PROBABILISTIC OPTIMUM FILTERING FOR ROBUST SPEECH RECOGNITION

Leonardo Neumeyer and Mitchel Weintraub

SRI International
Speech Research and Technology Program
Menlo Park, CA, 94025
USA

ABSTRACT

In this paper we present a new mapping algorithm for speech recognition that relates the features of simultaneous recordings of clean and noisy speech. The model is a piecewise linear transformation applied to the noisy speech feature. The transformation is a set of multidimensional linear least-squares filters whose outputs are combined using a conditional Gaussian model. The algorithm was tested using SRI's DECIPHER™ speech recognition system [1-5]. Experimental results show how the mapping is used to reduce recognition errors when the training and testing acoustic environments do not match.

1. INTRODUCTION

In many practical situations an automatic speech recognizer has to operate in several different but well-defined acoustic environments. For example, the same recognition task may be implemented using different microphones or transmission channels. In this situation it may not be practical to recollect a speech corpus to train the acoustic models of the recognizer. To alleviate this problem, we propose an algorithm that maps speech features between two acoustic spaces. The models of the mapping algorithm are trained using a small database recorded simultaneously in both environments.

In the case of steady-state additive homogenous noise, we can derive a MMSE estimate of the clean speech filterbank-log energy features using a model for how the features change in the presence of this noise [6-7]. In these algorithms, the estimated speech spectrum is a function of the global spectral SNR, the instantaneous spectral SNR, and the overall spectral shape of the speech signal. However, after studying simultaneous recordings made with two microphones, we believe that the relationship between the two simultaneous features is nonlinear. We therefore propose to use a piecewise-linear model to relate the two feature spaces.

There have been several algorithms in the literature which have focused on experimentally training a mapping between the noisy features and the clean features [8-13]. The proposed algorithm differs from previous algorithms in several ways:

- The MMSE estimate of the clean speech features in noise is trained experimentally rather than with a model as in [6, 7].
- Several frames are joined together similar to [13].
- The conditional PDF is based on a generic noisy feature not necessarily related to the feature that we are trying to estimate. For example, we could condition the estimate of the cepstral energy on the instantaneous spectral SNR vector.
- Multidimensional least-squares filters are used for the mapping transformation. This is used to exploit the correlation of the features over time and among components of the spectral features at the same time.
- Linear transformations are combined together without hard decisions.
- All delta parameters are computed after mapping the cepstrum and cepstral energy.
- The mapping parameters are trained using stereo recordings with two different microphones. Once trained, the mapping parameters are fixed.
- The mapping can be used to map either noisy speech features to clean features during training, or clean features to noisy features during recognition.

2. THE POF ALGORITHM

The mapping algorithm is based on a probabilistic piecewise-linear transformation of the acoustic space that we call *Probabilistic Optimum Filtering* (POF). Let us assume that the recognizer is trained with data recorded with a high-quality close-talking microphone (clean speech), and the test data is acquired in a different acoustic environment (noisy speech). Our goal is to estimate a clean feature vector $\hat{\boldsymbol{x}}_n$ given its corresponding noisy feature $\boldsymbol{y}_n$ where n is the frame index. (A list of symbols is shown in Table 1.) To estimate the clean vector we vector-quantize the clean feature space in I regions using the generalized Lloyd algorithm [14]. Each VQ region is assigned a multidimensional transversal filter (see Figure 1). The error between the clean vector and the estimated vectors produced by the i-th filter is given by

$$\boldsymbol{e}_{ni} = \boldsymbol{x}_n - \hat{\boldsymbol{x}}_{ni} = \boldsymbol{x}_n - \boldsymbol{W}_i^T \boldsymbol{Y}_n \tag{1}$$

where $\boldsymbol{e}_{ni}$ is the error associated with region i, $\boldsymbol{W}_i$ is the filter coefficient matrix, and $\boldsymbol{Y}_n$ is the tapped-delay line of the noisy vectors. Expanding these matrices we get

$$\boldsymbol{W}_i^T = \left[\boldsymbol{A}_{i,-p} \ \cdots \ \boldsymbol{A}_{i,-1} \ \boldsymbol{A}_{i,0} \ \boldsymbol{A}_{i,1} \ \cdots \ \boldsymbol{A}_{i,p} \ \boldsymbol{b}_i\right] \tag{2}$$

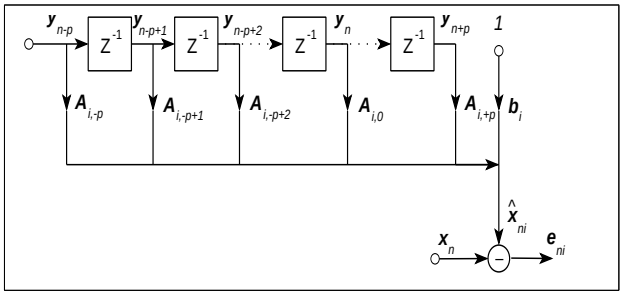

Figure 1: Multi-dimensional transversal filter for cluster *i*.

$$\mathbf{Y}_n^T = \left[\mathbf{y}_{n-p}^T \cdots \mathbf{y}_{n-1}^T \, \mathbf{y}_n^T \, \mathbf{y}_{n+1}^T \cdots \mathbf{y}_{n+p}^T \; 1 \right] \tag{3}$$

The conditional error in each region is defined as

$$E_i = \sum_{n=p}^{N-1-p} \left\| \mathbf{e}_{ni} \right\|^2 p(g_i|\mathbf{z}_n) \tag{4}$$

where $p(g_i|\mathbf{z}_n)$ is the probability that the clean vector $\mathbf{x}_i$ belongs to region g_i given an arbitrary conditional noisy feature vector $\mathbf{z}_n$. Note that the conditioning noisy feature can be any acoustic vector generated from the noisy speech frame. For example, it may include an estimate of the signal-to-noise ratio (SNR), energy, cepstral energy, cepstrum, etc.

The conditional probability density function $p(\mathbf{z}_n|g_i)$ is modeled as a mixture of I Gaussian distributions. Each Gaussian distribution models a VQ region. The parameters of the distributions (mean vectors and covariance matrices) are estimated using the corresponding $\mathbf{z}_n$ vectors associated with that region. The posterior probabilities $p(g_i|\mathbf{z}_n)$, are computed using Bayes' theorem and the mixture weights, $p(g_i)$, are estimated using the relative number of training clean vectors that are assigned to a given VQ region.

To compute the optimum filters in the mean-squared error sense, we minimize the conditional error in each VQ region. The minimum mean-squared error vector is obtained by taking the gradient of E_i defined in Eq. (4) with respect to the filter coefficient matrix and equating all the elements of the gradient matrix to zero. As a result, the optimum filter coefficient matrix has the form, $\mathbf{W}_i = \mathbf{R}_i^{-1} \mathbf{r}_i$ where

$$\mathbf{R}_i = \sum_{n=p}^{N-1-p} \mathbf{Y}_n \mathbf{Y}_n^T p(g_i|\mathbf{z}_n) \tag{5}$$

is a probabilistic non-singular auto-correlation matrix, and

$$\mathbf{r}_i = \sum_{n=p}^{N-1-p} \mathbf{Y}_n \mathbf{x}_n^T p(g_i|\mathbf{z}_n) \tag{6}$$

is a probabilistic cross-correlation matrix.

The algorithm can be completely trained without supervision and requires no additional information other than the simultaneous waveforms.

The run-time estimate of the clean feature vector can be computed by integrating the outputs of all the filters as follows:

$$\hat{\mathbf{x}}_n = \sum_{i=0}^{I-1} \mathbf{W}_i^T \mathbf{Y}_n p(g_i|\mathbf{z}_n) = \left\{ \sum_{i=0}^{I-1} \mathbf{W}_i^T p(g_i|\mathbf{z}_n) \right\} \mathbf{Y}_n \tag{7}$$

Symbol	Dimension	Description
n	*1*	frame index
i	*1*	region index
L	*1*	feature vector size
M	*1*	conditioning feature vector size
N	*1*	number of training frames
I	*1*	number of VQ regions
p	*1*	maximum filter delay
$\mathbf{e}_{ni}$	$L \times 1$	estimation error vector
$\mathbf{x}_n$	$L \times 1$	clean feature vector
$\hat{\mathbf{x}}_n$	$L \times 1$	estimate of clean feature vector
$\mathbf{y}_n$	$L \times 1$	noisy feature vector
$\mathbf{z}_n$	$M \times 1$	conditioning noisy feature vector
μ_i	$M \times 1$	mean vector of gaussian *i*
Σ_i	$M \times M$	covariance matrix of gaussian *i*
$\mathbf{W}_i$	$(2p+1)L+1 \times L$	transversal filter coefficient matrix
$\mathbf{Y}_n$	$(2p+1)L+1 \times 1$	tap input vector
$\mathbf{A}_{ik}$	$L \times L$	multiplicative tap matrix
$\mathbf{b}_i$	$L \times 1$	additive tap matrix
$\mathbf{R}_i$	$(2p+1)L+1 \times (2p+1)L+1$	auto-correlation matrix
$\mathbf{r}_i$	$(2p+1)L+1 \times L$	cross-correlation matrix

Table 1. List of Symbols.

3. EXPERIMENTS

3.1. Introduction

In this section we present a series of experiments that show how the mapping algorithm can be used in a continuous speech recognizer across acoustic environments. In all of the experiments the recognizer models are trained with data recorded with high-quality microphones and digitally sampled at 16,000 Hz. The analysis frame rate is 100 Hz.

The tables below show three types of performance indicators:

- *Relative distortion measure*. For a given component of a feature vector we define the relative distortion between the clean and noisy data as follows:

$$d = \sqrt{\frac{E\left[(x-y)^2 \right]}{\mathrm{var}(x)}} \tag{8}$$

- *Word recognition error.*
- *Error ratio.* The error ratio is given by E_n / E_c where E_n is the word recognition error for the test-noisy/train-clean con-

dition, and E_c is the word recognition error of the test-clean/train-clean condition.

3.2. Single Microphone

To test the POF algorithm on a single target acoustic environment we used the DARPA *Wall Street Journal* database [15] on SRI's DECIPHER™ phonetically tied-mixture speech recognition system [2]. The signal processing consisted of a filterbank-based front-end that generated six feature streams: cepstrum *(c1-c12)*, cepstral energy *(c0)*, and their first- and second-order derivatives. Cepstral-mean normalization [16] was used to equalize the channel. We used simultaneous recordings of high-quality speech (Sennheiser 414 head-mounted microphone with a noise-cancelling element) along with speech recorded by a standard speaker phone (AT&T 720) and transmitted over local telephone lines. We will refer to this stereo data as *clean* and *noisy* speech respectively. The models of the recognizer were trained using 42 male WSJ0 training talkers (3500 sentences) recorded with a Sennheiser microphone. The models of the mapping algorithm were trained using 240 development training sentences recorded by three speakers. The test set consisted of 100 sentences (not included in the training set) recorded by the same three speakers.

In this experiment we mapped two of the six features: the cepstrum *(c1-c12)* and the cepstral energy *(c0)* separately. The derivatives were computed from the mapped vectors of the cepstral features. For the conditioning feature we used a 13-dimensional cepstral vector *(c0-c12)* modeled with 512 Gaussians with diagonal covariance matrices. The results are shown in Table 2.

Filter Coefficients	Average Distortion	Recognition Error (%)	Error Ratio
No mapping	0.72	27.6	2.46
$\boldsymbol{A}_{i,0}=\boldsymbol{I}, \boldsymbol{b}_i$	0.62	18.1	1.62
$\boldsymbol{A}_{i,0}, \boldsymbol{b}_i$	0.57	17.0	1.52
$\boldsymbol{A}_{i,-1}, ..., \boldsymbol{A}_{i,-1}, \boldsymbol{b}_i$	0.51	17.3	1.54
$\boldsymbol{A}_{i,-2}, ..., \boldsymbol{A}_{i,-2}, \boldsymbol{b}_i$	0.50	16.4	1.46
$\boldsymbol{A}_{i,-3}, ..., \boldsymbol{A}_{i,-3}, \boldsymbol{b}_i$	0.49	15.9	1.42
$\boldsymbol{A}_{i,-4}, ..., \boldsymbol{A}_{i,-4}, \boldsymbol{b}_i$	0.49	16.1	1.44

Table 2. Performance of the POF algorithm for different number of filter coefficients. The number of Gaussian distributions is 512 per feature and the conditioning feature is a 13-dimensional cepstral vector.

The baseline experiment produced a word error rate of 27.6% on the noisy test set, that is, 2.46 times the error obtained when using the clean data channel. A 34% improvement in recognition performance was obtained when using only the additive filter coefficient $\boldsymbol{b}_i$. (Recognition error goes down to 18.1%.) The best result (15.9% recognition error) was obtained for the condition *p=3*, in which six neighboring noisy frames are being used to estimate the feature vector for the current frame. The correlation between the average relative distortion between the six clean and noisy features and the recognition error is 0.9.

3.3. Multiple Microphones

To test the performance of the POF algorithm on multiple microphones we used SRI's stereo-ATIS database. (See the companion paper [1] for details.) In this database, we recorded the clean channel with a Sennheiser microphone and the noisy channel with 10 different telephone handsets. For this set of experiments we also mapped the cepstrum vector *(c1-c12)* and the cepstral energy *(c0)*. The maximum delay of the filters was kept fixed at *p=2*, and the number of Gaussians was 512. We tried the following conditioning features:

- **Cepstrum**. Same conditioning feature used in the single microphone experiment *(c0-c12)*.
- **Spectral SNR**. This is an estimate of the instantaneous signal-to-noise ratio computed on the log-filterbank energy domain. The vector size is 25.
- **Cepstral SNR**. This feature is generated by applying the discrete cosine transform (DCT) to the spectral SNR. The transformation reduces the dimensionality of the vector from 25 to 12 elements.

The results are shown in Table 3. The baseline result is a 19.4% word error rate. This result is achieved when the same wide-band front-end is used for training the models with clean data and for recognition using telephone data. When a telephone front-end [1] is used for training and testing, the error decreases to 9.7%. The disadvantage of using this approach is that the acoustic models of the recognizer have to be re-estimated. However, the POF-based front-end operates on the clean models and results in better performance. The cepstral SNR produces the best result (8.7%). With this conditioning feature we combine the effects of noise and spectral shape in a compact representation.

Experiment	Word Error (%)	Error Ratio
Wide-band front-end	19.4	2.49
Telephone-bandwidth front-end	9.7	1.24
Mapping with cepstrum	9.4	1.20
Mapping with spectral SNR	8.9	1.14
Mapping with cepstral SNR	8.7	1.11

Table 3. Performance for the multiple-telephone handset test set.

3.4. Using POF in Either Training or Testing

The POF mapping can be applied to either the training data or the testing data. When applied to the training data, it makes the clean speech features look like the noisy speech features. During recognition, the standard signal processing of the noisy speech features may be used.

When applied to the testing data, it makes the noisy speech features look like the clean speech features. Training of the HMM acoustic models uses the standard signal processing.

Signal Processing		Word	Error
Training	Testing	Error	Ratio
Standard	Standard	31.4	2.7
Map Clean to Noisy	Standard	21.3	1.8
Standard	Map Noisy to Clean	20.0	1.7

Table 4: Training and Testing Paradigms using the Probabilistic Optimum Filter. Word Error is on AT&T Speaker Phone

The results in Table 4 show that equivalent performance is obtained when using the mapping either in training (21.3%) or in testing (20.0%). In both cases, this is a significant decrease from the performance without compensation (31.4%). The recognition numbers are slightly different from those in Table 2 since this experiment uses an earlier version of the mapping and recognizer.

4. CONCLUSIONS

We have presented a feature mapping algorithm capable of exploiting nonlinear relations between two acoustic spaces. We have shown how to improve the performance of the recognizer in the presence of a noisy signal by using a small database with simultaneous recordings in the clean and noisy acoustic environments.

The mapping algorithm has performed well on a speaker-dependent/single-microphone task and on a speaker-independent/multiple-microphone task. In both cases the target acoustic environment was known a priori. The POF algorithm efficiently exploited the correlations within and between frames, resulting in significant improvements over the unmapped systems.

The POF algorithm can be used only when a stereo database containing the clean and noisy speech is available. This requirement limits the use of the POF algorithm to applications in which the target acoustic environment is well defined and stable. These applications may include those for which the microphone, the channel or the background noise encountered in the field do not match the training conditions.

ACKNOWLEDGMENTS

This research was supported by a grant, NSF IRI-9014829, from the National Science Foundation.

REFERENCES

1. M. Weintraub and L. Neumeyer, “Constructing Telephone Acoustic Models from a High-Quality Speech Corpus,” 1994 IEEE ICASSP.
2. V. Digalakis and H. Murveit, “GENONES: Optimizing the Degree of Mixture Tying in a Large Vocabulary Hidden Markov Model Based Speech Recognizer,” 1994 IEEE ICASSP.
3. H. Murveit, J. Butzberger, V. Digalakis, and M. Weintraub, “Large-Vocabulary Dictation Using SRI’s DECIPHER Speech Recognition System: Progressive Search Techniques,” 1993 IEEE ICASSP, pp. II319-II322.
4. H. Murveit, J. Butzberger, and M. Weintraub, “Performance of SRI’s DECIPHER™ Speech Recognition System on DARPA’s CSR Task,” 1992 DARPA Speech and Natural Language Workshop Proceedings, pp. 410-414.
5. Murveit, H., J. Butzberger, and M. Weintraub, “Reduced Channel Dependence for Speech Recognition,” 1992 DARPA Speech and Natural Language Workshop Proceedings, pp. 280-284.
6. A. Erell and M. Weintraub, “Spectral Estimation for Noise Robust Speech Recognition,” 1989 DARPA SLS Workshop, pp. 319-324.
7. A. Erell and M. Weintraub, “Recognition of Noisy Speech: Using Minimum-Mean Log-Spectral Distance Estimation,” 1990 DARPA SLS Workshop, pp. 341-345.
8. B.H. Juang and L.R. Rabiner, “Signal Restoration by Spectral Mapping,” 1987 IEEE ICASSP, pp. 2368-2371.
9. H. Gish, Y.L. Chow, and J.R. Rohlicek, “Probabilistic Vector Mapping of Noisy Speech Parameters for HMM Word Spotting,” 1990 IEEE ICASSP, pp. 117-120.
10. K. Ng, H. Gish, and J.R. Rohlicek, “Robust Mapping of Noisy Speech Parameters for HMM Word Spotting,” 1992 IEEE ICASSP, pp. II-109-II-112.
11. A. Acero, “Acoustical and Environmental Robustness in Automatic Speech Recognition,” Ph.D. Thesis, Carnegie-Mellon University, September 1990.
12. R.M. Stern, F.H. Leu, Y. Ohshima, T.M. Sullivan, and A. Acero, “Multiple Approaches to Robust Speech Recognition,” 1992 International Conference on Spoken Language Processing, pp. 695-698.
13. A. Nadas, D. Nahamoo, and M. Picheny, “Adaptive Labeling: Normalization of Speech by Adaptive Transformations based on Vector Quantization,” 1988 IEEE ICASSP, pp. 521-524.
14. Y. Linde, A. Buzo, and R.M. Gray, “An Algorithm for Vector Quantizer Design,” IEEE Trans. Comm., vol. 28, pp. 84-95, January 1980.
15. G. Doddington, “CSR Corpus Development,” 1992 DARPA SLS Workshop, pp. 363-366.
16. S.F. Furui, “Cepstral Analysis Technique for Automatic Speaker Verification,” IEEE Trans. ASSP, Vol. 29, pp. 254-272, April 1981.

PROBABILISTIC OPTIMUM FILTERING FOR ROBUST SPEECH RECOGNITION

Leonardo Neumeyer and Mitchel Weintraub

SRI International
Speech Research and Technology Program
Menlo Park, CA, 94025
USA

ABSTRACT

In this paper we present a new mapping algorithm for speech recognition that relates the features of simultaneous recordings of clean and noisy speech. The model is a piecewise linear transformation applied to the noisy speech feature. The transformation is a set of multidimensional linear least-squares filters whose outputs are combined using a conditional Gaussian model. The algorithm was tested using SRI's DECIPHER™ speech recognition system [1-5]. Experimental results show how the mapping is used to reduce recognition errors when the training and testing acoustic environments do not match.

1. INTRODUCTION

In many practical situations an automatic speech recognizer has to operate in several different but well-defined acoustic environments. For example, the same recognition task may be implemented using different microphones or transmission channels. In this situation it may not be practical to recollect a speech corpus to train the acoustic models of the recognizer. To alleviate this problem, we propose an algorithm that maps speech features between two acoustic spaces. The models of the mapping algorithm are trained using a small database recorded simultaneously in both environments.

In the case of steady-state additive homogenous noise, we can derive a MMSE estimate of the clean speech filterbank-log energy features using a model for how the features change in the presence of this noise [6-7]. In these algorithms, the estimated speech spectrum is a function of the global spectral SNR, the instantaneous spectral SNR, and the overall spectral shape of the speech signal. However, after studying simultaneous recordings made with two microphones, we believe that the relationship between the two simultaneous features is nonlinear. We therefore propose to use a piecewise-linear model to relate the two feature spaces.

There have been several algorithms in the literature which have focused on experimentally training a mapping between the noisy features and the clean features [8-13]. The proposed algorithm differs from previous algorithms in several ways:

- The MMSE estimate of the clean speech features in noise is trained experimentally rather than with a model as in [6, 7].
- Several frames are joined together similar to [13].
- The conditional PDF is based on a generic noisy feature not necessarily related to the feature that we are trying to estimate. For example, we could condition the estimate of the cepstral energy on the instantaneous spectral SNR vector.
- Multidimensional least-squares filters are used for the mapping transformation. This is used to exploit the correlation of the features over time and among components of the spectral features at the same time.
- Linear transformations are combined together without hard decisions.
- All delta parameters are computed after mapping the cepstrum and cepstral energy.
- The mapping parameters are trained using stereo recordings with two different microphones. Once trained, the mapping parameters are fixed.
- The mapping can be used to map either noisy speech features to clean features during training, or clean features to noisy features during recognition.

2. THE POF ALGORITHM

The mapping algorithm is based on a probabilistic piecewise-linear transformation of the acoustic space that we call *Probabilistic Optimum Filtering* (POF). Let us assume that the recognizer is trained with data recorded with a high-quality close-talking microphone (clean speech), and the test data is acquired in a different acoustic environment (noisy speech). Our goal is to estimate a clean feature vector $\hat{\boldsymbol{x}}_n$ given its corresponding noisy feature $\boldsymbol{y}_n$ where n is the frame index. (A list of symbols is shown in Table 1.) To estimate the clean vector we vector-quantize the clean feature space in I regions using the generalized Lloyd algorithm [14]. Each VQ region is assigned a multidimensional transversal filter (see Figure 1). The error between the clean vector and the estimated vectors produced by the i-th filter is given by

$$\boldsymbol{e}_{ni} = \boldsymbol{x}_n - \hat{\boldsymbol{x}}_{ni} = \boldsymbol{x}_n - \boldsymbol{W}_i^T \boldsymbol{Y}_n \tag{1}$$

where $\boldsymbol{e}_{ni}$ is the error associated with region i, $\boldsymbol{W}_i$ is the filter coefficient matrix, and $\boldsymbol{Y}_n$ is the tapped-delay line of the noisy vectors. Expanding these matrices we get

$$\boldsymbol{W}_i^T = \left[\boldsymbol{A}_{i,-p} \ \cdots \ \boldsymbol{A}_{i,-1} \ \boldsymbol{A}_{i,0} \ \boldsymbol{A}_{i,1} \ \cdots \ \boldsymbol{A}_{i,p} \ \boldsymbol{b}_i\right] \tag{2}$$

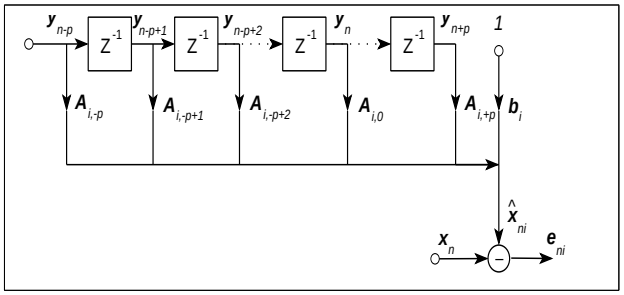

Figure 1: Multi-dimensional transversal filter for cluster *i*.

$$\mathbf{Y}_n^T = \left[\mathbf{y}_{n-p}^T \cdots \mathbf{y}_{n-1}^T \, \mathbf{y}_n^T \, \mathbf{y}_{n+1}^T \cdots \mathbf{y}_{n+p}^T \; 1 \right] \tag{3}$$

The conditional error in each region is defined as

$$E_i = \sum_{n=p}^{N-1-p} \left\| \mathbf{e}_{ni} \right\|^2 p(g_i|\mathbf{z}_n) \tag{4}$$

where $p(g_i|\mathbf{z}_n)$ is the probability that the clean vector $\mathbf{x}_i$ belongs to region g_i given an arbitrary conditional noisy feature vector $\mathbf{z}_n$. Note that the conditioning noisy feature can be any acoustic vector generated from the noisy speech frame. For example, it may include an estimate of the signal-to-noise ratio (SNR), energy, cepstral energy, cepstrum, etc.

The conditional probability density function $p(\mathbf{z}_n|g_i)$ is modeled as a mixture of I Gaussian distributions. Each Gaussian distribution models a VQ region. The parameters of the distributions (mean vectors and covariance matrices) are estimated using the corresponding $\mathbf{z}_n$ vectors associated with that region. The posterior probabilities $p(g_i|\mathbf{z}_n)$, are computed using Bayes' theorem and the mixture weights, $p(g_i)$, are estimated using the relative number of training clean vectors that are assigned to a given VQ region.

To compute the optimum filters in the mean-squared error sense, we minimize the conditional error in each VQ region. The minimum mean-squared error vector is obtained by taking the gradient of E_i defined in Eq. (4) with respect to the filter coefficient matrix and equating all the elements of the gradient matrix to zero. As a result, the optimum filter coefficient matrix has the form, $\mathbf{W}_i = \mathbf{R}_i^{-1} \mathbf{r}_i$ where

$$\mathbf{R}_i = \sum_{n=p}^{N-1-p} \mathbf{Y}_n \mathbf{Y}_n^T p(g_i|\mathbf{z}_n) \tag{5}$$

is a probabilistic non-singular auto-correlation matrix, and

$$\mathbf{r}_i = \sum_{n=p}^{N-1-p} \mathbf{Y}_n \mathbf{x}_n^T p(g_i|\mathbf{z}_n) \tag{6}$$

is a probabilistic cross-correlation matrix.

The algorithm can be completely trained without supervision and requires no additional information other than the simultaneous waveforms.

The run-time estimate of the clean feature vector can be computed by integrating the outputs of all the filters as follows:

$$\hat{\mathbf{x}}_n = \sum_{i=0}^{I-1} \mathbf{W}_i^T \mathbf{Y}_n p(g_i|\mathbf{z}_n) = \left\{ \sum_{i=0}^{I-1} \mathbf{W}_i^T p(g_i|\mathbf{z}_n) \right\} \mathbf{Y}_n \tag{7}$$

Symbol	Dimension	Description
n	*1*	frame index
i	*1*	region index
L	*1*	feature vector size
M	*1*	conditioning feature vector size
N	*1*	number of training frames
I	*1*	number of VQ regions
p	*1*	maximum filter delay
$\mathbf{e}_{ni}$	$L \times 1$	estimation error vector
$\mathbf{x}_n$	$L \times 1$	clean feature vector
$\hat{\mathbf{x}}_n$	$L \times 1$	estimate of clean feature vector
$\mathbf{y}_n$	$L \times 1$	noisy feature vector
$\mathbf{z}_n$	$M \times 1$	conditioning noisy feature vector
μ_i	$M \times 1$	mean vector of gaussian *i*
Σ_i	$M \times M$	covariance matrix of gaussian *i*
$\mathbf{W}_i$	$(2p+1)L+1 \times L$	transversal filter coefficient matrix
$\mathbf{Y}_n$	$(2p+1)L+1 \times 1$	tap input vector
$\mathbf{A}_{ik}$	$L \times L$	multiplicative tap matrix
$\mathbf{b}_i$	$L \times 1$	additive tap matrix
$\mathbf{R}_i$	$(2p+1)L+1 \times (2p+1)L+1$	auto-correlation matrix
$\mathbf{r}_i$	$(2p+1)L+1 \times L$	cross-correlation matrix

Table 1. List of Symbols.

3. EXPERIMENTS

3.1. Introduction

In this section we present a series of experiments that show how the mapping algorithm can be used in a continuous speech recognizer across acoustic environments. In all of the experiments the recognizer models are trained with data recorded with high-quality microphones and digitally sampled at 16,000 Hz. The analysis frame rate is 100 Hz.

The tables below show three types of performance indicators:

- *Relative distortion measure*. For a given component of a feature vector we define the relative distortion between the clean and noisy data as follows:

$$d = \sqrt{\frac{E\left[(x-y)^2 \right]}{\mathrm{var}(x)}} \tag{8}$$

- *Word recognition error.*
- *Error ratio.* The error ratio is given by E_n / E_c where E_n is the word recognition error for the test-noisy/train-clean con-

dition, and E_c is the word recognition error of the test-clean/train-clean condition.

3.2. Single Microphone

To test the POF algorithm on a single target acoustic environment we used the DARPA *Wall Street Journal* database [15] on SRI's DECIPHER™ phonetically tied-mixture speech recognition system [2]. The signal processing consisted of a filterbank-based front-end that generated six feature streams: cepstrum *(c1-c12)*, cepstral energy *(c0)*, and their first- and second-order derivatives. Cepstral-mean normalization [16] was used to equalize the channel. We used simultaneous recordings of high-quality speech (Sennheiser 414 head-mounted microphone with a noise-cancelling element) along with speech recorded by a standard speaker phone (AT&T 720) and transmitted over local telephone lines. We will refer to this stereo data as *clean* and *noisy* speech respectively. The models of the recognizer were trained using 42 male WSJ0 training talkers (3500 sentences) recorded with a Sennheiser microphone. The models of the mapping algorithm were trained using 240 development training sentences recorded by three speakers. The test set consisted of 100 sentences (not included in the training set) recorded by the same three speakers.

In this experiment we mapped two of the six features: the cepstrum *(c1-c12)* and the cepstral energy *(c0)* separately. The derivatives were computed from the mapped vectors of the cepstral features. For the conditioning feature we used a 13-dimensional cepstral vector *(c0-c12)* modeled with 512 Gaussians with diagonal covariance matrices. The results are shown in Table 2.

Filter Coefficients	Average Distortion	Recognition Error (%)	Error Ratio
No mapping	0.72	27.6	2.46
$\boldsymbol{A}_{i,0}=\boldsymbol{I}, \boldsymbol{b}_i$	0.62	18.1	1.62
$\boldsymbol{A}_{i,0}, \boldsymbol{b}_i$	0.57	17.0	1.52
$\boldsymbol{A}_{i,-1}, ..., \boldsymbol{A}_{i,-1}, \boldsymbol{b}_i$	0.51	17.3	1.54
$\boldsymbol{A}_{i,-2}, ..., \boldsymbol{A}_{i,-2}, \boldsymbol{b}_i$	0.50	16.4	1.46
$\boldsymbol{A}_{i,-3}, ..., \boldsymbol{A}_{i,-3}, \boldsymbol{b}_i$	0.49	15.9	1.42
$\boldsymbol{A}_{i,-4}, ..., \boldsymbol{A}_{i,-4}, \boldsymbol{b}_i$	0.49	16.1	1.44

Table 2. Performance of the POF algorithm for different number of filter coefficients. The number of Gaussian distributions is 512 per feature and the conditioning feature is a 13-dimensional cepstral vector.

The baseline experiment produced a word error rate of 27.6% on the noisy test set, that is, 2.46 times the error obtained when using the clean data channel. A 34% improvement in recognition performance was obtained when using only the additive filter coefficient $\boldsymbol{b}_i$. (Recognition error goes down to 18.1%.) The best result (15.9% recognition error) was obtained for the condition *p=3*, in which six neighboring noisy frames are being used to estimate the feature vector for the current frame. The correlation between the average relative distortion between the six clean and noisy features and the recognition error is 0.9.

3.3. Multiple Microphones

To test the performance of the POF algorithm on multiple microphones we used SRI's stereo-ATIS database. (See the companion paper [1] for details.) In this database, we recorded the clean channel with a Sennheiser microphone and the noisy channel with 10 different telephone handsets. For this set of experiments we also mapped the cepstrum vector *(c1-c12)* and the cepstral energy *(c0)*. The maximum delay of the filters was kept fixed at *p=2*, and the number of Gaussians was 512. We tried the following conditioning features:

- **Cepstrum**. Same conditioning feature used in the single microphone experiment *(c0-c12)*.
- **Spectral SNR**. This is an estimate of the instantaneous signal-to-noise ratio computed on the log-filterbank energy domain. The vector size is 25.
- **Cepstral SNR**. This feature is generated by applying the discrete cosine transform (DCT) to the spectral SNR. The transformation reduces the dimensionality of the vector from 25 to 12 elements.

The results are shown in Table 3. The baseline result is a 19.4% word error rate. This result is achieved when the same wide-band front-end is used for training the models with clean data and for recognition using telephone data. When a telephone front-end [1] is used for training and testing, the error decreases to 9.7%. The disadvantage of using this approach is that the acoustic models of the recognizer have to be re-estimated. However, the POF-based front-end operates on the clean models and results in better performance. The cepstral SNR produces the best result (8.7%). With this conditioning feature we combine the effects of noise and spectral shape in a compact representation.

Experiment	Word Error (%)	Error Ratio
Wide-band front-end	19.4	2.49
Telephone-bandwidth front-end	9.7	1.24
Mapping with cepstrum	9.4	1.20
Mapping with spectral SNR	8.9	1.14
Mapping with cepstral SNR	8.7	1.11

Table 3. Performance for the multiple-telephone handset test set.

3.4. Using POF in Either Training or Testing

The POF mapping can be applied to either the training data or the testing data. When applied to the training data, it makes the clean speech features look like the noisy speech features. During recognition, the standard signal processing of the noisy speech features may be used.

When applied to the testing data, it makes the noisy speech features look like the clean speech features. Training of the HMM acoustic models uses the standard signal processing.

Signal Processing		Word	Error
Training	Testing	Error	Ratio
Standard	Standard	31.4	2.7
Map Clean to Noisy	Standard	21.3	1.8
Standard	Map Noisy to Clean	20.0	1.7

Table 4: Training and Testing Paradigms using the Probabilistic Optimum Filter. Word Error is on AT&T Speaker Phone

The results in Table 4 show that equivalent performance is obtained when using the mapping either in training (21.3%) or in testing (20.0%). In both cases, this is a significant decrease from the performance without compensation (31.4%). The recognition numbers are slightly different from those in Table 2 since this experiment uses an earlier version of the mapping and recognizer.

4. CONCLUSIONS

We have presented a feature mapping algorithm capable of exploiting nonlinear relations between two acoustic spaces. We have shown how to improve the performance of the recognizer in the presence of a noisy signal by using a small database with simultaneous recordings in the clean and noisy acoustic environments.

The mapping algorithm has performed well on a speaker-dependent/single-microphone task and on a speaker-independent/multiple-microphone task. In both cases the target acoustic environment was known a priori. The POF algorithm efficiently exploited the correlations within and between frames, resulting in significant improvements over the unmapped systems.

The POF algorithm can be used only when a stereo database containing the clean and noisy speech is available. This requirement limits the use of the POF algorithm to applications in which the target acoustic environment is well defined and stable. These applications may include those for which the microphone, the channel or the background noise encountered in the field do not match the training conditions.

ACKNOWLEDGMENTS

This research was supported by a grant, NSF IRI-9014829, from the National Science Foundation.

REFERENCES

1. M. Weintraub and L. Neumeyer, “Constructing Telephone Acoustic Models from a High-Quality Speech Corpus,” 1994 IEEE ICASSP.
2. V. Digalakis and H. Murveit, “GENONES: Optimizing the Degree of Mixture Tying in a Large Vocabulary Hidden Markov Model Based Speech Recognizer,” 1994 IEEE ICASSP.
3. H. Murveit, J. Butzberger, V. Digalakis, and M. Weintraub, “Large-Vocabulary Dictation Using SRI’s DECIPHER Speech Recognition System: Progressive Search Techniques,” 1993 IEEE ICASSP, pp. II319-II322.
4. H. Murveit, J. Butzberger, and M. Weintraub, “Performance of SRI’s DECIPHER™ Speech Recognition System on DARPA’s CSR Task,” 1992 DARPA Speech and Natural Language Workshop Proceedings, pp. 410-414.
5. Murveit, H., J. Butzberger, and M. Weintraub, “Reduced Channel Dependence for Speech Recognition,” 1992 DARPA Speech and Natural Language Workshop Proceedings, pp. 280-284.
6. A. Erell and M. Weintraub, “Spectral Estimation for Noise Robust Speech Recognition,” 1989 DARPA SLS Workshop, pp. 319-324.
7. A. Erell and M. Weintraub, “Recognition of Noisy Speech: Using Minimum-Mean Log-Spectral Distance Estimation,” 1990 DARPA SLS Workshop, pp. 341-345.
8. B.H. Juang and L.R. Rabiner, “Signal Restoration by Spectral Mapping,” 1987 IEEE ICASSP, pp. 2368-2371.
9. H. Gish, Y.L. Chow, and J.R. Rohlicek, “Probabilistic Vector Mapping of Noisy Speech Parameters for HMM Word Spotting,” 1990 IEEE ICASSP, pp. 117-120.
10. K. Ng, H. Gish, and J.R. Rohlicek, “Robust Mapping of Noisy Speech Parameters for HMM Word Spotting,” 1992 IEEE ICASSP, pp. II-109-II-112.
11. A. Acero, “Acoustical and Environmental Robustness in Automatic Speech Recognition,” Ph.D. Thesis, Carnegie-Mellon University, September 1990.
12. R.M. Stern, F.H. Leu, Y. Ohshima, T.M. Sullivan, and A. Acero, “Multiple Approaches to Robust Speech Recognition,” 1992 International Conference on Spoken Language Processing, pp. 695-698.
13. A. Nadas, D. Nahamoo, and M. Picheny, “Adaptive Labeling: Normalization of Speech by Adaptive Transformations based on Vector Quantization,” 1988 IEEE ICASSP, pp. 521-524.
14. Y. Linde, A. Buzo, and R.M. Gray, “An Algorithm for Vector Quantizer Design,” IEEE Trans. Comm., vol. 28, pp. 84-95, January 1980.
15. G. Doddington, “CSR Corpus Development,” 1992 DARPA SLS Workshop, pp. 363-366.
16. S.F. Furui, “Cepstral Analysis Technique for Automatic Speaker Verification,” IEEE Trans. ASSP, Vol. 29, pp. 254-272, April 1981.