A ROBUST TEO-BASED SPEECH SEGMENTATION METHOD FOR AUTOMATIC SPEECH RECOGNITION

Igor S. Peretta, Gerson F. M. Lima, Josimeire Tavares, and Keiji Yamanaka
Department of Computer Engineering, Faculty of Electrical Engineering, Federal University of Uberlandia
P.O. Box 593, 38400-902, Uberlandia, MG, BRAZIL
{iperetta,gersonlima}@ieee.org, josycbelo@gmail.com, keiji@ufu.br

Abstract – Based on the Teager Energy Operator (TEO), the "TEO-based method for Spoken Word Segmentation" (TSWS) is presented and compared with two widely used speech segmentation methods: "Classical", that uses energy and zero-crossing rate computations, and "Bottom-up", based on the concepts of adaptive level equalization, energy pulse detection and endpoint ordering. The implemented Automatic Speech Recognition (ASR) system uses Mel-frequency Cepstral Coefficients (MFCC) as the parametric representation of the speech signal, and a standard multilayer feed-forward network (MLP) as the recognizer. A database of 17 different words was used, with a total of 3,519 utterances from 69 different speakers. Two in three of those utterances constituted the training set for the MLP, and one in three, the testing set. The tests were conducted for each of the TSWS, Classical or Bottom-up methods, used in the ASR speech segmentation stage. TSWS has enabled the ASR to achieve 99.0% of success on generalization tests, against 98.6% for Classical and Bottom-up methods. After, a white Gaussian noise was artificially added to the ASR inputs to reach a signal-to-noise ratio of 15dB. The noise presence alters the ASR performances to 96.5%, 93.6%, and 91.4% on generalization tests when using TSWS, Classical and Bottom-up methods, respectively.

Keywords – Automatic Speech Recognition, Speech Segmentation, Teager Energy Operator, Mel-frequency Cepstral Coefficients, Artificial Neural Network, and Multilayer Perceptron.

1. INTRODUCTION

Engineers and scientists have been researching spoken language interfaces for almost six decades. In addition to being a fascinating topic, speech interfaces are fast becoming a necessity. Advances in this technology are needed to enable the average citizen to interact with computers, robots, networks, and other technological devices using natural communication skills. As stated by Zue and Cole [1], "without fundamental advances in user-centered interfaces, a large portion of society will be prevented from participating in the age of information, resulting in further stratification of society and tragic loss of human potential". They also stated: "a speech interface, in a user's own language, is ideal because it is the most natural, flexible, efficient, and economical form of human communication".

Several different applications and technologies can make use of spoken input to computers. The conversion of a captured acoustic signal to a single command or a stream of words is the top of mind application for speech recognition, but we can also have applications for speaker's identity recognition, language spoken recognition or even emotion recognition.

This work is part of a research that aims to develop a speaker-independent automatic speech recognition (ASR) system for spontaneous spoken isolated words, with a small vocabulary, that could be embedded to several possible applications. This research intends to develop human-machine interfaces using Brazilian Portuguese language.

One of the most important aspects to this objective is to have a good speech segmentation algorithm. Speech segmentation is the core of speech recognition, because the recognizer needs to handle only with the speech fragments of a given audio signal. This work proposes a novel speech segmentation method to support speech recognition and presents an ASR system implemented with widely used stages to achieve 99.0% of successful recognition rates on generalization tests.

1.1 DATABASE

The adopted database for this project is derived from the one build by Martins [2]. We kept 17 of the original 50 words from Martins, the ones presented in Table 1. There are three utterances from each word, from 69 independent-speakers (46 men and 23 women, all adults). This project database is then constituted of 3,519 audio records.

In this work, we chose to represent Brazilian Portuguese words pronunciation using the International Phonetic Alphabet (IPA)[1]. By convention, we present the phonemic words between slashes (/. . ./).

The audio signals for all samples from this database had been captured by a DSP-16 Data Acquisition Processor, from Ariel manufacturer. The integrated bandpass filter has cutoff frequencies of 300Hz and 3,400Hz. The used sampling frequency is 8kHz and samples have word-length of 16 bits. For this work, all audio signals from this database were converted to *WAVEform Audio* format. Also, white noise was added to each audio file, to reach a Signal to Noise Ratio (SNR) of 30dB.

[1] IPA is an alphabet for phonetic notation and it has been devised by the International Phonetic Association as a standard. It aims to represent all possible human made sounds that support all spoken languages.

Table 1: Portuguese voice commands from the database.

ID	Command	English	IPA
0	ZERO	ZERO	/ˈzɛrʊ/
1	UM	ONE	/ˈũ/
2	DOIS	TWO	/ˈdoyʒ/
3	TRÊS	THREE	/ˈtreʒ/
4	QUATRO	FOUR	/ˈkwatrʊ/
5	CINCO	FIVE	/ˈsĩkʊ/
6	SEIS	SIX	/ˈseyʒ/
7	SETE	SEVEN	/ˈsɛtɩ/
8	OITO	EIGHT	/ˈoytʊ/
9	NOVE	NINE	/ˈnωvɩ/
10	MEIA	HALF [DOZEN] or SIX	/ˈmeyɐ/
11	SIM	YES	/ˈsĩ/
12	NÃO	NO	/ˈnãw/
16	VOLTAR	BACK	/volˈtaʀ/
17	AVANÇAR	FORWARD	/avãˈsaʀ/
20	OPÇÕES	OPTIONS	/opˈsõyʒ/
49	AJUDA	HELP	/aˈjudɐ/

2. BACKGROUND

2.1 DESIGNED ASR SYSTEM

The following ASR system was designed with widely used steps, but simple ones. Besides the proposed speech segmentation method, all other stages could be found in related literature. The main stages are, as for any ASR: preprocessing; speech segmentation; feature extraction; and recognizer. Some ASR systems also have a post-processing stage to solve possible recognition conflicts.

The speech segmentation method derived from this work may support several implementations of ASR systems, as the designed one described by the parameters shown in Table 2. Those parameters are based on the ones proposed by Zue, Cole and Ward [3].

Table 2: Designed ASR parameters.

Parameters	Range
Speaking Mode	Isolated words
Speaking Style	Spontaneous speech
Enrollment	Speaker-independent
Vocabulary	17 words
Language Model	Finite-state
SNR	Medium to High ($\approx$ 15 to 30dB)
Transducer	Electret condenser microphone

2.2 PREPROCESSING

The preprocessing stage adjusts the captured audio signal. Typically, this stage is headed to minimize the influence of bias (offset compensation), and to normalize the speech spectrum (pre-emphasis filtering).

2.3 SPEECH SEGMENTATION

As stated by Lamier, Rabiner, et al [4], “accurate location of the endpoints of an isolated word is important for reliable and robust word recognition”. Due to its importance, we had searched for a reliable and robust speech segmentation method. Finally, during this research, we have developed a novel method for speech segmentation, based on the Teager Energy Operator (TEO), also known as the Teager-Kaiser Operator. The proposed method was named “TEO-based method for Spoken Word Segmentation” (TSWS) [5].

2.3.1 The Teager Energy Operator

In the work of Teager and Teager[2] on nonlinear modeling of speech, referenced in Maragos et al. [6], an energy operator on speech-related signals is first presented.

In other work, Kaiser has discussed the properties of that Teager's energy-related algorithm — later designed as the Teager Energy Operator (TEO), or the Teager-Kaiser Operator — which, "by operating on-the-fly on signals composed of a single time-varying frequency, is able to extract a measure of the energy of the mechanical process that generated this signal" [7].

Kaiser [8] has also defined both TEO in the continuous and discrete domains as "very useful 'tools' for analyzing single component signals from an energy point-of-view" [emphasis in original].

TEO is then defined in the discrete domain by Equation (1) [8]. Note that this algorithm uses only three arithmetic operators applied to three adjacent samples of the signal for each time shift.

$$\Psi\left[x(n)\right] = x_n^2 - x_{n-1} \cdot x_{n+1}, \tag{1}$$

where Ψ is the TEO operator; and $x(n)$ is the n^{th} sample of the discrete signal.

2.3.2 Proposed TEO-Based Segmentation

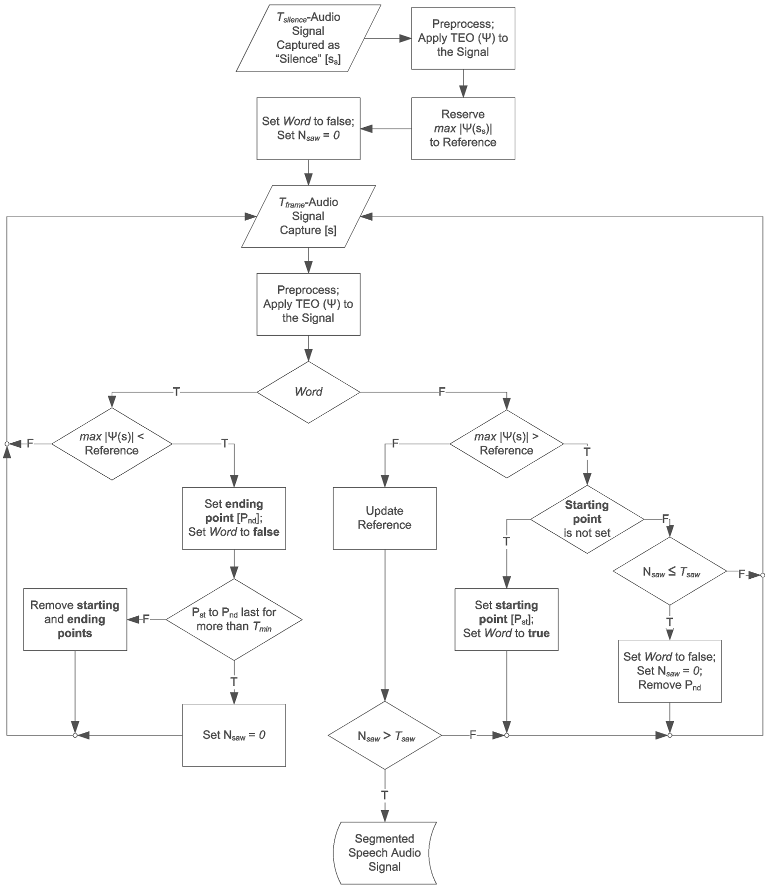
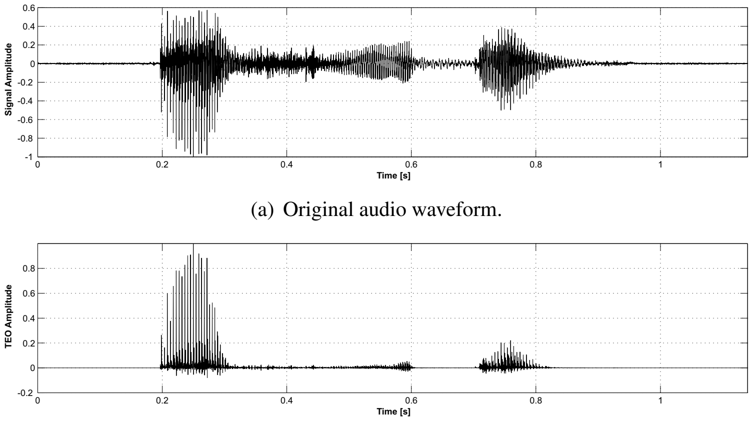

$T_{silence}$-Audio Signal Captured as "Silence" [s_s]
Preprocess; Apply TEO (Ψ) to the Signal
Reserve *max* |Ψ(s_s)| to Reference
Set *Word* to false; Set N_{saw} = *0*
T_{frame}-Audio Signal Capture [s]
Preprocess; Apply TEO (Ψ) to the Signal
Word
T
F
max |Ψ(s)| < Reference
F
T
max |Ψ(s)| > Reference
F
T
Set **ending point** [P_{nd}]; Set *Word* to **false**
Update Reference
Starting point is not set
T
F
N_{saw} ≤ T_{saw}
F
T
Remove **starting** and **ending points**
F
P_{st} to P_{nd} last for more than T_{min}
T
Set **starting point** [P_{st}]; Set *Word* to **true**
Set *Word* to false; Set N_{saw} = *0*; Remove P_{nd}
Set N_{saw} = *0*
N_{saw} > T_{saw}
F
T
Segmented Speech Audio Signal

Figure 1: Proposed TSWS word boundary detection algorithm (flowchart).

The development of the TSWS method has started with the awareness that TEO can give emphasis to speech regions in audio waveforms at the same time it understates the noise-only regions [9]. One aspect of the conclusions on the work of Kaiser [7] states that "it is as if the algorithm [of TEO] is able to extract the envelope function of the signal". This aspect give us the indication that the awareness into we are basing TSWS development is reliable. Figure 2 presents a given original audio waveform and the respective TEO resultant waveform. It can be seen in Figure 2(b) that, from an 'energy point-of-view', the speech carries much more information than the noise captured from the environment.

Application of the TSWS method to constantly incoming audio signals, instead of complete recorded audio signals, requires the use of a non-overlapping frame-by-frame approach. A non-overlapping frame of 25ms is then set (T_{frame}) to be the elementary structural constituent of the captured input audio. The first captured frames, for the length of 100ms ($T_{silence}$, chosen

[2] H. M. Teager and S. M. Teager, "Evidence for Nonlinear Production Mechanisms in the Vocal Tract," *NATO Advanced Study Institute on Speech Production and Speech Modeling*, Bonas, France, July 1989; Kluwer Acad. Publ., Boston, MA, 1990.



(a) Original audio waveform.

(b) Respective TEO resultant waveform.

Figure 2: Audio waveform for the speech "OPÇÕES" /opˈsõyʒ/ with white noise addition (SNR 30dB).

previously), are identified as "silence". At this moment, the TSWS method constructs a vector formed by TEO values of non-speech samples, named as silence vector. Now, there is the need for setting a reference value to the decision if the subsequent captured frames includes speech information or not. That reference value is evaluated by

$$REF = \max|\mathbf{\Psi}(\mathbf{s_s})| + A \cdot \sigma_{\Psi}, \tag{2}$$

where REF is the needed reference value; $\mathbf{\Psi}$ is the silence vector, evaluated when TEO is applied to the vector $\mathbf{s_s}$ constituted by audio sampled values for "silence"; A is a constant that depends on SNR[3]; and σ_{Ψ} is the *standard deviation* for TEO values from the silence vector. Table 3 shows the best empirically adjusted constant A values for each SNR level.

Table 3: Empirical SNR-dependent constant A from the TSWS method against SNR.

	SNR (WGN addition)			
	Clear	**30dB**	**15dB**	**5dB**
constant A	25	9	3	1.1

The decision if a given frame contains speech or non-speech information is taken by comparison with the reference value. If the maximum absolute TEO value from a given frame *is greater than* the the reference value (REF), it should contain speech information. Otherwise, it will be considered that this frame contains non-speech information. This inference enables the TSWS method to update reference value every time it gets a non-speech frame. This update is done by excluding samples from the first frame in the beginning of the silence vector, and appending to it the last non-speech frame captured. With this updated silence vector, the TSWS method applies equation (2) to reach an updated reference value.

To set the speech boundary inside the captured audio signal, a boolean control variable is set. The TSWS method identifies this variable as *Word* and uses it to keep control of last captured frame status. If this last frame was identified as "speech"[4] and *Word* is *false*, *Word* is set to *true*; if the last frame is considered a non-speech one and *Word* is *true*, *Word* is set to *false*. In other words, if the maximum absolute TEO value from a given frame is *greater than* the reference value and *Word* is *false*, the starting point of this frame is set as the "starting point of a possible spoken word". Reciprocally, if the maximum absolute TEO value from a given frame is *less than* the reference value and *Word* is *true*, the ending point of this frame is set as the "ending point of a possible spoken word".

The TSWS method has also the following adjustments incorporated:

- If a just found "spoken word" boundary is too short to be a phoneme, that boundary is discarded. The minimum lasting time inside boundary is identified as T_{min}, and, in this case, it is chosen to be 150ms.
- If the silence after a recently bounded "spoken word" is too short to mean the whole "spoken word" is bounded, the ending point of that boundary is discarded and *Word* is set to *true*. This means the method will carry on until finding a new ending point. The minimum lasting time for silence after a word is identified as T_{saw}, and, in this case, it is chosen to be 250ms.

[3] This constant has also dependency on the variability and the complexity of utterances, in a minor degree. SNR is the major factor of dependence.

[4] Note that not always a relative high absolute value for TEO in a given frame means it carries speech information. It could also be the capture of an interference.

2.4 FEATURE EXTRACTION

It is very important for any speech recognition system design to select the best parametric representation of acoustic data. This parametric representation is constituted of the features to be extracted from the speech signal. Some parametric representation starts from the study of how the speech is produced by human sound production system. Others starts from study of how the speech is perceived by human auditory system.

2.4.1 Framing and Windowing

To evaluate the features (or coefficients) to be extracted from the input speech signals, it is usual to divide those signals into non-overlapping frames. Each of those frames is widely used to be multiplied with a Hamming window, in order to keep the continuity of the first and the last points in the frame [10].

Martins [2] has used a method to keep the same numbers of coefficients extracted from each signal, regardless the length of the signal. In this method, each segmented speech signal is divided into a fixed number of 80 frames. A window of 20ms is then chosen to run through the frames. To ensure an overlapping relation between windows and frames, each window size must be greater than the frame size. If the evaluated frame size is greater than the window size, the window size is then adjusted to 1.5 times the frame size.

Each frame is zero padded to form an extended frame of 256 samples. Therefore, a Fast Fourier Transform (FFT) is performed to compute the magnitude frequency response (spectrum) of each "windowed" frame.

After, the magnitude frequency response obtained by FFT is multiplied by a set of 16 triangular bandpass filters, in order to get the log-energy of each filter respective output. The center frequencies of those filters are equally spaced along the Mel frequency scale.

Then comes to Mel-frequency Cepstral Coefficients (MFCC) extracted from the input speech signal. The choice of using MFCC as features to be extracted from speech signals comes from the widely use of those coefficients and the excellent recognition performance they can provide [2, 10]. MFCC, generalized from the ones computed by Davis and Mermelstein [10], is presented in Equation (3).

$$MFCC_i = \sum_{k=1}^{N} X_k \cdot \cos\left[i\left(k - \frac{1}{2}\right)\frac{\pi}{N}\right], \qquad i = 1, 2, \ldots, M, \tag{3}$$

where M is the number of cepstral coefficients; N is the number of triangular bandpass filters; and X_k represents the log-energy output of the k^{th} filter.

The set of MFCC constitutes the Mel-frequency Cepstrum (MFC), which is derived from a type of cepstral representation of the audio signal. The main difference from an usual cepstrum, termed as "the spectrum of the log of the spectrum of a time waveform" [11], is that MFC uses frequency bands equally spaced on the Mel scale (an approximation to the response of human auditory system) and usual cepstrum uses linearly-spaced frequency bands.

For a given input speech signal, we start dividing it into 80 frames and, for each frame, we end up evaluating 16 MFCC. Concatenating all those coefficients, we get 1,280 coefficients (the feature vector) to act as the input vector for the recognizer system.

2.5 ARTIFICIAL NEURAL NETWORK ACTING AS THE RECOGNIZER

An *artificial neural network* is defined by Fausett as "an information-processing system that has certain performance characteristics in common with biological neural networks" [12]. Fausett also characterizes an ANN by its architecture, its activation function, and its training (or learning) algorithm. The breadth of ANN's applicability is suggested by the areas in which they are currently being applied: signal processing, control, pattern recognition, medicine, business, among others. Speech recognition is also an area where ANNs are being applied with great success rates. However, some hybrid model solutions, as the ones which combine ANN with hidden Markov models (HMMs), have shown better results for speech recognition systems. Due to its relative simplicity, the authors opted for using an artificial neural network (ANN) model to act as the recognition system for the designed ASR.

Multilayer Perceptron (MLP) is an ANN architecture relatively simple to implement. It is very robust when recognizing different patterns from the ones used for training, and it has wide spread use to handle pattern recognition problems. Based on Rosenblatt's Perceptron neuron model [13], it typically uses an approach based on the Widrow-Hoff backpropagation of the error [12] as the supervised learning method. Note that the number of input units from a MLP is the same number of coefficients generated after the feature extraction stage. Likewise, the number of output units is generally the number of existing classes for pattern classification (or recognition).

The implemented recognizer is a single hidden layer MLP with $1,280$ input units, 100 hidden units and 17 output neurons. Each output neuron corresponds to a different voice command and its activation is equivalent to the recognition of its respective command (see Table 1 for the group of voice commands to be recognized). Regarding the database used in this project (section 1.1), audio files were divided by two sets: the *training set*, constituted of 2,346 patterns (the first and the second utterances of each word from each speaker), and *testing set*, constituted of 1,173 patterns (the third utterance of each).

Different types of algorithms were verified to enable a fast training for the recognizer. *Scaled Conjugate Gradient* [14] was then chosen as the supervised training algorithm and *Bayesian Regularization* algorithm[5] [15] was chosen to enable ANN's performance evaluation during the training.

3. EXPERIMENTAL RESULTS

To evaluate the performance of the ANN using TSWS method, as for the purpose of comparison, two other speech segmentation methods were implemented: *Classical* and *Bottom-up* methods. Classical method uses energy and zero-crossing rate computations [16], in order to detect the beginning and the ending of a spoken word present in a given audio signal. Bottom-up method, also known as Hybrid Endpoint Detector, was proposed by Lamel, Rabiner, et al [4] and uses concepts of adaptive level equalization, energy pulse detection, and ordering of found boundary. All mentioned methods were applied to the 3,519 audio files from the database.

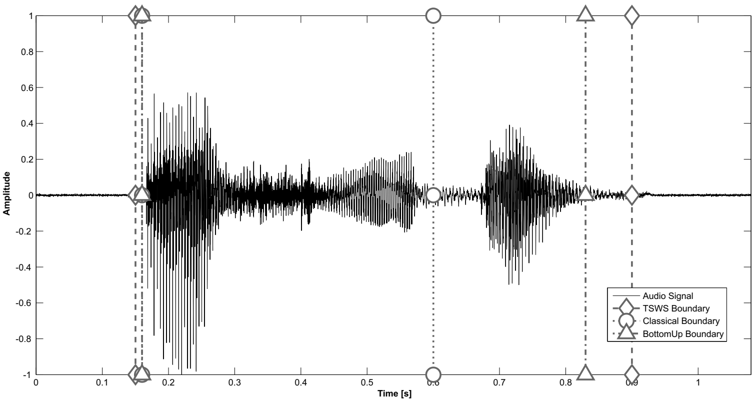


Figure 3: Audio waveform for the speech "OPÇÕES" /opˈsõyʒ/, with respective boundary set by TSWS, Classical and Bottom-up segmentation methods.

Another aspect considered during the preparation of this evaluation was to explore how those three segmentation methods react when facing different levels of noise. Actually, due to Lombard reflex[6] [17], no noise artificially added to a speech signal reflects the reality. This research tries to minimize this fact by artificially adding white Gaussian noise (WGN) to the audio signals from support database. The noise added this way could reflect a possible noise added by the transducer used to capture that audio signal. All above mentioned speech segmentation methods were also tested in the same SNR conditions. The conducted tests used the database audio files with WGN addition reaching a SNR of 30dB (first set) and 15dB (second test).

The resultant successful recognition rates presented here were found as the bests of a ANN training series. The conducted tests have used TSWS, Classical and Bottom-up methods in order to compare the contribution of the proposed segmentation method to the recognizer.

A successful recognition means the output neuron that represents the respective target class (command) of a given input pattern is strongly activated when compared to the others. The maximum output level is here considered to be the active output neuron. The division of training and testing patterns was kept here in order to evaluate the learning capability and the ability of generalization from the project recognizer.

Table 4 presents the overall successful recognition rates of the project recognizer (MLP) using 1,280 MFC coefficients as input patterns. Results from Classical and Bottom-up methods have shown no difference in recognition rates.

Table 4: Overall successful recognition rates in % for MFCC-MLP-recognizer.

Classical		**Bottom-up**		**TSWS**	
Train	**Test**	**Train**	**Test**	**Train**	**Test**
100.0	98.6	100.0	98.6	100.0	99.0

In this case, the TSWS method has supported MFCC-MLP recognizer to achieve 99.0% of overall successful recognition rates, in robusteness tests (testing set), besides the learning successful rate of 100.0% (using training set). Bottom-up method also has supported MFCC-MLP recognizer to achieve 100% with the training set, but achieved 98.6% on testing set. At this far on the scale, too close to 100.0% of succesful recognition rates, an error reduction of 28.6% is considered an important achievement.

[5] *Bayesian Regularization* is also known as *Mean Squared Error with Regularization*

[6] The Lombard reflex (or Lombard effect) is a noise-induced stress phenomenon that yields a modification of the speaker speech production in the presence of adverse conditions such as noise.

Figure 4 shows individual rates for each one of the 17 commands from the conducted tests. Only testing set results are here represented, because training set has achieved 100.0% of successful performance for all compared methods.

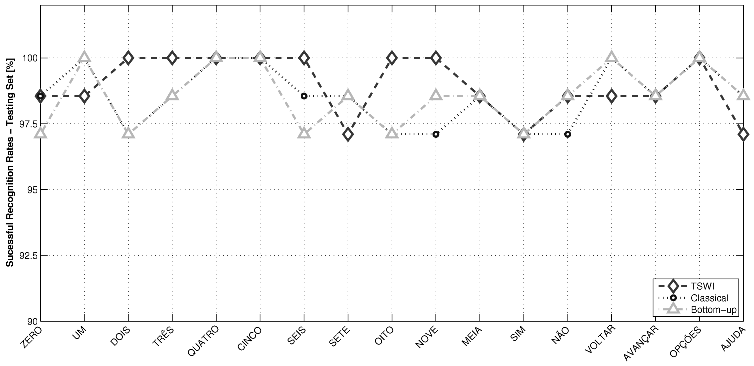


Figure 4: Comparison of the testing set successful recognition rates, in %, for MFCC-MLP-recognizer using TSWS, Classical and Bottom-up segmentation methods.

Another set of tests were conducted, this time with the artificial addition of white Gaussian noise to reach input audio signals with a SNR of 15dB. Table 5 presents the overall successful recognition rates achieved, using both Classical, Bottom-up and TSWS methods.

Table 5: Overall successful recognition rates in % for MFCC-MLP-recognizer with SNR of 15dB.

Classical		Bottom-up		TSWS	
Train	Test	Train	Test	Train	Test
99.8	93.6	99.9	91.4	100.0	96.5

As one can see, the TSWS method had significant contributions to the improvement of the recognition system, even in the presence of artificially added noise. An error reduction of 59.3% on nontrained patterns could be evaluated from the performance of the MFCC-MLP-recognizer when it is supported by TSWS method, instead of when supported by Bottom-up method. When comparing Classical and TSWS methods, we have achieved an error reduction of 45.3% in the performance on nontrained patterns successful recognition. Learning capability was also increased, achieving 100.0% only when the recognizer is supported by the TSWS method. Figure 5 shows individual rates for each one of the 17 commands from the conducted tests.

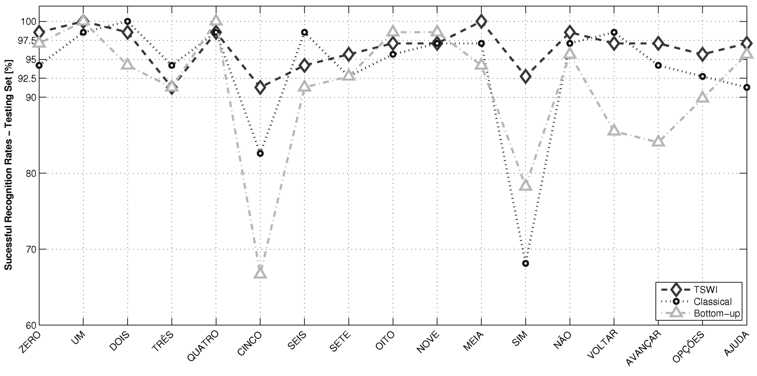


Figure 5: Comparison of the testing set successful recognition rates, in %, for MFCC-MLP-recognizer using TSWS, Classical and Bottom-up segmentation methods, with addition of WGN to achieve a SNR of 15dB.

4. CONCLUSION

In this work, the TSWS method for speech segmentation and an automatic speech recognition system based on Mel-frequency cepstral coefficients are presented. Comparisons are done in order to stabilish a criteria of understanding the influence of a TEO-based speech segmentation on success rates.

The speaker-independent speech recognition system presented here for isolated words from a limited vocabulary, supported by the proposed speech segmentation method, has achieved excellent recognition rates — 99.0% on average for the generalization of the smaller vocabulary case, against 98.6% of other two comparison methods (a 28.6% reduction on error rate). It also has presented a good generalization performance when dealing with noisy versions of the audio signals that constituted the smaller vocabulary case — achieving 96.5% on average of generalization successful rates when dealing with white Gaussian noise artificially added to audio signals (SNR of 15dB), against 93.6% when using Classical method (45.3% reduction on error rate) and 91.4% for Bottom-up method (59.3% reduction on error rate). Note that, for training sets in the SNR 15dB experiment, the TSWS was the only method which enabled the recognizer to achieve 100.0% of learning capability.

TEO has shown to support interesting tools to deal with speech signals. TSWS is a very robust TEO-based segmentation method and enables the increasing of success rates resulted from the implemented ASR system.

References

[1] V. Zue and R. Cole. *Survey of the State of the Art in Human Language Technology*, chapter Overview, pp. 1–3. Cambridge University Press and Giardini, web edition, 1997. [Online] available at `http://cslu.cse.ogi.edu/HLTsurvey/HLTsurvey.html`.

[2] J. A. Martins. "Avaliação de Diferentes Técnicas para Reconhecimento de Fala". Ph.D. thesis, Universidade Estadual de Campinas — UNICAMP, December 1997.

[3] V. Zue, R. Cole and W. Ward. *Survey of the State of the Art in Human Language Technology*, chapter Speech Recognition, pp. 3–10. Cambridge University Press and Giardini, web edition, 1997. [Online] available at `http://cslu.cse.ogi.edu/HLTsurvey/HLTsurvey.html`.

[4] L. Lamel, L. Rabiner, A. Rosenberg and J. Wilpon. "An improved endpoint detector for isolated word recognition". *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 29, no. 4, pp. 777–785, 1981.

[5] I. S. Peretta. "A novel word boundary detector based on the Teager Energy Operator for Automatic Speech Recognition". Master's thesis, Universidade Federal de Uberlândia - UFU, December 2010.

[6] P. Maragos, T. F. Quatieri and J. F. Kaiser. "Detecting Nonlinearities in Speech using an Energy Operator". In *Digital Signal Processing, Proceedings of 1990 IEEE International Workshop on*, pp. 1–2, September 1990.

[7] J. F. Kaiser. "On a simple algorithm to calculate the 'energy' of a signal". In *Acoustics, Speech, and Signal Processing, 1990. ICASSP-90., 1990 International Conference on*, volume 1, pp. 381–384, April 1990.

[8] J. F. Kaiser. "Some useful properties of Teager's energy operators". In *Acoustics, Speech, and Signal Processing, 1993. ICASSP-93., 1993 IEEE International Conference on*, volume 3, pp. 149–152, April 1993.

[9] I. S. Peretta, G. F. M. Lima, J. A. Tavares and K. Yamanaka. "A Spoken Word Boundaries Detection Strategy for Voice Command Recognition". *Learning & Nonlinear Models*, vol. 8, no. 3, pp. 148–156, 2010. [Online] journal available at `http://www.deti.ufc.br/~lnlm/index.php?v=8&n=3`.

[10] S. B. Davis and P. Mermelstein. "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences". *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 28, no. 4, pp. 357–366, August 1980.

[11] A. V. Oppenheim and R. W. Schafer. "From frequency to quefrency: a history of the cepstrum". *Signal Processing Magazine, IEEE*, vol. 21, no. 21, pp. 95–106, September 2004.

[12] L. Fausett. *Fundamentals of Neural Networks*. Prentice Hall, December 1993.

[13] F. Rosenblatt. "The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain". *Psychological Review*, vol. 65, no. 6, pp. 386–408, November 1958.

[14] M. F. Møller. "A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning". *Neural Networks*, vol. 6, no. 4, pp. 525–533, 1993.

[15] L. Tian and A. Noore. *Computational Intelligence in Reliability Engineering*, volume 39 of *Studies in Computational Intelligence*, chapter Computational Intelligence Methods in Software Reliability Prediction, pp. 375–397. Springer, 2007.

[16] L. R. Rabiner and M. R. Sambur. "Algorithm for determining the endpoints of isolated utterances". *The Journal of the Acoustical Society of America*, vol. 56, no. S1, pp. S31–S31, November 1974.

[17] J.-C. Junqua. "The influence of acoustics on speech production: A noise-induced stress phenomenon known as the Lombard reflex". *Speech Communication*, vol. 20, no. 1-2, pp. 13–22, 1996.