Acoustic-Phonetic Features for Refining the Explicit Speech Segmentation

Antonio Marcos Selmini, Fábio Violaro
Faculdade de Engenharia Elétrica e de Computação, UNICAMP, SP, Brasil
{selmini, fabio}@decom.fee.unicamp.br

Abstract
This paper describes the refinement of the automatic speech segmentation into phones obtained via Hidden Markov Models (HMM). This refinement is based on acoustic-phonetic features associated to different phone classes. The proposed system was evaluated using both a small speaker dependent Brazilian Portuguese speech database and a speaker independent speech database (TIMIT). The refinement was applied to the boundaries obtained by just running the Viterbi’s algorithm on the HMMs associated to the different utterances. Improvements of 30% and 13% were achieved in the percentage of segmentation errors below 20 ms for the speaker dependent and speaker independent databases respectively.

Index Terms: automatic speech segmentation, refining automatic speech segmentation, acoustic-phonetic features.

1. Introduction
With the development of systems that use speech for man-machine interface (text-to-speech synthesis, speech synchronized facial animation, etc.), reliable automatic speech segmentation is highly desired.

In all of the above applications, large segmented speech databases are required and a manual segmentation tends to be tedious, time consuming and may produce inconsistent results due to the subjectivity of the experts realizing the segmentation.

To overcome the manual segmentation problems and produce reliable speech segmentation, this paper describes the development and evaluation of a system for automatic explicit speech segmentation into phones. The system realizes the segmentation in two phases. In the first one, Viterbi’s algorithm is applied to the HMM of a given utterance in order to align the input speech parameters with the phone HMMs. In the second one, the segmentation produced in the first phase is refined based on the acoustic-phonetic features of the phone class at the left side of a given boundary.

2. Automatic Speech Segmentation Overview
Basically the automatic speech segmentation can be divided into two categories: implicit or explicit segmentation [1], [2], [3]. This classification depends on the type of information used to carry out the utterance segmentation.

In the implicit segmentation, all the information used is extracted only from the utterance. For the explicit segmentation, the phonetic transcription (explicit information) is additionally required. In the implicit segmentation it is very common the occurrence of boundary insertions or deletions. The explicit segmentation does not suffer insertion or deletion problems, but the boundaries between two phones may be quite distant from their correct positions.

Many different techniques to provide automatic speech segmentation have been proposed in the literature. The most efficient is based on the use of HMMs, due to its capacity to model the temporal variations of the speech signal.

3. Automatic Speech Segmentation Using an HMM-Based Approach
An automatic speech segmentation using a HMM-based approach consists of two phases: i) training the phone HMMs ii) segmentation using Viterbi’s alignment.

In the first phase, a set of basic sub-units (phones) is modeled by HMMs and the models are trained using a speech database. Normally each phone is modeled by a three-state HMM and a continuous density with diagonal covariance matrix is used to model the probability density function of symbol emission.

The second phase consists in applying Viterbi’s algorithm to find out an optimal state sequence from the symbols (speech parameters) associated to the input utterance. The input speech parameters are then aligned to the phone models.

Segmentation carried out by an HMM-based approach has several limitations such as the quantization error of one frame (normally around 10 ms) as the input speech parameters are calculated at each frame. Also, there can be misplaced boundaries. This limitation occurs because the Viterbi’s alignment used with HMM tries to find out the best HMM sequence and not the optimal boundaries between two phones.

4. Automatic Speech Segmentation Refinement
Refining or improving the automatic speech segmentation consists in carrying out an automatic processing in the segmented utterance in order to approximate the boundaries to the ones that would result from a handmade segmentation executed by an expert.

During the last years of research in automatic speech segmentation, several techniques have been developed. Among them we can mention: segmentation based on spectral transition measure, maximum likelihood segmentation [4], rule-based detection of phonetic boundaries [5], artificial neural networks as phonetic boundary detectors and fuzzy systems.

In the proposed system, the refinement process is carried out using specific acoustic-phonetic phone features. The goal is to simulate the same process used by experts to segment utterances, just hearing the speech file and looking at the speech waveform and spectrogram.
5. Rule-Based System for Automatic Speech Segmentation Refinement

The fundamentals of the system described in this paper were first proposed by Selmin and Violaro [6]. Basically the developed system is composed of three modules: i) training module of the HMMs associated to the different phones, ii) explicit segmentation module of the utterances using the Viterbi’s algorithm and iii) refinement module.

An additional module was added to the system. This new module is responsible for applying a systematic error correction (bias removal) to the segmented utterance. Figure 1 shows a block diagram of the proposed system.

![Block diagram of the proposed system.](image)

Figure 1: Block diagram of the proposed system.

5.1. Training module

In the training module, context-independent phones are modeled with a three-state HMM for speaker dependent speech database and five-state HMM for speaker independent speech database, with left-to-right topology. A multidimensional Gaussian mixture models the probability density function of symbol emission. The number of components in the mixture is dependent on the size of the available database used to train the HMMs. In our simulations, 4 Gaussians per state were employed for training the speaker dependent HMMs (Brazilian Portuguese speech database) and 8 Gaussians for the speaker independent HMMs (TIMIT database).

For the small Brazilian Portuguese speaker dependent database (SDD), just 38 different phones were considered (it was not made a distinction between reduced and plain phones). For the speaker independent TIMIT database (SID), 48 different phones were considered.

For each utterance of the training database, the DC level is first removed. After that a pre-emphasis is carried out (1 - 0.95 z⁻¹) and the speech is submitted to a 20 ms Hamming window. At each 10 ms (frame interval), a new set of parameters is calculated. Normally 12 mel-frequency cepstral coefficients, a normalized log-energy coefficient, and their first and second derivatives (delta and delta-delta parameters considering 1 frame on the left and 1 frame on the right) are calculated. All these parameters are then joined into a single acoustic vector of dimension 39. The HMM training is provided by the Baum-Welch’s algorithm.

5.2. Explicit segmentation module

During the segmentation process, the segmentation module uses the previously trained HMM system and the utterance phonetic transcription. Initially, on the basis of the phonetic transcription, the complete HMM of the input utterance to be segmented is generated. At each frame (10 ms) the acoustic parameters are calculated.

The sequence of parameter vectors corresponding to the whole utterance is then aligned to the corresponding HMM by using the Viterbi’s algorithm. This algorithm chooses the best path (among all possibilities) that maximizes the likelihood of the composite model emitting the input symbols. Given the best path, the number of frames associated to each phone model is then determined and it is possible to calculate the number of samples associated to each phone and, consequently, estimate the boundary between adjacent phones.

5.3. Bias removal module

Some authors [7, 8] suggest the use of post-processing techniques to remove a possible bias from the estimated boundaries that normally occurs between the automatic and manual segmentation. This bias depends on the phone class,

Some authors [7, 8] suggest the use of post-processing techniques to remove a possible bias from the estimated boundaries that normally occurs between the automatic and manual segmentation. This bias depends on the phone class, consequently, estimate the boundary between adjacent phones.

5.4. Refinement module

To refine the boundaries estimated by the segmentation module and shifted by the bias removal module, two pieces of information are necessary: i) the phonetic transcription of the utterances and ii) the acoustic parameters of the phone classes.

As the refinement is based on acoustic-phonetic features, firstly the 38 different phones used for the SDD and the 48 different phones used for the SID are grouped into 11 classes. These classes are: silence, voiced and unvoiced fricatives, voiced and unvoiced plosives, nasal consonants, nasal vowels, front vowels, median vowels, back vowels and liquid consonants.

For each class, a given set of parameters is calculated. The used parameters are the most representative of each class and are largely used for phone classification. For each parameter a threshold that represents the most significant value where the
transition occurs between two phones is calculated. These most significant values are determined using a manually speech segmented database in two steps. Firstly all the values for each parameter of each phone are calculated. Second, all the values are distributed into histograms and a detailed analysis is done in order to determine the best threshold values. The thresholds were calculated based on the speaker dependent Brazilian Portuguese database.

For calculating the acoustic parameters, the speech signal is first normalized to the peak value of each utterance in order to avoid problems with recording level.

The parameters used for each class are now listed [6]:

- **Silence** - Only the total energy of the analysis window is used (threshold: -35.8 dB). The boundary between the silence and other classes on the right is set up at the frame where the total energy becomes greater than the threshold.

- **Vowels (median, front, back, nasal)** - Four parameters are used (total energy of the analysis window, first (F1) and second (F2) formant values and energy profile). The transition between vowels and the other classes is determined by using the total energy of the analysis windows (transition is set where the energy is below -28 dB). Energy profile, F1 and F2 values are used to separate vowels in diphthongs. To determine the formants, a LPC analysis with order 14 was carried out using the Levinson-Durbin’s algorithm. F1 is used to separate median vowels from back and front vowels, and the boundary is set up at the frame where F1 is below 673 Hz. Energy profile and F2 value are used to separate front vowels from back vowels. The transition is determined at the frame where F2 is below 1845 Hz and the energy profile is below 2106 Hz. Energy profile represents the frequency band carrying a given percentage of the total energy and is calculated from the DFT of the windowed speech signal.

- **Fricatives (voiced and unvoiced)** - Two parameters are used: zero crossing rate (thresholds: 0.35 for voiced fricatives and 0.62 for unvoiced fricatives) and spectral energy ratio and energy profile respectively. The zero crossing rate represents the frequency under which 50% of the total energy of the windowed signal is concentrated. The transition from fricatives to other classes is determined at the frame where the parameters values are below the thresholds.

- **Plosives (voiced and unvoiced)** - Three parameters are employed: energy in the frequency bands [0-F3] and [F1-F2/2] and the first order derivative of F2, where F2 and F3 represent the second and third formant frequencies and f is the sampling frequency. As the derivative of F2 exhibits a peak at the transition from plosives to other classes, the peak position represents the boundary. Energy is combined with the derivative to avoid spurious peaks. The energy in the frequency band [0-F1] for voiced plosives is above of -5 dB and in the frequency band [F1-F2/2] is above of -2 dB. For unvoiced plosives the energy is above of 5 dB and 10 dB for the bands [0-F1] and [F1-F2/2] respectively.

- **Nasal consonants** - Two parameters are used: F1 value (threshold: 280 Hz) and the ratio between the spectral energy in the frequency bands [0-353 Hz] and [358-5373 Hz] (threshold: 0.87). When the F1 value is greater than 280 Hz and the spectral energy ratio is below 0.87, a transition has occurred from nasal consonant to another class.

- **Liquids** - Two parameters are employed: spectral energy band [0-2600 Hz] (threshold: above 6.5 dB) and its first order derivative. Transition from liquid to another class tends to exhibit a peak in the first derivative of the spectral energy.

The peak determines the transition and at this frame the spectral energy threshold must be below 6.5 dB.

Initially it is necessary to define the interval over which the boundaries under analysis can be moved. This interval, in this work, is called refinement interval and comprises the interval between the previous boundary on the left and the next one on the right. After that, based on the phonetic transcription, it is possible to determine the phone classes present on the left and on the right side and a specific transition rule is then applied.

The parameters specified by the rule are calculated over the refinement interval. These parameters are calculated using a 20 ms Hamming window and a 1 ms frame step. The center position of the window whose parameters cross the thresholds is defined as the new position for the boundary that is being refined.

### 6. Experiments

As it was described in the last section, the first stage in the proposed system is the training of the HMM context-independent phone models.

For training the phone models in the SDD, 1026 utterances were used and, for the segmentation tests, another 200 utterances were used. A male speaker recorded the database using a sampling frequency of 22.05 kHz. And, for TIMIT database (SID) 3696 utterances were used in the training phase and 1344 in the segmentation test.

The parameters used for training the HMMs are the same described in subsection 5.1.

### 7. Results and Discussion

In Table 1, the first results for the SDD and SID are shown. The table shows the segmentation errors (in percentage) below some pre-defined thresholds, obtained by just making a forced alignment using the Viterbi’s algorithm (HMM module).

<table>
<thead>
<tr>
<th>Threshold (ms)</th>
<th>Error (%) - SDD</th>
<th>Error (%) - SID</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;= 5</td>
<td>26.87</td>
<td>35.09</td>
</tr>
<tr>
<td>&lt;= 10</td>
<td>42.39</td>
<td>50.23</td>
</tr>
<tr>
<td>&lt;= 20</td>
<td>65.95</td>
<td>79.83</td>
</tr>
<tr>
<td>&lt;= 30</td>
<td>74.10</td>
<td>87.20</td>
</tr>
<tr>
<td>&lt;= 40</td>
<td>78.82</td>
<td>91.68</td>
</tr>
<tr>
<td>&lt;= 50</td>
<td>82.76</td>
<td>93.50</td>
</tr>
<tr>
<td>&lt;= 100</td>
<td>95.33</td>
<td>99.27</td>
</tr>
</tbody>
</table>

Analyzing Table 1, it can be observed that the results obtained for the speaker independent database are better than those obtained for the speaker dependent one. This can be explained by the greater number of utterances available for the HMM’s training and by the more detailed modeling of the phonetic sub-units considered (38 for the SDD and 48 for the SID).

The segmentation results produced by Viterbi’s alignment (segmentation module) are then applied to the bias removal module. The results are shown in Table 2.

Analyzing Tables 1 and 2 it can be observed that, after the bias removal, there is an improvement of about 15.31% for segmentation errors below 20 ms for the SDD and of 9.07% for the SID. The average deviation for the speaker dependent
HMM was estimated using the SDD and for the speaker independent HMM it was estimated using the SID.

Table 2. Results of the automatic segmentation after bias removal.

<table>
<thead>
<tr>
<th>Threshold (ms)</th>
<th>Error (%) - SDD</th>
<th>Error (%) - SID</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;= 5</td>
<td>55.77</td>
<td>41.70</td>
</tr>
<tr>
<td>&lt;= 10</td>
<td>58.82</td>
<td>55.50</td>
</tr>
<tr>
<td>&lt;= 20</td>
<td>81.26</td>
<td>88.90</td>
</tr>
<tr>
<td>&lt;= 30</td>
<td>86.09</td>
<td>92.45</td>
</tr>
<tr>
<td>&lt;= 40</td>
<td>90.69</td>
<td>94.00</td>
</tr>
<tr>
<td>&lt;= 50</td>
<td>92.38</td>
<td>96.17</td>
</tr>
<tr>
<td>&lt;= 100</td>
<td>97.80</td>
<td>99.88</td>
</tr>
</tbody>
</table>

The segmented utterances, after bias removal, are then submitted to the refinement module. After the refinement process, there is a significant decrease in the segmentation errors for all thresholds. These results are shown in Table 3.

Table 3. Results of the automatic segmentation after the refinement.

<table>
<thead>
<tr>
<th>Threshold (ms)</th>
<th>Error (%) - SDD</th>
<th>Error (%) - SID</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;= 5</td>
<td>61.00</td>
<td>56.75</td>
</tr>
<tr>
<td>&lt;= 10</td>
<td>73.89</td>
<td>69.12</td>
</tr>
<tr>
<td>&lt;= 20</td>
<td>95.55</td>
<td>92.95</td>
</tr>
<tr>
<td>&lt;= 30</td>
<td>97.23</td>
<td>94.76</td>
</tr>
<tr>
<td>&lt;= 40</td>
<td>98.70</td>
<td>96.75</td>
</tr>
<tr>
<td>&lt;= 50</td>
<td>98.90</td>
<td>97.98</td>
</tr>
<tr>
<td>&lt;= 100</td>
<td>100</td>
<td>100</td>
</tr>
</tbody>
</table>

Results observed in Table 3 show that the refinement based on the acoustic-phonetic features can highly improve the segmentation quality on both databases. The best results are obtained, as expected, for the SDD.

Another experiment was done to analyze the results of the refinement process without applying the bias removal. These results are shown in Table 4.

Table 4. Results of the automatic segmentation after the refinement without bias removal.

<table>
<thead>
<tr>
<th>Threshold (ms)</th>
<th>Error (%) - SDD</th>
<th>Error (%) - SID</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;= 5</td>
<td>60.43</td>
<td>57.10</td>
</tr>
<tr>
<td>&lt;= 10</td>
<td>73.00</td>
<td>68.69</td>
</tr>
<tr>
<td>&lt;= 20</td>
<td>95.99</td>
<td>92.97</td>
</tr>
<tr>
<td>&lt;= 30</td>
<td>96.98</td>
<td>94.50</td>
</tr>
<tr>
<td>&lt;= 40</td>
<td>98.23</td>
<td>96.32</td>
</tr>
<tr>
<td>&lt;= 50</td>
<td>98.50</td>
<td>97.90</td>
</tr>
<tr>
<td>&lt;= 100</td>
<td>100</td>
<td>100</td>
</tr>
</tbody>
</table>

From Tables 3 and 4 we can observe that the refinement results before and after the bias removal are very close. As the refinement module searches for the real transition in the refinement interval, a small correction before the refinement is not worthwhile. Besides the thresholds of the refinement module have been calculated just over the speaker dependent database, its application to the speaker independent database has also resulted in a good improvement.

The results shown in this paper are comparable with the best systems reported in the literature. For the SDD, Toledano et al [2] has achieved 96.01% of automatic boundary marks with errors smaller than 20 ms and we have achieved 95.99%. For the SID the best results are around 92%. Hosom [10] has achieved 92.57% and we have achieved 92.97%.

8. Conclusions

This paper relates some improvements in the system originally proposed in [6] for automatic speech segmentation (phone level) followed by a refinement process. The system provides an explicit segmentation based on Viterbi’s algorithm and, after that, a process of boundary refinement is carried out to reduce the segmentation errors. As it can be observed, the additional module for bias removal can produce an improvement but it is not necessary once the refinement proposed tries to find out the optimal boundary.

As future works we can mention: i) new parameters must be included, mainly for diphthongs composed by two back or front vowels, ii) new techniques for bias removal must be tested, iii) testing the system with context-dependent phones and iv) inclusion of sex-independent parameters.

9. References