BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **14**, No 2

Sofia • 2014

Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2014-0025

Telephone Speech Endpoint Detection Using Mean-Delta Feature

Atanas Ouzounov

Institute of Information and Communication Technologies, 1113 Sofia
Email:atanas@iinf.bas.bg

Abstract: *In the study the efficiency of three features for trajectory-based endpoint detection is experimentally evaluated in the fixed-text Dynamic Time Warping* (*DTW*) *– a based speaker verification task with short phrases of telephone speech. The employed features are Modified Teager Energy* (*MTE*)*, Energy-Entropy* (*EE*) *feature and Mean-Delta* (*MD*) *feature. The utterance boundaries in the endpoint detector are provided by means of state automaton and a set of thresholds based only on trajectory characteristics. The training and testing have been done with noisy telephone speech* (*short phrases in Bulgarian language with length of about 2 s*) *selected from BG-SRDat corpus. The results of the experiments have shown that the MD feature demonstrates the best performance in the endpoint detection tests in terms of the verification rate.*

Keywords: *Speech detection, endpoint detection, spectral entropy.*

1. Introduction

The aim of the endpoint detection is to locate the beginning and the ending points of the speech message. This detection is a crucial preprocessing stage in automatic speech and speaker recognition systems designed to operate in noisy real-world environments. The wrong endpoint detection increases the cases when the system processes the part of a speech message or the message, prolonged with non-speech frames. This leads to increase of the recognition error or/and amount of computations.

The endpoint detection algorithms are based on the speech/non-speech detection paradigm and can be divided into two general groups. The first one comprises the algorithms for analyzing the time variations (trajectories) of selected parameters. These algorithms utilize a combination of state automaton and a set of thresholds (fixed or adaptive) in order to produce utterance endpoints based only on the trajectories characteristics [3, 6, 7, 21]. The second group comprises algorithms based on some type of pattern recognition technique. In this case, during the training mode the reference models for two classes (i.e., speech and non-speech) are created based on selected features. In the classification mode, each frame is associated with one of the classes based on some kind of similarity function. Then the first and the last frames (i.e., the endpoints) are located using additional rules [18, 20, 24, 25].

The most frequently used feature for endpoints detection is the energy of the speech signal [3]. This feature is efficient for clean conditions but does not have robustness in noisy real-world environments. To improve the noise robustness of the endpoint detection, a lot of features are developed, such as energy and spectral entropy combinations [4, 5], modifications of the spectral entropy [6, 21], features based on wavelets [19], bispectrum [9], etc.

In the study the Mean-Delta (MD) parameter [14] is proposed as a feature for trajectory-based real-world endpoint detection. Two additional features: the Modified frame Teager Energy (MTE) [4, 5] and the Energy-Entropy feature (EE) [4] are included for comparative purposes only. The starting and ending frames of an utterance are estimated by means of state automaton and a set of thresholds and based only on trajectory characteristics.

In order to validate the performance of the proposed endpoint detection algorithms, two experiments are carried out. In the first one the accuracy was evaluated in terms of frame difference between manually labelled and detected endpoints. In the second experiment the performance of the endpoints detection features in terms of the recognition rate is estimated in the Dynamic Time Warping (DTW) fixed-text speaker verification task with short noisy telephone phrases in Bulgarian language [16]. The verification results are compared with those obtained by the manual endpoint detection.

The Z_{HTER}-test method proposed in [1] is applied to check whether the verification rate obtained by a given endpoint detection feature is statistically significantly different from the rate provided by another one. To illustrate the verification results the Receiver Operating Characteristics (ROC) curves are plotted [8].

2. Endpoint detection parameters

2.1. Mean-Delta feature

The Mean-Delta (MD) feature was proposed in [14] and it is defined as the mean absolute value of the delta spectral autocorrelation function of the power spectrum of the speech signal. In order to remove the slope of the spectral autocorrelation

function and enhance the peaks, a parameter obtained in a way similar to the delta cepstrum evaluation was proposed in [14]. It is named Delta Spectral AutoCorrelation Function (DSACF). This parameter is computed as an orthogonal polynomial fit of the first-order derivative (in the correlation domain) of the spectral autocorrelation function. For a particular frame, the DSACF is computed utilizing only the frame's spectral autocorrelation lags. For the *n*-th frame, the DSACF $\Delta R_P(n,l)$ is

$$\Delta R_{\mathrm{p}}(n,l) = \frac{\sum_{q=-Q}^{Q} qR_P(n,l+q)}{\sum_{q=-Q}^{Q} q^2}, \tag{1}$$

where $l = 0,...,L$; L is the number of correlation lags; $n = 0,...,N-1$, N is the number of frames and $R_{\mathrm{p}}(n,l)$ is the biased spectral autocorrelation function defined with the power spectrum. The parameter Q determines the window width around the lag l and its effect on the accuracy of the approximation.

For *n*-th frame the MD feature $m_d(n)$ is computed as follows:

$$m_{\mathrm{d}}(n) = \left[\sum_{l=0}^{L} \left| \Delta R_{\mathrm{p}}(n,l) \right| \right]^{0.5}, \tag{2}$$

where $\Delta R_{\mathrm{p}}(n,l)$ is the DSACF in (1) for lag l, L is the number of lags. For more details about MD feature, see [14].

Up to now the above described MD feature was not used in trajectory-based endpoint detection algorithm. Its vector version is utilized in a speech detection module as a part of speaker recognition tasks [15].

As shown in [14, 15], MD feature estimation is based on the spectral autocorrelation function defined with the power spectrum. The results from preliminary experiments have revealed that for the endpoint detection scheme utilized in this study, the MD feature provides better performance when its estimation is based on the spectral autocorrelation function, defined not with the power but with the magnitude spectrum. This is due to the difficulties in endpoints detection of some phonemes as weak fricatives, nasals and the end, etc. In this case the trajectory of MD feature based on the magnitude spectrum represents more accurate similar low-level phonemes.

For each frame, the magnitude-based version of the MD feature is computed as follows:

- compute the magnitude spectrum $|X(k)|$ of the Hamming-windowed speech signal via the Fast Fourier Transform (FFT) with size *K*;
- compute the average magnitude spectrum – over all frames in the utterance;
- apply mean normalization – the frame magnitude spectrum is divided by the average magnitude spectrum;

- compute the non-normalized biased spectral autocorrelation function with lags $L = K/4$ using the mean normalized frame magnitude spectrum;
- compute the delta spectral autocorrelation function by (1) with Q=3;
- perform a trajectory smoothing for delta spectral autocorrelation function (inter-frame processing) by J-order long-term spectral envelope algorithm with J=3 [17]; the obtained smoothed version of $\Delta R_{\mathrm{p}}(n, l)$ is denoted as $\Delta R_{\mathrm{p}}^{\mathrm{s}}(n, l)$;
- compute $m_{\mathrm{d}}(n)$ by equation (2) using $\Delta R_{\mathrm{p}}^{\mathrm{s}}(n, l)$.

2.2. Modified frame Teager Energy feature

The modified frame Teager energy is computed according to the algorithm described in [4, 5]. In this algorithm the spectrum of the signal is used rather than the instantaneous energy. The algorithm for the Modified frame Teager Energy (MTE) feature calculation includes the following steps for each frame:

- calculate the power spectrum;
- weight each sample in the power spectrum with the square of the frequency;
- take the square root of the sum of the weighted power spectrum.

The result of the last step is the MTE feature for the particular frame.

2.3. Energy Entropy Feature

A feature for isolated word endpoints detection, obtained by combination of the energy and the spectral entropy, is proposed in [4]. This feature is similar to the one described in [5], but without the step of subtracting the average (over the first 10 frames). This subtraction is done in an attempt to reduce the effect of background noise. In the current study the feature described in [4] is to be used. This Energy-Entropy (EE) feature is computed for every speech frame as follows (for simplicity, the frame index is omitted):

- compute the energy E

$$E = \sum_{i=0}^{I-1} x(i)^2 , \tag{3}$$

where I is the number of the samples in the frame;

- estimate the probability density function $P(k)$ for the frequency component k as

$$P(k) = \frac{\left|X(k)\right|^2}{\sum_{k=0}^{K/2} \left|X(k)\right|^2}, \tag{4}$$

where K is the FFT-size;

- compute the negative entropy

$$H = \sum_{k=0}^{K/2} P(k) \log(P(k)); \tag{5}$$

- compute the EE feature as

(6) $$\mathrm{EE} = \sqrt{(1 + |E \times H|)} .$$

In order to make correct comparisons among different features, the limitation of the frequency range from 250 Hz up to 3750 Hz (as done in [4]), was not applied in our case.

3. Endpoints detection algorithm

The proposed Endpoint Detection (ED) algorithm is intended for location of the beginning and ending frames of a word or a single phrase of a short length (few seconds). It is supposed that the length of a single pause between the words, within the phrase, is less than a second and the phrase or word starts and ends within the speech record. This algorithm is based on the trajectory variations for a single parameter and is using thresholds and detection rules to take a decision for the beginning and ending frames.

3.1. Thresholds' setting

Usually the ED algorithms utilize two types of thresholds – fixed and adaptive [3]. The fixed thresholds are set beforehand and do not change during the detection, whereas the adaptive thresholds do change along the utterance according to some selected rules. Only fixed thresholds will be used in the study.

Typically there are two ways to estimate the fixed thresholds. In the first one it is assumed that there are not any speech activities during the few hundred milliseconds from the beginning of the utterance [3]. The values of the analyzed parameters in this period are used to calculate the thresholds values. But, if there is a speech in the adaptation period, this leads to wrong thresholds setting and endpoints errors. In the second, the analysis is performing over the entire utterance in order to find the parameters values for thresholds setting. In this study the second approach is used. Here no assumptions are made about the place of speech and non-speech fragments in the utterance.

The aim of the fixed threshold is to separate the noise frames from the noise and speech frames based only on the value of a selected parameter. The method proposed in [3, 11] is based on the observation that the histograms of the log energy of noisy speech have a clean bimodal distribution corresponding to "noise only" and "noise + speech" parts of the signal. In this case the distribution can be approximated by two Gaussian densities that allow deriving a statistically optimal threshold [3, 11]. To do an accurate histogram, several seconds of speech in noise (4 s according to [11]) is required. It is supposed that the speech dominates the noise and this modeling is suitable only for cases with significant positive Signal-to-Noise Ratio (SNR) and stationary background noise.

In the paper a simpler algorithm for fixed thresholds settings is proposed. The preliminary experiments show its reliable work in moderate noise levels. The algorithm sets two thresholds (T_{low} and T_{high}) and includes the following steps:

- compute the values of the selected parameter E_n, $n = 1, \cdots, N,$ where N is the number of frames in analyzed utterance;
- compute mean μ and standard deviation σ values of E_n;
- compute the base threshold t_{base}

(7) $$t_{\text{base}} = \mu + \sigma;$$

- compute the mean value μ_{down} as

(8) $$\mu_{\text{down}} = E\{E_n < t_{\text{base}}, n = 1, \cdots, N\};$$

- compute the mean value μ_{up} as

(9) $$\mu_{\text{up}} = E\{E_n \geq t_{\text{base}}, n = 1, \cdots, N\};$$

- if μ_{down} is close to zero

(10) $$\text{if } \frac{\mu_{\text{down}}}{\mu_{\text{up}}} < \gamma \text{ then } \mu_{\text{down}} = \gamma \mu_{\text{up}};$$

- compute the low threshold T_{low}

(11) $$T_{\text{low}} = \mu_{\text{down}} + \alpha(\mu_{\text{up}} - \mu_{\text{down}});$$

- compute the high threshold T_{high}

(12) $$T_{\text{high}} = \beta T_{\text{low}}.$$

The coefficients α, β and γ are experimentally determined and their typical values are $\alpha = 0.03$, $\beta = 1.5$ and $\gamma = 0.05$.

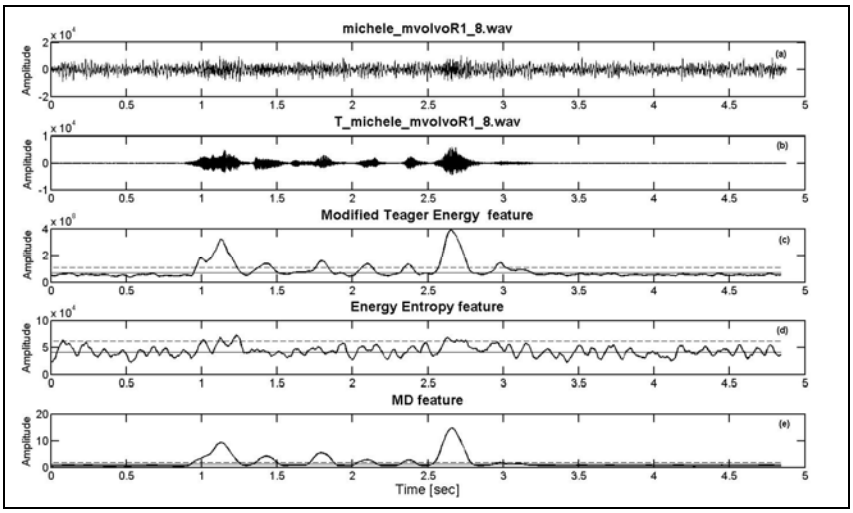


Fig. 1. An example of noisy speech: noisy speech data (a); clean speech reference (b); modified Teager energy contour (c); energy entropy feature contour (d); MD feature contour (e)

For illustration Fig. 1 shows the trajectories of the described above features for a car noise example selected from the "Lombard Speech" section in the SpEAR database [26]. The example has a clean speech reference and a corresponding noisy

version (time-aligned). It contains speech corrupted with noise, recorded inside a driving car (Volvo 340). For the clean reference SNR = 27.00 dB and for its noisy version SNR = –14.58 dB. The clean and noisy versions are downsampled to 8 kHz. Figs. 1 (c), (d) and (e) demonstrate the features trajectories of the noisy example in Fig. 1 (a) and both thresholds estimated according to the algorithm above described. In Fig. 2 the histograms of the amplitudes for all features placed in Fig. 1 are shown. It can be seen in Figs 1 and 2 that for this noisy example the EE feature is not suitable for trajectory-based endpoint detection.

3.2. Detection algorithm

The detection algorithm used in the study is designed for end pointing of a single word or a short phrase. It works off-line and is an improved version of the algorithm developed by the author in [12]. A brief description of it is given in the text below. The algorithm is based on six-state automaton. The six states are: *scan data, scan start, maybe in, scan end, maybe out* and *end found*. The transition from one state to another is controlled by rules based on the feature values, two thresholds scheme and some duration constraints. These constraints are included in order to filter (to some extent) the prolonged low-level and short high-level non-speech events before and after the speech utterance.

In *scan data* the algorithm scans the values of the feature until they become greater than the lower threshold. If this is the case, the number of the frame is remembered as a beginning point candidate and the algorithm goes to the next state. If not, the search continues until the maximum length of the phrase is reached (in this version it is set to 7 s). If it is reached and there are not any values greater than the lower threshold, then an error occurs – for no speech message.

In *scan start* the algorithm scans the feature values while they are between the two thresholds. If the value is smaller than the lower threshold, the algorithm returns to the previous state. If it is between two thresholds for a time longer than a pre-specified period of time, then an error occurs – for low level signal. If the value is over the higher threshold, the algorithm goes to the next state.

In *maybe in* the algorithm estimates the period when the value is greater than the higher threshold. If this period is less than the prespecified period of time, the algorithm returns to the previous state. If it is not, then it is considered that there is an actual speech message. In this case the starting point of the message is estimated by analyzing the sequence of beginning point candidates. Then the algorithm goes to the next state.

In *scan end* the algorithm searches a frame when the feature value becomes smaller than the lower threshold for a prespecified period of time. The frame when this is fulfilled is remembered and this is the first endpoint candidate. Then the algorithm goes to the next state.

In *maybe out* the algorithm analyzes the features values and the periods when they are located above/below the thresholds and generates a sequence of endpoints candidates. Then based on specific rules the actual endpoint is estimated.

In *end found* the algorithm checks whether the length of the speech message (estimated between the beginning and ending points) is within acceptable limits. If

this is the case, the algorithm ends successfully and sends endpoints for further processing. If not, it generates an error – for a very short or very long speech message.

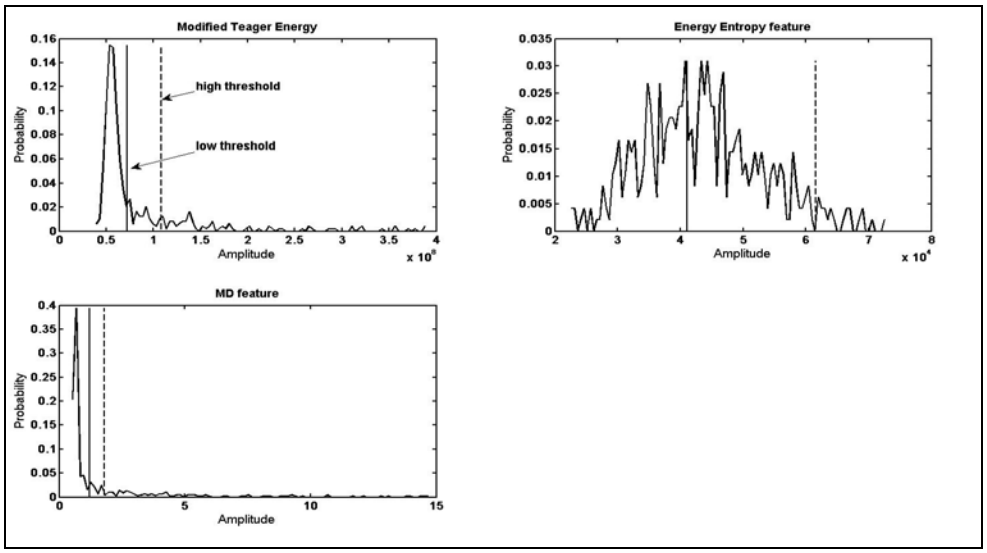


Fig. 2. Histograms and thresholds for the features shown in Fig. 1

4. Experiments and discussion

In the study three endpoint detection algorithms based on the described above features are experimentally evaluated. To validate the performance of the proposed endpoint detection algorithms, two experiments were carried out. In the first one the accuracy was evaluated in terms of the frame difference between manually labelled and detected endpoints. The second experiment was conducted to evaluate the endpoint algorithms in terms of the speaker verification performance.

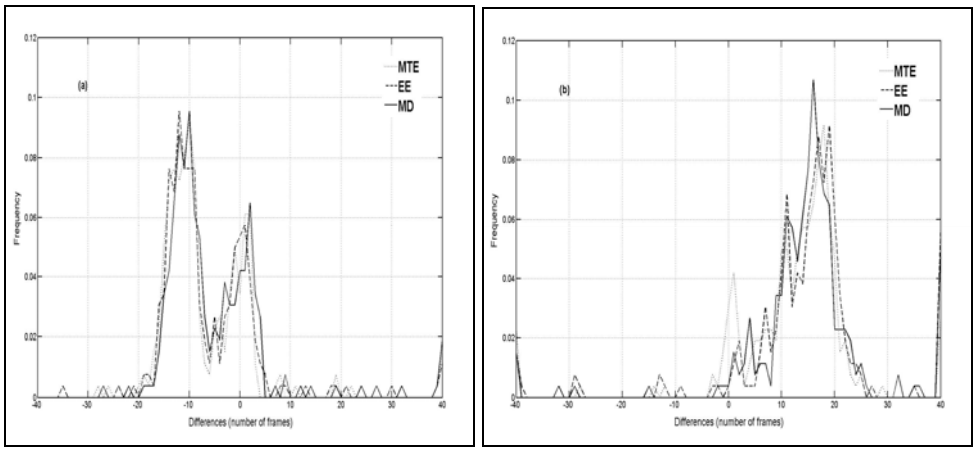
The speech data used in the experiments are selected from the BG-SRDat corpus [13]. This corpus is in Bulgarian language and it is recorded over noisy telephone channels and intended for speaker recognition. The speech data is collected from different types of telephone calls and various acoustical environments. The data are sampled with a frequency of 8 kHz at 16 bits, PCM format, and mono mode. The telephone speech data used in this study are recorded in real-world environment. Most of the speech records are obtained from street pay phones and they are noisy. The length of the phrase is about 2 s and the length of the single record is about 2.5-3 s.

It is worth to make some clarifications about the used phrase in Bulgarian language. It starts with voiced fricative *"z"* and ends with unvoiced fricative *"s"*. The phrase is: „*Zdravei Manolov. Kak se chuvstvash dnes?"*. Its English meaning is "*Hello Manolov! How are you today?*". The pronunciation (roughly) is – *"[zdra`vei:] [ma`nolov]! [kak] [se] [`tʃuvstvaʃ] [dnes]?"*[13]. In addition, the manual labelling of the endpoints of all speech data is done in order to have reference endpoints for comparative purposes.

4.1. Endpoint accuracy

In this experiment the endpoints accuracy was evaluated in terms of frames difference between manually labelled and detected endpoints [22].

The histograms of the differences for beginning and ending points are shown in Fig. 3. Table 1 presents the statistical information of these histograms. Each value in the table shows the rate of the distribution (in %) for less than 10-frames and 20-frames difference, respectively. The phrase used begins with the following two phonemes *"z"* and *"d"* (it is the Bulgarian word *"zdravei"*). The histogram in Fig. 3 (a) has two modes. This is due to the fact that for some records all algorithms miss the voiced fricative *"z"* and set the beginning point at the voiced stop consonant *"d"* (after the voice bar). These errors correspond to the left mode with a mean value of the difference of about –12 frames, whereas the right mode corresponds to the correct beginning points. As seen in Table 1, for beginning points the rate is highest for the MD feature. The histogram in Fig. 3 (b) indicates that for most files the algorithms set endpoints about 15 frames before the manual label. The phrase ended with unvoiced fricative *"s"* which is difficult to detect in noise due to its noise-like characteristics. According to Table 1, the maximum rate belongs to the MTE feature.



(a) (b)

Fig. 3. Histograms of the differences (number of frames) between manually labeled and detected points: beginning points (a); endpoints (b)

Table 1. The rate of distribution in %

Features	The differences (number of frames)			
	Beginning points		Endpoints	
	≤ 10	≤ 20	≤ 10	≤20
1. MTE	54.19	95.80	28.62	88.54
2. EE	55.34	96.56	18.32	82.06
3. MD	61.45	95.80	17.55	82.44

4.2. Speaker verification performance

The proposed endpoint detector is examined as a part of the fixed-text DTW-based speaker verification system. Since the detailed analysis of this system is out of the scope of this paper, only a brief description of the speaker verification scheme is included in the text below.

The speech data used in the study includes 262 records of a phrase collected from 12 male speakers. Each speaker utters the phrase at least 16 times. As the speech corpus is not large enough, we cannot use two separate data sets in a training mode – one for the reference template creation (a training set) and another for thresholds settings (a validation set). So in the study the training set is used directly as a validation set. There are two limitations, which must be taken into account, when using available speech data in the verification task. First, there are different numbers of records per a speaker – from 16 up to 34. Second, it is necessary to use an equal number of records for speaker's reference creation [23]. Considering these limitations in the study 10 records [16] per a speaker are randomly selected from speaker's data for reference creation, while the rest of his data are used for testing. This procedure is repeated 5 times. In the verification mode each time there are 142 client accesses or false rejection tests and 1562 impostor accesses or false acceptance tests. After 5 repeats, the total tests are: for false rejection – 710, and for false acceptance – 7810.

In the preprocessing step the Hamming-windowed frames of 30 milliseconds are utilized, with a frame rate of 10 milliseconds. The number of Mel-Frequency Cepstral Coefficients (MFCC) is 14. These cepstral coefficients are calculated using 24 Mel-frequency spaced filters. The 0-th cepstral coefficient is not used. In addition, cepstral mean subtraction is applied (for each file separately) to obtain the MFCC feature. For endpoint detection features, FFT-size of 512 points is chosen [16].

In the study, the DTW algorithm, named as the normalize-wrap method is applied [10]. In this algorithm, the length normalization of both the reference and the test pattern are used before performing the actual DTW algorithm. In the DTW, the relaxed endpoints constraints, Itakura's form of local constraints and the root power sum – the cepstral distance as a local distance are implemented. The speaker's reference is obtained by averaging (after dynamic time warping alignment) of his training utterances [23]. The individual speakers' verification thresholds are estimated by using the cohort normalization method [2].

For each ED algorithm in the study, a separate speaker verification task is carried out, i.e., a single classifier is considered. An additional verification task is performed with manually labelled endpoints. Actually, the efficiency of various endpoints detection features are compared via the verification results.

It is known that the single error value is not a reliable estimation of the speaker's verification performance [1]. This is true especially for real-world tasks where the available data are limited and the error value can depend on the data size. Since this is our case, it was decided to apply the methodology for performance estimation of the speaker verification proposed in [1]. The verification results are presented as rate ratios – False Rejection Rate (FRR), False Acceptance Rate (FAR)

and Half Total Error Rate (HTER) [1]. Besides, the 95% Confidence Interval (CI) for the HTER is shown computed according to [1]. The Z_{HTER}-test method proposed in [1] is applied to verify whether the given classifier (i.e., ED feature in our case) is statistically significantly different than another. Table 2 presents the speaker verification results in rates and a confidence interval for the HTERs. These rates are obtained for each feature and also for the manual end pointing. As seen from the table, MD feature performs best among the feature set.

Table 2. Speaker verification results

No	Features	FRR (%)	FAR (%)	HTER (%)	95% CI
1	Manual	8.30	4.94	6.62	±0.010
2	MD	7.04	10.57	8.80	±0.010
3	EE	10.84	11.61	11.22	±0.011
4	MTE	13.94	10.62	12.28	±0.013

Table 3. Confidence values

Value	[MD, EE]	[MD, MTE]
δ	99.76%	99.72%
σ	0.0079	0.0090

Table 3 shows the confidence values δ and the standard deviations σ obtained from the Z_{HTER}-tests (independent case) [1]. [A, B] denotes the two endpoints detection features A and B being tested. As seen from the table, the MD feature is statistically significantly different from the EE and MTE features. For both tests the confidence value δ on their HTERs difference is greater than 95%.

The average ROC curves are plotted in Fig. 4 to show the verification performance for each ED feature. Each curve is a vertical average of the five ROC curves of the five tests for 12 speakers. It is clearly seen that the MD feature curve is closer to the reference curve (manual ED) than the other two.

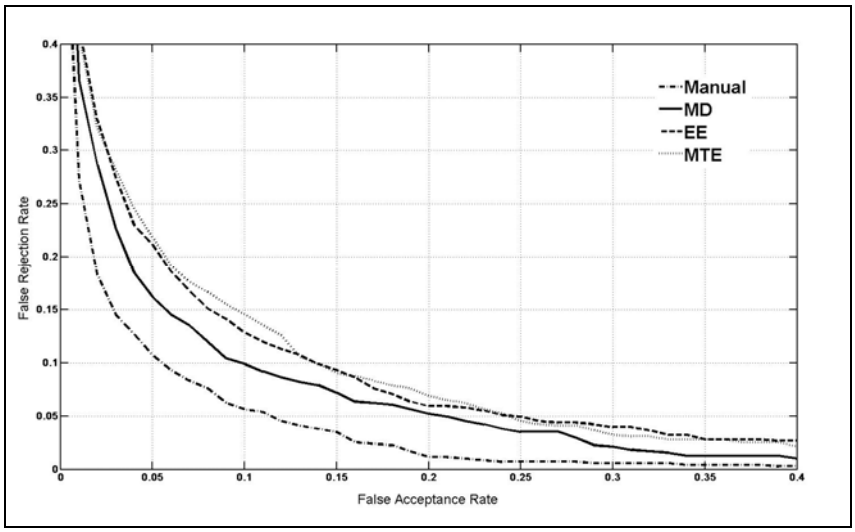


Fig. 4. ROC curves for different ED features

It can be seen in Table 1 that MD feature possesses the maximal rate for 10-frames difference for beginning points, but not for ending ones. Nevertheless, this feature provides the best verification rate as seen in Tables 2 and 3 and Fig. 4. For most files, as seen from Fig. 3(b), all algorithms set endpoints about 15 frames before the manual label of the end of the unvoiced fricative *"s"*. In fact the ED algorithms fail to detect the correct ending point of the phrase. It turns out, that these endpoint errors have a little impact on the recognition rate of the speaker verification scheme used in the present study.

5. Conclusions

The efficiency of three spectrum-based features for endpoint detection is experimentally evaluated in the fixed-text DTW-based speaker verification task with short phrases of telephone speech. As seen from Tables 2 and 3, and Fig. 4 the MD feature demonstrates the best performance in endpoint detection tests in terms of the verification rate.

The future research in this area will be focused on two main objectives – developing of a trajectory-based feature, which in combination with the MD feature will improve the endpoint detection accuracy for weak phonemes and examination of the developed endpoint detector in the hidden Markov models framework for short phrases.

References

1. Bengio, S., J. Marietho z. A Statistical Significance Test for Person Authentication, ODYSSEY. – In: The Speaker and Language Recognition Workshop, 2004, 237-244.
2. Burileanu, C., D. Moraru, L. Bojan, M. Puchiu, A. Stan. On Performance Improvement of a Speaker Verification System Using Vector Quantization, Cohorts and Hybrid Cohort-World Models. – International Journal of Speech Technology, 2002, No 5, 247-257.
3. Gerven, S., F. Xie. A Comparative Study of Speech Detection Methods. Eurospeech, 1997, 1095-1098.
4. Gu, L., S. Zahorian. A New Robust Algorithm for Isolated Word Endpoint Detection. – IEEE ICASSP, Vol. **IV**, 2002, 4161-4164.
5. Huang, L., C. Yang. A Novel Approach to Robust Speech Endpoint Detection in Car Environment. – In: IEEE ICASSP, 2000, 1751-1754.
6. Jia, C., B. Xu. An Improved Entropy Based Endpoint Detection Algorithm. – ISCSLP, 2002, 96-100.
7. Li, Q., J. Zheng, A. Tsai, Q. Zhou. Robust Endpoint Detection and Energy Normalization for Real-Time Speech and Speaker Recognition. – IEEE Transaction on SAP, Vol. **10**, 2002, No 3, 146-157.
8. Fawcett, T. An Introduction to ROC Analysis. – Pattern Recognition Letters, Vol. **27**, 2006, No 8, 861-874.
9. Mesa-Navarro, J., A. Moreno-Bilbao, E. Lleida-Solano. An Improved Speech Endpoint Detection System in Noisy Environments by Means of Third-Order Spectra. – IEEE Signal Processing Letters, Vol. **6**, 1999, No 9, 224-226.
10. Myers, C., L. Rabiner, A. Rosenberg. Performance Tradeoffs in Dynamic Time Warping Algorithms for Isolated Word Recognition. – IEEE Transactions on ASSP, Vol. 28, 1980, No 6, 623-635.

11. M c A u l a y, R., M. M a l p a s s. Speech Enhancement Using a Soft-Decision Noise Suppression Filter. – IEEE Transactions on ASSP, Vol. **28**, 1980, No 2, 137-145.
12. O u z o u n o v, A. Endpoint Detection Algorithm. – In: VoicePass – Real-Time Speaker Verification System Based on DSP Board Linkon FC-3000. Internal Report AdVoice, Ltd., 1996.
13. O u z o u n o v, A. BG-SRDat: A Corpus in Bulgarian Language for Speaker Recognition over Telephone Channels. – Cybernetics and Information Technologies, Vol. **3**, 2003, No 2, 101-108.
14. O u z o u n o v, A. A Robust Feature for Speech Detection. – Cybernetics and Information Technologies, Vol. **4**, 2004, No 2, 3-14.
15. O u z o u n o v, A. Robust Features and Neural Network for Noisy Speech Detection. – Cybernetics and Information Technologies, Vol. **6**, 2006, No 3, 75-84.
16. O u z o u n o v, A. Cepstral Features and Text-Dependent Speaker Identification – A Comparative Study. – Cybernetics and Information Technologies, Vol. **10**, 2010, No 1, 1-12.
17. R a m i r e z, J., J. S e g u r a, C. B e n í t e z, A. D e l a T o r r e, A. R u b i o. Efficient Voice Activity Detection Algorithms Using Long-Term Speech Information. – Speech Communication, Vol. **42**, 2004, No 3-4, 271-287.
18. R a m i r e z, J., P. Y e l a m o s, J. G o r r i z, J. S e g u r a e t. SVM-Based Speech Endpoint Detection Using Contextual Speech Features. – Electronics Letters, Vol. **42**, 2006, No 7, 426-428.
19. S e o k, J., K. B a e. A Novel Endpoint Detection Using Discrete Wavelet Transform. – IEICE Transaction on Inf. & Syst., Vol. **E82-D**, 1999, No 11, 1489-1491.
20. S h i n, W., B. L e e, Y. L e e, J. L e e. Speech/Non-Speech Classification Using Multiple Features for Robust Endpoint Detection. – IEEE ICASSP, 2000, 1399-1402.
21. W u, B. F., K. C. W a n g. Robust Endpoint Detection Algorithm Based on the Adaptive Band-Partitioning Spectral Entropy in Adverse Environments. – IEEE Transactions on SAP, Vol. **13,** 2005, No 5, 762-775.
22. Y a m a m o t o, K., F. J a b l o u n, K. R e i n h a r d, A. K a w a m u r a. Robust Endpoint Detection for Speech Recognition Based on Discriminative Feature Extraction. – In: IEEE ICASSP, Vol. **I**, 2006, 805-808.
23. Z e l i n s k i, R., F. C l a s s. A Learning Procedure for Speaker-Dependent Word Recognition System Based on Sequential Processing of Input Tokens. – In: IEEE ICASSP, 1983, 1053-1056.
24. Z h a o H., L. Z h a o, K. Z h a o, G. W a n g. Voice Activity Detection Based on Distance Entropy in Noisy Environment. – In: Fifth International Joint Conference on INC, IMS and IDC, 2009, 1364-1367.
25. Z h a n g, Z., S. F u r u i. Noisy Speech Recognition Based on Robust End-Point Detection and Model Adaptation. – In: IEEE ICASSP, Vol. **1**, 2005, 441-444.
26. Center for Spoken Language Understanding, Speech Enhancement and Assessment Resource (SpEAR) Database. Oregon Graduate Institute of Science and Technology. **http://www.cslu.ogi.edu/nsel/data/SpEAR_lombard.html**