Automatic Blind Syllable Segmentation for Continuous Speech

Rudi Villing$^{\phi}$, Joseph Timoney*,Tomas Ward$^{\phi}$ and John Costello$^{\phi}$

$^{\phi}$ Department of Electronic Engineering
NUI Maynooth,
Maynooth, Co. Kildare
E-mail:
rudi.villing@eeng.may.ie;tomas.ward@eeng.may.ie

** Department of Computer Science,*
NUI Maynooth
Maynooth, Co. Kildare
E-mail: $^{\phi}$ jtimoney@cs.may.ie

Abstract -- **In this paper a simple practical method for blind segmentation of continuous speech into its constituent syllables is presented. This technique which uses amplitude onset velocity and coarse spectral makeup to identify syllable boundaries is tested on a corpus of continuous speech and compared with an established segmentation algorithm. The results show substantial performance benefit using the proposed algorithm.**

Keywords – **Syllable segmentation, syllable boundary detection, speech perception.**

I. INTRODUCTION

Syllables, it has been argued are one of the most important elements in human speech perception but until recently most speech analysis, from the point of view of synthesis has been on a phoneme, diphone or triphone level. This is a legacy of both the history of speech science and engineering tractability. The number of possible syllables in any language is several times larger than the aforementioned units yielding concatenative synthesis databases of unwieldy proportions. Nonetheless, considering its accepted status as the smallest pronounceable unit in any language, its role in speech perception cannot be over estimated. Indeed integration of syllabic information is yielding significant improvements in many aspects of speech engineering such as recognition and synthesis.

Our interest in the syllable lies in its potential to help understand the integration of emotional information and natural speech. Such knowledge will have an impact in the development of more expressive synthesis in artificial speech. Research has shown that emotion is encoded in contextual and prosodic manners of the utterance and of these one of the most germane is rhythm. It appears that the timing of the syllabic units underlying an utterance is important in conveying emotional sentiment. However, accurate modelling of the perceived timing of syllables is still an open problem in speech science. While research continues on improved models of temporal perception of isolated syllables [1], we are currently seeking to apply current models to continuous speech. In order to do this, the automatic segmentation of continuous speech into syllabic segments is required.

There are many advanced techniques for this at present but as yet there are no robust efficient methods that can yield acceptable results. We propose a robust, straightforward technique that will achieve this. The technique to be presented here has application in speech synthesis, particularly in corpora labelling such as for unit selection synthesis and studies of syllabic co-articulation and other prosodic features pertinent to speech research in general.

II. SYLLABLE SEGMENTS

The idea of a syllable is one that most people, be they speech researchers or not, understand on an intuitive level. However although an important linguistic element, there is no exact scientific definition of what constitutes a syllable. In fact currently most linguistics use rather vague descriptions based on a central peak of sonority (usually a vowel), and the consonants that cluster around it. The combination of allowable segments and typical sound sequences, is language specific and constitutes the syllable structure.

In linguistic terms the syllable consists of a number of parts as shown in Figure 1.

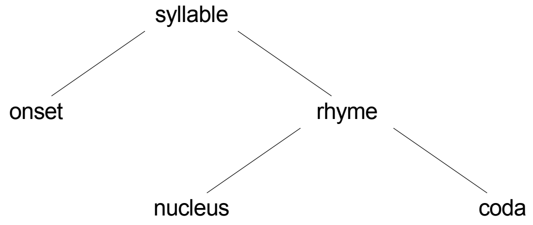

Figure 1 Syllable Structure

Of these parts, the syllable onset is generally best preserved in continuous speech while the nucleus vowel may be reduced or altered to fit the speaking rate and adjacent syllables. The syllable coda may be lost entirely.

Within this definition of syllable structure many different permutations of syllables are possible. Table 1 lists some examples for the English language.

Light	Has a non-branching rhyme (short vowel). Some languages treat syllables with a short vowel(nucleus) followed by a consonant (coda) as light.	CV, CVC
Closed	Ends with a consonant coda.	CVC, CVCC, VC
Open	Has no final consonant	CV

Table 1 Some syllable types in the English language

There are many other such types of syllable that linguists judge to exist and it is this variability in their manifestation that make them so difficult to define precisely from a speech processing point of view. Indeed a working definition for the purposes of this work is that a syllable is a speech segment consisting of a cluster of phones surrounding an vowel like energy peak.

For segmentation purposes therefore the detection of this energy peak leads to the detection of the syllable. Detecting vowel like energy peaks is relatively straightforward using spectral methods but to complete the segmentation syllable boundaries must also be detected. It is this boundary detection problem that has proven very difficult to do in a reliable and simple manner. The assignment of consonants among syllable centres is not easily determined and seems to be a function of the pauses between energy bursts introduced by the articulation that produced the utterance to be segmented. The location of these pauses is as a result of very complex relationships between the linguistic, grammatical, contextual and etymological variables. The difficulty of this problem however has not deterred speech scientists and engineers from making progress in this area and consequently a number of fairly acceptable techniques exist.

One of the more favoured techniques is due to [2]. As most syllables onsets are characterised by synchronised patterns of rising adjacent sub-band intensity this technique segments on full band intensity minima subject to segment length and energy change magnitude criteria. This technique has proven to be 75-85% consistent with manual labelling. It has also been the basis for a whole family of techniques such as those by [3,4] which integrate additional acoustic features into a weighted threshold model. These models have the advantage of ease of implementation although their performance has not been adequate for many researchers.

This has lead to the development of more sophisticated techniques such as that by [5] which uses self organising maps and techniques by [6,7] which both use complex models of the relationships between the linguistic and speech processes involved. These techniques appear to yield improvements in segmentation have the disadvantage of being difficult to implement and none have been accepted as an algorithm of choice for speech researchers.

Interestingly from a practical point of view the difficulty of the problem can be tempered if one considers that recent research has shown that for English at least, most everyday words in conversational use are monosyllalbic in nature. It is also useful to realise that for practical applications most syllables (again in the English language) are of the canonical CV,VC,V or CVC varieties. With this in mind and perceptual timing in speech as outlined in the introduction as a long term goal, a method was sought to aid our syllabification task subject to the caveats that some hand correction of labels would be acceptable and that the technique must be simple and fast.

Before describing the technique in detail some rigorous mathematical formulation of the problem is presented based on [8].

We have an alphabet **S** of syllable segments from which our speech waveforms constitute rendered utterances. Consequently each utterance can be expressed in the form

$$u = s_1 \dots s_{l(u)} \qquad (1)$$

where *l(u)* denotes the length of the utterance *u* in syllable segments. Such an utterance manifests itself as a waveform w of sample length *l(w)*.

The set of syllable-built utterances used for testing constitutes a set ***U***, which together with the set ***W*** of waveforms defines our problem corpus ***D*** that is to be segmented.

$$\boldsymbol{D}=\{(u,w);\ u\in\boldsymbol{U},\ w\in\boldsymbol{W}\} \qquad (2)$$

Hand labelling of data yields a set of boundaries (which manifest themselves as sample numbers in the waveform) assigned to every (u,w) ∈ D as an (n+1)-tuple (x_0, …, x_n). We will refer to these as $X_{(u,w)}$. Obviously then $X_{(u,w)}$ represents syllable segment boundary points of the waveform *w* in accordance with its syllable-labelled form *u*.

The goal therefore is to derive a boundary estimation function ε that assigns to any (u,w) ∈ D an (m+1)-tuple (y_0, …, y_m) which we will refer to as $Y_{(u,w)}$.

An ideal segmentation function ε will yield

$$X_{(u,w)} = Y_{(u,w)} \qquad (3)$$

although in reality insertion and deletion errors will occur. Such a set theoretic approach gives rise to the following scoring systems based on cardinality.

$$\text{Fraction correct} = \frac{\#(X_{(u,w)} \cap Y_{(u,w)})}{\#(X_{(u,w)})} \qquad (4)$$

$$\text{Fraction deletions} = \frac{\#(X_{(u,w)} / Y_{(u,w)})}{\#(X_{(u,w)})} \qquad (5)$$

$$\text{Fraction insertions} = \frac{\#(Y_{(u,w)} / X_{(u,w)})}{\#(X_{(u,w)})} \qquad (6)$$

Candidate boundary estimation functions ε are described and evaluated in the following sections.

III. MERMELSTEIN SYLLABLE DETECTOR

Mermelstein's algorithm [2] is a popular rule based syllable boundary detection algorithm that does not rely on a statistically oriented back end such as a neural network. The implementation of the algorithm used in this paper can be broadly outlined as follows:

1. Band pass filter the speech signal to the range 500 to 4000Hz using a second order Butterworth filter
2. Low pass filter the square of the resulting signal at 12Hz to obtain the intensity envelope. Bidirectional filtering using a first order Butterworth filter ensures zero phase shift.
3. Calculate the convex hull of the intensity envelope
4. Subtract the intensity envelope from the convex hull. The difference has peaks corresponding to troughs in the intensity envelope.
5. Subject to some constraints outlined below the point of maximum difference is selected as a syllable boundary and the algorithm (from step 3 onwards) is recursively applied to the subintervals delimited by the boundary.
6. Recursion stops when no suitable boundary can be found in an interval.

Several constraints may prevent a candidate boundary being selected as a syllable boundary:

- The maximum difference between the intensity envelope and the convex hull must be greater than 2dB
- The subintervals on the left and right of the boundary must both be longer than 80ms
- The difference between the peak intensity of each subinterval and the peak intensity of the entire signal cannot be more than 25dB
- The zero crossing measurement of the signal at the peak intensity location of each subinterval must be less than 5000 crossings/sec

A variation that is claimed to improve on Mermelstein's algorithm was described by [9]. Two simple modifications are made:

- The original signal is low pass filtered to 650Hz rather than band pass filtered between 500 to 4000Hz. This approximates the intensity envelope of the first formant.
- The zero crossing constraint is made redundant by the low pass nature of the signal and is removed

The Mermelstein algorithm generally performs well with clearly articulated syllables but does not handle short unstressed syllables very well. The thresholds used to avoid incorrect syllable insertions are largely responsible for this. Unfortunately simply relaxing the thresholds does not solve the problem as the algorithm has no good way of pruning the inevitable non-syllable boundaries which would then be detected.

This paper investigates whether performance of the Mermelstein algorithm can be improved upon by integrating perceptual pre-processing with simple spectral classification of boundaries to yield a lower error rate.

IV. PROPOSED ALGORITHM

Like Mermelstein's algorithm we propose intensity peaks as syllable nucleus candidates and intensity troughs as candidate syllabic boundaries. However the method used to score and prune candidates is based on the envelope velocity and coarse spectral makeup rather than a convex hull.

Figure 2 illustrates the signal processing procedures performed on the acoustic waveform.

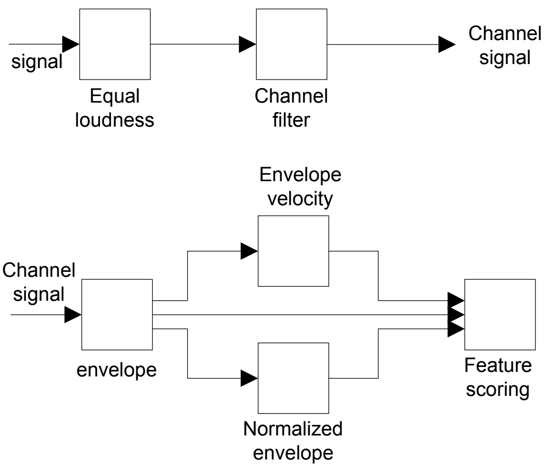

Figure 2 Signal processing steps for syllable segmentation

a) *Perceptual Signal Processing*

The speech waveform is first resampled if necessary. All subsequent operations assumed the 16kHz sample frequency required for accurate transmission of wideband speech.

To approach the performance of a human listener more closely the speech is filtered to simulate the perception of equal loudness in human hearing. In general the ear very much attenuates low frequencies while frequencies centered around 500Hz and 4000Hz are enhanced. A comfortable sound listening level is generally acknowledged to be around 72dB SPL. For that reason the 70dB equal loudness curve measured by [10] was used to design an equal loudness filter. The 70dB equal loudness curve describes the sound pressure level required at each frequency to be perceived as equal in loudness to a 1kHz tone at 70dB SPL. The gain of the equal loudness filter is the difference between the inverted curve and the constant 70dB level as shown in Figure 3.

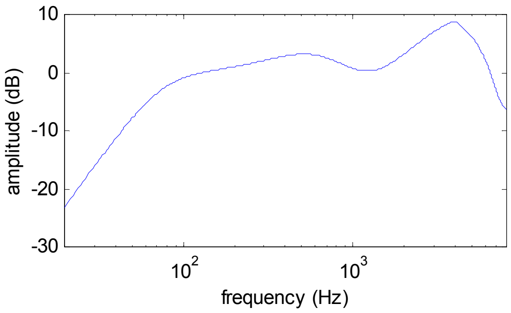

Figure 3 Frequency response of equal loudness filter cascade

Using the approach of [11] the filter was implemented as a cascade of a high pass second order butterworth filter (cutoff at 150Hz) and an IIR filter of order 8 designed to match the loudness gain response above 150Hz.

The signal is next filtered into channels using second order butterworth filters. Inspired by approach in [12] an ever widening low pass filter is used to obtain the channel signals. In practice just 3 channels were found to be satisfactory. The first two channels have cutoff frequencies of 1000Hz and 3000Hz approximating the maximum frequencies of the formants F1 and F2 respectively. The third channel is simply the entire signal and is limited by the sampling rate to a maximum frequency of 8000Hz.

The envelope for each channel is obtained by full wave rectification of the channel signal, followed by low pass filtering at 12Hz, downsampling to 100Hz and raising to the power of 0.3 to simulate human sensitivity to loudness. The low pass filter used was a first order butterworth filter and the filter was applied bidirectionally to ensure zero phase shift.

A normalized channel envelope is obtained by calculating the ratio of each channel's envelope to the envelope of the total signal. These normalized envelopes represent the coarse spectral content of the signal.

Finally the onset velocity is calculated for the fullband channel. The envelope velocity is calculated as the first difference of the envelope. Then the onset velocity is obtained by half wave rectification of the envelope velocity.

b) *Candidate Boundary Locations*

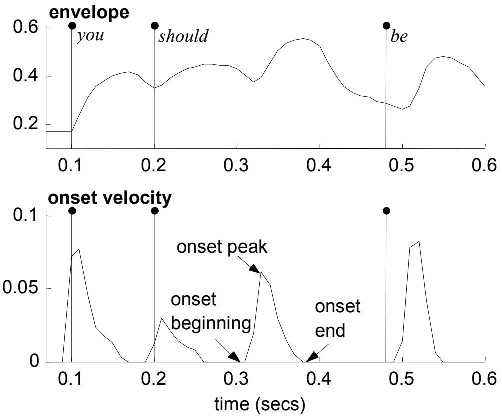

Figure 4 Envelope and onset velocity for a portion of one test utterance. The vertical bars indicate hand labelled boundaries.

Associated with each onset are three significant locations: the onset start, onset peak and onset end.

An onset start is defined as the sample index at which onset velocity first rises above zero. It corresponds to point of transition from a trough to an onset in the original envelope and is considered to be a candidate syllable boundary location. The vector of onset starts for a waveform is denoted **os**.

Conversely, an onset end is defined as the sample index at which the velocity finally drops back to zero. It corresponds to a peak in the original envelope and is a candidate syllable nucleus location. The vector of onset ends is denoted **oe**.

Finally an onset peak is the sample index at which the onset velocity reaches its maximum. It provides a measure of the abruptness of the onset. The vector of onset peaks is denoted **op**.

c) Candidate Boundary Scoring

A general score function is defined in equation (7) based on the method described in [13].

$$\mathbf{s} = score(\mathbf{x}, \mathbf{idx}, c_1, c_2) \qquad (7)$$

where **s** is defined as follows:

$$\mathbf{s}_i = \left\{ \begin{array}{l} 0, \text{if } \mathbf{x}_j < c_1 \\ 1, \text{if } \mathbf{x}_j \geq c_2 \\ \dfrac{\mathbf{x}_j - c_1}{c_2 - c_1}, \text{if } c_1 \leq \mathbf{x}_j < c_2 \end{array} \right\}, j = \mathbf{idx}_i \qquad (8)$$

In other words the score function evaluates values of the vector **x** at indexes defined by the vector **idx** against a lower threshold c$_1$ and an upper threshold c$_2$, returning scores in the range 0 to 1 at each index.

Thresholds which are required for the various scores were determined by empirical analysis. The following table defines the values used for subsequent score calculation.

Threshold	Minimum	Maximum
b	0.01	0.1
s	0.6	0.7
c	0.85	0.97
vp	0.01	0.1

Table 2 Thresholds used for feature score calculation

Boundary scores, **bs**, in the a signal are computed based on the onset velocity, **vel**, as follows:

$$\mathbf{bs} = score(\mathbf{vel}, \mathbf{op}, b_{\min}, b_{\max}) \qquad (9)$$

This score eliminates boundaries which precede onsets that peak at a low velocity (less than b$_{\min}$).

Vowel scores, **vs**, are computed based on the onset velocity, **vel**, at each onset peak and the spectral content at each onset end obtained from the normalized F1 envelope **f1n**.

$$\mathbf{vs}_i = \mathbf{ss}_i(1 - \mathbf{cs}_i)\mathbf{vps}_i \qquad (10)$$

where

$$\mathbf{ss} = score(\mathbf{f1n}, \mathbf{oe}, s_{\min}, s_{\max}) \qquad (11)$$

$$\mathbf{cs} = score(\mathbf{f1n}, \mathbf{oe}, c_{\min}, c_{\max}) \qquad (12)$$

$$\mathbf{vps} = score(\mathbf{vel}, \mathbf{op}, vp_{\min}, vp_{\max}) \qquad (13)$$

In order to prevent voiced consonants at the start of syllables being incorrectly identified as vowels the **vs** vector is convolved with a temporal window that causes the largest scoring vowel to suppress smaller vowel scores less than 100ms before or after it as shown in Figure 5.

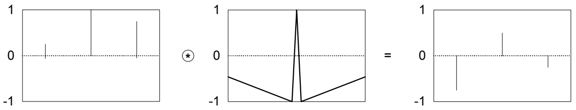

Figure 5 Convolution of vowel score with temporal window

d) Final Boundary Selection

Syllable boundaries are selected from the **oe** vector using the following simple steps:

1. The best boundary to date, bb, is set to be empty.
2. If the current boundary score, **bs**$_i$, is better than that of the best boundary to date, **bs**$_{bb}$, and the signal envelope at this boundary, **env**(**oe**$_i$)is a deeper trough than at the the best boundary **env**(**oe**$_{bb}$), set bb=i.
3. If the vowel score, **vs**$_i$, corresponding to the current boundary is non-zero it indicates that a vowel has been encountered and bb is added to the set of detected boundaries ***Y***. Repeat from step 1.

V. Results

The test corpus ***D*** consisted of 12 phonetically balanced sentences with a total of 74 syllables [14]. These were produced by a single Irish male speaker at a moderate speaking rate. The sentences (studio recorded at 44.1kHz, 16 bit) were downsampled to 16kHz for our experiments.

Each algorithm was implemented in MATLAB and run on the entire set of test data. The output of each algorithm was a set of detected syllable boundaries, ***Y***. These were compared with the hand labelled reference syllable boundaries ***X*** to count the number of correct detections, insertion errors and deletion errors based on equations (4), (5) and (6) respectively. A detected boundary y$_i$ within a tolerance of 50ms before or after a reference boundary x_i is considered to be a match. The results obtained are tabulated in Table 3.

	Correct	Insert	Delete
Mermelstein	76.1%	15.6%	23.9%
Howitt	78.9%	12.8%	21.1%
Proposed	93.1%	5.6%	6.9%

Table 3 Syllable boundary detection scores by algorithm

Where a number of candidate boundaries exist between adjacent syllables the most common error by all algorithms was to select the "wrong" candidate boundary, i.e. a boundary other than a reference boundary. Our scoring scheme records a matching insertion and deletion error in such a case. If these errors are excluded, the results will only include gross errors due to inserting or removing entire syllables. By way of comparison, these results are listed in Table 4.

	Correct	Insert	Delete
Mermelstein	87.5%	4.2%	12.5%
Howitt	90.3%	1.4%	9.7%
Proposed	98.6%	0%	1.4%

Table 4 Gross syllable detection scores by algorithm

Remaining deletion errors in the Mermelstein and Howitt algorithms are due to coalescence of adjacent syllables. The Howitt algorithm seems to coalesce syllables delimited by voiced consonants where the degree of voicing does not change significantly.

Finally it is worth noting that certain syllable combinations in the test corpus give rise to consonant clusters which would not be legal in an English word, for example the utterance "Jack was" is articulated more like "Jac kw's". The segmentation algorithms generally put the boundary before the 'k' while the human labelled reference boundary was placed after the 'k' leading to a detection error. It remains to be seen to what extent the linguistic bias overrides the acoustic boundary and whether the detected boundary is in fact more correct in this case.

VI. CONCLUSION

Substantial improvement in performance over the algorithm of Mermelstein and the modified algorithm of Howitt was gained by using onset velocity as the key indicator of boundary significance with coarse spectral makeup used to identify syllabic nuclei and prune boundaries. The algorithm has the advantage of straightforward implementation and will serve as a starting point for a continuous speech front end for P-Centre detection [1].

REFERENCES

[1] R. Villing, T. Ward and J. Timoney, "P-Centre Extraction from Speech: the need for a more reliable measure", *Irish Signals and Systems Conference*, Limerick. July 1-2, 2003.

[2] P. Mermelstein, "Automatic segmentation of speech into syllabic units." *J. Acoust. Soc. Am.*, 58(4):880-883, 1975

[3] H Meinedo, J.P. Neto and L.B. Almeida. "Syllable onset detection applied to the Portuguese language", Proceedings *EUROSPEECH'99*, Budapest, Hungary, September 1999.

[4] S.L. Wu, B.E.D. Kingsbury, N. Morgan, and S. Greenberg. "Incorporating information from syllable-length time scales into automatic speech recognition" *ICASSP*, 721-724, 1998.

[5] A. Noetzel, "Robust Syllable Segmentation of Continuous Speech Using Neural Networks", *IEEE Electro International Conference Record*, New York, 580-585, April 1991.

[6] L. Shastri, S. Chang and S. Greenberg "Syllable Detection and Segmentation Using Temporal Flow Neural Networks", *Int. Cong. of Phonetic Sciences*, San Francisco, 3:1721-1724, August 1999.

[7] C. Chandrasekhar, "Neural Network Models for Recognition of Stop Consonant-Vowel (SCV) segments in Continuous Speech". PhD thesis, Dept. of Computer Science and Engineering, Indian Institute of Technology Madras, Apr. 1996.

[8] I. Kopeček, "Automatic segmentation into syllable segments", *Proceedings of First International Conference on Language Resources and Evaluation*, Granada, Spain, 1275-1279, May 1998

[9] A.Howitt, "Vowel landmark detection", Proceedings *EUROSPEECH'99*, Budapest, Hungary, 2777-2780, September 1999

[10] D.W.Robinson and R.S.Dadson, "A re-determination of the equal-loudness relations for pure tones", *British Journal of Applied Physics* 7:166-181, May 1956

[11] D.Robinson, "Replay Gain – A Proposed Standard", [Online Document] July 2001, [cited Mar 1, 2004], Available at http://replaygain.hydrogenaudio.org/

[12] Ta-Hsin Li and J.D.Gibson, "Time-correlation analysis of non stationary signals with application to speech processing", *IEEE International Symposium Time-Frequency and Time-Scale Analysis*, Paris, France,1996

[13] D.G. Childers, "Speech Processing and Synthesis Toolboxes", New York: John Wiley & Sons, 2000.

[14] S.R. Quackenbush., T.P. Barnwell, and M.A. Clements, "Objective Measures of Speech Quality", New York: Prentice-Hall, 1988