On the Processing of Fuzzy Patterns for Text Independent Phonetic Speech Segmentation

Luis D. Huerta-Hernández[1,2] and Carlos A. Reyes-García[1]

[1] Instituto Nacional de Astrofísica Óptica y Electrónica (INAOE)
Luis Enrique Erro No. 1, Sta. Ma. Tonanzintla, Puebla, 72840, México
{luisdh2, kargaxxi}inaoep.mx
[2] Instituto Tecnológico Superior de Acatlán de Osorio,
Unidad Tecnológica, Acatlán de Osorio, Puebla, 73440, México

Abstract. In this work we propose an algorithm for continuous speech segmentation with text independency. In our approach we do not use feature vectors in order to detect phoneme boundaries, instead we only make use of the intensity measure. Obtaining with this a remarkable reduction in the amount of information needed and simplified rules on the processing. In the process only a pre-emphasis filter, and one strategy based on a distance measure with normalized fuzzy memberships over the signal patterns are used. In the preliminary results the method reaches up to 77.54% of correct segmentation with a 20 msec. accuracy and an over segmentation rate near to 0%. The algorithm implementation, the experiments, as well as some results are shown.

1 Introduction

From the arrival of computers, we have the need to communicate with them, and the recent tendency is to try to do it by natural means, like through the use of speech. We need to implement methods to communicate with machines, by developing friendly interfaces. In order to understand human oral expressions by mean of machines, they have to perform speech recognition. One way to do it is by first performing speech segmentation and later recognizing the found segments. The continuous speech recognition process is highly dependent of the segmentation process, being a crucial factor for automatic speech recognition (ASR) systems. We need to develop methods with features aimed to increase the performance and speed, such as the use of reduced speech units to be treated; reduced amount of information extracted from the speech, and simplified processing. Currently, many speech recognition systems are using phoneme like units because they bring the following advantages: phonemes are linguistically well defined units and can be looked up easily on a dictionary; pronunciation variability due to linguistic context, accent or dialogues can be easily represented by applying rules to basic forms; the number of units is small; and the phonemes require significantly less data to train than would be needed for whole word modeling [1]. There have been reported phoneme segmentation methods; with acceptable results, but with some of the following restrictions imposed: restricted vocabulary [1], speaker dependency [2], isolated words [3], and text dependency [4]. There are some

J.F. Martínez-Trinidad et al. (Eds.): CIARP 2006, LNCS 4225, pp. 437 – 445, 2006.

reported works, with complex processing based on rules derived from acoustic phonetic knowledge for phoneme segmentation [5, 6, 7, 8]. Recently, new methods have been proposed for phoneme segmentation, without the restrictions mentioned above but having to deal with over-segmentation and text-independency. They have minimized the complexity of the processing; however, they use speech feature vectors or set of features per sequence of time [2, 9, 10,11] and a post processing in order to reduce the insertion rate, but increasing the response time. The feature of speech commonly used with success in recognition and segmentation are LPC, and bank filter of models like MFCC, PCBF (Perceptual Critical Band Features) to mention some. Nonetheless, the encoded representation given on vectors is extracted from basic features presented on time domain.

The main goal of this work is to develop a phoneme speech segmentation algorithm with text independency and low computational cost, that can obtain high phoneme boundary detection rates without over segmenting, using a simplified approach on the feature extraction, segmentation process, fuzzy pattern and distance measure.

We tested the utility of basic features, like intensity, to get phoneme segmentation, and we found similar performance to the one reported on the state of art without presenting over-segmentation. It is important to remark that the intensity is one basic feature easily obtained from speech with light processing. Two remarkable issues of the proposed method are; the reduced information and the simple rules used, obtaining with them an almost real time phoneme segmentation. The testing was done over the same corpus and under similar conditions to the used in [9, 10].

2 The Auditory System Model

The human ear is able to perceive a range of frequency between 20 and 20000 Hz approximately. Since the speech wave is composed by many frequencies, these are not perceived with the same sensitivity. The high and low frequencies are perceived with less intensity. In general, following the Bark scale, the variations of sensitivity below 1000 Hz follow a constant variation with 100 Hz. bands, and when the frequency increases above the 1000 Hz, the sensitivity of frequencies follows a logarithmic scale. These obtained scales, show that low frequencies are increased significantly between 100 and 1000 Hz, and are based on the hearing functioning.

Methods to obtain cepstral vectors using filter banks have been developed, like the previously mentioned, which model the hearing functioning. In recent phoneme segmentation algorithms, the features of speech have been extracted by following some model of the auditory system, having an encoded speech in form of time sequence vectors, which has been taken in [10] as a constraint. Depending of the codification scheme used, the feature extraction might involve a time consuming process, and might obtain too much information to be treated. Doubtlessly some of these schemes like MFCC have reported success on speech recognition, because they give a detailed representation on the speech wave. For segmentation these details are not totally necessary, to show this we have used only scalar values of intensity per sequence of time.

3 The Basic Features of Speech

The Sound is composed of waves of pressure variations that oscillate from positive to negative relative to the surrounding medium, usually the air. The number of air pressure oscillations per second, determines the pitch of the sound, whose physical correlate is frequency. The amplitude is defined like the pressure applied by the vibration, on the elastic mean. If we have a sine wave, the *y*-value for any given *x*-value is the amplitude of the sine wave at that point in time. The amplitude is given in Pascal (Pa) units. On the other hand, the intensity is the size of the pressure vibrations determining the loudness of the sound. Acoustic scientists measure the intensity in a base 10 logarithmic scale called decibels (dB) [13]. The term intensity is used to refer to the overall power of a sound.

The phonemes used in words, have different intensity, for example, most aperture of the mouth is required for relative long time in order to pronounce vowels, releasing most energy and resulting in high intensity, in contrast with the intensity of the majority of consonants, and although plosives have high intensity their duration is very short.

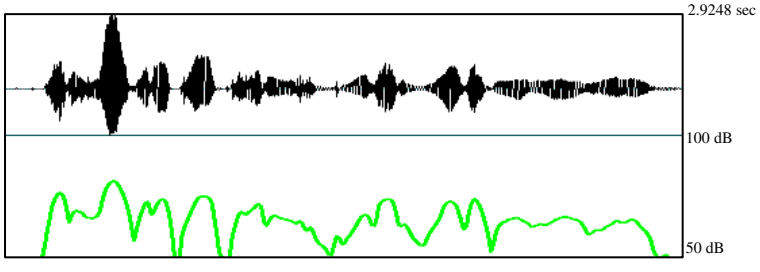

Fig. 1. Speech waveform and its respective intensity

Basic features as zero crossing rate, energy and pitch have been used in previous works in order to obtain sentence segmentation without speech recognition; details of one of those works are presented in [15]. The authors remark its accuracy comparable to methods using speech recognition but to a lower computational cost. Our work is oriented in a similar way, it is, to obtain phoneme segmentation based only on intensity changes without phoneme recognition.

4 The Fuzzy Algorithm

Some kind of speech codification scheme like the previously mentioned was avoided; instead, minimal information like the obtained from the intensity was used. In order to carry out the phoneme segmentation, we implemented a fuzzy distance measure between contiguous frames, and a set of simples rules aimed to detect significant distances, which could be tried like phoneme transition changes on continuous speech. We took advantage of fuzzy memberships of the intensity in order to obtain details on cases where the differences between frames are vague. Different from

other approaches [2], this fuzzy algorithm does not require any training and has a reduced computational load due to its simplicity. The details are next described.

4.1 Preprocessing

First, the pre-emphasis filter on the speech signal was applied. The pre-emphasis filter gives a resultant sound with a high spectral slope. A frequency *F,* above which the spectral slope will be increased by 6 dB/octave, is given. The pre-emphasis factor α is computed as:

$$\alpha = \exp\left(-2\pi F \Delta t\right) \tag{1}$$

Where Δt is the sampling period of the sound. The resultant sound y_i is obtained with (2).

$$y_i = x_i - \alpha x_{i-1} \tag{2}$$

Where every sample x_i of the sound is changed, going down from the last sample [14]. According to the acoustical theory, in order to approximate the unequal sensitivity of human hearing at different frequencies [12], pre-emphasis process is used. In contrast, with the smoothing preprocessing techniques, we can enhance certain frequency intervals from another ones containing less relevant information. In our case, the pre-emphasis filter setting 50 Hz to the *F* argument in (1) was applied.

4.2 Phoneme Segmentation

The pre-emphasized signal is used in order to obtain the intensity with a minimal pitch of 93 Hz, and 3 msec. frames without overlapping were used. We also tested 4 and 5 msec. frame size, the results are shown in the experiments section.

For each signal, the maximum and minimum intensity were obtained, in order to establish the fuzzy space. The average between the maximum and minimum intensity, in order to obtain the medium point of the fuzzy space, was calculated. Three triangular fuzzy functions, representing low, median and high intensity, were applied. Membership values from the fuzzy sets are obtained for each intensity measure, and then they are normalized as follows: $\lambda = max(M)$, where M represents the fuzzy membership obtained from the compared frames. The maximum fuzzy membership denoted as λ is obtained. Then $\mu_{i=} \mu_i / \lambda \ \forall \mu_i \in M$ is applied. Since our strategy is based on the difference between contiguous frames, in order to detect phoneme boundaries, the normalized memberships are used in (3), and we denote the values obtained as *V*.

$$D(f_t, f_{t-2}) = \sqrt{\left(\mu_{\text{high}}(f_t) - \mu_{\text{high}}(f_{t-2})\right)^2 + \left(\mu_{\text{mid}}(f_t) - \mu_{\text{mid}}(f_{t-2})\right)^2 + \left(\mu_{\text{low}}(f_t) - \mu_{\text{low}}(f_{t-2})\right)^2} \tag{3}$$

Our approach is simple, because we focus on detecting the local maxima on the V values, which indicate the significant differences between the compared frames, and, therefore, the presence of a phoneme boundary. The rules used to detect the local maxima are the following:

1) $V_t > V_{t-1}$ & $V_t > V_{t+1}$
2) $V_t > 46.8$ *dB*
3) $V_t > \Phi$

In condition 1) a V value at time t is treated as local maximum if it is greater than the previous and following V value on the sequence of time. Condition 2) is used to discriminate potential boundary values, because low intensity V_t values generally are not representative of phoneme boundaries. Condition 3) selects local maxima representing significant changes, when they are over the threshold Φ. The last two conditions are used in order to avoid unwanted insertions; although some valid phoneme boundaries are incorrectly discarded by them. On the other hand, erroneous points detected by the algorithm are rejected, resulting in a competitive performance. A limitation of this algorithm is that segments shorter than 0.021 msec are not allowed, this condition reduces the over segmentation problem and, at the same time, some of the valid phoneme boundaries are sacrificed too.

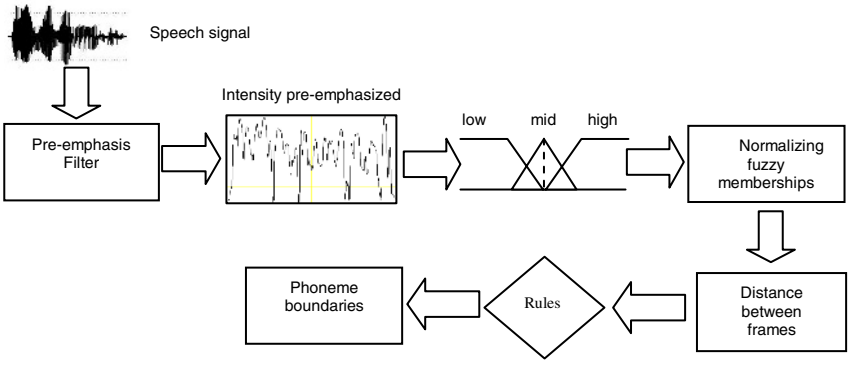

Fig. 2. Algorithm diagram

5 Implementation and Experiments

The feature extraction and segmentation processing was implemented using the freeware PRAAT v.4.4.04 [14].

For the experiments we used continuous speech expressed naturally, and with text and speaker independency. The algorithm was tested with 544 speech signals sampled at 16 kHz of the American English DARPA-TIMIT database, corresponding to 68 speakers (34 males and 34 females) of all dialect regions. The phoneme segmentation performance of the algorithm was compared with the true phoneme boundaries obtained from the transcription associated to the speech signals, which was made manually by phonetician experts. The subset of sentences has a total of 20647 phonetic boundaries.

5.1 Measurement Performance

The performance of the algorithm was evaluated with commonly used measures, like the one used in [9, 10, 11].

$$D = 100 \cdot \left(\frac{S_d}{S_t} - 1 \right) \tag{4}$$

Where D is the measure of over-segmentation, S_d is the number of segmentation points detected by the algorithm, and S_t is the number of the true segmentation points.

$$P_c = 100 \cdot \left(\frac{S_c}{S_t} \right) \tag{5}$$

Where P_c is the percentage of correct detection, and S_c is the number of correct segmentation points. The segmentation points detected by the algorithm are defined as correct if its distance from the true segmentation point is within the range of ± 20 msec.

5.2 Experiments and Results

Many changes aimed to improve, were implemented on the algorithm. Results of different algorithm versions are shown, in order to remark the effects of the experimented modifications. The first version of our algorithm, was using a frame size of 5msec, without normalized fuzzy memberships, and using the measure computed by (6), which obtains the distance between adjacent frames. The results are shown in table 1.

When the frame size was 5 msec without normalized fuzzy memberships, a poor performance was observed, and while the frame size was reduced the performance was slightly improved. The best observed performance was obtained when a frame size of 3 msec was used. That is why, in the remaining experiments, a 3 msec frame size was used.

$$D(f_t, f_{t-1}) = \sqrt{\left(\mu_{\text{high}}(f_t) - \mu_{\text{high}}(f_{t-1})\right)^2 + \left(\mu_{\text{mid}}(f_t) - \mu_{\text{mid}}(f_{t-1})\right)^2 + \left(\mu_{\text{low}}(f_t) - \mu_{\text{low}}(f_{t-1})\right)^2} \tag{6}$$

Table 1. Algorithm performance, with different frame size and its respective parameters

Size	Φ	Sd	Sc	% Correct Detection	% Over Segmentation
5 msec	0.040	20946	15046	72.87	1.44
4 msec	0.036	21039	15353	74.35	1.89
3 msec	0.032	20910	15485	74.99	1.27

Another remarkable improvement was when the fuzzy memberships were normalized, which increased the correct detection rate and reduced over segmentation to near to 0%. The results are shown in table 2.

Table 2. Algorithm performance, using normalized fuzzy memberships

Φ	Sd	Sc	% Correct Detection	% Over Segmentation
0.050	20642	15567	75.39	-0.02

When the distance between adjacent frames is being used, many phoneme changes are not detected, because their significant difference does not appear in adjacent

frames. So a frame distance between compared frames was used in order to detect those slowly reflected changes, applying (3). The frame distance between compared frames is referred in this work as inter-frame. Finally, we used a minimum intensity of 25 dB instead the relative minimum intensity of each signal, in order to obtain the fuzzy space, increasing the performance. The results are shown in table 3.

Table 3. Algorithm performance, using inter frame

Φ	Sd	Sc	% Correct Detection	% Over Segmentation
0.0855	20668	16010	77.54	0.09

This modification to the distance measure and the minimum intensity of the fuzzy space, results on a correct detection increased above of 2 %, maintaining a percentage of over segmentation of near to 0%. Starting, in the preliminary experiments, with a correct detection performance of near to 73% and an over segmentation rate above of 1%, with simple modifications like the frame size, normalization of fuzzy membership functions, using an inter frame between compared frames and a minimum intensity of 25 dB in the fuzzy space, we improved the correct detection rate to near to 5%, and the over segmentation rate was reduced in more than 1%.

5.3 Comparison with Similar Works

The results are compared to algorithms alike [9,10,11], dealing with text and speaker independency, with over segmentation and some other previously mentioned conditions. Relevant aspects like amount of phoneme boundaries treated, information extracted from the speech, number of speakers and percentage of correct detection are used for the comparison, they are shown in table 4.

Table 4. Algorithm comparatives on different issues

Used features	Extracted values per 20 msec	Treated Speaker	Treated Phoneme Boundaries	% Correct detection
Intensity	6.66	68	20647	77.54
PCBF [9]	15	48	17930	73.56
Mel spectrum [10]	8	48	17930	76.53
PCBF [11]	15	20	6200	75.80

In the first row, significant aspects of our algorithm are presented, the remaining rows contain the reported aspects of the mentioned algorithms in the listed order as presented in [9, 10, 11]. The compared algorithms use feature vectors on frames of 20 msec, with 10 msec overlapping; and the “jump” term is used to denote significant changes (peaks) between compared frames; algorithms [10,11] are modifications of the algorithm presented in [9].

These algorithms have a fundamental process to combine, in a unique indication of phoneme boundary, the “jump” events detected around the same frame. The process is called “fitting” and was introduced to place the segmentation boundary in the

middle of a cluster of quasi-simultaneous "jumps". The fitting process is used after the "jump" detection.

On the other hand, our approach use scalar values per time sequence, and no fitting process is used. Although the difference in number of features used among the algorithms is insignificant, we are not using overlapping frames, and our process to extract the features is remarkably simple and effective. Generally, the compared and the proposed algorithm present difficulties to detect vowel-vowel phoneme boundaries, and, specifically the proposed algorithm rejects those consecutive correct boundaries separated by only 0.021 msec.

7 Conclusions

The proposed phoneme segmentation algorithm has shown some advantages over others due to its lower computational cost in the extraction of features and boundaries detection. Our approach achieves competitive performance with fast execution due to the reduced information used and its simplified processing. In the preprocessing phase, only a pre-emphasis filter to enhance spectral changes was used. The strategy of using a fuzzy distance measure between frames shows to be simple and effective. The use of fuzzy normalized membership in an Euclidean distance, in order to obtain details of vague phoneme boundaries difficult to detect, lead to an increase in the overall algorithm performance. The use of inter frames was useful to detect phoneme boundaries, which present slow changes. The algorithm detected 77.54% of the boundaries without over segmentation. As future works we will try to improve the performance by enhancing the algorithm with more efficient strategies and rules.

References

1. Hu Z., Schalwyk J., Bernard E., Cole R., "Speech recognition using syllabe-like units", ICSLP '96, 2: 1117—1120, 1996.
2. Suh Y. and Lee Y., "Phoneme Segmentation of Continuous speech using multi-layer perceptron", IEEE Trans. Speech and Audio Proc., 7(6):697-708, 1999.
3. Ratsameewichai S., Theera N., Vilasdechanon J., Uatrongjit S., and Likit-Anurucks K., "Thai phoneme segmentation using dual-band energy contour", ITC- CSCC-2002.
4. Pellom B., Hansen J., "Automatic segmentation of speech recorded in unknown noisy channel characteristics", Speech Communication, 1998, 25, 97-116.
5. Schwartz R. And Makhoul J., "Where the Phonemes Are: Dealing with Ambiguity in Acoustic Phonetic Recognition", IEEE Trans. ASSP, Vol. 23, pp 50-53, Feb. 1975.
6. Zue V., "The Use of Speech Knowledge in Automatic Speech Recognition", Proceeding of the IEEE, Vol. 73, pp. 1602-1615, Nov. 1985.
7. Weinstein C., McCandless S., Mondstein L., and Zue V., "A system for Acoustic-Phonetic Analysis of Continuous Speech", IEEE Trans. ASSP, Vol. 23, pp. 54-67, Feb. 1975.
8. Grayden D. and Scordilis M., "Phonemic Segmentation of Fluent Speech, "Proc. ICASSP-94, pp. 73-76, 1994.
9. Aversano G. and Esposito A., "A new text-independent method for phoneme segmentation". in Proc. the 44th IEEE Midwest Symposium on Circuits and Systems, vol. 2, pp. 516--519, 2001.

10. Aversano G. and Esposito A., “Automátic Parameter Estimation for a Context-Independent Speech Segmentation Algorithm", TSD 2002, LNAI 2448, pp. 293-300, 2002 Springer Verlag Berlin Heidelberg 2002
11. Saraswhati S., Geetha T.V, and Saravanan K., “Integrating Language Independent Segmentation and Language Dependent Phoneme Based Modeling for Tamil Speech Recognition System”, Asian Journal of Information Technology 5 (1) : 38-43, 2006.
12. Gold B., Morgan N., “Speech and audio signal processing”, John Wiley & Sons Inc., 2000
13. Rodman R., “Computer Speech Technology”, Artech House Inc., 1999.
14. Boersma, P. “Praat, a system for doing phonetics by computer”. Glot International 5:9/10, 341-345.
15. Wang D., Le L., Zhang H., “Speech segmentation without speech recognition”. Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP03), Vol. I, pp. 468-471, Hong Kong, 2003.
16. Petek B., Andersen O., Dalsgaard P., “On the robust automatic segmentation of spontaneous speech”, in proceedings of ICSLP’96, 1996, pp. 913-916.