Performance Limits for Envelope based Automatic Syllable Segmentation

Rudi Villing[ϕ], Tomas Ward[ϕ†] and Joseph Timoney*

[ϕ]Department of Electronic Engineering, National University of Ireland, Maynooth, IRELAND

**Department of Computer Science, National University of Ireland, Maynooth, IRELAND*

*E-mail:[ϕ]rudi.villing@eeng.nuim.ie, [†]tomas.ward@eeng.nuim.ie, *jtimoney@cs.may.ie*

In this paper the upper performance limits of automatic syllable segmentation algorithms using single or multiple frequency band envelopes as their primary segmentation feature are explored. Each algorithm is tested against the TIMIT corpus of continuous read speech. The results show that candidate matching rates as high as 99% can be achieved by segmentation based on a simple envelope, but only at the expense of as many as 13 non-matching candidates per syllable. We conclude that a low total error rate requires an algorithm which can reject many candidates or which uses features other than those based on envelope alone to generate fewer, more accurate candidates.

Keywords – Syllable, syllabification, syllable segmentation, speech perception

I INTRODUCTION

A syllable is one of the most fundamental units of speech and an important structural unit in language production and perception. Syllabic processing has been used to improve the accuracy of speech recognition [1] and is proposed as a tool to aid labelling of large recorded speech corpora for concatenative synthesis [2]. While there are no phonetic definitions for the syllable which are universally agreed upon, it is possible to identify some of its most salient features.

All syllables have a nucleus, consisting of a sonorant, usually a vowel. This may optionally be preceded by an onset, consisting of one or more consonants: a consonant cluster. The nucleus may also be succeeded by a consonant cluster, labelled the coda. Based on their typical phonemic constituents (consonant clusters and sonorants that function like vowels) and the presence or absence of onset and coda, it is common to represent the various syllable possibilities as CV, CVC, VC and V. Variants which make the number of onset or coda consonants explicit are also used. For example the word "scratched" may be represented as CCCVCC.

Listeners do not usually find it difficult to syllabify a phonetic string, segmenting it into syllables, and will generally agree on the number of syllables. However some inconsistency in the placement of syllable boundaries does arise [3]. Specifically, in the sequence —VCV—, individual listeners may choose to consider the intervocalic consonant to be the coda of the first syllable or the onset of the second. Some phonological descriptions specifically allow an intervocalic consonant to be affiliated with both the previous and following syllable, a concept referred to as ambisyllabicity [4].

A single boundary cannot be simultaneously located both before and after some intervocalic consonant. If it is assumed that there is a single boundary between syllables and that syllables do not overlap in time, then ambisyllabicity may imply that the location of the boundary is ambiguous or that no categorical boundary exists. An alternative interpretation is that there isn't a single boundary between syllables; instead syllables overlap in time such that the end of one syllable may be located after the beginning of the next. In this interpretation, when listeners syllabify speech, locating syllable onsets and offsets constitutes two distinct operations. The onset hypothesis [5] then assumes that when there is a conflict between onset and offset preferences, the onset decision dominates. Throughout the remainder

of this paper the term syllable boundary will be taken to mean a syllable onset.

Automatic blind syllable segmentation attempts to identify syllabic segment boundaries based on acoustic features of a speech waveform. Algorithms can be broadly classified as either rule based (data independent) or trained (data dependent) and may use a variety of features to identify syllable boundaries. In this paper we will evaluate a small number of algorithms which use the waveform envelope or the envelope in multiple frequency bands as their primary segmentation feature. These algorithms have the benefit of being straightforward to implement and integrate into larger systems.

A syllable segmentation algorithm generally consists of two main processing stages: candidate boundary generation and final boundary selection. In general much of the algorithm complexity can be attributed to the final boundary selection stage and this stage is also usually most sensitive to the training data used and the tuning of algorithm parameters. In this paper, therefore, we examine the performance of only the candidate generation stage of each algorithm. This simplification makes it easier to compare algorithms and gain insight into the factors which can affect the upper limit of segmentation performance.

II TEST CORPUS

The TIMIT corpus of read speech was designed to provide acoustic and phonetic speech data for the development and evaluation of automatic speech recognition systems [6]. It consists of 6300 utterances: 10 spoken by each of 630 speakers representing 8 major dialects of American English. The corpus includes time-aligned orthographic, phonetic and word transcriptions and a 16-bit, 16kHz speech waveform file for each utterance. It does not, however, include syllabic transcriptions.

Syllabic transcriptions were generated for each TIMIT utterance using tsylb2 [7], a programme for the automatic syllabification of phonetic transcriptions implementing the algorithm described in [4]. TIMIT phonetic transcriptions are not directly compatible with tsylb2 so the following rules are used to prepare a converted phonetic transcription that is compatible with tsylb2:

1. TIMIT closure labels are deleted if followed by a matching plosive or affricate phoneme (e.g. /dcl jh/ becomes /jh/), or rewritten as the corresponding phoneme otherwise (e.g. /gcl l/ becomes /g l/).
2. The sequence /hv w/ is rewritten as /wh/.
3. Pauses are converted to tsylb2 word boundaries.
4. The TIMIT phonemes /ax-h/, /hv/, /eng/, /ng/ and stress marks are converted to their tsylb2 equivalents.
5. Time alignment data is removed.

The tsylb2 software is then used to create a syllabic transcription based on the input phoneme transcription and a specified rate of speech. Different rates of speech cause tsylb2 to produce different syllabifications of the same input phoneme sequence.

As TIMIT is a corpus of read speech just two of the five rates supported were deemed suitable for syllabification of the corpus: rate 2 denotes “formal, monitored, self-conscious speech” while rate 3 denotes “ordinary conversational speech”. While rate 2 seems to be most compatible with the manner in which the TIMIT corpus was recorded, syllabic transcriptions were also generated for rate 3. The syllabic transcriptions of the corpus are referred to as rate 2 syllables and rate 3 syllables throughout the remainder of this paper.

The rate 3 syllables differ from those of rate 2 primarily by whether intervocalic consonants are considered part of the previous or following syllable. The most visible side effect is that many of the syllables which take a CV form at rate 2 instead take a VC form at rate 3 as the intervocalic consonant is considered part of the previous syllable. For example, the phonetic transcription of the partial utterance “she had your dark suit…” is syllabified as /[sh ix] [hh eh d] [jh ih] [d ah k] [s ux] [q]/ at rate 2 but as /[sh ix hh] [eh d] [jh ih d] [ah k] [s ux q]/ at rate 3 (where ‘[‘ denotes a syllable onset and ‘]’ denotes a syllable offset).

The syllabic transcription generated by tsylb2 is not time aligned so the following rules were used to generate time aligned syllabic transcriptions:

1. Where tsylb2 generates more than one possible syllabification, the final option is selected
2. A sequence of one or more phonemes not surrounded by ‘[‘ and ‘]’ are grouped and considered to be a syllable
3. The onset time of the first phoneme after a syllable onset delimiter is considered to be the syllable onset time
4. Syllable offsets are ignored
5. Phonemes that are ambisyllabic are assigned to the following syllable in the syllabic transcription

III ALGORITHMS

The candidate boundary generation stages of a number of algorithms were implemented and the details of these implementations are described in the following subsections.

a) *Mermelstein Minima*

Mermelstein proposed a syllable boundary detection algorithm which uses the difference between the convex hull of the envelope and the envelope itself to identify candidate boundaries [8]. The outline implementation of the candidate boundary generation stage used in our evaluation is as follows:

1. Preemphasise the speech signal using a 1st order FIR filter with a slope of approximately 6dB per octave
2. Bandpass the preemphasised signal with a 4th order Butterworth filter giving an attenuation of -12dB per octave below 500Hz and above 4000Hz
3. Full wave rectify the band passed signal
4. Low pass filter the rectified signal at the envelope cutoff frequency: 40Hz. Bidirectional filtering with a 2nd order Butterworth filter produces a result equivalent to a zero phase shift 4th order filter.
5. Down-sample the low passed envelope to a sampling frequency of 500Hz.
6. Identify candidate boundaries as the times of minima in the down-sampled envelope.

b) *Multichannel Envelope Minima*

A syllable segmentation algorithm was proposed in [9] which used the envelope (and envelope ratios) in three frequency bands to identify syllable boundaries. A slightly modified version of the candidate boundary generation stage of this algorithm can be outlined as follows:

1. Pre-filter the speech signal with one of three filtering options: no filter, the preemphasis filter used for Mermelstein Minima or the simplified equal loudness filter described in [9].
2. Decompose the signal into 3 frequency bands: 0-1000Hz, 0-3000Hz and the full frequency range. A 2nd order Butterworth filter is used to low pass filter the two narrower bands.
3. Full wave rectify the signal in each band.
4. Low pass filter the rectified signal in each band at the envelope cutoff frequency using bidirectional filtering with a 2nd order Butterworth filter.
5. Down-sample each band to a sampling frequency of 500Hz.
6. Identify candidate boundaries as the union of Envelope Minima times in all bands.

c) *Envelope Minima*

The Envelope Minima algorithm is a simplified version of the Mermelstein candidate boundary generation stage. The primary difference is that there is no band pass filter step. Candidate boundaries are identified using the envelope of the (possibly pre-filtered) speech signal. The algorithm has the following outline:

1. Pre-filter the speech signal with one of three filtering options: no filter, the preemphasis filter used for Mermelstein Minima or the simplified equal loudness filter described in [9].
2. Full wave rectify the possibly filtered signal
3. Low pass filter the rectified signal at the envelope cutoff frequency using bidirectional filtering with a 2nd order Butterworth filter.
4. Down-sample the low passed envelope to a sampling frequency of 500Hz.
5. Identify candidate boundaries as the times of minima in the down-sampled envelope.

d) *Wu Maxima*

The Wu Maxima algorithm is a significantly modified version of of the candidate boundary generation of the algorithm described in [1]. In the original data dependent algorithm, features derived from two dimensional filtering of the power spectrum are combined with log-RASTA features and used as input to neural network classifier for estimating syllable onsets. The data independent implementation outlined below excludes both the log-RASTA features and subsequent neural net classification:

1. Resample the speech signal at 8000Hz.
2. Compute the magnitude squared of the 512 point Short Term Fourier Transform (STFT), evaluated on a 25ms Hanning window, calculated every 10ms.
3. Filter each STFT band across all time samples using a 61 point Gaussian derivative that emphasises changes on the order of 150ms and correct for the average group delay.
4. Filter across the STFT bands at each time sample using a 61 point Gaussian low pass filter and correct for the average group delay.
5. Half wave rectify the signal in each STFT band.
6. At each time sample, map from equal size STFT bands to 9 critical bands, by taking the mean of all STFT bands whose centre frequency is within the range of the critical band.
7. Identify candidate boundaries as the union of signal maxima times in all critical bands.

IV RESULTS

The test corpus consisted of the acoustic waveform data and syllabic transcriptions (generated as described in section II) of all 6300 utterances in the TIMIT corpus. The syllabic transcriptions contained a total of 80897 rate 2 syllables and 80134 rate 3 syllables.

For each utterance in the corpus a strictly monotonically increasing sequence of reference syllable onset times, $\{r_1,..,r_J\}$, can be extracted from the corresponding time aligned syllabic transcriptions for rate 2 and rate 3 syllables. Each algorithm outlined in section III was implemented in MATLAB and returns a monotonic sequence of candidate syllable onset times, $\{c_1,..,c_K\}$, when executed on an utterance waveform. We define the sequence of matching candidate syllable onsets, $\{m_1,..,m_L\}$, to be a monotonic subsequence of $\{c_k\}$ such that equations (1) and (2) hold.

$$\min\left\{\left|c_k - r_j\right|\right\} < 0.05, \quad 1 \le k \le K,\ 1 \le j \le J \qquad (1)$$

Table 1: The performance of each algorithm under test. The results are first divided by tsylb2 rate, then grouped by envelope smoothing frequency (f_{env}). In each group, the results listed are the reference syllable match rate expressed as a percentage, the mean Δt between matching candidate and reference boundaries, and the insertion rate (number of non-matching candidate boundaries inserted per reference boundary). The temporal filtering of the Wu Maxima algorithm is unlike the envelope smoothing of the other algorithms but nevertheless most similar to envelope smoothing at 10Hz.

Algorithm	f_{env}=10Hz		f_{env}=20Hz		f_{env}=40Hz	
	Match % ($\overline{\Delta t}$ ms)	Ins. Rate	Match % ($\overline{\Delta t}$ ms)	Ins. Rate	Match % ($\overline{\Delta t}$ ms)	Ins. Rate
rate 2 syllables						
Envelope Minima, no prefilter	81.7 (25)	0.5	92.1 (20)	1.4	98.6 (12)	5.6
Envelope Minima, premphasis	80.8 (26)	0.6	93.0 (21)	1.5	99.1 (12)	5.6
Envelope Minima, equal loudness	81.9 (25)	0.5	92.2 (20)	1.4	98.7 (12)	5.7
Multi-Channel Minima, no prefilter	67.1 (21)	2.4	93.4 (18)	4.3	99.1 (11)	10.5
Multi-Channel Minima, premphasis	77.0 (19)	3.0	96.3 (16)	5.5	99.7 (8)	13.3
Multi-Channel Minima, equal loudness	69.5 (21)	2.6	93.8 (18)	4.6	99.2 (10)	11.4
Mermelstein Minima	—	—	—	—	99.4 (11)	6.1
Wu Maxima	84.3 (17)	4.3	—	—	—	—
rate 3 syllables						
Envelope Minima, premphasis	71.2 (27)	0.7	88.7 (21)	1.6	98.7 (13)	5.6
Multi-Channel Minima, premphasis	89.1 (18)	2.9	96.6 (12)	5.5	99.7 (7)	13.4
Mermelstein Minima	—	—	—	—	99.3 (10)	6.2

$$\text{len}\{m_i\} \le \text{len}\{r_j\} \qquad (2)$$

From equation (1), each candidate syllable onset time in $\{m_i\}$ is within ±50ms of a reference syllable onset time in $\{r_j\}$. There may be reference syllable onsets where equation (1) does not hold, and some reference onsets may not have a matching candidate onset, hence equation (2).

We can now define the match rate, insertion rate, deletion rate, Total Error Rate (TER) and mean Δt ($\overline{\Delta t}$) as follows:

$$matchRate = \frac{\text{len}\{m_i\}}{\text{len}\{r_j\}} \qquad (3)$$

$$deletionRate = 1 - matchRate \qquad (4)$$

$$insertionRate = \frac{\text{len}\{c_k\} - \text{len}\{m_i\}}{\text{len}\{r_j\}} \qquad (5)$$

$$TER = insertionRate + deletionRate \qquad (6)$$

$$\overline{\Delta t} = \frac{\sum_{i,k} \min\{|m_i - r_k|\}}{\text{len}\{m_i\}}, \quad |m_i - r_k| \le 0.05 \qquad (7)$$

The key results of executing the algorithms under test on all utterances are tabulated in Table 1. The deletion rate and TER (not included in the table) can be calculated simply using equations (4) and (6).

The match rate of each algorithm improves as the low pass cut off frequency used for envelope smoothing is increased. For rate 2 syllables the match rate is higher than 99% at 40Hz. For rate 3 syllables the same trend is maintained. The algorithm choice has little effect on the match rate performance at 40Hz, with the best and worst algorithms differing by just over 1%.

The pre-filtering of the speech signal has an effect on the match rate which depends on both the algorithm and envelope smoothing frequency. For rate 2 syllables and an envelope smoothing frequency of 10Hz, the simplest algorithm, the Envelope Minima algorithm with no pre-filtering, has a match rate which is almost as good as the best match rate. The more complex Multi-Channel Minima algorithm with no pre-filtering has a match rate performance as much as 10% worse. The situation is reversed when segmenting rate 3 syllables. In this case the match rate performance of the Multi-Channel Minima algorithm is almost 18% better than the Envelope Minima algorithm.

The insertion rate of each algorithm increases faster than the match rate as the envelope smoothing frequency is increased. This means that increasing the envelope smoothing frequency improves matching performance, but only at the expense of a significant increase in the number of candidate syllable onsets generated. Table 2 shows that the increasing insertion rate quickly dominates the TER. A large TER at the candidate generation stage can make development of a robust syllable segmentation algorithm more difficult as the boundary selection stage must reject many more, often very similar, candidates.

Table 2: TER versus envelope smoothing frequency for the Envelope Minima algorithm with no pre-filtering segmenting rate 2 syllables.

	Ins. Rate	Del. Rate	TER
f_{env}=10Hz	0.53	0.18	0.71
f_{env}=20Hz	1.40	0.08	1.48
f_{env}=40Hz	5.61	0.01	5.62

It is instructive to examine an utterance that exhibits a poor match rate in more detail. Figure 1 depicts the spectrogram for the utterance "he will allow a rare lie". Figure 2 depicts the utterance segmented using the Envelope Minima algorithm after envelope smoothing at 10Hz, while Figure 3 depicts the same utterance segmented after envelope smoothing at 40Hz.

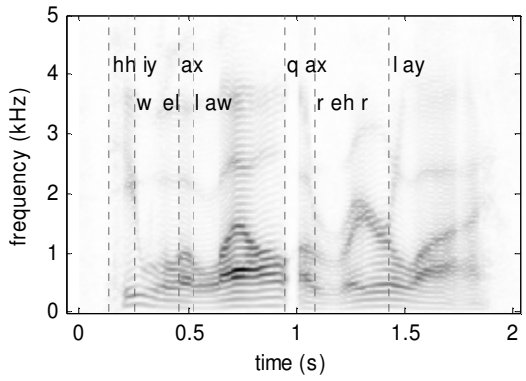

Figure 1: Spectrogram for the utterance "he will allow a rare lie". The vertical dotted lines mark syllable onsets derived from the rate 2 syllabic transcription.

At 10Hz, there are relatively few candidates and hence relatively few insertions. There are several occurrences of a deletion followed (or preceded) by an insertion within 50 to 100ms. This pattern occurs when the segmentation algorithm chooses the "wrong" location for the boundary rather than missing the boundary altogether. The problem phonemes in this utterance are liquids and glides which appear to have an envelope minimum within the main body of the phoneme rather than at its labelled boundaries. The syllables /w el/ and /l aw/ exhibit this behaviour.

At 40Hz, there are a large number of candidates, many of which result from relatively low amplitude high frequency ripples in the smoothed envelope. It appears that the improved matching performance at 40Hz may be attributed to the greater number of candidates and shorter time between them, providing a more complete sampling of the possible boundary space. An algorithm whose selection stage primarily uses the envelope for candidate rejection (such as the convex hull algorithm described in [8]) will have difficulty distinguishing between good and bad candidates. For example the onset of the syllable /l ay/ is marked by an envelope minimum that is not very different from the minimum that immediately precedes it. The syllables /w el/ and /r eh r/ are not marked by any envelope minimum.

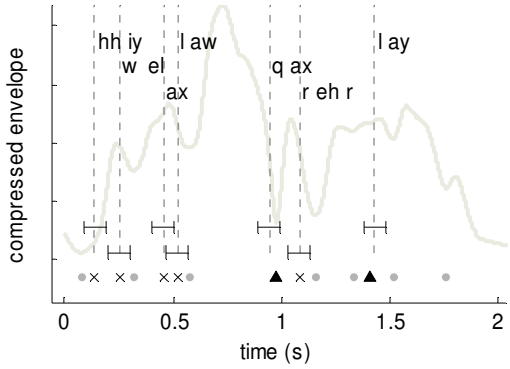

Figure 2: "He will allow a rare lie" segmented using Envelope Minima with f_{env}=10Hz. The solid line is the envelope, the vertical dotted lines are the reference syllable onsets, the horizontal error bars are the range within which candidate boundaries can match, the triangles are matched candidates, the 'x' marks are deletions and the filled circles are insertions.

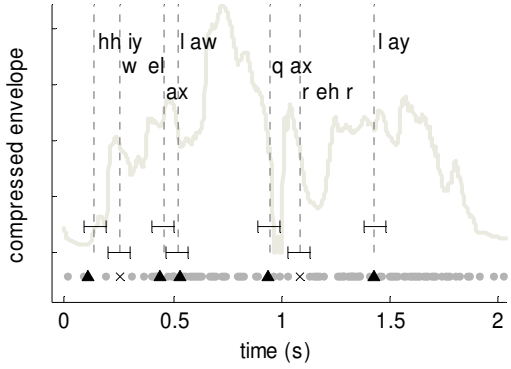

Figure 3: "He will allow a rare lie" segmented using Envelope Minima with f_{env}=40Hz. Figure markings are as described for Figure 2.

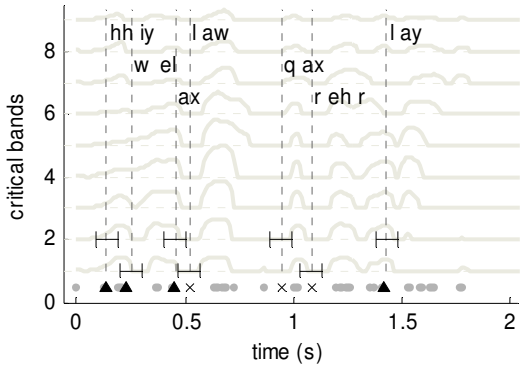

Figure 4: "He will allow a rare lie" segmented using Wu Maxima. The temporal and channel filtered envelopes are half wave rectified and then averaged into critical bands. The solid gray lines are the band values after compression (by taking the 4th root) and normalization for plotting. The band pass form of the temporal filter means that it is not possible to directly compare the channel values with the envelopes in Figure 2 and Figure 3.

Figure 4 depicts the same utterance as before, segmented using the Wu Maxima algorithm. The combination of a band pass temporal filtering and half wave rectification results in critical band maxima being located in the vicinity of the transition from local minima to rising edges in the low pass filtered envelope. While this approach enhances changes in envelope it still fails to generate good candidate onsets for the syllables /w el/ and /r eh r/. Furthermore the greater frequency resolution obtained by generating candidate boundaries in multiple critical bands does not appear to significantly improve the performance. One reason for this is that the bands are highly correlated as a result of the channel filtering performed in the algorithm. Therefore individual bands are not adding much information.

V CONCLUSIONS

The results show that the matching rate performance of envelope based syllable segmentation algorithms generally seems to improve as the envelope smoothing frequency is increased. However this apparent improvement is far exceeded by the corresponding increase in the insertion rate (and TER). Within the range of parameters examined above, very near optimum algorithm performance measured in terms of TER can be achieved by the simplest algorithm, Envelope Minima with no pre-filtering, at the lowest envelope smoothing frequency. However the matching rate of this algorithm and configuration is just 82%. We interpret this result as suggesting that envelope based syllable segmentation must be supplemented by syllable segmentation based on other acoustic features in order to achieve a higher matching rate without the significant increase in TER. Manual inspection of the spectrogram in Figure 1 indicates direction changes in the formant tracks in the vicinity of labelled syllable boundaries. A straightforward extension of envelope based techniques with formant track features may yield improved performance and an investigation of this hypothesis is for future study.

ACKNOWLEDGEMENTS

The authors wish to thank Nick Campell for his valuable discussions and feedback while portions of the work described above were carried out at ATR Human Information Science Laboratories and subsequently.

The authors also wish to thank Graham O'Brien and Radostin Getzov for their valuable contribution to the generation of the test corpus and implementation of segmentation algorithms.

REFERENCES

[1] S.-L. Wu, M. L. Shire, S. Greenberg, and N. Morgan, "Integrating syllable boundary information into speech recognition," presented at ICASSP, Munich, 1997.

[2] P. Mokhtari and N. Campbell, "Automatic measurement of pressed/breathy phonation at acoustic centres of reliability in continuous speech," IEICE Transactions on Information and Systems, vol. E86-D, pp. 574-582, 2003.

[3] J. Goslin, A. Content, and U. H. Frauenfelder, "Syllable segmentation: are humans consistent?," presented at Eurospeech '99, Budapest, 1999.

[4] D. Kahn, "Syllable based generalizations in English phonology," Ph.D. dissertation, Department of Linguistics and Philosophy, Massachusetts Institute of Technology, Cambridge, 1976.

[5] A. Content, R. K. Kearns, and U. H. Frauenfelder, "Boundaries versus Onsets in Syllabic Segmentation," Journal of Memory and Language, vol. 45, pp. 177-199, 2001.

[6] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, N. L. Dahlgren, and V. Zue, "TIMIT Acoustic-Phonetic Continuous Speech Corpus," Linguistic Data Consortium, University of Pennsylvania., 1993.

[7] W. M. Fisher, "tsylb2," National Institute of Standards and Technology, 1996. Available: http://www.nist.gov/speech/tools.

[8] P. Mermelstein, "Automatic segmentation of speech into syllabic units," Journal of the Acoustical Society of America, vol. 58, pp. 880-883, 1975.

[9] R. Villing, J. Timoney, T. Ward, and J. Costello, "Automatic Blind Syllable Segmentation for Continuous Speech," presented at Irish Signals and Systems Conference 2004, Belfast, 2004.