ARCHIVES OF ACOUSTICS
36, 1, 29–47 (2011)

DOI: 10.2478/v10168-011-0003-2

Phoneme Segmentation Based on Wavelet Spectra Analysis

Bartosz ZIÓŁKO[(1)], Suresh MANANDHAR[(2)],
Richard C. WILSON[(2)], Mariusz ZIÓŁKO[(1)]

[(1)] *AGH University of Science and Technology*
Department of Electronics
Al. Mickiewicza 30, 30-059 Kraków, Poland
e-mail: {bziolko,ziolko}@agh.edu.pl

[(2)] *University of York*
Department of Computer Science
Heslington, YO10 5DD, York, UK
e-mail: {suresh,wilson}@cs.york.ac.uk

(*received February 4, 2010; accepted October 19, 2010*)

A phoneme segmentation method based on the analysis of discrete wavelet transform spectra is described. The localization of phoneme boundaries is particularly useful in speech recognition. It enables one to use more accurate acoustic models since the length of phonemes provide more information for parametrization. Our method relies on the values of power envelopes and their first derivatives for six frequency subbands. Specific scenarios that are typical for phoneme boundaries are searched for. Discrete times with such events are noted and graded using a distribution-like event function, which represent the change of the energy distribution in the frequency domain. The exact definition of this method is described in the paper. The final decision on localization of boundaries is taken by analysis of the event function. Boundaries are, therefore, extracted using information from all subbands. The method was developed on a small set of Polish hand segmented words and tested on another large corpus containing 16 425 utterances. A recall and precision measure specifically designed to measure the quality of speech segmentation was adapted by using fuzzy sets. From this, results with F-score equal to 72.49% were obtained.

Keywords: speech recognition, speech segmentation, discrete wavelet transform.

1. Introduction

Speech signals typically need to be divided into small segments before starting a recognition procedure. Analysis of these frames can determine the likelihood of

a particular phoneme being present within the frame. Speech is non-stationary in the sense that frequency components change continuously over time, but it is generally assumed to be a stationary process within a single frame. Naturally, this causes recognition difficulties if the frame contains the end of one phoneme and the beginning of another. Segmentation methods currently used in speech recognition do not consider where phonemes begin and end. Uniform segmentation causes conflicting information to appear at the boundaries of phonemes. For more accurate modeling, non-uniform phoneme segmentation can be useful in speech recognition (Glass, 2003).

Phonetic segmentation can be also successfully used in automatic labeling of time-aligned data (eg. subtitle cues generation) (Cardinal *et al.*, 2005) and information retrieval from temporal data. Automatic segmentation of speech corpora can be used in unit-selection speech synthesis (Hunt and Black, 1996).

A phoneme segmentation method presented in this paper is more sophisticated than that described in (Ziółko *et al.*, 2006), as more scenarios are covered and the results are evaluated in a better way. Results were obtained from a much larger corpus. Our method is based on analyzing the envelopes and the rate-of-change of the Discrete Wavelet Transform (DWT) subband power.

The outline of the paper is as follows. Section 2 describes several possible approaches to phoneme segmentation. Section 3 presents some rudiments of the DWT. In Sec. 4, the general idea of our segmentation approach is described. Section 5 contains the exact algorithm and its explanation. Details of all scenarios are presented in Sec. 6. The data used in the experiment are described in Sec. 7. The evaluation method is explained in Sec. 8 along with the reasons for which it was used. Finally, the results are commented in Sec. 9. The paper is summed up with conclusions.

2. Phoneme segmentation

Constant-time segmentation, i.e. framing, for example into 23.2 ms blocks (Young, 1996), is commonly used to divide the speech signal for processing. This method benefits from the simplicity of implementation and easy comparison of blocks, which are of the same length. However, it is perceptually unnatural because the duration of phonemes varies significantly.

In fact, human phonetic categorization is also very poor for such short segments (Morgan *et al.*, 2005). Moreover, boundary effects provide additional distortions (which are partially reduced by applying the Hamming window), and such short segments create many more boundaries than there are phonemes in the speech. The boundary effects can cause errors in speech recognition. Additional difficulties appear because of the mixing of two phonemes in a single frame. A smaller number of boundaries means a smaller number of errors because of the aforementioned effects. Therefore, constant segmentation, though

straightforward, risks losing valuable information about the phonemes because of the merging of different sounds into a single block. Moreover, the complexity of individual phonemes cannot be represented in short frames. The important advantage of nonuniform segmentation rely on that the length of a phoneme can also be used as an additional parameter in speech recognition, improving the accuracy of the whole process. A comparison of constant framing and phoneme segmentation is presented in Fig. 1.

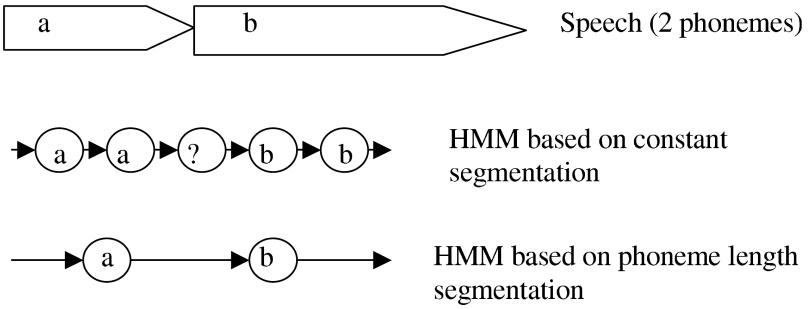

Fig. 1. Comparison of the frames produced by uniform segmentation and segmentation that results on phoneme length.

Models based on processing information over long time ranges have already been introduced. The RASTA (RelAtive SpecTrAl) methodology (Hermansky, Morgan, 1994) is based on relative spectral analysis and the TRAPs (TempoRAl Patterns) approach (Morgan *et al.*, 2005) is based on multilayer perceptrons with the temporal trajectory of logarithmic spectral energy as the input vector. It allows one to generate class posterior probability estimates.

A number of approaches have been previously suggested (Stöber, Hess, 1998; Grayden, Scordilis, 1994; Weinstein *et al.*, 1975; Zue, 1985; Toledano *et al.*, 2003) to find phoneme boundaries from the time-varying speech signal properties. These approaches utilize features derived from acoustic knowledge of the phonemes. For example, the solution presented in (Grayden, Scordilis, 1994) analyzes a number of different subbands in the signal using its spectra. Phoneme boundaries are extracted by comparing the percentage of signal power in different subbands. The Toledano *et al.* (2003) approach is based on spectral variation functions. Such methods need to be optimized for particular phoneme data and cannot be performed in isolation from phoneme recognition itself. Neural Networks (NNs) (Suh, Lee, 1996) have also been tested, but they require time-consuming training.

Segmentation can be applied by the Segment Models (SMs) instead of the Hidden Markov Models (HMMs) (Ostendorf *et al.*, 1996; Russell, Jackson, 2005). The SM solution differs from HMM by searching paths through sequences of segments of different lengths rather than frames. Such a solution means that

segmentation and recognition are conducted at the same time and there is a set of possible observation lengths. In a general SM, the segmentation is associated with a likelihood and in fact describes the likelihood of a particular segmentation of an utterance. The SM for a given label is also characterized by a family of output densities, which gives information about observation sequences of different lengths. These features of SM solution allow the location of boundaries only at several fixed positions, which are dependent on framing (i.e. on an integer multiple of the frame length).

The typical approach to phoneme segmentation for creating speech corpora is to apply the dynamic programming (Rabiner, Juang, 1993; Holmes, 2001). The dynamic programming is a tool that guarantees one to find the cumulative distance along the optimum path without having to calculate the distance along all possible paths. In speech segmentation, it is used for time alignment of boundaries. The common practice is to provide a transcription done by professional phoneticians for one of the speakers in the given corpus. Then, it is possible to automatically create phoneme segmentation of the same utterances for other speakers. This method is very accurate, but demands transcription and hand segmentation to start with. For this reason, it is not very useful for any other application than creating a corpus.

3. Analysis using the discrete wavelet transform

The human hearing system plays a role of a frequency-processing system in the first step of sound analysis. While the details are still not fully understood, it is clear that a frequency-based analysis of speech reveals important information. This encourages us to use DWT as a method of speech analysis, since the DWT may work more similarly to the human hearing system than other methods (Wang, Narayanan, 2005; Daubechies, 1992).

Details of the wavelet transform are beyond the scope of this paper, but a brief overview of the method is presented. The wavelet transform provides a time-frequency analysis. The original speech signal $s(n)$ and its wavelet spectrum are of 16 bits accuracy. To obtain DWT (Daubechies, 1992), the coefficients of the approximation of signal $s(n)$ as series

$$s_{m+1}(n) = \sum_i c_{m+1,i}\, \phi_{m+1,i}(n) \tag{1}$$

which is the approximation of signal $s(n)$, are computed, where $\phi_{m+1,i}$ is the i-th wavelet function at the $(m+1)$-th resolution level. Thanks of the orthogonality of wavelet functions approximation

$$c_{m+1,i} = \sum_{n \in D_i} s(n)\phi_{m+1,i}(n) \tag{2}$$

is used, where D_i are supports of $\phi_{m+1,i}$. The coefficients of the lower level are calculated by applying the well-known (DAUBECHIES, 1992; RIOUL, VETTERLI, 1991) formulae

$$c_{m,k} = \sum_i h_{i-2k}\, c_{m+1,i}, \tag{3}$$

$$d_{m,k} = \sum_i g_{i-2k}\, c_{m+1,i}, \tag{4}$$

where h and g are the constant coefficients, which depend on the scale function ϕ and wavelet ψ (e.g. functions presented in Fig. 2). The speech spectrum is decomposed by using digital filtering and downsampling procedures defined by Eqs. (3) and (4). It means that given the wavelet coefficients $c_{m+1,i}$ of the $(m+1)$-th resolution level, Eqs. (3) and (4) are applied to compute the coefficients of the mth resolution level. The elements of the DWT for a particular level may be collected into a vector, for example $\mathbf{d}_m = (d_{m,1}, d_{m,2}, \ldots)^{\mathrm{T}}$. The coefficients of other resolution levels are calculated recursively by applying formulae (3) and (4). The multiresolution analysis gives a hierarchical and fast scheme for the computation of the wavelet coefficients for a given speech signal s. In this way, the values

$$\mathrm{DWT}(s) = \{\mathbf{d}_M, \mathbf{d}_{M-1}, \ldots, \mathbf{d}_1, \mathbf{c}_1\} \tag{5}$$

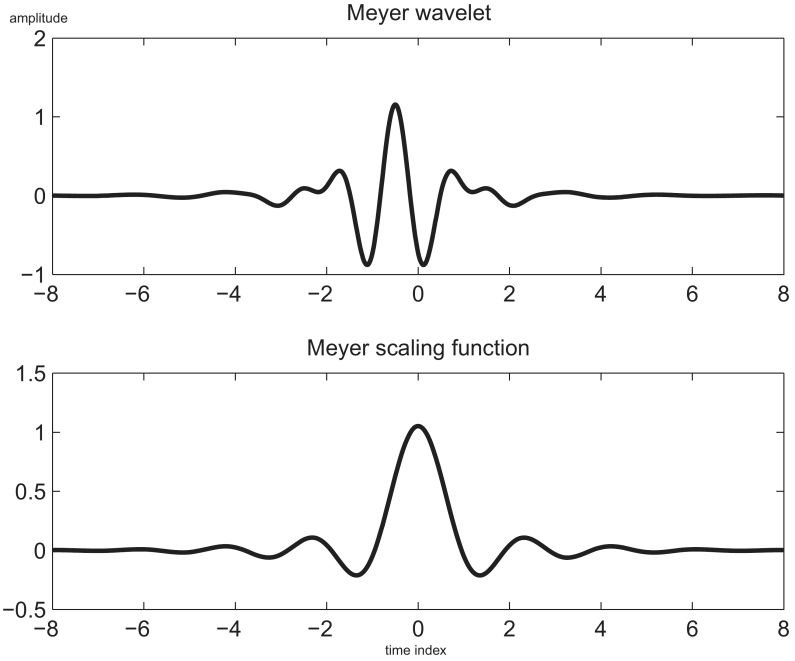

Fig. 2. The discrete Meyer wavelet $\psi(n)$ and its scale function $\phi(n)$.

of the DWT for M+1 levels are obtained. Each signal

$$s_{m+1}(n) = s_m(n) + s_m^d(n) \text{ for all } n \in Z \tag{6}$$

on the resolution level m+1 is split into approximation (coarse signal)

$$s_m(n) = \sum_k c_{m,k}\phi_{m,k}(n) \tag{7}$$

on the lower m resolution level and high frequency details

$$s_m^d(n) = \sum_k d_{m,k}\psi_{m,k}(n), \tag{8}$$

where $\phi_{m,k}(n) = 2^{m/2}\phi(2^m n\Delta t - k)$ and $\psi_{m,k}(n) = 2^{m/2}\psi(2^m n\Delta t - k)$ and Δt is sampling density. The frequency density Δf depends on the support of wavelet $\phi(t)$ defined in the continuous domain. For the case when wavelet support

$$\text{supp } \phi(t) = \{\overline{t : \phi(t) \neq 0}\} \tag{9}$$

is compact (let us denote its width by $2T$), we obtain $\Delta f = 0.5/T$. In practice the support can be always limited to the segment $[-T, T]$, where

$$T = \max\{t \in \mathbb{R} : |\phi(t)| \geq h\}. \tag{10}$$

The threshold h should depend on the extreme value of the scale function, e.g $h = \max_t |\phi(t)| /1000$. In that way, the support of scale function was bounded to obtain the reasonable compromise: fast computations in real time and relatively small errors.

The number of samples should be the smallest integer value N which satisfies inequality $(N-1)\Delta t \geq 2T$, that is $N \geq 1 + 32000T$ because the sampling frequency $f_s = 1/\Delta t = 16000$ [Hz]. The sampling density in the frequency domain $\Delta f = 0.5/T$ and $(N-1)\Delta f \geq 16000$ [Hz] because the whole frequency band is spread from -8000 to 8000 [Hz].

The wavelet transform can be viewed as a tree. The root of the tree consists of the coefficients of wavelet series (1) of the original speech signal. The first level of the tree is the result of one step of (4). Subsequent levels in the tree are constructed by recursively applying (3) and (4) to split the spectrum into the low (approximation $c_{m,n}$) and high (detail $d_{m,n}$) parts. Experiments undertaken by us show that the speech signal decomposition into $M = 6$ levels is sufficient (see Fig. 3) to cover the frequency band of a human voice (see Table 1). The energy of the speech signal above 8 kHz and below 125 Hz is very low and can be neglected. There is a wide variety of possible basis functions from which a DWT can be derived. To determine the optimal choice of wavelet, we analyzed six different wavelet functions: *Meyer* (Fig. 2), *Haar, Daubechies wavelets* of three different orders and *symlets*. Our results, described in Sec. 9, show that the discrete *Meyer wavelet* gives the best results.

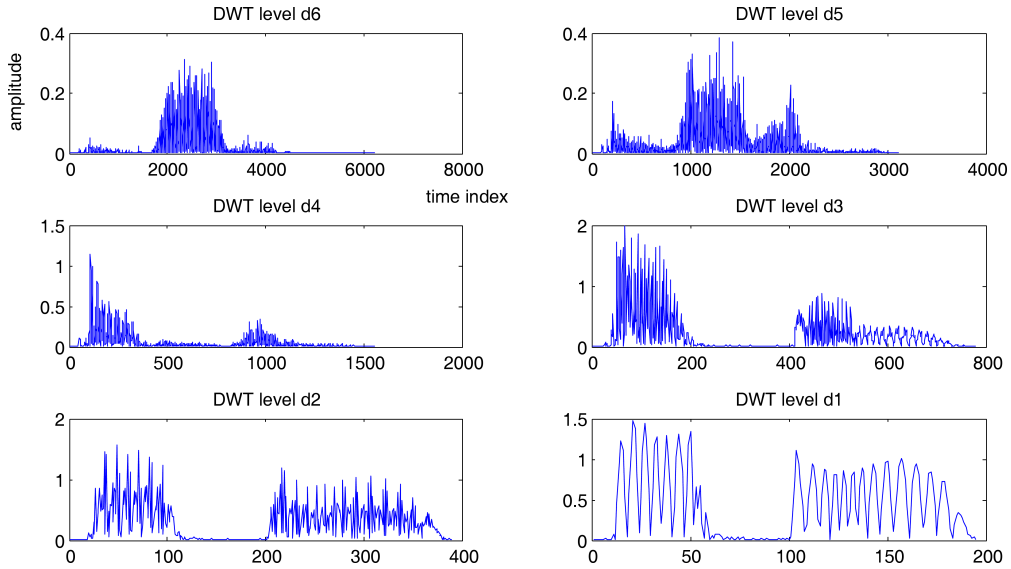

Fig. 3. Subband amplitude DWT spectra of the Polish word 'osiem' (Eng. eight). The number of samples depends on a resolution level.

Table 1. Bandwidths of the DWT levels and widths of their envelopes.

Level	Band (kHz)	No. of samples	Window
d_6	4–8	32	5
d_5	2–4	16	5
d_4	1–2	8	5
d_3	0.5–1	4	3
d_2	0.25–0.5	2	3
d_1	0.125–0.25	1	3

4. Principles

Phonemes are characterized by frequency content, so we would expect changes in the power of wavelet resolution levels between phonemes. Clearly, it would be easier to analyze the absolute value of the rate-of-change of power and expect it to be large at the beginning and at the end of phonemes. However, this does not uniquely define start and end points for two reasons. Firstly, the power can rise over a considerable length of time at the start of a phoneme, leading to an ambiguous start time. Secondly, there may also be rapid changes in power in the middle of a segment. A better method of detecting the boundary of phonemes relies on power transitions between the DWT subbands.

The amount $2^{-M+m-1}N$ of wavelet spectrum samples in the mth level (where $m = 1, \ldots, M$) depends on the length N of the speech signal in time domain,

assuming N is a power of 2. Table 1 presents their number at each level according to the lowest resolution level. The power waveform

$$p_m(n) = \sum_{j=1}^{2^{m-1}} d^2_{m,j+n2^{m-1}} \quad \text{where} \quad n = 0, \ldots, 2^{-M}N-1, \tag{11}$$

is computed in a such way that the equal number of power samples for each m-level decomposition is obtained.

The DWT subband amplitude shows rapid variations (see Fig. 3) and despite smoothing (11), the power waveforms still change rapidly. It makes the first-order differences in the power inevitably noisy, so we calculate the envelopes $p^{en}_m(n)$ for power fluctuations in each subband (Fig. 4) by choosing the highest values of $p_m(n)$ in a window of given size ω (see Table 1). Next, a smoothed differencing operator was used, the subband power p_m is convolved with the mask $[1, 2, -2, -1]$ to obtain smoothed power rate-of-change $r_m(n)$.

To improve accuracy, a minimum threshold $p_{\min}$ was introduced for a subband DWT power. This threshold was chosen experimentally as 0.0002 for the test corpus. This prevents us from analyzing noise where the power of the speech signal is very small (i.e. in the areas of 'silence'), even though noise is very low in the test corpus. Threshold $p_{\min}$ can be set based on the power of noise. The start and end of a phoneme is usually marked by an initially small but rapidly rising power level in one or more of the DWT levels. In other words, the derivative can be expected to be approximately as large as the power. This is why phoneme boundaries can be detected searching for n-points for which the inequality

$$p \geq |\beta|r_m(n)| - p^{en}_m(n)| \tag{12}$$

holds. Constant p is a value of threshold that accounts for the time scale and sensitivity of the crossing points. We found that setting the threshold $p = 0.1$ gave the best results. A power and its derivative have different physical units. This is why the rate-of-change function r_m is multiplied by scaling factor β, approximately equal to 1 [s], to subtract the power from the product $\beta|r_m(n)|$.

5. Phoneme detection algorithm

Without any additional refinement, the method presented here may not be able to detect the phoneme boundaries precisely. There are several reasons for this. Firstly, the exact locations of the boundaries may vary slightly between subbands. For some phonemes, only one frequency band may show significant variations in power; for others, a few of them may do so. Sometimes, analysis will detect slightly separate boundaries for different subbands. Secondly, despite smoothing the derivative, there may be a number of transitions going up and down, which represent the same boundary. This problem was approached by holding indexes of situations, which are very likely to happen for phoneme

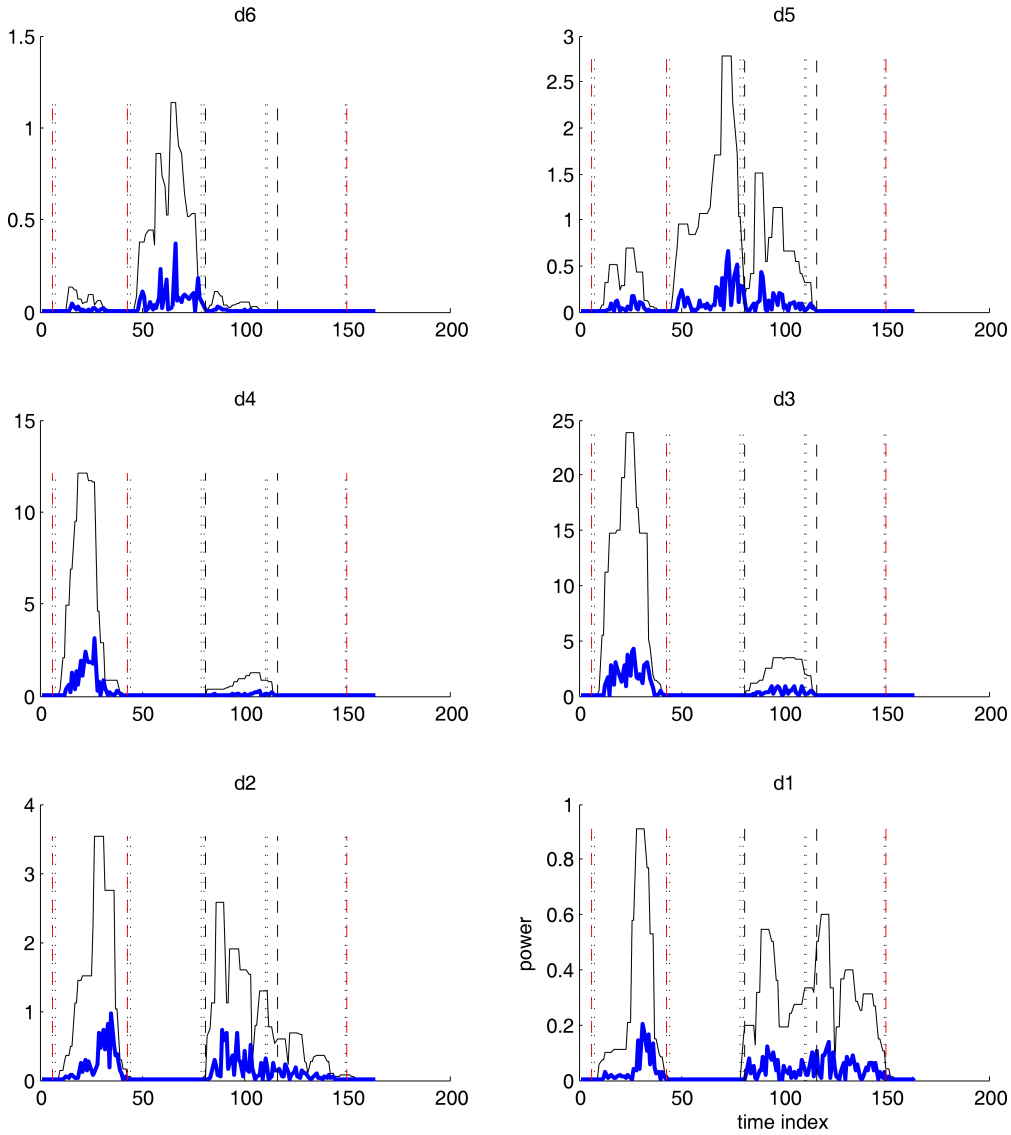

Fig. 4. Segmentation of the Polish word 'osiem' (Eng. eight) based on DWT subbands. Dotted lines are hand segmentation boundaries; dashed lines are automatic segmentation boundaries, thin lines are envelopes and bold lines are smoothed rate-of-change.

boundaries, using event function $e(n)$. Such an approach enables one to consider several scenarios and aspects of potential phoneme boundaries. It also allows for improving the method easily by adding additional events to the existing list.

The suggested events are presented in Table 2 and explained in detail later. Surprisingly, pre-emphasis filtering was found to deteriorate quality, thus it was

not used in the final version of the algorithm. The algorithm steps are listed below:

1. Normalize a speech signal by dividing it by its maximum value in an analyzed fragment of speech.
2. Decompose a signal into six levels of the DWT.
3. Calculate the sum (11) of power samples in all frequency subbands to obtain the power representations $p_m(n)$ of the mth subband.
4. Calculate the envelopes p_m^{en} (Fig. 4) for power fluctuations in each subband by choosing the highest values of p_m in a window of a given size ω according to Table 1.
5. Calculate the rate-of-change function (Fig. 4) $r_m(n)$ by filtering $p_m(n)$ with $[1, 2, -2, -1]$ mask.
6. Create an event function $e(n) = 0$ for all n. In the next step, the function value will be increased to record events for which $r_m(n)$ and $p_m^{en}(n)$ look like a phoneme boundary for a given n.
7. Analyze $r_m(n)$ and $p_m^{en}(n)$ for each DWT subband to find the discrete time n for which the event conditions described in Table 2 hold. Add the value of the event importance (Table 2) to the event function $e(n)$ (Fig. 5) for a given discrete time n according to Table 2. If several events occur for a single discrete time, then summarize the event importances of all of them. Repeat the step for all discrete times n. In this way, we have a boundary distribution-like function
$$e(n) = \begin{cases} 0 & \text{no condition fullfiled for } n, \\ \sum_i w_i & \text{otherwise,} \end{cases} \tag{13}$$
where w_i are importance weights (see Table 2) for events that occurred for n in all subbands.
8. Find a discrete time n starting from $n = 1$ which the event function is higher than a decision threshold τ. A value of $\tau = 4$ was chosen experimentally.
9. Find all the discrete times t_i for which
$$\begin{cases} e(t_i) > \tau - 1, \\ t_i > n, \\ t_i - t_{i+1} < \alpha, \end{cases} \tag{14}$$
where n is the last index analyzed in the previous step and α is associated with minimal phoneme length ($\alpha = 4$ gives approximately 20 [ms]). Organize all the discrete times t_i in separate groups of those fulfilling the above conditions.
10. Calculate the weighted mean discrete time
$$b = \frac{\sum_i t_i w_i}{\sum_i w_i} \tag{15}$$

for every set of the discrete times t_i grouped in the previous step. Parameter b is the hypothesis of the detected phoneme boundary. Discrete timing of DWT level d_1 is used in the algorithm for all other subbands by summing samples.

11. Repeat the previous three steps for next discrete time values n until the largest n with non-zero value of event function $e(n)$ is obtained.

Table 2. Types of events associated with phoneme boundary. Mathematical conditions are based on power envelope $p_m^{en}(n)$, rate-of-change information $r_m(n)$, a threshold p of the distance between $r_m(n)$ and $p_m^{en}(n)$ and a threshold $p_{\min}$ of minimal $p_m^{en}(n)$ and $\beta = 1$. Event values in the last four columns are for different DWT levels (d_1, d_2, from d_3 to d_5 and for d_6 level).

Description	Mathematical condition	Importance w_i			
Quasi-crossing point	$\lvert\beta\lvert r_m(n)\rvert - p_m^{en}(n)\rvert < p$ AND $(\lvert\beta\lvert r_m(n+1)\rvert - p_m^{en}(n+1)\rvert > p$ OR $\lvert\beta\lvert r_m(n-1)\rvert - p_m^{en}(n-1)\rvert > p)$ AND $p_m^{en}(n) > p_{\min}$	1	3	4	1
Crossing point first case	$\beta\lvert r_m(n)\rvert > p_m^{en}(n) + p$ AND $\beta\lvert r_m(n+1)\rvert < p_m^{en}(n+1) - p$ AND $p_m^{en}(n) > 5\,p_{\min}$	1	3	4	1
Crossing point second case (opposite one)	$\beta\lvert r_m(n)\rvert < p_m^{en}(n) - p$ AND $\beta\lvert r_m(n+1)\rvert > p_m^{en}(n+1) + p$ AND $p_m^{en}(n) > 5\,p_{\min}$	1	3	4	1
Rate-of-change higher than power envelope	$\beta\lvert r_m(n)\rvert > p_m^{en}(n)$ AND $p_m^{en}(n) > 2\,p_{\min}$	1	2	2	1

Table 2 describes the events that can be expected to occur in the power of DWT subbands. Some of them are more crucial than others. In our previously published work (Ziółko *et al.*, 2006), only the first of them was used. Additionally, different weights were given to events with respect to a subband in which

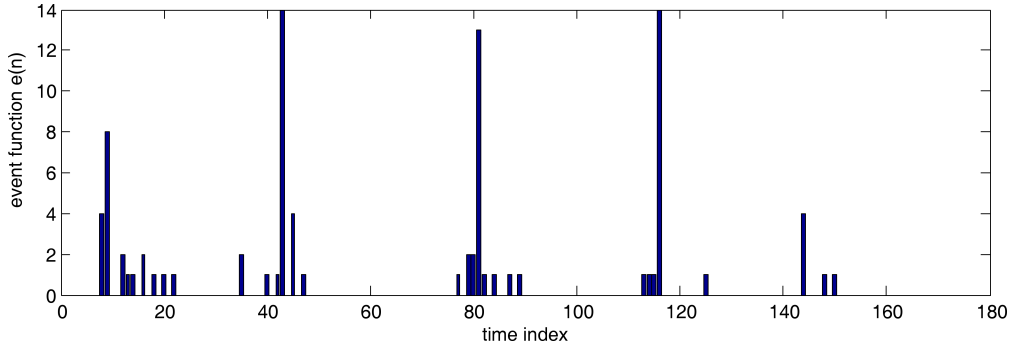

Fig. 5. The event function vs. time (in [ms]) of the word presented in Fig. 4. High event scores mean that a phoneme boundary is more likely.

they occur. It is a perceptually motivated idea, which was very successfully used in PLP (Perceptual Linear Predictive) (HERMANSKY, 1990). As per this study, information in relatively high and low frequency subbands are not so important for the human ear as information in the bands from 345 Hz to 2756 Hz. Briefly, the Hermansky solution (HERMANSKY, 1990) and (HERMANSKY, MORGAN, 1994) used a window to modify speech, decreasing frequencies not crucial for the human ear and amplifying the most important ones. The same aim was achieved in our solution by assigning low weights to events occurring in detectable, but not the most important frequencies, and the higher ones for the most sensitive bands of human hearing system. Six DWT subbands were used. The third, fourth, and fifth were grouped together, as the most crucial ones. As a result, in Table 2 the last four columns with importance values (weights) are presented (the first one for level d_1, the second one for level d_2, the third for the levels from d_3 to d_5 and the last one for level d_6).

6. Segmentation scenarios

There are four possible events presented in Fig. 6 and described in Table 2. Second is a mirrored version of the third one, which will be described in details later. The first one is a weaker condition for a similar scenario as the second and third event. It has to be stressed that for some discrete times and subbands, more than one event can occur (typically two and very rarely more). In these cases, weights of both events are taken into account to the event function $e(n)$. Also, the weights from all subbands are summed. In all cases, the values of rate-of-change information $|r_m(n)|$ are multiplied by scaling factor β equal to 1 [s].

The first event is called quasi-crossing point. It is the most general and common one. The mathematical condition for this event detects discrete times for which the power envelope $p_m^{en}(n)$ and the absolute value of rate-of-change $|r_m(n)|$ cross or approach each other very closely (on a distance of threshold p). Additionally, the power envelope $p_m^{en}(n)$ has to be higher than the threshold $p_{\min}$.

The second and third events are twin events and represent rarer cases, namely the crossing of the power envelope $p_m^{en}(n)$ and the absolute value of rate-of-change $|r_m(n)|$ when $p_m^{en}(n)$ is higher than five times the minimum threshold, i.e. $5\,p_{\min}$. It means that the second and third cases are used to detect and include in consideration more specific situations than the first one, because typically fulfilling one of those conditions means fulfilling the first one as well. As we sum all event importances for a given n, this will cause a higher value of event function $e(n)$ than just the first event. In these cases, one of the functions of $p_m^{en}(n)$ and $|r_m(n)|$ starts with higher level than the other and goes below the level of the second one, suggesting a phoneme boundary very strongly.

The fourth event is also quite rare and is designed for situations where the DWT spectrum changes very rapidly, what happens for changes in speech content

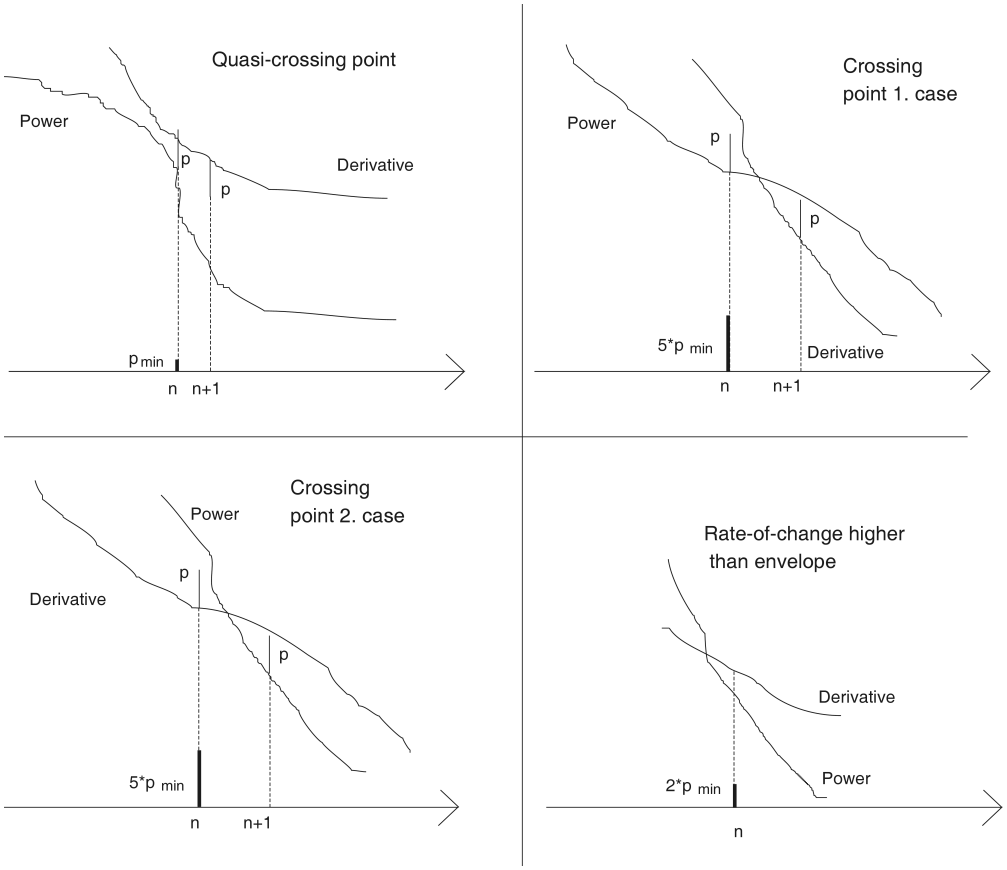

Fig. 6. Simple examples of four events described in Table 2. They are characteristic of phoneme boundaries. Images present power envelope $p_m^{en}(n)$ and rate-of-change information (derivative) $r_m(n)$.

like phoneme boundaries. In this situation, a level of $p_m^{en}(n)$ can be relatively low. The absolute value of the rate-of-change information $|r_m(n)|$ being higher than the power envelope $p_m^{en}(n)$ and the power envelope $p_m^{en}(n)$ being higher than double the minimum threshold are searched for.

The fourth event is different, as it does not describe anything similar to crossing used in the general description of the method in the previous section. However, if $|r_m(n)|$ is so high, it also indicates that a phoneme boundary may occur. It is less strict and more general, so a lower weight was given. The values of thresholds in the first three events were chosen to make the second and third events more difficult to fulfill than the first one. The threshold in the fourth type event was chosen experimentally.

The method is designed so that it would be easy to improve it, by introducing additional conditions. For example, a new condition will add or subtract addi-

tional values to $e(n)$. Subtracting would introduce negative events, which imply that boundaries did not occur in particular n. They are not included in the presented solution, but generally are possible. Another aspect of the 'intelligence' of the method is that even though it consists of several conditions, the sensitivity can be easily changed by setting another decision threshold. The decision threshold is lowered by one for finding the following discrete times (compared with the first one in the group) owing to a hysteresis rule. The application of hysteresis for the threshold produces better results.

The algorithm was implemented in Matlab and not optimized for time efficiency. In its current version, it needs 14 [min] to segment the whole corpus using Haar wavelet (the lowest order of filters) and 20 [min] for discrete Meyer wavelet (the highest order of filters, namely 50). The corpus has 16 425 utterances (some of them are words and others are sentences), which gives 0.05 [s] per utterance for the version with Haar wavelet and 0.07 [s] for the Meyer one. The properly optimized code in C++ would be much more time-efficient. The experiment was conducted on a computer with AMD Athlon 64 processor 3500+990 [MHz], 1.00 [GB] of RAM.

7. Database

The method was developed on a set of 50 hand-segmented Polish words with the sampling frequency $f_0 = 11\,025$ [Hz], equivalent to a sampling period $t_0 = 90.7$ [μs]. To assess the quality of our results, the method was tested on a much larger set, called CORPORA, created under the supervision of Stefan Grocholewski from the Institute of Computer Science, Poznań University of Technology in 1997 (Grocholewski, 1995).

Speech files in CORPORA were recorded with the sampling frequency $f_0 = 16$ [kHz] equivalent to sampling period $t_0 = 62.5$ μs. Speech was recorded in an office with a working computer in the background, which makes the corpus not perfectly clean. Signal to Noise Ratio (SNR) is not stated in the description of the corpus. It can be assumed that SNR is very high for actual speech, but minor noise is detectable for periods of silence.

The database contains 365 utterances (33 single letters, 10 digits, 200 names, 8 simple computer commands, and 114 short sentences), each spoken by 11 females, 28 males, and 6 children (45 people), giving 16 425 utterances in total. One set spoken by a male and one by a female were hand-segmented. The rest were segmented by the dynamic programming algorithm, which was trained on hand-segmented ones, and manually checked afterwards. The quality of all transcriptions can be assumed to be as good as hand-made transcription. None of the CORPORA utterances were in the original set used during development. Hand-segmentation was done by different people in the small development set and for CORPORA.

8. Evaluation method

Detected boundaries may have various degrees of accuracy with respect to hand-segmentation of speech. There are a number of factors that must be considered, including the accuracy of hand-segmented boundaries, since hand-segmentation is not in itself an entirely accurate process because of uncertainties in human perception of the phoneme boundaries. Additionally, overlapping phonemes or partially merged phonemes are a natural phenomena. There is, therefore, a degree of uncertainty in the precision of the boundaries of the phonemes.

Simply assigning a Boolean value (correct or incorrect) is not really a sensitive measure of segmentation quality. For this reason, fuzzy logic was used, which produces a graded rating of boundary locations in a more sensitive and human-like way. The concept of fuzzy sets and logic was used to derive recall and precision scores. The reasons for why we believe this evaluation is better than the typical ones are presented in (Ziółko *et al.*, 2007).

Let us begin with two assumptions. Firstly, correct (hand) segmentation is presented as a set of narrow ranges (typically 5 [ms]). Although it might be tempting to interpret each range as the end of the previous phoneme and the beginning of the next one, the real situation is that neighboring phonemes overlap with each other in these ranges. Detected boundaries are represented as a set of single discrete times. Secondly, perfect detection of silence is assumed. Silence segments may be of almost any length. This is why, including them in evaluation would cause serious inaccuracy. There are other very good methods for speech and silence separation (Zheng, Yan, 2004). Our goal is then to match the detected and hand-segmented boundaries in pairs and assess the quality of the recovered boundaries.

Let us define three sets. Set A contains the predicted boundaries (the retrieved set). Set G contains the correct hand-segmented boundaries (the relevant set). Finally, let us define the set of correctly found boundaries C (i.e. the boundaries that are both retrieved and relevant). This is essentially the intersection of A and G. The membership of C is fuzzy; if x is a detected boundary from A, then $f(x)$ describes the degree of membership of C. If $f(x) = 1$, then $x \in A$ is a correct boundary and therefore $x \in C$, whereas if $f(x) = 0$, then $x \in A$ is incorrectly detected. Values between 0 and 1 indicate partially correct boundaries.

Each boundary in A is paired with the closest hand-segmented boundary from G. For matched boundaries, when the detected boundary is inside the hand-segmented boundary range, the boundary is correct and $f(x) = 1$. Otherwise, it is a fuzzy case and we set $f(x) = 1 - b(x)/a(x)$ where a is half of the duration of the phoneme in which the boundary x resides and $b(x)$ stands for the difference between the nearest end of hand-segmented boundary and the detected one. This grades the membership from 0, when x is precisely halfway between two hand-segmented boundaries to 1, when it lies exactly in the range of one

of the boundaries. Then, the traditional precision and recall measures can be redefined:

$$\textit{Fuzzy Precision} = \begin{cases} 1 & \text{when } \dfrac{\sum_{x\in A} f(x)}{|G|} > 1, \\ \dfrac{\sum_{x\in A} f(x)}{|G|} & \text{otherwise;} \end{cases} \tag{16}$$

$$\textit{Fuzzy Recall} = \frac{\sum_{x\in A} f(x)}{|A|}, \tag{17}$$

where $|A|$ and $|G|$ are cardinalities of sets A and G, respectively. Recall (17) and precision (16) can be used to give a single evaluation grade in many different ways according to their importance. A widely used way is the F-score (VAN RIJSBERGEN, 1979)

$$F = \frac{2\,P\,R}{P+R}, \tag{18}$$

where P is fuzzy precision and R is fuzzy recall. F is the measure from 0 to 1, which we use in our experiments, where higher results mean better ones.

9. Experimental results

Our first set of results looks at the usefulness of the six wavelet functions for analyzing phoneme boundaries. The obtained results for different wavelets (see Table 3) shows the differences in their efficiency. The results show that discrete Meyer wavelet (Fig. 2) (ABRY, 1997) performs the best in this case, probably because of its symmetry in time domain, which helps in synchronization of the subbands. Asynchronization in time domain can be caused by ripples in frequency domain. An experiment using two wavelets (Meyer and sym6) one after another was also conducted. As it might be expected, it improved results only a little, while it doubled the time of calculations. Analyzing seven subbands was also checked, where the seventh one was from 125 [Hz] to 62.5 [Hz].

Table 3. Comparison of the proposed method using different wavelets.

Method	av. recall	av. precision	F-score
Meyer	0.7096	0.7408	0.7249
db2	0.6770	0.7562	0.7144
db6	0.7029	0.7414	0.7217
db20	0.7034	0.7408	0.7216
sym6	0.7015	0.7426	0.7215
haar	0.6377	0.8042	0.7113
Meyer+sym6	0.6825	0.7936	0.7339
Meyer 7subbands	0.6449	0.6714	0.6579

The accuracy of our phoneme detection technique was then compared with some standard framing techniques (see Table 4) like constant segmentation methods, where the speech is broken into fixed length segments, and where the speech signal is segmented randomly. Accuracy of constant segmentation for many multiplications of 5.8 [ms] (the time length between neighboring discrete times) was evaluated, but we only present results for 23 [ms], as it corresponds to typical length of frames in speech recognition and for 92.8 [ms] for which the result is the best of all constant segmentations. It was not possible to compare our method with any referred segmentation method because we do not have access to the software and corpora used by other researchers. We made attempts to make such collaboration but we were refused.

Table 4. Comparison of some other segmentation strategies and the proposed method.

Method	av. recall	av. precision	F-score
Const 23.2 ms	0.9651	0.1431	0.2493
Const 92.8 ms	0.7635	0.4659	0.5787
SVM	0.50	0.33	0.40
Wavelet	0.7096	0.7408	0.7249

We also trained the support vector model (SVM) using powers and derivatives from DWT subbands. Features for SVM included analyzed part of speech as well as left and right context. No other phoneme segmentation method available for comparison was found. While constant segmentation is able to find most of the boundaries with a 23 [ms] frame, this is only at the expense of very short segments and many irrelevant boundaries. The overall score of our method is much superior to the constant segmentation approach.

Several researchers claim that syllables are better basic units for ASR than phonemes. It is probably true in terms of their content, but it seems not to be the same for detecting unit boundaries. Our method is not perfect, but the observed DWT spectra of speech (e.g. Fig. 3) clearly show that boundaries between phonemes can be extracted. Boundaries between syllables seem not to differ from phoneme boundaries in observed DWT spectra, while obviously there are fewer syllable boundaries than phoneme ones. It is, therefore, difficult to detect syllable boundaries without also finding phoneme boundaries when analyzing DWT spectra.

10. Conclusions

Because of the uniform segmentation, most of the ASR systems do not use information about boundaries of phonetic units like phonemes. A method based on the DWT to find such boundaries is presented. The method is language-agnostic, as it does not rely on any phonetic models, but relies purely on the

analysis of the power spectrum and hence has applicability to any language. For the same reason, it can be easily introduced to most of the existing systems, as it does not depend on any exact configuration or does not need training of the speech model. It can also be used to provide additional information or primal hypothesis for segmentation methods based on models like in Ostendorf's et al. solutions (Ostendorf *et al.*, 1996). Our method is constructed in a way that additional conditions or changing weights can be applied in need of search for solutions for specific applications, noisy data, etc.

The use of several wavelet functions were compared and our results show that Meyer wavelets are better than the others. Fuzzy recall and precision measures were introduced for segmentation to evaluate the method with more sensitivity, grading errors more smoothly than in the commonly used evaluation methods. Our results give approximately 0.72 F-score for Meyer and slightly less for other wavelets.

References

1. Abry P. (1997), *Ondelettes et turbulence (eng. Wavelets and turbulence)*, Diderot ed., Paris.
2. Cardinal P., Boulianne G., M. Comeau (2005), *Segmentation of recordings based on partial transcriptions*, Proceedings of Interspeech, 3345–3348.
3. Daubechies I. (1992), *Ten lectures on Wavelets*, Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania.
4. Glass J. (2003), *A probabilistic framework for segment-based speech recognition*, Computer Speech and Language, **17**, 137–152.
5. Grayden D.B., Scordilis M.S. (1994), *Phonemic segmentation of fluent speech*, Proceedings of ICASSP, Adelaide, 73–76.
6. Grocholewski S. (1995), *Assumptions of acoustic database for Polish language* [in Polish: *Założenia akustycznej bazy danych dla języka polskiego* (CD-ROM), Mat. I KK: Głosowa komunikacja człowiek-komputer, Wrocław, 177–180.
7. Hermansky H. (1990), *Perceptual linear predictive (PLP) analysis of speech*, Journal of the Acoustical Society of America, **87**, 4, 1738–1752.
8. Hermansky H., Morgan N. (1994), *RASTA processing of speech*, IEEE Transactions on Speech and Audio Processing, **2**, 4, 578–589.
9. Holmes J.N. (2001), *Speech Synthesis and Recognition*, Taylor and Francis, London.
10. Hunt A., Black A. (1996), *Unit selection in a concatenative speech synthesis system using a large speech database*, Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 1996, ICASSP-96, **1**, 373–376.
11. Morgan N., Zhu Q., Stolcke A., Sonmez K., Sivadas S., Shinozaki T., Ostendorf M., Jain P., Hermansky H., Ellis D., Doddington G., Chen B., Cretin O., Bourlard H., Athineos M. (2005), *Pushing the envelope – aside*, IEEE Signal Processing Magazine, **22**, 81–88.

12. Ostendorf M., Digalakis V.V., Kimball O.A. (1996), *From HMM's to segment models: A unified view of stochastic modeling for speech recognition*, IEEE Transactions on Speech and Audio Processing, **4**, 360–378.

13. Rabiner L., Juang B.H. (1993), *Fundamentals of speech recognition*, PTR Prentice-Hall, Inc., New Jersey.

14. Rioul O., Vetterli M. (1991), *Wavelets and signal processing*, IEEE Signal Processing Magazine, **8**, 11–38.

15. Russell M., Jackson P.J.B. (2005), *A multiple-level linear/linear segmental HMM with a formant-based intermediate layer*, Computer Speech and Language, **19**, 205–225.

16. Stöber K., Hess W. (1998), *Additional use of phoneme duration hypotheses in automatic speech segmentation*, Proceedings of ICSLP, Sydney, 1595–1598.

17. Suh Y., Lee Y. (1996), *Phoneme segmentation of continuous speech using multi-layer perceptron*, Proceedings of ICSLP, Philadelphia, 1297–1300.

18. Toledano D.T., Gómez L.A.H., Grande L.V. (2003), *Automatic phonetic segmentation*, IEEE Transactions on Speech and Audio Processing, **11**, 6, 617–625.

19. van Rijsbergen C.J. (1979), *Information Retrieval*, Butterworths, London.

20. Wang D., Narayanan S. (2005), *Piecewise linear stylization of pitch via wavelet analysis*, Proceedings of Interspeech, Lisboa, 3277–3280.

21. Weinstein C.J., McCandless S.S., Mondshein L.F., Zue V.W. (1975), *A system for acoustic-phonetic analysis of continuous speech*, IEEE Transactions on Acoustics, Speech and Signal Processing, **23**, 54–67.

22. Young S. (1996), *Large vocabulary continuous speech recognition: a review*, IEEE Signal Processing Magazine, **13**, 5, 45–57.

23. Zheng C., Yan Y. (2004), *Fusion based speech segmentation in DARPA SPINE2 task*, Proceedings of ICASSP, Montreal, I–885–888.

24. Ziółko B., Manandhar S., Wilson R.C., Ziółko M. (2006), *Wavelet method of speech segmentation*, Proceedings of 14th European Signal Processing Conference EUSIPCO, Florence.

25. Ziółko B., Manandhar S., Wilson R.C. (2007), *Fuzzy recall and precision for speech segmentation evaluation*, Proceedings of 3rd Language and Technology Conference, Poznań.

26. Zue V.W. (1985), *The use of speech knowledge in automatic speech recognition*, Proceedings of the IEEE, **73**, 1602–1615.