transactions on Signal Processing*. 39(4), 1008–1012.

Viterbi, A. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*. 13(2), 260–269.

Wang, L., Zhao, Y., Chu, M., Zhou, J., Cao, Z. (2004). Refining segmental boundaries for TTS database using fine contextual-dependent boundary models. In *Proceedings of the 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing* (*ICASSP 2004*), vol. 1, pp. 641–644.

Wightman C.W., Talkin, D.T. (1997). The aligner: text-to-speech alignment using Markov models. In van Santen, J.P.H., Sproat, R.W., Olive, J.P., Hirschberg, J. (Eds.): *Progress in Speech Synthesis*, Springer, pp. 313–323.

Young, S., Evermann, G., Gales, M., Hain, T., Kershaw, D., Liu, X., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V., Woodland, P. (2006). *The HTK Book* (*for HTK Version 3.4*). Cambridge University Engineering Department.

Ziolko, B., Manandhar, S., Wilson, R.C. (2006). Phoneme segmentation of speech. In *Proceedings of the 18th International Conference on Pattern Recognition* (*ICPR 2006*), pp. 282–285.

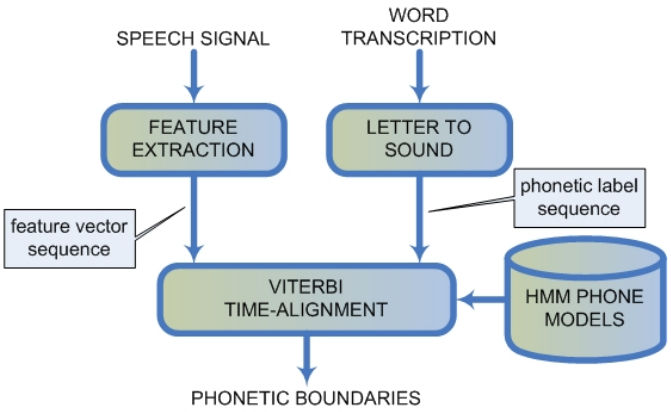

Fig. 1. Block diagram of the HMM-based phonetic segmentation method for the explicit case.

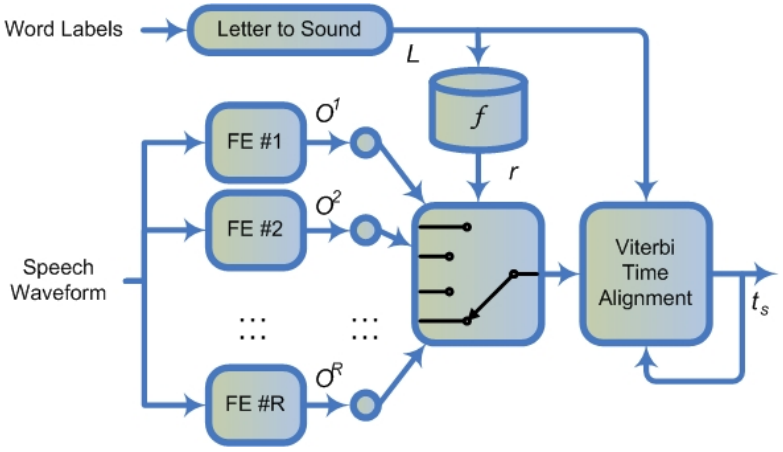

Fig. 2. Block diagram of the VSMSF phonetic segmentation method. FE denotes feature extraction.

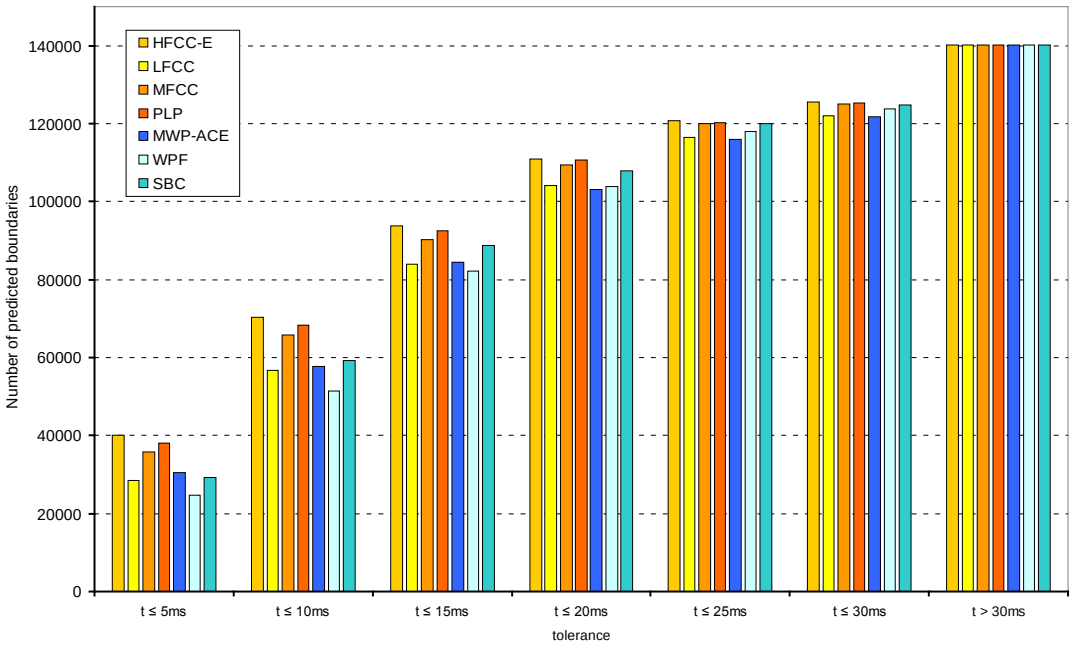

Fig. 3. Number of predicted boundaries with respect to their distance from the real ones for each of the evaluated parameterization techniques.

Table 1. Segmentation accuracy, in percentages, of the evaluated parameterization techniques for different tolerances and number of mixtures.

t ≤ 5 ms	Train Subset				Test Subset			
# mixtures	1	2	4	6	1	2	4	6
HFCC-E	***28.92***	21.63	20.47	19.72	***28.77***	21.42	19.84	19.14
LFCC	*20.88*	18.68	19.50	19.71	*20.64*	18.62	19.24	19.37
MFCC	*26.35*	20.75	20.43	19.75	*25.92*	20.38	20.05	19.23
PLP	*27.61*	22.21	21.31	20.80	*27.43*	21.82	20.90	20.23
MWP-ACE	*22.16*	21.26	21.25	20.99	*22.10*	21.27	21.19	20.62
WPF	*18.15*	17.38	16.94	16.69	*17.86*	16.81	16.58	16.50
SBC	*21.86*	19.99	19.37	18.59	*21.14*	19.60	18.66	18.04
t ≤ 10 ms	**Train Subset**				**Test Subset**			
# mixtures	***1***	**2**	**4**	**6**	***1***	**2**	**4**	**6**
HFCC-E	***50.21***	42.51	40.64	39.21	***50.07***	42.20	40.16	38.72
LFCC	*40.96*	37.01	38.40	39.24	*40.60*	36.82	38.37	39.02
MFCC	*47.13*	40.79	40.37	39.39	*47.04*	40.62	40.42	39.16
PLP	*48.71*	42.80	41.66	41.40	*48.74*	42.95	41.72	41.04
MWP-ACE	*41.26*	40.63	40.40	39.88	*41.38*	40.55	40.71	40.01
WPF	*37.47*	34.74	34.44	34.28	*36.97*	34.34	34.08	33.95
SBC	*42.65*	39.10	38.34	37.02	*42.44*	38.86	38.02	36.79
t ≤ 15 ms	**Train Subset**				**Test Subset**			
# mixtures	***1***	**2**	**4**	**6**	***1***	**2**	**4**	**6**
HFCC-E	***66.79***	62.17	60.37	59.38	***66.54***	62.15	60.33	59.57
LFCC	*59.84*	56.54	58.12	58.92	*59.86*	56.59	58.05	58.81
MFCC	*64.42*	60.24	59.88	59.11	*64.25*	60.29	59.72	59.03
PLP	*65.88*	62.65	61.47	61.27	*65.34*	62.32	61.06	61.12
MWP-ACE	*60.26*	59.89	60.25	59.81	*60.30*	59.75	59.98	59.54
WPF	*58.63*	55.28	55.31	55.14	*58.38*	55.30	55.06	54.99
SBC	*63.22*	59.79	59.13	57.77	*62.91*	59.92	59.12	57.91
t ≤ 20 ms	**Train Subset**				**Test Subset**			
# mixtures	***1***	**2**	**4**	**6**	***1***	**2**	**4**	**6**
HFCC-E	***79.11***	76.05	74.09	73.68	***78.54***	75.79	73.76	73.38
LFCC	*74.25*	71.92	73.04	73.30	*74.14*	71.60	72.71	73.04
MFCC	*78.03*	74.90	74.17	73.62	*77.54*	74.61	73.66	73.24
PLP	*78.86*	76.62	75.09	75.12	*77.99*	75.81	74.51	74.73
MWP-ACE	*73.54*	74.04	74.55	74.34	*73.33*	73.67	74.07	73.89
WPF	*74.02*	71.74	71.52	71.26	*73.42*	71.20	70.80	70.79
SBC	*76.93*	75.19	74.30	73.62	*76.16*	74.55	73.65	72.96
t ≤ 25 ms	**Train Subset**				**Test Subset**			
# mixtures	***1***	**2**	**4**	**6**	***1***	**2**	**4**	**6**
HFCC-E	***85.97***	84.56	81.67	81.21	***85.02***	83.60	81.15	80.89
LFCC	*82.98*	81.99	81.56	80.75	*82.44*	81.20	80.92	80.31
MFCC	*85.51*	83.61	81.62	81.37	*84.80*	82.82	81.16	81.05
PLP	*85.79*	84.45	82.23	82.39	*84.95*	83.48	81.66	81.94
MWP-ACE	*82.68*	83.54	83.91	83.92	*81.94*	82.82	83.12	83.07
WPF	*84.11*	83.29	83.27	82.77	*83.34*	82.30	82.28	81.79
SBC	*85.45*	84.36	83.38	83.00	*84.54*	83.48	82.80	82.28
t ≤ 30 ms	**Train Subset**				**Test Subset**			
# mixtures	***1***	**2**	**4**	**6**	***1***	**2**	**4**	**6**
HFCC-E	***89.40***	88.14	85.81	85.45	***88.74***	87.39	85.33	85.04
LFCC	*86.95*	86.19	85.65	84.95	*86.43*	85.56	85.19	84.62
MFCC	*89.18*	87.59	85.91	85.74	*88.63*	86.90	85.55	85.39
PLP	*89.28*	88.03	86.29	86.44	*88.62*	87.26	85.90	86.13
MWP-ACE	*86.83*	87.69	87.89	88.06	*86.01*	86.89	87.08	87.16
WPF	*88.20*	87.58	87.65	87.29	*87.42*	86.86	86.82	86.43
SBC	*89.00*	88.27	87.49	87.30	*88.19*	87.54	87.00	86.66

Table 2. Best parameterization technique for each pair of phonetic classes. Rows and columns indicate the left and right context of the phonetic boundary type, respectively.

t ≤ 5ms	AFF	FRI	NAS	GLI	SIL	STO	VOW
AFF	MWP-ACE	MWP-ACE	MFCC	PLP	PLP	MWP-ACE	HFCC-E
FRI	PLP	MWP-ACE	HFCC-E	HFCC-E	SBC	MWP-ACE	HFCC-E
NAS	HFCC-E	MFCC	HFCC-E	PLP	LFCC	HFCC-E	PLP
GLI	MWP-ACE	LFCC	MWP-ACE	LFCC	HFCC-E	HFCC-E	HFCC-E
SIL	HFCC-E	MFCC	PLP	PLP		HFCC-E	PLP
STO	MFCC	MFCC	HFCC-E	HFCC-E	MFCC	MWP-ACE	HFCC-E
VOW	HFCC-E	HFCC-E	SBC	SBC	WPF	MWP-ACE	PLP
t ≤ 10ms	**AFF**	**FRI**	**NAS**	**GLI**	**SIL**	**STO**	**VOW**
AFF	MWP-ACE	MWP-ACE	HFCC-E	HFCC-E	PLP	PLP	HFCC-E
FRI	HFCC-E	MWP-ACE	SBC	HFCC-E	SBC	HFCC-E	HFCC-E
NAS	HFCC-E	PLP	HFCC-E	HFCC-E	WPF	HFCC-E	PLP
GLI	HFCC-E	LFCC	HFCC-E	MWP-ACE	HFCC-E	PLP	HFCC-E
SIL	SBC	HFCC-E	HFCC-E	PLP		HFCC-E	PLP
STO	MFCC	HFCC-E	HFCC-E	HFCC-E	MFCC	LFCC	HFCC-E
VOW	HFCC-E	HFCC-E	SBC	HFCC-E	WPF	HFCC-E	MWP-ACE
t ≤ 15ms	**AFF**	**FRI**	**NAS**	**GLI**	**SIL**	**STO**	**VOW**
AFF	SBC	HFCC-E	SBC	MWP-ACE	WPF	MFCC	HFCC-E
FRI	SBC	MWP-ACE	WPF	HFCC-E	SBC	HFCC-E	HFCC-E
NAS	HFCC-E	PLP	HFCC-E	HFCC-E	WPF	HFCC-E	HFCC-E
GLI	SBC	LFCC	HFCC-E	MWP-ACE	SBC	HFCC-E	MFCC
SIL	HFCC-E	HFCC-E	MWP-ACE	PLP		PLP	PLP
STO	SBC	HFCC-E	MWP-ACE	HFCC-E	MFCC	HFCC-E	HFCC-E
VOW	HFCC-E	HFCC-E	SBC	HFCC-E	WPF	HFCC-E	SBC
t ≤ 20ms	**AFF**	**FRI**	**NAS**	**GLI**	**SIL**	**STO**	**VOW**
AFF	MWP-ACE	HFCC-E	SBC	LFCC	PLP	MFCC	HFCC-E
FRI	PLP	MWP-ACE	SBC	HFCC-E	SBC	PLP	HFCC-E
NAS	HFCC-E	MFCC	HFCC-E	HFCC-E	MWP-ACE	HFCC-E	HFCC-E
GLI	SBC	LFCC	HFCC-E	HFCC-E	HFCC-E	MFCC	MFCC
SIL	MFCC	HFCC-E	MWP-ACE	HFCC-E		PLP	PLP
STO	SBC	MFCC	SBC	HFCC-E	MFCC	MFCC	HFCC-E
VOW	HFCC-E	HFCC-E	MFCC	LFCC	WPF	PLP	SBC
t ≤ 25ms	**AFF**	**FRI**	**NAS**	**GLI**	**SIL**	**STO**	**VOW**
AFF	WPF	HFCC-E	SBC	LFCC	MFCC	MFCC	SBC
FRI	WPF	MWP-ACE	SBC	HFCC-E	SBC	PLP	HFCC-E
NAS	MFCC	MFCC	HFCC-E	HFCC-E	MWP-ACE	PLP	PLP
GLI	PLP	LFCC	PLP	SBC	SBC	WPF	MFCC
SIL	SBC	SBC	MWP-ACE	HFCC-E		PLP	PLP
STO	SBC	SBC	SBC	HFCC-E	MFCC	MWP-ACE	HFCC-E
VOW	HFCC-E	SBC	MFCC	LFCC	SBC	MFCC	PLP
t ≤ 30ms	**AFF**	**FRI**	**NAS**	**GLI**	**SIL**	**STO**	**VOW**
AFF	WPF	HFCC-E	SBC	WPF	WPF	MWP-ACE	HFCC-E
FRI	SBC	MWP-ACE	MWP-ACE	MFCC	SBC	MWP-ACE	MFCC
NAS	SBC	PLP	HFCC-E	HFCC-E	MWP-ACE	WPF	PLP
GLI	PLP	LFCC	SBC	HFCC-E	MFCC	MFCC	MFCC
SIL	SBC	SBC	MWP-ACE	MFCC		PLP	PLP
STO	SBC	PLP	HFCC-E	HFCC-E	MFCC	MWP-ACE	HFCC-E
VOW	HFCC-E	SBC	MFCC	HFCC-E	SBC	SBC	PLP

Table 3. Segmentation accuracy, in percentages, of the proposed VSMSF method and the best performing parameterization technique HFCC-E, for different tolerances.

tolerance (ms)	t ≤ 5	t ≤ 10	t ≤ 15	t ≤ 20	t ≤ 25	t ≤ 30
VSMSF	29.13	50.55	67.17	79.24	85.45	88.87
HFCC-E	28.77	50.07	66.54	78.54	85.02	88.74
Difference	0.36	0.48	0.63	0.70	0.43	0.13

Table 4. Paired *t*-test between the VSMSF method and the baseline alignment with HFCC-E (critical value 1.9743, for confidence level 95%).

tolerance	t -value	p-value
≤ 5 ms	2.4191	0.166e-1
≤ 10 ms	3.3954	0.856e-3
≤ 15 ms	4.3510	0.235e-4
≤ 20 ms	5.2362	0.488e-6
≤ 25 ms	4.4455	0.159e-4
≤ 30 ms	1.2794	0.2025

List of Figures

List of Tables