# Locating Phone Boundaries from Acoustic Discontinuities using a Two-staged Approach

*Pairote Leelaphattarakij* *Proadpran Punyabukkana* *Atiwong Suchato*

Department of Computer Engineering, Faculty of Engineering,
Chulalongkorn University, Bangkok, Thailand
E-mail: u45pll@cp.eng.chula.ac.th, {Proadpran.P, Atiwong.S}@chula.ac.th

## Abstract

Ability to automatically align phonetic transcriptions with their associated acoustic signal is crucial to the development of computer-assisted speech training system where, it is frequently needed to locate phone boundaries from a known transcription in the signal. In this paper, an attempt to locate phone boundaries in the case where only the numbers of boundaries are known is described. Phone boundaries are hypothesized based solely on acoustic discontinuities without knowing exact transcriptions. This allows speech segmentation to be performed without large number of in-domain speech data for training. The boundary identification is done in two stages. Candidates for possible boundaries are selected in the first stage via local maxima of spectral changes. Dynamic programming is used to search for the best locations of phone boundaries from the candidate list. Allowing at most 20 ms. deviation from the actual boundaries, approximately 75% accuracy is achieved on a Thai continuous speech corpus.

**Index Terms**: speech recognition, speech segmentation

## 1. Introduction

Automatic speech segmentation, where acoustic boundaries between sound units in speech signals are located in unsupervised fashions, plays an important role in many area of speech research. It is needed in a wide range of applications, from the creation of prosodically labeled databases, to corpus preparation for concatenative speech synthesis as well as training statistical speech recognizers. Furthermore, the development of computer-assisted speech training systems, in which computer programs are used for detecting mistakes in the trainees' pronunciation of given transcripts based on associated acoustic speech signals, is also benefited from the advancements in automatic speech segmentation techniques.

Developing a speaker-independent computer-assisted speech training system so that it can automatically align the acoustic signal of an utterance according to a given word-level transcription with a sequence of sound units associated with the transcription involves two steps. First, we have to know the sequence of associated sound units. Then, we need to find the boundaries of those units. The sequence of sound units can be obtained by using a pronunciation dictionary or by applying a set of pronunciation rules to the word-level transcription of the utterance. The second step involves automatic segmentations which usually require a large amount of in-domain speech data for acoustic models training. While it is often affordable to obtain such a large amount of speech data in many tasks nowadays, in some task, such as automatic segmentation for a computer-assisted speech training system for hearing-impaired children, it is neither economical nor efficient to collect appropriate speech data.

Arguably, speech is the result of a sequential acoustically linking of phones. A phone boundary can serve as a pointer to approximate the timing where one sound ends and another begins. Acoustic properties of speech signals change, to some degree of rapidness, at phone boundaries. These property changes serve as cues to locating phone boundaries. Given only an exact number of sound units in a speech utterance, spectrogram readers, novice or elite, can identify boundaries of associated sound units quite correctly, if not perfectly, by visually observe acoustic discontinuities.

In this paper, we describe our attempt to locate phone boundaries utilizing measurements capturing acoustic discontinuities together with dynamic programming techniques for finding the most likely set of such discontinuities that are corresponding to a known number of phone boundaries, while avoiding the large training corpus requirement and the need for knowledge of exact transcriptions.

## 2. Related works

A large number of previous studies on automatic speech segmentation for computer-assisted speech training systems have focused on generative models of speech signal using Hidden Markov Models (HMM). Kipp, Wesenick, and Schiel [1] implemented an HMM system for automatic speech segmentation when only word-level transcriptions are available. They reported 84% agreement within 20 ms. tolerance. Brugnara et al. [2] developed an HMM forced-alignment system that used spectral variation features in addition to the standard cepstral-domain features for computing state occupation likelihoods. The alignment result was evaluated on the TIMIT database, and 75.3% agreement within 10 ms. tolerance was reported. Ronen, Neumeyer, and Franco [3] developed a computer-assisted speech training system using an HMM-based forced alignment for boundary detection and transcription aligning. They also performed segmental scoring at the phonetic level. 80.2% recognition rate was achieved.

Although the HMM-based approaches enable automatic phone segmentation, their performances highly rely on the speech corpora used in the parameter training phase. Large amount of speech data is required to estimate a large number of parameters. Some alternative methods focusing on detecting phonetic boundaries using acoustic features intended to capture acoustic discontinuities were experimented. Most of these features were based on spectral-domain representations and energy information in different frequency bands [5]. Wang, Lu and Zhang [6] proposed a speech segmentation approach

without performing speech recognition. Various features such as speech segment duration, Rate of Speech (ROS) and prosody together with phonetic sequence information were used to identify speech boundaries. An accuracy of 82.3% was reported.

In this research, inventing a novel technique in order to achieve the best accuracy is not our primary goal. What we are more interested in is to observe measurements for acoustic discontinuity which were shown to be successful in phone boundary detection and how to apply them to the development of a speaker-independent speech training system under the lack of training speech data constraint.

## 3. Two-staged approach

Given a known number of phone boundaries, $k$, we are facing with the problem of placing these $k$ boundaries at $k$ time instances so that the placement optimizes a certain criterion. The criterion used in this work is based on the assumptions that acoustic properties change abruptly across phone boundaries, in other words, acoustic properties immediately on both sides of a boundary must be significantly different, and acoustic properties of the signal located within the same pair of boundaries should remain stationary. Therefore, the best placement must produce the biggest total acoustic difference across boundaries while in-segment differences must be minimal. Exhaustive search for the best placement is not feasible computationally. Thus, dynamic programming is utilized.

Here, finding phone boundaries was performed in two stages, as shown in figure 1. The motivation behind this was to reduce the number of possible boundary placements, based on that some methods are more likely to generate false alarm candidates than miss detections, so that we could identify the placement that optimizes our criterion more rapidly and more accurately. In the first stage, the speech signal was windowed into 10ms-long non-overlapping frames, where features, to be described in 3.1, were calculated for each frame. A frame would be included in the candidate list for phone boundary if its feature values implied the occurrence of large acoustic differences locally. In the second phase, the best placement was hypothesized based on the candidate list obtained from the first stage.

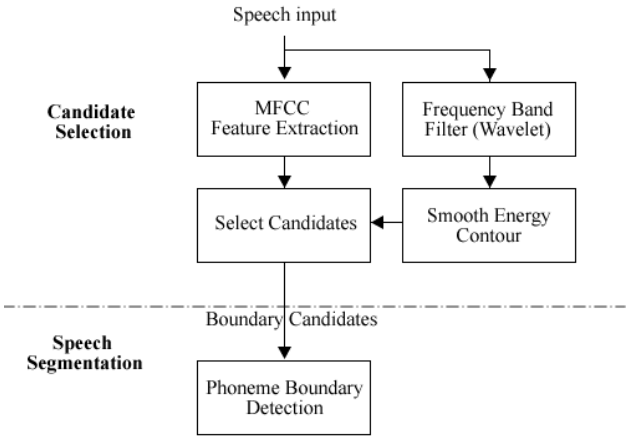


Figure 1. *An illustration of two-staged approach for phoneme boundary detection.*

### 3.1. Candidate selection

In order to select the phone boundary candidates, two types of acoustic features were used for capturing discontinuities. The first type utilized energy-based measurements while the other was a function based on Mel-frequency cepstral coefficients (MFCC). Here, we denote a candidate list as a sequence of start-time $\bar{c} = (c_1, \ldots, c_k)$, where $c_i$ is the start-time of the $i^{th}$ candidate, measured in frame number, and $k \leq N$, the total frame number.

#### *3.1.1. Energy-based measurements*

Speech signal was decomposed into five components in with different frequency ranges using wavelet decomposition. The frequency ranges were 0 to 500 Hz, 500 to 1,000 Hz, 1,000 to 2,000 Hz, 2,000 to 4,000 Hz and 4,000 to 8,000 Hz. Five energy-based features were measure for each frame. Each of the five features was the difference between the normalized squared energy of the current frame to the one of the frame immediately to the right in each of the five frequency ranges. Frames included in the candidate list were the frames exhibiting local maxima in any of the feature values. An energy threshold was set at mean energy in the utterance to prevent the inclusion of rapidly-changing low energy signal. Figure 2 shows an example of candidate selection based on energy measurements.

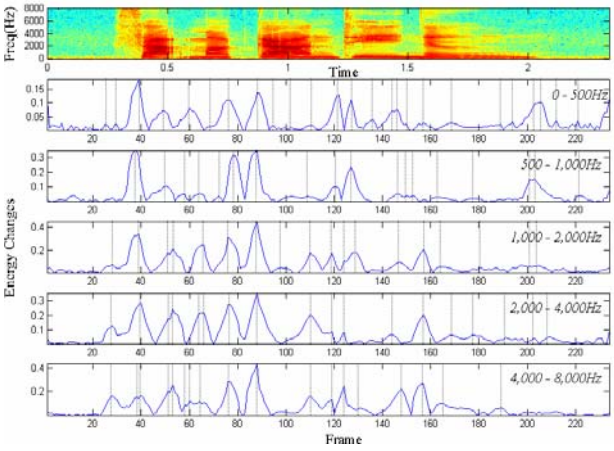


Figure 2. *An illustration of spectrogram for the word "/s/ /a/ /m/ /a/ /sp/ /k/ /aa/ /n^/ /th/ /ii/ /n/ /v/ /ng^/" and changes in energy from five frequency bands. Vertical lines indicate candidates.*

#### *3.1.2. MFCC-based functions*

Alternatively to the energy-based candidate selection, values based on MFCC difference were used. Thirteen-dimensional MFCC feature vector was measured for each frame. Euclidean distances between the MFCC vector of the current frame and the one of the frame immediately to the right is measured. Frames included in the candidate list were the frames exhibiting local maxima in the Euclidean distance measured. Figure 3 shows an example of candidate selection based on distance between MFCC vectors.

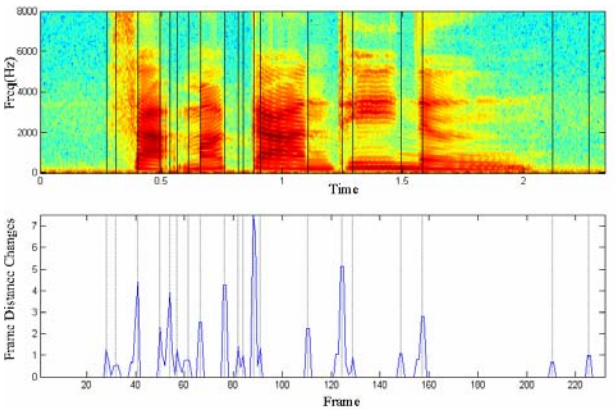


Figure 3. *An illustration of spectrogram for the word "/s/ /a/ /m/ /a/ /sp/ /k/ /aa/ /n^/ /th/ /ii/ /n/ /v/ /ng^/" and euclidean distance changes between consecutive frames. Vertical lines indicate candidates*.

### 3.2. Identifying phone boundaries

In this section we describe the problem of identifying phone boundaries and present the details of our algorithm which was used to find the true phoneme boundaries from the candidate list obtained from the candidate selection stage.

We were given a speech utterance along with a phonetic representation of the utterance. MFCC's were extracted from the speech signal. We denote the domain of the acoustic feature vectors by $X \subset R^D$ where $D$ is the number of MFCC coefficients. The acoustic feature representation of a speech signal is therefore a sequence of vectors $\bar{x} = (x_1, \ldots, x_T)$, where $x_t \in X$ for all $1 \le t \le T$ where $T$ is the length of the sequence $\bar{x}$ (measured as frame number). The list of true boundaries is a sequence of start-times $\bar{y} = (y_1, \ldots, y_k)$ where $y_i \in N$ is the start-time of phone $i$ in the acoustic signal and $k$ is the number of phones which varies from one utterance to another. The input is a pair $(\bar{x}, \bar{c})$ where $\bar{x}$ is an acoustic representation of the speech signal and $\bar{c}$ is a sequence of each candidate boundary start-times.

In order to predict the phoneme boundaries from $\bar{c}$, we defined our feature functions aiming at detecting phone boundaries. These feature functions were also based on the Euclidean distance, $d$, between frames of the signal and can be defined as

$$\phi_s(\bar{x}, \bar{c}, \bar{y}) = \sum_{i=1}^{|\bar{y}|} d(x_{y_{i-s}}, x_{y_{i+s}}), \; y_i \in \bar{c}, s \in \{1,2,3,4\} \quad (1)$$

The feature function in Eq. (1) returns a scalar which represents the confidence in acoustic discontinuity. By using four different feature functions (M = 4) in Eq. (2), we propose our alignment function of the form

$$f(\bar{x}, \bar{c}) = \underset{\bar{y}}{\operatorname{argmax}} \sum_{s=1}^{M} \phi_s(\bar{x}, \bar{c}, \bar{y}) \quad (2)$$

The function $f$ returns a suggestion for phone boundary sequence by maximizing a sum of scores returned from every feature functions. Note that calculating $f$ requires solving the optimization problem. Since possible boundary sequence $\bar{y}$ is no longer exponentially large due to the candidate selection stage, we can efficiently calculate $f$ in polynomial time using dynamic programming.

## 4. Speech corpus

To evaluate the effectiveness of the proposed approach we performed experiments on a corpus called LOTUS [10] which is a publicly available large vocabulary continuous speech corpus in Thai language. This corpus contains 801 phonetically distributed sentences with manually labeled phone boundaries in a set called PD. These sentences were uttered by 24 male and 24 female speakers. The total number of phonemes and vocabularies are 38,106 and 2,269, respectively. The sentences were recorded with a dynamic close-talk microphone in a clean environment. The speech was digitized and down-sampled to 16 kHz. with 16 bit resolution. Table 1 shows statistics about the number of boundaries per utterances for the PD set.

Table 1. *Statistics about the number of true boundaries per utterance for the speech data in the PD set*.

| | Mean | SD | Min | Max |
|---|---|---|---|---|
| # boundary /utterance | 52.7 | 30.4 | 12 | 190 |

## 5. Evaluation framework

Speech data in the entire PD set of the LOTUS corpus was used as the test set for the evaluation of the candidate selection performance and the overall performance of the phone boundary identification. The two methods used for candidate selection described earlier were evaluated first. Since the algorithm performed in the second stage was independent of the method chosen for the first stage, we only performed the phone boundary identification on the candidate list obtained from the better candidate selection method. After the second stage was performed on the candidate list, phone boundary detection accuracy percentage, *%Acc*, was determined from:

$$\%Acc = \frac{\#\text{correctly positioned boundaries}}{\text{Total \# of boundaries}} \times 100 \quad (3)$$

Performances were evaluated at four levels of tolerance value, $\tau \in \{10 \text{ ms}, 20 \text{ ms}, 30 \text{ ms}, 40 \text{ ms}\}$.

## 6. Results and discussion

Table 2 shows the results of the candidate selection stage using the two methods under the four tolerance levels. It can be seen that, at every tolerance level, the energy-based method yielded higher percentages of the actual boundaries get included in the candidate lists than the MFCC-based one. These percentages were the upper bound of what the boundary identification can achieve at last. If we look at the tolerance level of 20 ms., 93.0% of the actual boundaries were included in the candidate list for the energy-based case, while it was almost 90% for the MFCC-based case. In this aspect, the former method seemed to be a better method. However, table 3 in which ratios between the number of the actual boundaries included in the candidate list to the total number of candidates are shown, suggests that the MFCC-based method tends to select more reliable candidates.

At the tolerance level of 20 ms., almost half of the candidates are the actual boundaries in the MFCC-based case, while less than one-forth of the candidates are the actual boundaries. Table 4 reports some statistics of the number of candidates per utterance selected by each method. The average number of candidates per utterance for the energy-based method is more than twice the number for the MFCC-based method. Clearly, we can see that the energy-based method tended to over-generate candidates. From these results, we used the candidate lists from the MFCC-based method.

Table 2. *Percentage of the actual boundaries included in the candidate list at four levels of tolerance.*

| Tolerance (ms) | Energy-based | MFCC-based |
|---|---|---|
| 10 | 77.1 % | 78.3 % |
| 20 | 93.0 % | 86.9 % |
| 30 | 97.0 % | 89.9 % |
| 40 | 98.3 % | 91.9 % |

Table 3. *Ratio (%) between the number of included actual boundaries to the total number of candidates at four levels of tolerance.*

| Tolerance (ms) | Energy-based | MFCC-based |
|---|---|---|
| 10 | 19.2% | 44.8% |
| 20 | 23.1% | 49.8% |
| 30 | 24.1% | 51.4% |
| 40 | 24.4% | 52.6% |

Table 4. *Statistics about the number of candidates per utterance*.

| # candidates/utterance | Mean | SD | Min | Max |
|---|---|---|---|---|
| Energy-based | 221.9 | 121.2 | 58 | 873 |
| MFCC-based | 92.0 | 53.3 | 23 | 342 |

Table 5 shows the result of the second stage when the candidate lists were constructed from the MFCC-based method in the first stage. At the tolerance level of 10 ms., which is considered very strict, our two-stage approach can achieve the phone boundary identification accuracy percentage of 64.5%. When a weaker but still reasonable tolerance level at 20 ms. was used, the accuracy percentage achieved is 74.5%. The accuracy percentages increase to almost 80% if the deviations were allowed up to 30ms. and 40 ms. While such levels of tolerance might be acceptable for some low-precision tasks, the phone boundaries obtained using such high tolerance levels are usually not precise enough for most tasks, including the computer-assisted speech training task that we are interested in.

Table 5. *Percentage of correctly positioned boundaries.*

| Tolerance | 10 ms | 20 ms | 30 ms | 40 ms |
|---|---|---|---|---|
| *%Acc* | 64.5 % | 74.5 % | 77.7 % | 79.6 % |

Compared to other automatic speech segmentation experiments reviewed in section 2, the accuracy percentages obtained using our approach are slightly lower. With comparable tolerance level of 20 ms., we achieved around 75% accuracy while the accuracies obtained from those works were in the low 80's. However, it might not be very useful to make direct comparisons since our experiments were done without constructing any acoustic models trained from some training speech data.

## 7. Conclusion and future works

Although the two-staged approach for locating phone boundaries in this paper performed reasonably well compared to reported results in related works, given the fact that it operated under the lack of training data constraint, its performance was still far from human's ability to perform such a task from spectrograms. Therefore, there are still rooms for improvements. Additional feature functions containing different information reflecting acoustic discontinuity should be explored. If the constraint on the training process is relaxed, one could utilize trainable acoustic models of different phones and/or acoustic models of different boundary types in the forms of additional feature functions. Support Vector Machine could also be used to explore the contribution of each feature function in order to weigh their importance differently.

## 8. References


[1] Kipp, A., Wesenick, M. B., and Schiel, F., "Automatic Detection and Segmentation of Pronunciation Variants in German Speech Corpora," in *Proc. ICSLP'96*, 1996, pp.106-109

[2] Brugnara, F., Falavigna, D., and Omologo, M., "Automatic Segmentation and Labeling of Speech Based on Hidden Markov Models," in *Speech Communication*, 1993, pp. 357-370.

[3] Ronen, O., Neumeyer, L., and Franco, H., "Automatic Detection of Mispronunciation for Language Instruction," in *Proc. Eurospeech' 97*, 1997, pp. 645-648.

[4] Wutiwiwatchai, C., Cotsomrong, P., Subevisai, S., and Kanokphara, S., "Phonetically Distributed Continuous Speech Corpus for Thai Language," in *LREC*, 2002, pp. 869-872.

[5] Chaiareerat, J., and Santiprabhob, P., "Fuzzy-based Thai Syllable Segmentation for Connected Speech using Energy and Different Cepstral," in *Proc. InTech/VJFuzzy*, 2002, pp. 334-7.

[6] Wang, D., Lu, L., and Zhang H.-J., "Speech Segmentation without Speech Recognition," in *Proc. IEEE ICASSP'03*, Vol. I, 2003, pp. 468-471.

[7] Kominek, J., Bennet, C., and Black, A. W., "Evaluating and correcting phoneme segmentation for unit selection synthesis," in *Proc. Eurospeech'03*, 2003, pp. 313–316.

[8] Crammer, K., Dekel, O., Shalev-Shwartz, S., and Singer, Y., "Online passive aggressive algorithms," In *NIPS*, 2003.

[9] Rabiner, L., and Juang, B.H., *Fundamentals of Speech Recognition*. Prentice Hall, 1993.

[10] Kasuriyam, S., Sornlertlamvanich, V., Cotsomrong, P., Kanokphara, S., and Thatphithakkul, N., "Thai Speech Corpus for Thai Speech Recognition," in *Proc. COCOSDA'03*, 2003, pp 54-61.

[11] Santen, V., J.P.H., and Sproat, R.W., "High-Accuracy Automatic Segmentation," in *Proc. of Eurospeech '99*, Vol. 6, 1999, pp. 2809-2812.