SLAM v1.0 for Windows: a Simple PC-Based Tool for Segmentation and Labeling

Piero COSI
Centro di Studio per le Ricerche di Fonetica, C.N.R.
WWW: http://www.csrf.pd.cnr.it

Abstract

The first official release of SLAM [1], [2], a semi-automatic segmentation and labeling environment, especially developed for Windows©-based[1] Personal Computers, is described. The system was designed with the aim of being extremely user-friendly and supporting speech scientists in assessing the very heavy and time-consuming task of segmenting and labeling a big amount of speech material such as that caused by the tremendous spread of new and always bigger speech data-bases. SLAM is based on the Multi-Level Segmentation theory [3] and the segmentation process works with various spectral representations of speech. (FFT, LPC…), even if it was especially created for an auditory-based representation [4].

Introduction

Phonetic or phonemic labeling 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 labeling experts in the results of labeling the same waveform [5]. In fact, due to human variability of visual and acoustic perceptual capabilities and to the difficulty in finding a clear common labeling strategy, the manual labeling procedure is implicitly incoherent. Another important drawback of manual intervention in labeling speech signals is that it is extremely time consuming. Considering these and other disadvantages, the development of methods for semi-automatic or automatic labeling of speech data is becoming increasingly important especially considering the present tremendous spread of new and always bigger speech data-bases. Moreover, even if segmentation and labeling are avoided by most of the more successful Automatic Speech Recognition (ASR) systems, generally based on Hidden Markov Model techniques, a completely labeled 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 labeling systems minimize assessment time of input/output speech databases 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

SLAM segmentation strategy is entirely based on the Multi-Level Segmentation

[1] All words followed by the © sign refer to trademarks of their respective companies: Pentium (Intel Corp.), Windows-3.11, Windows-95. Windows-NT4.0 (Microsoft Corp.), C++7.0, Visual-C1.5 (Microsoft Corp.), SoundBlaster-16, AWE32, AWE64 (Creative Labs. Inc.), Oros AU22 DSP (Oros Inc.).

(MLS) 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. SLAM makes use of the Multi Level Segmentation (MLS) algorithm [3-4] illustrated in Table 1.

1) Find boundaries $\{b_i, 0 \le i \le N\}, t_i < t_j, \forall\, i < j$
2) Create initial region set
$R_0 = \{r_0(i),\ 0 \le i<N\},\ r_0(i) \circ r(i,i+1)$
3) Create initial distance set
$D_0 = \{d_0(i), 0 \le i<N\},\ d_0(i) \circ d(r_0(i), r_0(i+1))$
4) Until $R_N = \{r_N(0)\} \circ r(0,N)$
For any k such that:
$d_j(k-1) > d_j(k) < d_j(k+1)$
(a) $r_{j+1}(i) = r_j(i),\ 0 \le i<k$
(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) $R_{j+1} = \{r_{j+1}(i),\ 0 \le i<N-j-1\}$
(e) $d_{j+1}(i) = d_j(i),\ 0 \le i<k-1$
(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) $D_{j+1} = \{d_{j+1}(i),\ 0 \le i<N-j-1\}$

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 [6], pp. 47).

A joint Synchrony/Mean-Rate (S/M-R) model of auditory speech processing (AM in the following), proposed by S. Seneff [4], providing an adequate and efficient basis for phonetic segmentation and labeling, is used as pre-processing module, and in particular, both Envelope and Synchrony Detector parameters are simultaneously considered. Other spectral representations such as FFT-based or LPC-based ones can obviously be adopted as preprocessing schemes for segmentation by SLAM, but the above-cited auditory representation resulted the most effective one in our previous segmentation experiments [7].

Following MLS theory, 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 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 [6]. After all frames have been analyzed 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 characterized by short events that are often quite distinct from their local environment. This hierarchical technique, incorporating some kind of temporal constraint, is quite useful in order to rank appropriately the significance of acoustic events. The clustering scheme utilized 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 neighbor 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 neighbors. The process is repeated until the whole utterance is analyzed and described by a single acoustic event. By keeping track of the distance at which two regions merge into one, a multi-level structure usually called dendrogram [6], graphically illustrated in Figure 1, can be constructed.

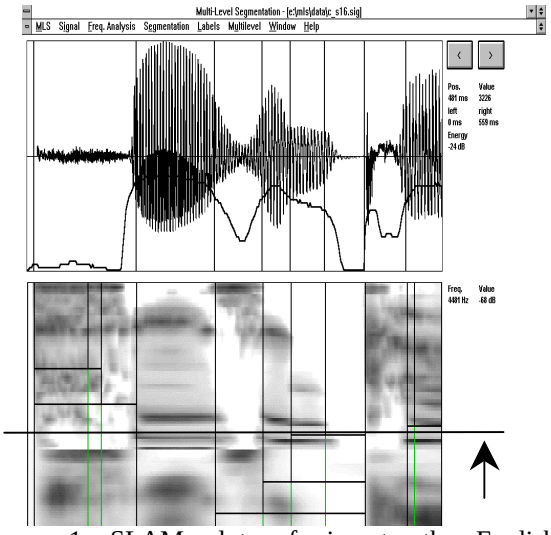

Figure 1. SLAM plot referring to the English sentence "Susan ca (n't)" uttered by a female speaker. Time waveform, energy and final segmentation are plotted in the top, while AM spectrogram and its corresponding dendrogram are illustrated in the bottom. The arrow in the bottom also points the chosen segmentation line.

The final target segmentation could be automatically extracted by suited pattern recognition techniques, the aim of which should be that of finding the optimal segmentation path given the dendrogram structure and the target phonemic transcription of the input sentence. At the present time the final target segmentation is extracted, with minimal human intervention, by exclusively fixing the vertical point determining the target segmentation boundaries corresponding to those 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 behavior, it is capable of analyzing the speech signal virtually instantaneously.

Software Implementation

SLAM was originally built under Windows-for-Workgroup-3.11© using C++7.0© but has been recently ported on VisualC++1.5©. SLAM was tested on Pentium©-based personal computers, running Windows3.1© Windows3.11©, Windows95© or Windows-NT4.0© operative systems, equipped with SuperVGA boards with at least 256 colors and at least 4 Mbytes of RAM. If the audio functionality is activated SLAM makes use of SoundBlaster16© or AWE32/64© boards, even if other A/D-D/A hardware could be easily considered such as the OROS-AU22© DSP board. Various operations can be executed by SLAM as illustrated by its main MENU depicted in Table 2. Signal waveform files can be easily displayed together with their corresponding FFT, LPC, or AM-based spectrogram, energy, pitch (computed by AMDF [8] and SIFT [9]), and

zero-crossing files, all computed by specialized algorithms. Originally, in order to use SLAM, other appropriate off-line software should have already created all files, but in this first official release on-line creation is included in SLAM together with off-line or batch creation. A part from the signal waveform, the user is free to visualize any combination of the corresponding analysis files.

SLAM Segmentation and Labeling
SLAM! Signal Edit Analysis Process Seg&Lab Multilevel Window Help

Table 2. SLAM main MENU.

Various editing operations can be executed on the signal, such as LISTEN (only if adequate hardware is available), ZOOM, SCROLL, CUT, PASTE, CLEAR, COPY, FADE-in/out, NORMALIZE, and other MATH-operations. Thus the system is not only a segmentation and labeling tool, which represents however its most important feature, but also a general speech assessment system. While moving the mouse within the various windows all the corresponding values of active representations, such as signal amplitude or time position, energy, pitch or frequency, are instantaneously visualized. In order to segment and label speech signals, their corresponding spectral representation (FFT, LPC, AM based) is computed, if required, and visualized 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 [7]. 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. As already underlined the use of AM versus FFT-based spectrogram greatly reduces this kind of manual intervention [7] thus emphasizing the importance of using an adequate signal representation when dealing with speech segmentation, especially in noisy environment. A labeling capability is also included in SLAM where SAMPA [10] labels can be attached to each segmentation mark or modified by the user using a three-buttons mouse. MDI (Multiple Document Interface) was adopted thus allowing the user to open simultaneously more than one window in order to visualize multiple signals and their related parameters, as well as to open more than one segmentation session, as illustrated in Figure 2. The only limitation is given by the available amount of RAM.

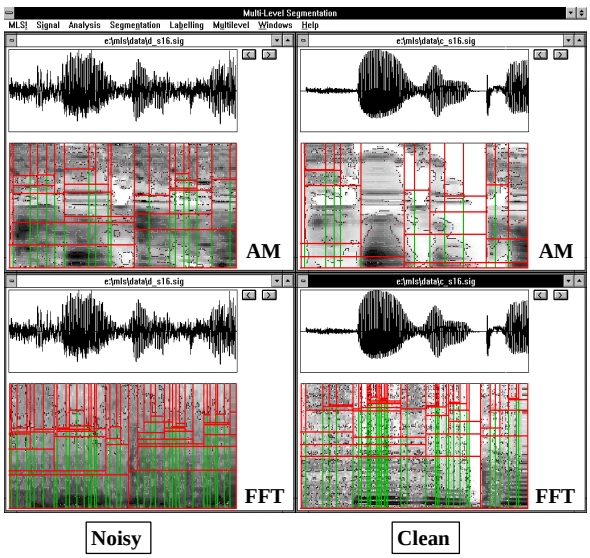

Figure 2. Use of SLAM with four simultaneous opened segmentation sessions. The same signal of Figure 1, “Susan ca(n't)”, recorded in noisy (left) and clean (right) condition, is analyzed and segmented by SLAM using two different spectral analysis based on usual FFT (bottom) or auditory model (AM) (see text).

Conclusions and Future Trends

SLAM's main feature 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 labeling system such as the one used in [11] leaving the best segmentation hypothesis to the system and permitting a human intervention in case of system errors.

Acknowledgements

This work has been made possible exclusively thanks to S. Seneff and J.R. Glass who gave me important suggestions for implementing the joint Synchrony/Mean-Rate (S/M-R) model of Auditory Speech Processing (ASP) [4], and for developing the MLS segmentation strategy [6].

References

[1] P. Cosi (1993), "SLAM: Segmentation and Labeling Automatic Module", in *Proc.EUROSPEECH-93*, Berlin, 21-23 September, Vol. 1, 1993, pp. 88-91.

[2] P. Cosi (1995), "SLAM: a PC-Based Multi-Level Segmentation Tool", in *Speech Recognition and Coding. New Advances and Trends*, A.J. Rubio Ayuso and J.M. Lopez Soler edts, NATO ASI Series, Computer and Systems Sciences, Springer Verlag, Vol. F 147, 1995, pp. 124-127.

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

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

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

[7] P. Cosi (1992), "Ear Modeling 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.

[8] M.J. Ross, H.L. Shaffer, A. Cohen, R. Freudberg and H.J. Manley (1974), "Average magnitude difference function pitch extractor", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSO-22, pp. 565-572

[9] J.D. Markel (1972), "The SIFT algorithm for fundamental frequency estimation", *IEEE Trans. Audio Elettroacoust.*, Vol. AU-20, pp. 367-377.

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

[11] V.W. Zue, J. Glass, M. Philips and S. Seneff (1989), "Acoustic Segmentation and Phonetic Classification in the SUMMIT System", *Proc. IEEE ICASSP-89*, pp. 389-392.