FUSION BASED SPEECH SEGMENTATION IN DARPA SPINE2 TASK

*Chengyi Zheng, Yonghong Yan**

OGI School of Science & Engineering, Oregon Health & Science University
Computer Science & Engineering Department
20000 NW Walker Rd., Beaverton, OR 97006, USA
{chengyi,yan}@cse.ogi.edu

ABSTRACT

We report a new fusion based segmentation approach using multiple filter bank coefficients. This approach takes advantage of current feature extraction procedure, with little additional computation cost. Another level of fusion was performed by combining several segmentation systems. Evaluation was conducted on the second SPeech In Noisy Environments (SPINE2) task. Experiments show our fusion based approaches significantly reduced the WER compared to two classifier-based approaches. Compared to the manual segmentation, our approach only has 0.3% WER increase.

1. INTRODUCTION

The input speech stream to an ASR system is a continuous flow of speech signal without any type of boundary information. For recognition efficiency, the speech stream is first transformed into a sequence of audio segments. The basic task of speech segmentation is chopping long periods of speech into short ones and removing non-speech events at the same time. Additional tasks of a speech segmenter may also include segmenting and clustering the speech stream according to speaker identities, environmental and channel conditions. In this paper, we only focus on the basic task, segmenting the speech stream at the boundaries of speech/non-speech events. This process is also commonly referred to as endpoint (or silence) detection.

Speech segmentation is necessary for an LVCSR system due to memory and speed restrictions of speech recognition. From the accuracy point of view, current technology still performs poorly when facing non-speech events, especially when there is strong background noise. Non-speech events are often mis-recognized as words, causing insertion errors. Another type of errors is the substitution error, when speech events were corrupted by the neighboring non-speech events, causing the recognizer mis-recognized both regions.

Since manual segmentation of speech is time consuming and unrealistic in most conditions, various approaches on automatic speech segmentation have been proposed. According to [1, 2], these approaches can be categorized as follows:

1. Metric-based segmentation. This approach is based on the acoustic distance measurement between every two contiguous windows along the speech signal. The maximum distances are detected as potential segmentation points, and final segmentation decision is based on some thresholds.
2. Classifier-based segmentation. This approach builds separate models for speech and non-speech events. The segmentation problem becomes a classification task. Gaussian mixture models or HMM are trained to model each class, and final segmentation decision is based on the change point of classes. Some classifier-based approaches perform actual decoding to generate phoneme or word sequences. The final segmentation decision is based on the silence locations generated from the decoder.

The metric-based approaches generally cannot compete with the classifier-based approaches on segmentation accuracy. However, classifier-based segmentations require complex computation and cause large latency, thus are not suitable for real time applications.

In this paper, we report a novel fusion based approach that is highly accurate and demands little computation. We compared our approach with two classifier-based methods, which will be introduced in Section 3.1 and Section 3.2.

2. EVALUATION TASK

The task used to evaluate our approach is the DARPA SPINE task. The purpose of SPINE task is to evaluate the current state of the art in speech recognition under noise especially military noise. The second evaluation (SPINE2) was conducted in November 2001 [3]. The test data comprises 128 speaker-environment pairs with 8 different noise environments. Each pair of speakers participated in a Milton Bradley™ battleship game. Each pair of speaker work in a cooperative way to locate and attack a target. Each conversation sessions are complicated by introduction of noise and the confusable military words. Each pair of speaker are located in separate sound recording rooms and use military headsets. The raw audio data has a total of 7 hours (423 minutes) of speech comprised of 128 unsegmented conversations with an average duration of 200 seconds. The overall reference words are 24,015.

In SPINE1, four scenarios are combined by realistic noise, handsets, communication channels and vocoders from the military operations, as shown in Table 1. SPINE2 added two additional noise, military tank and helicopter environments.

The system used to evaluate segmentation approaches is our Large Vocabulary Continuous Speech Recognition (LVCSR) system [4]. It uses decision tree based context clustering, and supports within word and cross word context-dependent phonemes (triphones). The decoder uses a two pass search strategy: the first pass generates a word graph using a simpler acoustic model (within word triphones) and language model (bigram); the second

*Also with Institute of Acoustics, Chinese Academy of Science, Beijing 100080, P.R. China

Table 1. Scenarios in SPINE1

	Recording Room 1		Recording Room 2	
Scenario	Noise	Handset	Noise	Handset
DOD	Quiet	STU-III	Office	STU-III
Navy	Aircraft Carrier	TA840	Office	STU-III
Army	HMMWV	H250	Quiet	STU-III
Air Force	E3A AWACS	R215	MCE	EV M87

pass re-scores the word graph using a more detailed acoustic model (cross word triphone) and language model (trigram). The decoder uses the common language models provided by CMU. The best official evaluation results of SPINE2 for using common language model is 38.1%.

The best way to evaluate a speech segmentation algorithm for a LVCSR task is to use its standard measurement: Word Error Rate (WER). In our experiments, we use the same recognizer with MFCC feature for all segmentation approaches.

3. TWO CLASSIFIER-BASED SEGMENTATIONS

Segmentation is important in SPINE task because there is lots of noise. Failing to exclude long periods of non-speech noise not only causes a large amount of insertion errors but also disrupts the search continuance.

Speech segmentation is the major interest in the SPINE1 workshop and remains an important topic in the SPINE2 workshop and following conferences. Almost all nine participants in SPINE2 evaluation used classifier-based segmentation [2, 3, 6, 7, 8, 9, 10].

3.1. TRAPS Based Segmentation

The segmentation that we used in the official SPINE2 evaluation is a TRAPS based approach proposed by Dr. Hermansky's group [3, 9]. The TRAPS based segmentation is based on two main processing steps. In the first step, learning the distribution of the temporal patterns of speech/non-speech present in each critical band independently. This was performed by training a Multi-Layer Perceptron (MLP) in each critical band. The input to the MLPs is a one second long temporal trajectory of critical band energy. The temporal trajectories were mean subtracted, variance normalized and hamming windowed before given as input to MLPs. The output layer of the MLP consists of two nodes targeting speech/non-speech respectively. In the second step, combining the outputs from each band-specific MLP and trained another MLP as a merging classifier. The output layer of this MLP again targets speech/non-speech events. The size of the hidden units is kept at 300 for band-specific MLPs and at 50 for the merging MLP. The output from this merging MLP was then median filtered to give final decisions.

3.2. Gaussian Mixture Classifier Based Segmentation

We also obtained a segmentation from Dr. Richard Stern and Dr. Rita Singh of CMU [2, 3]:

"A two-class speech/non-speech Gaussian mixture classifier was trained with KLT features from the SPINE2 development data. To train the classifier, the training data were segmented using Viterbi alignment. Feature vectors from segments corresponding to speech events (i.e. words and filled pauses) were used to train the speech distributions. All segments not corresponding to speech were used to train the non-speech distributions. Each of the distributions was a mixture of 32 Gaussians.

During segmentation, the likelihood of each of the two classes was computed over a sliding window corresponding to 0.5 seconds of speech, where the window was advanced in steps of 20 ms. Histograms of the difference in the likelihoods of the classes were derived and the inflexion points between the modes representing speech/non-speech events were located. The likelihood difference at the inflexion point was used as the threshold for the likelihood difference that separated speech from non-speech."

4. PROPOSED SEGMENTATION APPROACHES

4.1. Segmentation using Filter Bank (Subbands) Based Fusion

Filter bank calculation is a necessary step in many feature extraction algorithms. Filter bank is a set of band-pass filters that span the whole frequency spectrum. Each filter bank corresponds to a subband of the speech spectrum. In the MFCC case, certain number (we used 24 in our system) of mel scale triangular filters cover the whole frequency analysis spectrum. The filters have 50% overlap with their neighboring filters to obtain a smoothed frequency estimation. The magnitude coefficients in the SFFT spectrum are transformed into mel scale by correlating with these filter banks. The mel scale adopted in our system is

$$mel(f) = 1127log(1 + \frac{f}{700}), \quad (1)$$

which is designed to normalize 1000Hz correspond to 1000 mels.

According to equation 1, 24 mel scale filter bank coefficients were calculated, which represent a weighted sum of the spectral magnitude in that subband[1]. These coefficients were combined into a single feature vector, and each coefficient describes part of the information carried by the speech signal. In a traditional ASR system, the entire feature vector is used as one entity for training and classification. In our work, however, we treated each filter bank coefficient independently (Fig. 1). Using filter bank coefficients in speech segmentation has several advantages over the traditional energy based approach on detecting non-speech:

1. Each coefficient comes from the short-term spectral vector and represents the energy of the speech signal in a given frequency subband. The noise may corrupt some frequency bands but the majority of them are still useable. Based on this assumption, when majority filter bank coefficients drop to a local minimum, we can assume it is a possible non-speech frame.
2. The noise in SPINE task varies with types and distributions. Some are spread through the whole conversation but some appear only in speech or non-speech segments. They are also not just a simple additive noise that can be removed by spectral subtraction. The traditional energy based method treats the entire feature vector as a single entity, thus noise is no different from speech in their contribution on energy calculation. Even a single noise corrupted subband spectral can falsely signal non-speech event as a speech event.

Based on the analysis above, we designed a fusion based segmentation approach. The basic algorithm is as follows:

[1] 1 subband $\Rightarrow$ 6 filter banks $\Rightarrow$ 3 MFCC coefficients. These coefficients are not independent. And some methods to decorrelate them may necessary.

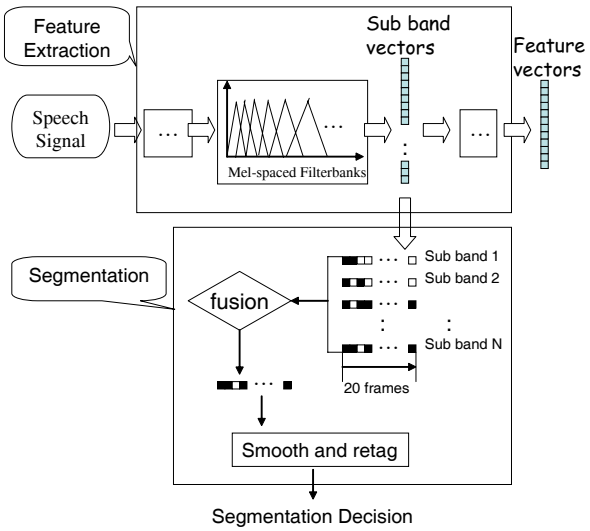

Fig. 1. Subbands Fusion Based Segmentation

1. For each frame *t*, obtain *N* filter bank coefficients from the normal feature extraction routine.
2. Form a fusion window *l*, which covers *T* consecutive frames and ends at frame *t*. Find the minimum filter bank coefficients for each filter bank i $(1 \leq i \leq N)$ within that window:

$$min_l(i) = \min_{t'=t-T+1}^{t} mel_{t'}(i) \quad (2)$$

   Note: This is similar to the minimum statistic algorithm proposed by Martin, in which the minimum of smoothed power within a finite length window is used to estimate the noise power [11].
3. Fuse the statistical information on all the filter bank coefficients within window *l* for each frame $t^{'}$ $(t-T+1 \leq t^{'} \leq t)$:

$$x_{t'}(i)(= \begin{cases} 1 & \text{if } (mel_{t'}(i) = min_l(i)) \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

$$num_{min}(t^{'}) = \sum_{i=1}^{N} x_{t'}(i) \quad (4)$$

   Note: *N* is the number of filter banks and $num_{min}(t^{'})$ is the number of minimum filter bank coefficients occurring at frame $t^{'}$.
4. This step we will tag each frame as speech or non-speech. First, compare $num_{min}(t^{'})$ with threshold Θ,[2] if

$$num_{min}(t^{'}) \geq \Theta, \quad (5)$$

   then we think this frame $t^{'}$ as a potential non-speech point. It still has several possibilities considering its relative location to the speech segments:

   (a) Frame $t^{'}$ is the last frame of this *l* frames window. $\Rightarrow$ It is the start point of a non-speech segment following a speech segment.

   (b) Frame $t^{'}$ is the first frame of this *l* frames window. $\Rightarrow$ It is located at the end of a non-speech segment and is followed by a speech segment.

   (c) It could be located anywhere within the *l* frames window. $\Rightarrow$ It is a non-speech frame surrounded by other non-speech frames. It occurs when low level noise appeared inside a high level noise period.

   Scan from the first frame and tag each frame as speech or non-speech according to the neighboring tagging status and value of $num_{min}(t^{'})$.
5. Re-scan the tagging information from the first frame and connect the neighboring short periods of speech together. Because in continuous speech, there are always small periods of non-speech parts existing between spoken words. We do not want to segment the whole input speech into individual words but rather separate it into utterances/sentences. Word level segmentation is not only more error prone but also loses the benefit of language model constraint.

   Similarly we connect the closely located non-speech parts together in this step.
6. Scan again starting from the first frame and produce segmented speech files according to the tagging result. We found our method is quite accurate at detecting the change of speech/non-speech events. We extend the speech duration by certain number of frames on both directions to append some silence. We would rather include a short period of silence than lose part of the speech events.

The TRAPS segmentation was used in our official evaluation system. After the evaluation, we tried CMU's segmentation and reported some results at the following SPINE2 workshop. This is the first time that our fusion based approach was reported.

After the official evaluation, we conducted a series of experiments to compare these three segmentation algorithms. Table 2 shows the number of files generated after the segmentations and their total file size.

Table 2. Comparison on Segmented Speech Files

Segmentation	Number of files	Total files size
RAW[3]	64	779M
TRAPS	4591	315M
Gaussian Mixture	5682	315M
Subbands	5563	305M

Our fusion based approach generates the least amount of speech data but results in the best recognition performance (Table 3). Further analysis shows the performance gain comes from:

1. Reduction on insertion errors which are caused by noise. It measures the accuracy of excluding non-speech (including silence, noise, etc.) events.
2. Reduction on deletion errors which are caused by discarded speech. It measures the accuracy of tagging speech event.

The TRAPS based segmentation is also using multiple bands of the speech spectrum. However, it is not performed as well as our subband based approach. We speculate the following differences

[2] $0 \leq \Theta \leq N$

[3] The raw speech data files contain both channels of conversation. There is only one participant who is supposed to speak at one time, most of the time, only one channel contains speech data. So roughly only half of the 779M data contains speech.

may be the reason. First, the MLPs used in the TRAPS approach were trained on all kinds of noise conditions; thus it is less accurate to a specific noise condition especially for the unseen testing noise conditions. Second, the TRAPS approach has a large amount of parameters in the MLPs which requires sufficient training data and carefully optimization. Third, TRAPS approach uses 15 critical bands on a down-sampled 8kHz speech while we use 24 bands on the original 16kHz speech. Another difference is that the TRAPS approach uses 101-frame window compared to our 8-frame window.

4.2. Fusion on Several Segmentations

Another level of fusion is achieved by combining the result from these three segmentations (Fig. 2). We tried several fusion methods here:

1. Majority Vote: The speech/non-speech tag of each frame is decided by the majority of segmentations.
2. Weighted Combination: A set of weights α_i are obtained from a development data set. The final tag is decided by comparing the threshold τ with the following value:

$$\sum_{i}^{N}(\alpha_i \cdot P_i(t)) \tag{6}$$

α_i is the weight value for segmentation i, $P_i(t)$ is the probability of frame t is speech estimated by segmentation i, in our case, due to lack data from other two segmentations, $P_i(t)$ is a value of 0/1. Ideally a probability or confidence score can give a more reliable combined score.

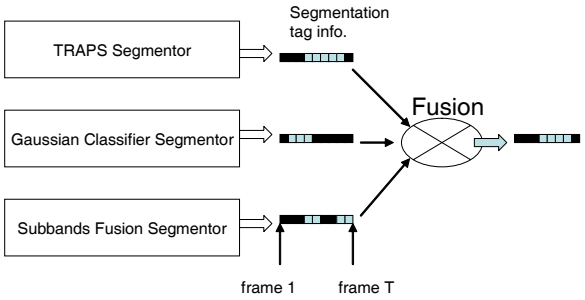

Fig. 2. Fusion Across Several Segmentations

Table 3. WER Comparison on SPINE2 Task

Segmentation Approaches	WER
TRAPS	41.6%
Gaussian Mixture Classifier	39.3%
Subbands Fusion	38.4%
Majority Vote Fusion	38.2%
Weighted Combination Fusion	38.1%
Manual	37.8%

The weighted combination fusion achieves the lowest WER among automatical approaches, which is only 0.3% higher compared to the manual segmentation. Significance tests show these two systems have no statistical difference at the level of p=0.05, and they are both significantly better (at level p=0.001) than the TRAPS and Gaussian Mixture Classifier systems.

5. CONCLUSIONS AND DISCUSSION

Our fusion based segmentation has a clear advantage over others due to its simplicity and fast execution. The filter bank coefficients were already available from feature extraction and the additional calculation is negligible, so our approach can be easily integrated into the front end of an ASR system and be performed on-the-fly. Another level of fusion was performed by combining it with two classifier-based segmentations. Our approach demonstrated its efficiency under a very challenge LVCSR task. Compared to manual segmentation, our approach only has a 0.3% WER difference. Furthermore, this work is integrated into our run time fusion framework [5] and is part of our effort on performing fusion in an LVCSR system.

6. ACKNOWLEDGEMENT

The authors thank Dr. Hermansky and his students for providing the TRAPS-based segmentation. Thanks also to Dr. Singh and Dr. Stern for providing the Gaussian mixture based segmentation.

7. REFERENCES

[1] S. Chen and P. S. Gopalakrishnan, "Speaker, environment and channel change detection and clustering via the bayesian information criterion," in *Proceedings of Broadcast News Transcription and Understanding Workshop*, Feb 1998.

[2] B. Raj and R. Singh, "Classifier-based non-linear projection for adaptive endpointing of continuous speech," *Computer Speech and Language*, vol. 17, no. 1, 2003.

[3] NRL, "http://elazar.itd.nrl.navy.mil/spine," in *Proceedings of The Second Speech in Noisy Environments Evaluation and Workshop*, 2001.

[4] X. Wu, C. Liu, Y. Yan, D. Kim, S. Cameron, and R. Parr, "The 1998 *OGI-FONIX* broadcast news transcription system," in *Proceedings of Broadcast News Transcription and Understanding Workshop*, Feb 1999.

[5] C. Zheng and Y. Yan, "Run time information fusion in speech recognition," In *Proceedings of ICSLP2002*, Sep 2002.

[6] R. Gadde, A. Stolcke, D. Vergyri, J. Zheng, K. Sonmez, and A. Venkataraman, "Building an *ASR* system for noisy environments:*SRI*'s 2001 *SPINE* evaluation system," in *Proceedings of ICSLP2002*, Sep 2002.

[7] Ö.Çetin, H. Nock, K. Kirchhoff, J. Bilmes, and M. Ostendorf, "The 2001 *GMTK*-based *SPINE ASR* system," in *Proceedings of ICSLP2002*, Sep 2002.

[8] B. Pellom and K. Hacioglu, "Recent improvements in the *CU SONIC ASR* system for noisy speech: The *SPINE* task," in *Proceedings of ICASSP2003*, Apr 2003.

[9] B.Kingsbury, P.Jain, and A.Adami, "A hybrid *HMM/TRAPS* model for robust voice activity detection," in *Proceedings of ICSLP2002*, Sep 2002.

[10] J. Zhang and S. Nakamura, "Modeling varying pauses to develop robust acoustic models for recognizing noisy conversational speech," in *Proceedings of ICSLP2002*, Sep 2002.

[11] R. Martin, "Noise Power Spectral Density Estimation Based on Optimal Smoothing and Minimum Statistics," in *IEEE Trans. Speech and Audio Processing*, Vol. 9, No. 5, July 2001, pp. 504-512.