PAPER *Special Issue on Speech Information Processing*

Noise and Channel Distortion Robust ASR System for DARPA SPINE2 Task

Konstantin MARKOV[†], ***Nonmember***, **Tomoko MATSUI**[†], ***Member***, **Rainer GRUHN**[†], **Jinsong ZHANG**[†], ***Nonmembers***, ***and*** **Satoshi NAKAMURA**[†], ***Member***

SUMMARY This paper presents the ATR speech recognition system designed for the DARPA SPINE2 evaluation task. The system is capable of dealing with speech from highly variable, real-world noisy conditions and communication channels. A number of robust techniques are implemented, such as differential spectrum mel-scale cepstrum features, on-line MLLR adaptation, and word-level hypothesis combination, which led to a significant reduction in the word error rate.

key words: *noise robustness, online adaptation, hypothesis combination, robust features*

1. Introduction

The second "Speech in Noisy Environments" (SPINE2) evaluation was conducted by the Naval Research Laboratories (NRL) in October 2001. The purpose of the evaluation was to provide a continuing forum for assessing the state-of-the-art practices in speech recognition technology for noisy military environments and for exchanging information on innovative speech recognition technology in the context of fully implemented systems that perform realistic tasks.

The task consisted of recognizing speech recorded during battleship games with realistic military noises playing in the background. The main feature of the evaluation is the variety of background noises and communication channels.

ATR took part in this evaluation to objectively test the performance of our speech recognition system in noisy environments on a common evaluation basis. Our system is based on several robust techniques including robust feature extraction, on-line acoustic model adaptation, and combining word hypotheses produced by two parallel recognition systems using different feature parameters. We assume that different features represent different "views" of the speech signal and thus contain complementary information. Through various experiments, we investigated the proper set of feature parameters as well as the effectiveness of on-line adaptation and hypothesis combination.

In the following sections, we describe the SPINE2 data, the ATR system and the experimental results. In Section 5, we discuss the problems arising in automatic speech detection of conversational speech. In Sect. 6, we present our conclusions.

Manuscript received June 30, 2002.

[†]The authors are with the ATR Spoken Language Translation Research Labs., Kyoto-fu, 619-0228 Japan.

2. SPINE2 Data

The SPINE2 data [1] is organized in conversations between two speakers collaborating in a task of seeking and shooting at targets. They speak freely, but the total vocabulary is fairly limited. Each speaker is seated in a different sound chamber in which previously recorded military background noise as the environment is accurately reproduced. The participants use a microphone and handset that carefully simulate the particular environment. The communication channel between speakers includes vocoders that degrade the speech quality. The speech data is recorded both directly from the microphones and through the vocoders. Thus, two sets of speech data - non-vocoded and vocoded are available. However, for the evaluation using vocoded data, simulated vocoded data is used instead of data from the real (hardware) vocoders. These data are produced by passing the non-vocoded speech through software versions of three vocoder types: CELP, MELP and LPC.

Training data consists of 324 dialogs involving 20 speakers (10 male and 10 female). There are about 28000 utterances with an average length of 4 seconds. Total duration of speech data for training is about 15 hours. There are 11 types of noisy environments including quiet, office, aircraft carrier, street, car, helicopter, tank, fighter jet and others. Some of these noises are highly variable in both acoustic level and spectral characteristics. There are sounds of whistles, rings, additional tones, background speech and so on. Furthermore, some segments of the recordings exhibit dropouts, where short segments of speech within an utterance are deleted. The signal-to-noise ratio (SNR) varies from 5 dB to 20 dB.

As test data, we used 32 conversations between 2 male and 2 female talkers, who were not among the training speakers, in the following four noise environments: quiet, office, helo (helicopter) and bradley (tank). Total speech duration is about 1.5 hours and the total number of utterances is about 2,900.

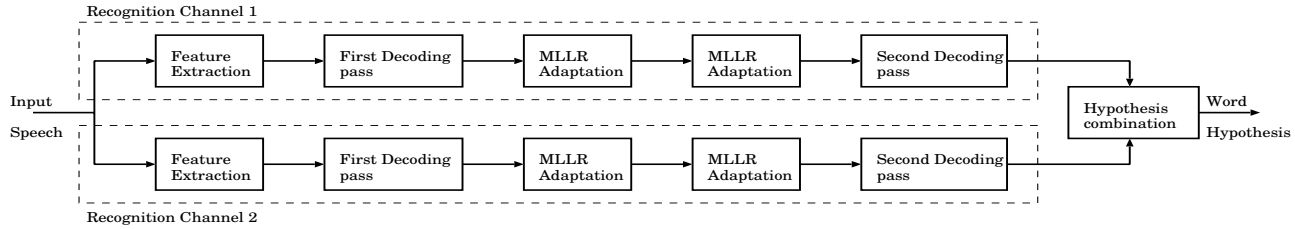

Fig. 1 ATR system block diagram.

3. System Description

A block diagram of the ATR SPINE2 system is shown in Figure 1. The system has two recognition channels (or subsystems) that recognize input speech parameterized with different feature parameters. Recognition hypotheses from each channel are combined to produce the system's final output. The two channels share the same decoder module but have specific acoustic models corresponding to the particular feature type. For each channel, recognition is done by passing data through the decoder two times. After the first pass, two iterations of MLLR adaptation are performed, and updated acoustic models are used for the second recognition pass. Finally, the hypothesis combination module combines hypotheses from the two recognition channels to produce the final result.

Using this system, we investigated the proper set of feature parameters as well as the effectiveness of on-line adaptation and hypothesis combination. In the following sections, each module of the system is described in greater detail and the results of some development experiments are also provided.

3.1 Feature extraction

For feature extraction, we compared four feature extraction methods. The first one is the standard mel-scale cepstrum (MFCC).

Our previous research showed that some modifications of the MFCC algorithm can yield better performance in noisy speech conditions [2]. The so-called differential spectrum MFCC is calculated from the differential power spectrum of speech, which is defined as:

$$D(i,k) = Y(i,k) - Y(i,k+1) \tag{1}$$

where $D()$ is the differential spectrum, $Y()$ is the power spectrum for the i^{th} frame and k is the spectrum bin index. This simple modification was efficient for the AURORA2 task [2] and as described in Section 4 was effective for the SPINE2 evaluation as well. We denote this type of differential spectrum MFCC feature as MFCC_DS.

Another type of MFCC-like feature we tried is based on the spectral subtraction idea, where the noise spectrum estimate is given by:

$$\hat{N}(f) = \frac{1}{N}\sum_{j=1}^{N} Y(j,f) \tag{2}$$

where N is the number of frames for a given utterance and $Y(j,f)$ is the power spectrum of the j^{th} frame [2]. The motivation for this is that under the assumption that the noise is stationary, the long-term spectrum average should follow the noise spectrum. We refer to this type of feature as MFCC_LTR.

The last feature extraction method we experimented with is the standard PLP-RASTA approach in the belief that this type of feature would be more robust for channel distortions [3].

All speech data were sampled at 16 kHz, and a frame size of 20 ms and a frame shift of 10 ms were common for all feature extraction methods. For MFCC, MFCC_DS and MFCC_LTR, a cepstral mean normalization (CMN) was also applied. For each utterance, the cepstral mean was subtracted to cancel the effect of the transmission channel.

3.2 Acoustic model

Cross-word triphones are usually used for context modeling in current speech recognition systems. They are commonly developed through a flat-start training procedure evolving from monophone HMMs, such as the one presented in the HTK book [4]. However, since the variability here in the training data is much more than that in normal clean speech, it is questionable whether this approach will work well. Consequently, we compared the performances of the following context modeling methods in experiments with a small non-vocoded data set of two dialogs which include 184 utterances. The channel conditions include pairs of quiet and office noise environments. The language model is word bigram.

- AM01: crossword triphone HMMs
- AM02: crossword right context-dependent diphone HMMs
- AM03: crossword left context-dependent diphone HMMs
- AM04: intraword triphone HMMs

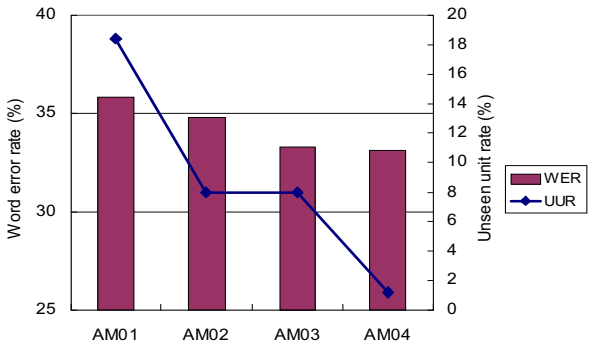

Fig. 2 Relationship between word error rate (WER) and unseen unit rate (UUR) for different methods of context modeling.

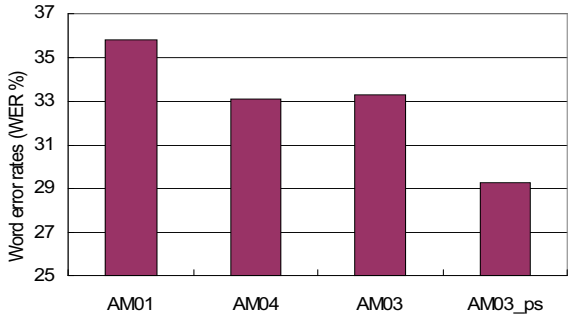

Fig. 3 Overview of the baseline HMMs.

Figure 2 shows the word error rate for each context modeling method. All of the acoustic models are decision-tree-based state-tying HMMs having a similar number of roughly 2000 states, with 16 Gaussian mixtures in each state. We can see that the intraword triphone HMMs, AM04, achieved the lowest WER, while the crossword triphone HMMs; AM01, got the highest WER. We attributed the reason partially to the highest *unseen unit rate (UUR)* for the triphone models. The UUR is defined as the ratio of unseen units desired by the testing data but do not appearing in the training data to all units desired by the testing data.

Figure 2 also shows the results for two kinds of diphone modeling. They have worse UURs than intraword triphones but much better ones than crossword triphones. Their capability to model crossword context still led that of AM03, the left context dependent diphone HMMs, and achieved comparable performance to AM04. We finally chose it as our context modeling method for one other advantage: a much smaller number of allophones (1.8 k) compared to intraword triphones (8.1 k).

Another important issue we addressed during the baseline model development is how to model inter-word pauses. A tee model *sp* with one skip-able state HMM is usually used to model them. However, we questioned its capability to model the extremely variable noisy pauses in the environments we studied. After collecting timing information from segmented data, we found that the *sp* segments have a mean duration longer than 40 ms. From the viewpoint of coarticulation, a long *pause* may block the coarticulation effects between the two crossword segments. So we finally adopted two additional symbols, *ps* and *np*, to explicitly model the long *quiet* and *noisy* interword pause segments. They are modeled by a 3-state left-to-right HMM as the "sil" symbol is. Including these additional symbols led to an error reduction of absolute 4%, as illustrated by model AM03_ps in Figure 3, when compared to the AM03. The AM03 reduced by absolute 6.5% errors compared to the primary crossword triphones developed by the standard flat-start procedure.

This left context-dependent diphone model AM03_ps served as the baseline model for further adaptations in the later stages.

3.3 Language model

The language model training data and task vocabulary were provided by the CMU and were common for all sites participating in this evaluation. The vocabulary consisted of about 5700 words. Using 43 phonemes to define word pronunciations based on this vocabulary we created a pronunciation lexicon of about 12000 entries.

Using the provided language model's training data, we trained both word and phrase bigram language models.

3.4 Parallel hypothesis combination

We investigated two approaches to combining hypotheses from the parallel recognition channels. Both of them are based on a word graph constructed from the two word hypotheses and finding the best path through the graph. However, their word graphs and scores on the paths are different.

3.4.1 Combination using likelihoods

Initially, each word in each of the hypotheses is represented by an arc in the graph. The acoustic likelihood score of that word is associated with the arc. In the next step, all arcs representing identical words hypothesized between the same time instants are collapsed into a single arc. Finally, nodes are formed between all arc pairs where the word-end time of one arc and the word-start time of the next arc are within 30 ms of each other. Figure 4 illustrates the formation of the word graph.

After the word graph is constructed in this manner, acoustic likelihood and language model probability for each word were combined to form the word (or arc) score. Finally, DP search is performed to find the best scoring path through the graph. A similar approach for hypothesis combination was presented in [5].

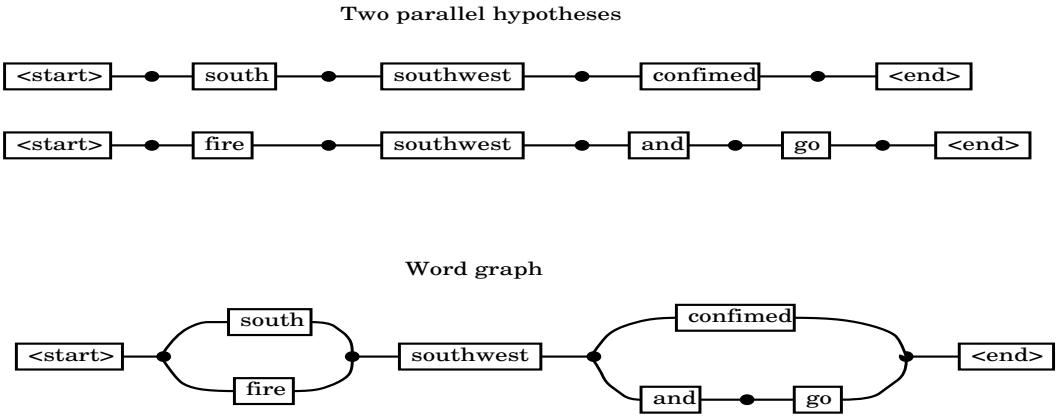

Fig. 4 Example of graph construction using likelihoods

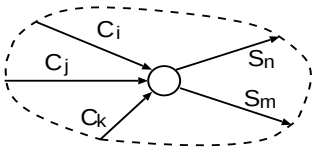

Fig. 5 Segment of word lattice for one node.

3.4.2 Combination using confidence measure

In this approach, each word in each of the hypotheses is assigned a confidence score. The confidence score is calculated by using acoustic likelihood, language model probability, and lattice path density. An example is shown schematically in Fig 5 for a given node in the lattice.

The three words (arcs) coming to this node have confidence scores of C_i, C_j and C_k, respectively. For the words (arcs) coming out of the node and having combined acoustic and language model scores S_n and S_m, confidence scores C_n and C_m are found from the following expressions:

$$\begin{cases} C_n + C_m &= C_i + C_j + C_k \\ C_n : C_m &= S_n : S_m \end{cases} \tag{3}$$

Further, words from the two hypotheses are aligned by a DP algorithm. Note that in contrast to the previous approach, during alignment no information about the word-start or word-end time is used. A word graph is made from the aligned word sequences by making nodes at the alignment boundaries as shown in Fig. 6. The final combined hypothesis is composed from those arcs that have the highest accumulated confidence scores. Confidence-score-based techniques for hypothesis combination are also used in some spoken dialog systems, for example [6].

4. Recognition Results

In this section we provide the experimental results obtained with the test data. Our baseline system consists of only one decoding pass without model adaptation or hypothesis combination.

In all experiments presented here we used speech utterances as input to the system. That is, speech portions of the dialogs are extracted in advance according to their actual time instances. Automatic recognition of whole dialogs implies another problem apart from the difficulty of recognizing speech in the presence of noise. That is a speech detection problem, which exceeds the scope of this task. Our approach to speech detection and several evaluative experiments are discussed in Section 5.

4.1 Non-vocoded data results

The baseline system word error rates (WER) for each kind of feature parameters are shown in Table 1. Each row shows the result for a particular noise environment, and the last row shows the average WER.

For relatively noise free environments (Quiet and Office), MFCC and MFCC-like features perform quite well, but for the heavy noise cases (Helo and Bradley), the WER nearly doubles. The best performing feature is MFCC_DS. MFCC_LTR is even worse than the standard MFCC, probably because the noise stationarity assumption is not proper for this data. PLP_RASTA features performed the worst.

Tables 2 and 3 show the WER at the first and second iterations of MLLR adaptation, respectively. As can be seen, the first MLLR iteration decreased WER dramatically and the second one improved the results only a little.

Results from the hypothesis combination experiments are presented in Table 4. Although the number of all possible combinations out of the four different features is six, we show only four of them since the other combinations did not perform better. The two columns of the table show the WER for the two techniques for hypothesis combination described in sections 3.4.1 and

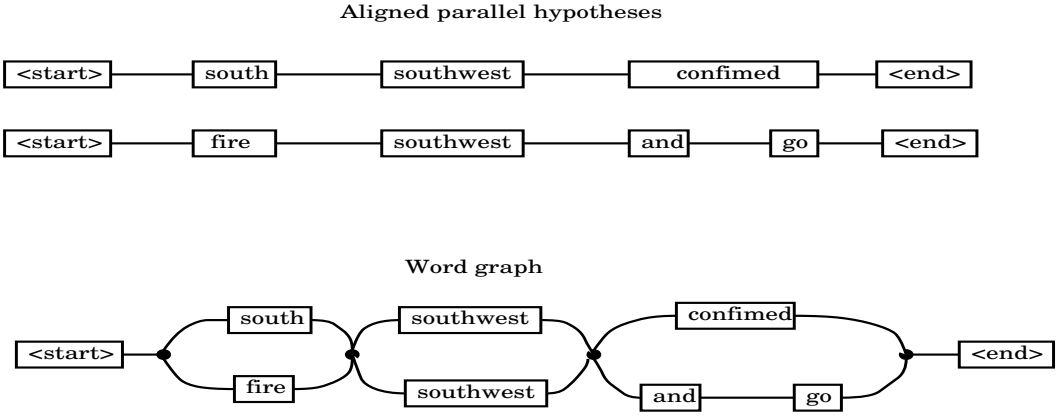

Fig. 6 Example of graph construction using confidence measures

Table 1 Baseline WER (%)

Noise Type	Feature type			
	mfcc	mfcc_ds	mfcc_ltr	plp_rasta
Quiet	29.1	26.9	30.1	39.5
Office	26.4	26.7	27.7	33.2
Helo	52.1	51.8	54.6	71.5
Bradley	57.2	56.8	56.4	69.6
Total	40.3	39.4	41.2	51.9

Table 2 WER (%) at the first MLLR iteration

Noise Type	Feature type			
	mfcc	mfcc_ds	mfcc_ltr	plp_rasta
Quiet	24.6	24.1	28.2	34.0
Office	23.6	22.8	25.3	30.5
Helo	42.1	42.2	43.9	55.3
Bradley	39.8	40.8	43.8	56.2
Total	31.7	31.6	34.6	43.1

Table 3 WER (%) at the second MLLR iteration

Noise Type	Feature type			
	mfcc	mfcc_ds	mfcc_ltr	plp_rasta
Quiet	24.5	23.5	27.6	34.3
Office	23.4	22.2	25.1	30.3
Helo	41.7	42.1	43.6	54.9
Bradley	38.9	39.3	43.4	53.7
Total	31.3	30.9	32.2	42.3

3.4.2, respectively. For comparison, the first row shows the WER when a single MFCC feature is used, and the following rows show the results of combining the MFCC feature with the other three kinds: MFCC_DS, MFCC_LTR and PLP_RASTA. The hypothesis combination based on likelihoods proved best for this task, improving the WER in all cases; however, the other hypothesis combination techniques showed improvement only when combining MFCC and MFCC_DS features, which alone provide similar performance. The lower performance of the confidence based hypothesis combination can be explained with the fact that in this case (as opposed to likelihoods based hypothesis combination) word graph is build using DP matching where word end timings are not considered and words with significant duration differences could reside at parallel branches of the graph.

Table 4 WER (%) for different hypothesis combination approaches

Feature Type	Hypothesis combination	
	Likelihood	Confidence
mfcc	31.3	31.3
mfcc+mfcc_ds	28.3	30.5
mfcc+mfcc_ltr	30.1	32.6
mfcc+plp_rasta	30.8	37.5

Table 5 Baseline WER (%) for vocoded data

Vocoder Type	Feature type			
	mfcc	mfcc_ds	mfcc_ltr	plp_rasta
CELP	49.1	47.9	53.5	57.4
MELP	51.2	49.6	54.7	60.1
LPC	56.3	55.1	59.9	66.5
Total	52.2	50.9	56.0	61.3

Table 6 WER (%) at the second MLLR iteration for vocoded data

Vocoder Type	Feature type			
	mfcc	mfcc_ds	mfcc_ltr	plp_rasta
CELP	43.6	41.9	48.5	52.4
MELP	46.4	44.7	47.7	54.2
LPC	52.3	50.8	54.9	61.5
Total	47.4	45.8	50.3	56.0

4.2 Vocoded data results

Here we present experimental results for vocoded test data using models build with vocoded variant of the training data. Tables 5 and 6 show the WER before and after two iterations of MLLR adaptation. The CELP and MELP vocoders do not differ much in their channel characteristics and thus recognition results. However, the LPC vocoder distorts speech signals more and the

WER is lower by 5-6%. Compared to the non-vocoded speech, vocoded speech WER are roughly 1.5 times higher. Surprisingly, PLP_RASTA features did not perform well even in this case. Hypothesis combination based on likelihoods between MFCC and MFCC_DS features slightly improved WER to 45.0%. As in the case of non-vocoded data, MLLR adaptation proved most effective.

5. Speech detection problem

The heavy, dynamic and unpredictable noises in SPINE2 make speech detection a challenging task. We experimented with the following three algorithms:

EPD : an energy- and energy-derivation-based approach

SPD : an algorithm using Gaussian mixture models (GMMs) to make a frame-wise speech — non-speech decision

combination : EPD for basic speech area selection and SPD for block-wise confirmation

5.1 EPD

The EPD algorithm is the baseline speech detection system, which showed fair but sub-optimal success. EPD detects speech area start points by comparing an energy-based and a context-smoothed energy-derivation-based metric with manually pre-set thresholds. The endpoint is detected when the energy-derivation-based measure falls below a certain threshold. Such an approach is described for example in [7].

For the EPD-only method, the thresholds for EPD were set for each target so that there would be as few misfires as possible, even at the price of losing an occasional very quiet utterance.

5.2 SPD

This approach is based on two GMMs, one trained on speech data and the other on noise segments. Training data were the manually segmented parts of the SPINE2 training data. GMMs for vocoded and unprocessed tasks were trained separately.

For each frame, the log likelihoods of speech and noise GMMs are calculated. The classification step itself is a comparison of the differences of those log likelihoods from a threshold [5][8].

5.3 Combination method

EPD occasionally misfires and classifies a noise area as speech. Some of these mistakes can be detected by their short length. The combination of EPD and SPD targets these misfired segments. After EPD has provided a basic speech — non-speech segmentation, each frame in a segment is classified with SPD. If a majority of all frames are speech frames, the whole segment is accepted; if there are fewer speech frames than noise frames, the whole segment is discarded.

5.4 Evaluation and Comparison

The approaches were evaluated with the baseline system by using a subset of the data consisting of 6 dialogs representing all types of the non-vocoded testing data noise environments: Quiet, Office, Bradley (tank) and Helo (helicopter).

Figure 7 shows the performance of the SPD approach for various threshold settings. Each of the GMMs (for speech and noise) has 64 Gaussian distributions. The best threshold is different for each noise type: for clean conditions, a low threshold is important, for Bradley noise a high threshold performs better. An acceptable threshold for Helo noise could not be found.

Figure 8 gives word accuracies for the various segmentation approaches depending on noise type. SPD performs best in a silent condition. For moderate noise, all algorithms show similar performance. SPD com-

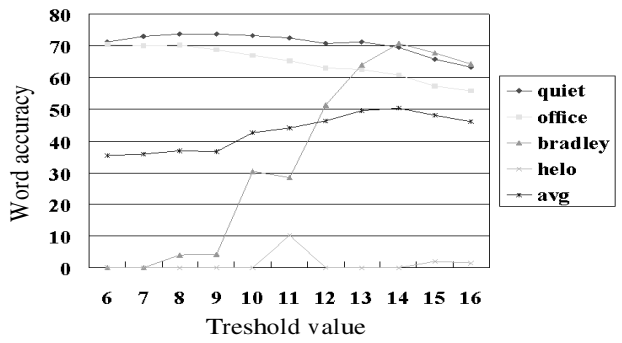

Fig. 7 SPD performance (word accuracy) depending on threshold setting.

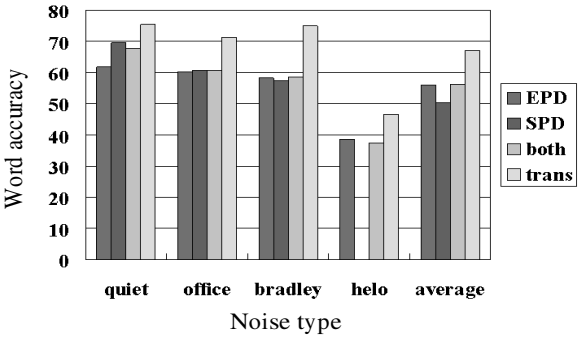

Fig. 8 Word accuracy archived with various segmentation methods for (from left) clean condition, office, bradley and helo noise. The rightmost bars show the average performance. "trans" means manual segmentation.

Table 7 Word accuracy for the baseline separation by transcription, EPD, and combination method

EPD	SPD	Combination	Transcription
56.0	50.3	56.2	67.1

pletely fails for Helo noise. This is probably because of the acoustical difference between Helo and the other noise types. Even single noise GMM trained on all noises was not adequate for Helo noise.

Table 7 lists the word accuracy rates for each of the approaches. In average, the combination method performs better than the automatic segmentations and achieves a word accuracy of 56.2%, which is fair considering the strong noise conditions. However, this is more than 10% below the rate of manual segmentation.

6. Conclusions

We presented the ATR speech recognition system developed for the DARPA SPINE2 evaluation task. Our main goal was high robustness with respect to real-world variable military noises.

In our system, we implemented several robust techniques in feature and model domains as well as combinations of multiple recognition outputs, which yielded significant reductions in word error rates. Our baseline system result of 39.4% was reduced by more than 10% absolute to 28.3% WER.

7. Acknowledgments

We would like to thank Mr. Hirofumi Yamamoto, Mr. Norbert Binder, and Mr. Masaki Ida for their help and valuable discussions during this task.

The research reported here was supported in part by a contract with the Telecommunications Advancement Organization of Japan entitled, "A study of speech dialogue translation technology based on a large corpus."

References

[1] URL http://elazar.itd.nrl.navy.mil/spine/

[2] K. Yao, J. Chen, K. K. Paliwal, and S. Nakamura. Feature extraction and model-based noise compensation for noisy speech recognition evaluated on AURORA 2 task. In *Eurospeech*, volume I, pages 233–236, Sept. 2001.

[3] H. Hermansky and N. Morgan. Rasta Processing of Speech. In *IEEE Trans. on SAP*, volume 2, number 4, pages 578–589, Oct.1994.

[4] http://htk.eng.cam.ac.uk/docs/docs.shtml.

[5] R. Singh, M. L. Seltzer, B. Raj, and R. M. Stern. Speech in noisy environments: Robust automatic segmentation, feature extraction, and hypothesis combination. In *Proc. ICASSP*, volume I, pages 273–276, May 2001.

[6] R. San-Segundo, B. Pellom, K. Hacioglu, and W. Ward. Confidence measures for spoken dialog systems. In *Proc. ICASSP*, volume I, pages 393–396, May 2001.

[7] J.-C. Junqua, B. Mak, and B. Reaves. A robust algorithm for word boundary detection in the presence of noise. *Proc. IEEE*, 2(3):406–412, 1994.

[8] N. Binder et al., Speech—Non-Speech Separation with GMMs. *Proc. Autumn Meeting Acoust. Soc. Jap.*, pages 141–142, 2001.

Konstantin P. Markov was born in Bulgaria. He received his B.E. degree in electrical engineering from the Department of Cybernetics, St. Petersburg Technical University, Russia in 1984. In 1996 and 1999 he received his M.E. and D.E. in Computer Science from Toyohashi University of Technology, Japan. From 1999 to 2000, he was a research engineer in the ATR Research and Development Center and from 2000 he joined ATR Spoken Language Translation Research Laboratories as invited researcher. Dr. Markov is a member of the Acoustic Society of Japan. His main research interests include automatic speech recognition, noise robustness, speaker identification, and statistical pattern recognition in general.

Tomoko Matsui received the Ph. D. degree in computer science from the Tokyo Institute of Technology, Tokyo, in 1997. She has been a permanent researcher at NTT since 1988. From January to June 2001, she was a member of the Acoustic and Speech Research Department, Bell Laboratories, Murray Hill, NJ, as a visiting researcher working on confidence measure for speech recognition. She is currently visiting ATR, Kyoto, as a senior researcher working on speech recognition. Her research interests include speech and speaker recognition. She received a paper award from the Institute of Electronics, Information and Communication Engineers of Japan (IEICE) in 1993.

Rainer Gruhn received the M.E. degree from the Friedrich-Alexander-University of Erlangen-Nuremberg, Germany, in 1998. Since 1998, he has been with Advanced Telecommunication Research (ATR) Institute, working in research and development of speech recognition systems. He is a member of the Acoustical Society of Japan.

Jinsong Zhang was born in China on October 4, 1968. He received his B.E. degree in electronic engineering from Hefei University of Technoloy, China in 1989, the M.E. degree from the University of Science and Technology of China (USTC) in 1992, and the Ph.D. degree from the University of Tokyo, Japan in 2000. From 1992 to 1996 he worked as a teaching assistant and lecturer in the department of electronic engineering of USTC. Since 2000, he joined ATR Spoken Language Translation Research Laboratories as an invited researcher. Dr. Zhang is a member of Acoustic Society of Japan. His main research interests include speech recognition, prosody information processing, and speech synthesis.

Satoshi Nakamura was born in Japan on August 4, 1958. He received the B.S.degree in electronics engineering from Kyoto Institute of Technology in 1981 and the Ph.D. degree in information science from Kyoto University in 1992. Between 1981-1986 and 1990-1993, he worked with the Central Research Laboratory, Sharp Corporation, Nara, Japan, where he was engaged in speech recognition research. During 1986-1989, he was a researcher of the speech processing department at ATR Interpreting Telephony Research Laboratories. From 1994-2000, he was an associate professor of the graduate school of information science, Nara Institute of Science and Technology, Japan. In 1996, he was a visiting research professor of the CAIP center of Rutgers, the state university of New Jersey USA. He is currently the head of Department 1 in ATR Spoken Language Translation Laboratories, Japan. He also serves as a guest professor for Toyohashi University of Technology and Ritsumeikan University from April, 2002. His current research interests include speech recognition, speech translation, spoken dialogue systems, stochastic modeling of speech, and microphone arrays. He received the Awaya Award from the Acoustical Society of Japan in 1992, and the Interaction2001 best paper award from the Information Processing Society of Japan in 2001. He is a member of the Acoustical Society of Japan, Institute of Electrical and Electronics Engineers (IEICE), Information Processing Society of Japan, and IEEE. He is currently a member of the Speech Technical Committee of the IEEE Signal Processing Society and an editor for the Journal of the IEICE Information and System Society.