SLAM: a PC-Based Multi-Level Segmentation Tool

Piero COSI

Centro di Studio per le Ricerche di Fonetica, C.N.R.
Phone: +39 49 8755106, FAX: +39 49 8754560
P.zza G. Salvemini, 13 - 35131 PADOVA, ITALY

Abstract

An interactive Segmentation and Labelling Automatic Module (SLAM), especially developed for Windows-based Personal Computers, is described. The system is extremely user-friendly and it was designed with the aim of supporting speech scientists in assessing the very heavy and time-consuming task of segmenting a big amount of speech material such as that caused by the tremendous spread of new and always bigger speech data-bases. The system, which is based on the Multi-Level Segmentation theory, was built using Microsoft C++ and Windows 3.1 SDK software[1], and runs preferably on Intel 386/486-based personal computers running DOS 5.00 or higher and equipped with VGA and SuperVGA boards.

Introduction

Phonetic or phonemic labelling of speech signals is normally performed manually by phoneticians or speech communication experts. Even if various attractive graphic and acoustic tools are simultaneously available, there will always be some disagreement among skilled human labelling experts in the results of labelling the same wave form[1]. In fact, due to human variability of visual and acoustic perceptual capabilities and to the difficulty in finding a clear common labelling strategy, the manual labelling procedure is implicitly incoherent. Another important drawback of manual intervention in labelling speech signals is that it is extremely time consuming. Considering these and other disadvantages, the development of methods for semi-automatic or automatic labelling of speech data is becoming increasingly important [2] especially considering the present tremendous spread of new and always bigger speech data-bases. Moreover, even if segmentation and labelling are avoided by most of the more successful Automatic Speech Recognition (ASR) systems, generally based on Hidden Markov Model techniques, a completely labelled true continuous speech database will always be of interest for other classes of ASR systems, such as those based on Neural Networks techniques, or for linguistic and phonetic research.

Complete automatic labelling systems minimise assessment time of input/output speech data-bases and are at least implicitly coherent. In fact, using the same strategy, if they make some errors they always make them in a coherent way. Unfortunately, at the present time highly reliable automatic segmentation systems are still not on the market. The semi-automatic system being described constitutes an attempt to cover the gap between reliable but time consuming manually created segmentation data and those produced by fast but still unreliable automatic systems.

Segmentation Strategy

The system is based on the Multi-Level Segmentation theory[3]. Speech is considered as a temporal sequence of quasi-stationary acoustic segments, and the points within such segments are more similar to each other than to the points in adjacent segments. Following this viewpoint, the segmentation problem can be simply reduced to a local clustering problem where the decision to be taken regards the similarity of any particular frame with the signal immediately preceding or following it. Using only relative measures of acoustic similarity, this technique should be quite independent of the speaker, vocabulary, and background noise.

The implemented segmentation algorithm was originally developed by J.R. Glass and V.W. Zue [3-4] and is called *Multi Level Segmentation* (MLS) algorithm. A joint *Synchrony/Mean-Rate* (S/M-R) model of *Auditory Speech Processing* (ASP), proposed by S. Seneff [5], providing an adequate and efficient basis for phonetic segmentation and labelling, is used as pre-processing module, and in particular, both Envelope and Synchrony Detector parameters are simultaneously considered. Advantages of using Auditory Models (AM) Vs classical "short-term" analysis approaches for automatic speech segmentation have been shown in literature, especially in adverse conditions [6]. For each target frame, within its left and right window of Δ frames length (Δ can be set to different values), an average value for each analysis vector component is computed. Depending on an Euclidean-based similarity measure, forward and backward distances between the current frame and the right and left window are calculated and a decision is taken in associating the current frame to its immediate

[1] The following are trademarks of their respective companies: Intel 386/486 (Intel Corp.) Microsoft, DOS, Windows 3.1, MS SDK, MS C++ , MDI (Microsoft Corp.), PC-AT, VGA (IBM Corp.), Oros (Oros Inc.).

past or to its immediate future. Various strategies can be adopted in defining forward and backward distances allowing the possibility of adapting the sensitivity of the association to the local environment [4]. After all frames have been analysed various adjacent regions are created. These initial 'seed regions' constitute the basis for the following 'hierarchical structuring' segmentation procedure (see Table 1) suggested by the fact that the speech signal is characterised by short events that are often quite distinct from their local environment.

Algorithm:

```
1) Find boundaries $\{b_i,\ 0 £ i £ N\},\ t_i < t_j,\ " i < j$
2) Create initial region set
    $R_0 = \{r_0(i),\ 0 £ i < N\},\ r_0(i) ° r(i,i+1)$
3) Create initial distance set
    $D_0 = \{d_0(i),\ 0 £ i < N\},\ d_0(i) ° d(r_0(i), r_0(i+1))$
4) Until $R_N = \{r_N(0)\} ° r(0,N)$
    For any $k$ such that $d_j(k-1) > d_j(k) < d_j(k+1)$
    (a) $r_{j+1}(i) = r_j(i),\ 0 £ i < k$
    (b) $r_{j+1}(k) = merge(r_j(k), r_j(k+1))$
    (c) $r_{j+1}(i) = r_j(i+1),\ k < i < N-j-1$
    (d) $R_{j+1} = \{r_{j+1}(i),\ 0 £ i < N-j-1\}$
    (e) $d_{j+1}(i) = d_j(i),\ 0 £ i < k-1$
    (f) $d_{j+1}(k-1) = max(d_j(k-1), d(r_j(k-1), r_{j+1}(k)))$
    (g) $d_{j+1}(k) = max(d_j(k+1), d(r_{j+1}(k), r_j(k+1)))$
    (h) $d_{j+1}(i) = d_j(i+1),\ k < i < N-j-1$
    (i) $D_{j+1} = \{d_{j+1}(i),\ 0 £ i < N-j-1\}$
```

Definitions:

- **b_i is a boundary occuring at time t_i.**
- **$r(i,j)$ is a region spanning times t_i to t_j.**
- **$r_j(i)$ is the i^{th} region of the j^{th} iteration.**
- **$d(i,j)$ is the distance between regions i and j.**
- **$d_j(i)$ is the i^{th} distance of the j^{th} iteration.**
- **$merge(r(i,j), r(j,k))$ combines two adjacent regions to produce a region $r(i,k)$ spanning times t_i to t_k.**
- **The distances $d_j(-1)$ and $d_j(N-j)$ are infinite.**

Table 1. Algorithmical structure of multi-level hierarchical segmentation strategy (by J.R. Glass [4], pp. 47).

This hierarchical technique, incorporating some kind of temporal constraint, is quite useful in order to appropriately rank the significance of acoustic events. The clustering scheme utilised to produce a multi-level description of the speech signal is based essentially on the same framework used for locating 'seed acoustic events'. In fact, starting from previously calculated initial 'seed regions', each region is associated with either its left or right neighbour using an Euclidean-based similarity measure, where the similarity measure is computed with a distance measure applied to the average spectral analysis vector of each region. Two regions are merged together to form a single region when they associate with each other and this new created region subsequently associates itself with one of its neighbours. The process is repeated until the whole utterance is analysed and described by a single acoustic event. By keeping track of the distance at which two regions merge into one, a multi-level description usually called *dendrogram*[4] can be constructed (see Fig.1).

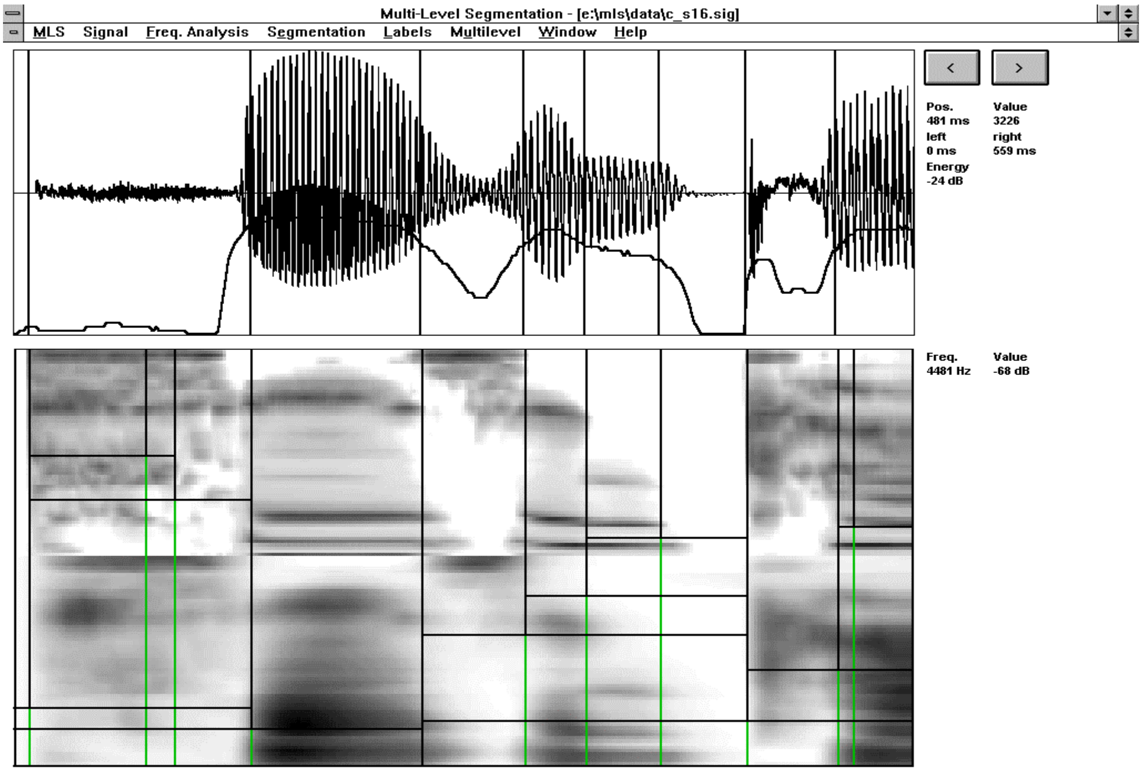
The final target segmentation can be automatically extracted [7] by appropriate pattern recognition techniques the aim of which is to find the optimal segmentation path given the dendrogram structure and the target phonemic transcription of the input sentence, but also with minimal human intervention, which is limited exclusively on fixing the vertical point determining the final target segmentation (corresponding to that found on the horizontal line built on this point), and eventually deleting over-segmentation landmarks forced by this choice. Even when using the above described manual intervention, segmentation marks are always automatically positioned by the system and never adjusted by hand. Nevertheless, the manual positioning of segmentation boundaries is always permitted to the user should this be requested in special cases. As for the computation complexity of the MLS algorithm, considering the fact that it does not make use of the entire utterance for emitting segmentation hypotheses but that it shows a local behaviour, it is capable of analysing the speech signal virtually instantaneously.

SLAM Software Implementation

As for the software implementation of SLAM, it is built using Microsoft C++ and Windows 3.1 SDK software, and runs preferably on Intel 386/486-based personal computers, running DOS 5.00 or higher and equipped with VGA or

SuperVGA boards and at least 4 Mbytes of RAM. Only for audio facilities the present implementation makes use of the OROS-AU22 DSP board, but other A/D-D/A hardware could be easily considered.

Signal wave form files can be easily displayed together with their corresponding FFT, LPC, or AM-based spectrogram, energy, pitch and zero-crossing files. At the present time, in order to use SLAM, all files should have already been created by other appropriate off-line software, but in the future their on-line creation will be included in SLAM.
A part from the signal wave form, the user is completely free to visualise any combination of the related files. Various editing operations can be executed on the signal such as LISTEN (only if adequate hardware is available), ZOOM, SCROLL, CUT, PASTE, CLEAR, and COPY, making the system, not only a segmentation and labelling tool, which represents however its most important feature, but also a general speech assessment system. One important feature of SLAM, considered as a simple speech visualising system, is represented by the possibility to move the mouse within the various windows and to instantaneously visualise the corresponding values of active representations, such as signal amplitude or time position, energy, pitch or frequency. In order to segment and label speech signals, their corresponding spectral representation (FFT, LPC, AM based) is visualised by SLAM. On the basis of the chosen spectral information, the MLS algorithm can be applied in order to create various signal alignment hypotheses and the user can easily choose the best by using the mouse and clicking in any position within the dendrogram structure (see Figure 1). The performance of the SLAM segmentation system when applied to a simple but significant segmentation task is reported in [6].



Pos. 481 ms
Value 3226
left 0 ms
right 559 ms
Energy -24 dB

Freq. 4481 Hz
Value -68 dB

Figure 1. SLAM plot referring to the English sentence "Susan ca(n't)" uttered by a female speaker. Time wave form, Energy and final segmentation are plotted in the top, while AM spectrogram and its corresponding dendrogram are illustrated in the bottom.

The user can also manually add new markers, besides those explicitly set by choosing a particular alignment hypothesis based on the dendrogram structure, in case of under-segmentation, or delete some markers in case of over-segmentation. The use of AM versus FFT-based spectrogram greatly reduces this kind of manual intervention [6] thus emphasising the importance of using an adequate signal representation when dealing with speech segmentation, especially in noisy environment. A labelling capability is also included in SLAM where SAMPA [7] labels can be attached to each segmentation mark or modified by the user.

Since Windows 3.1 MDI (Multiple Document Interface) standard was adopted in building SLAM, it is possible to open more than one window in order to visualise mulltiple signals and their related parameters, as well as to open more than one segmentation session, as illustrated in Fig. 2. The only limitation is given by the available amount of RAM.

Conclusions and Future Trends

SLAM's main feature, a part from performance [6], is its user-friendliness and given the great amount of speech databases this characteristic is very important for any useful segmentation system. In order to reduce manual intervention, SLAM will be transformed in a completely automatic segmentation and labelling system such as the one used in [8] leaving the best segmentation hypothesis to the system and permitting a human intervention in case of system errors.

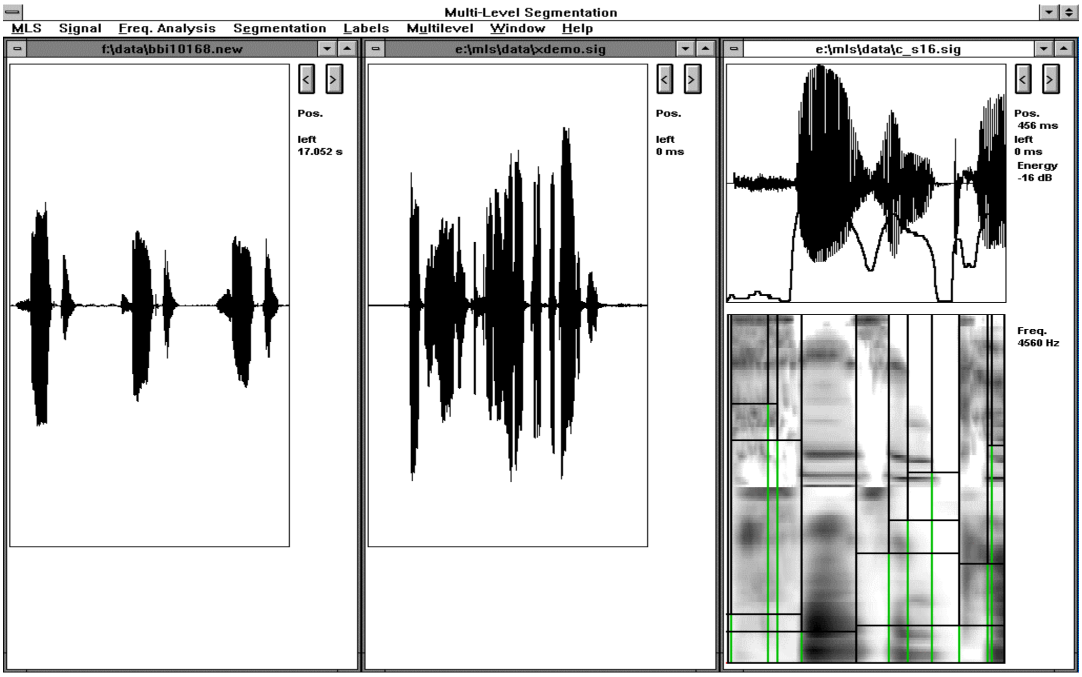


Figure 2. Use of SLAM with three simultaneous opened sessions.

Acknowledgements

This work has been made possible exclusively thanks to S. Seneff for giving me important suggestions for implementing her joint *Synchrony/Mean-Rate* (S/M-R) model of *Auditory Speech Processing* (ASP) [5], and thanks to J.R. Glass for sending me his important work "Finding Acoustic Regularities in Speech: Application to Phonetic Recognition", which was essential for developing the segmentation strategy.

References

1) P. Cosi, D. Falavigna and M. Omologo, "A Preliminary Statistical Evaluation of Manual and Automatic Segmentation Discrepancies", *Proceedings of EUROSPEECH-91*, Genova, 24-26 September 1991, pp. 693-696.

2) E. Vidal and A. Marzal, "A Review and New Approaches for Automatic Segmentation of Speech Signals", *Signal Processing V: Theories and Applications*, L.Torres, E.Masgrau, and M.A. Lagunas (eds.), Elsevier Science Publisher B.V., 1990, pp. 43-53.

[3] J.R. Glass and V.W. Zue (1988), "Multi-Level Acoustic Segmentation of Continuous Speech", Proc. IEEE ICASSP-88, New York, N.Y., April 11-14, 1988, pp. 429-432.

[4] J.R. Glass (1988), "Finding Acoustic Regularities in Speech: Application to Phonetic Recognition", Ph. D Thesis, May 1988, MIT press.

[5] S. Seneff (1988), "A joint synchrony/mean-rate model of auditory speech processing", Journal of Phonetics, Vol. 16(1), January 1988, pp. 55-76.

[6] P. Cosi (1992), "Ear Modelling for Speech Analysis and Recognition", in *Visual Representation of Speech*, M. Cooke, S. Beet and M. Crawford eds., John Wiley & Sons Ltd., 1992, pp. 205-212.

[7] A.J. Fourcin, G. Harland, W. Barry and W. Hazan eds., "Speech Input and Output Assessment, Multilingual Methods and Standards ", Ellis Horwood Books in Information Technology, 1989.

[8] V.W. Zue, J. Glass, M. Philips and S. Seneff, "Acoustic Segmentation and Phonetic Classification in the SUMMIT System", Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-89), pp. 389-392.