SPEECH ENHANCEMENT BASED UPON
HIDDEN MARKOV MODELING

Yariv Ephraim, David Malah1, and Bling-Hwang Juang

AT&T Bell Laboratories
Murray Hill, NJ 07974

ABSTRACT

A maximum a-posteriori approach for enhancing speech signals which have been degraded by statistically independent additive noise is proposed. The approach is based upon statistical modeling of the clean speech signal and the noise process using long training sequences from the two processes. Hidden Markov models (HMM's) with mixtures of Gaussian autoregressive (AR) output probability distributions are used to model the clean speech signal. A low order Gaussian AR model is used for the wide-band Gaussian noise considered here. The parameter set of the HMM is estimated using the Baum or the EM (estimation-maximization) algorithm. The enhancement of the noisy speech is done by means of reestimation of the clean speech waveform using the EM algorithm. An approximate improvement of 4.0-6.0 dB in signal to noise ratio (SNR) is achieved at 10 dB input SNR.

I. Introduction

In [1], a model based approach for enhancing speech signals degraded by statistically independent additive noise was proposed. In this approach the unknown probability distributions (PDF's) of the speech signal and the noise process are first estimated from long training sequences from the two processes, and then estimation of the clean speech signal is applied using the estimated statistics. Hidden Markov models (HMM's) were used for the clean speech signal, and a low order Gaussian autoregressive (AR) model was used for the noise process. The noise process considered in [1] was a wide-band Gaussian noise. The parameter set of the HMM for the speech signal was estimated using the segmental k-means algorithm, which is an approximate maximum likelihood (ML) modeling approach. The estimation of the clean speech signal was done using an approximate maximum a-posteriori (MAP) estimation approach, in which the joint probability density function (PDF) of the state sequence, the clean signal, and the noisy speech is locally maximized over all state sequences and clean signals.

In this paper we examine ML hidden Markov modeling of the clean speech and exact MAP estimation of the clean speech given the noisy speech. The modeling is done using the well known Baum algorithm, and enhancement is performed using the EM (estimation-maximization) algorithm [2], [3]. The MAP algorithm locally maximizes the conditional PDF of the clean speech given the noisy speech. The algorithm starts from the given noisy speech and generates a sequence of speech sample functions with non-decreasing likelihood values by maximizing in each iteration an appropriately defined auxiliary function. The enhancement algorithm developed here is examined and compared with the approximate MAP approach of [1].

II. Problem Formulation

A. HMM's for Clean Speech

Let \( p_{\lambda} \) be the PDF of an HMM for the clean speech signal, where \( \lambda \) denotes the parameter set of the model. We consider HMM's with \( M \) states and mixtures of \( L \) Gaussian AR output processes at each state. Let \( y_{T} \) be a sequence of \( K \)-dimensional vectors which represent the output from the model. Let \( x_{T} \) be a sequence of states corresponding to \( y \). Let \( h_{T} \) be a sequence of mixture components corresponding to \( (x,y) \). The PDF \( p_{\lambda} \) is given by

\[
p_{\lambda}(y) = \sum_{x} \sum_{h} p_{\lambda}(x,h,y)
\]

(1)

where \( p_{\lambda}(x) \) is the PDF of the sequence of states \( x \), \( p_{\lambda}(h \mid x) \) is the PDF of the sequence of mixture components \( h \) given the sequence of states \( x \), and \( p_{\lambda}(y \mid h,x) \) is the PDF of the output sequence \( y \) given \( x \). The probability \( p_{\lambda}(x) \) is given by

\[
p_{\lambda}(x) = \prod_{t=1}^{T} a_{x_{t} \mid x_{t-1}}
\]

(2)

where \( a_{x_{t} \mid x_{t-1}} \) denotes the transition probability from state \( x_{t-1} \) at time \( t-1 \) to state \( x_{t} \) at time \( t \), and \( \rho_{x_{t} \mid x_{t-1}} \) denotes the probability of the initial state \( x_{0} \). For \( p_{\lambda}(h \mid x) \), and \( p_{\lambda}(y \mid h,x) \), we make the following standard assumptions:

\[
p_{\lambda}(h \mid x) = \prod_{t=1}^{T} \frac{T}{\prod_{t=0}^{T-1} c_{h_{t} \mid x_{t}}}
\]

(3)

and

\[
p_{\lambda}(y \mid h,x) = \prod_{t=1}^{T} \frac{T}{\prod_{t=0}^{T-1} b_{y_{t} \mid h_{t},x_{t}}}
\]

(4)

where \( c_{h_{t} \mid x_{t}} \) is the probability of choosing the mixture component \( h \) given that the process is in state \( x_{t} \), and \( b_{y_{t} \mid h_{t},x_{t}} \) is the PDF of the output vector \( y_{t} \) given \( h_{t},x_{t} \). For zero mean \( N_{1} \)-th order Gaussian AR output processes, we have

\[
b_{y_{t} \mid h_{t},x_{t}} = \frac{\exp(-{\frac{1}{2} \hat{\sigma}_{y_{t}}^{2} y_{t}^{2}})}{(2\pi)^{T/2} \hat{\sigma}_{y_{t}}^{T/2}}
\]

(5)

where \# denotes vector transpose, \( \hat{\sigma}_{y_{t}} = \sigma_{y_{t}}^{2} + \alpha_{1}^{2} \), \( \sigma_{y_{t}}^{2} \) is the variance of the innovation process of the AR source, and \( \hat{\sigma}_{y_{t}}^{2} \) is a \( K \times K \) lower triangular Toeplitz matrix in which the first \( N_{1}+1 \) elements of the first column constitute the coefficients of the AR process, \( \hat{\sigma}_{y_{t}}^{2} = \sum_{i=0}^{N_{1}} \hat{\sigma}_{y_{t}}^{2} \). 

The modeling problem is that of estimating the parameter set \( \lambda_{t} = (a_{\lambda}, \alpha, \beta, \gamma) \), where \( a_{\lambda} = (a_{0}, a_{1}, \ldots, a_{L}) \), \( \alpha = (\alpha_{1}, \ldots, \alpha_{L}) \), \( \gamma = (\gamma_{0}, \ldots, \gamma_{L}) \), and \( \beta = (\beta_{0}, \ldots, \beta_{L}) \), for \( \alpha, \beta, \gamma \), and \( \lambda_{t} \). Given a training sequence \( y \) from the speech signal. An ML estimate of the parameter set \( \lambda_{t} \) is obtained from

\[
\max_{\lambda} \ln p_{\lambda}(y) = \max_{\lambda} \sum_{x} \sum_{h} p_{\lambda}(x,h,y)
\]

(6)

and this maximization is locally performed by the Baum reestimation algorithm. The segmental \( k \)-means algorithm for estimating the parameter set of the model used in [1], assumes that the double sum in (6) is dominated by a unique sequence of states and mixture components. Hence, the parameter set of the model is estimated along with the most likely sequence of states and mixture components by

\[
\max_{\lambda} \ln p_{\lambda}(x,h,y)
\]

(7)

The ML estimation procedure is described in Section III-A.

B. AR Model for the Noise Process

Let \( p_{\lambda} \) be the PDF of the model for the noise process, where \( \lambda_{n} \) is the parameter set of the model. For the Gaussian noise with a
components by enhancement procedure developed in [l] assumes that the double dent. Note that since multiple training sequences which are assumed to be statistically independent have been used. Hence, we provide the algorithm for the more general case of modeling using N training sequences of speech. Let \( \gamma_T \) be a sequence of \( T+1 \) \( K \)-dimensional vectors, and let \( \gamma_T \) be the set of \( N \) such sequences. Let \( \gamma_T \) and \( h_T \) be, respectively, the sequences of states and mixture components corresponding to the \( n \)-th utterance of the training sequence. Finally, let \( X \) be a sequence of \( \gamma_T \), \( n=1, \ldots, N \) and \( h_T \) be \( \{ h_T \}, n=1, \ldots, N \).

### A. Baum reestimation algorithm

The likelihood function to be maximized is given by

\[
\log p_{\lambda}(y) = \sum_{n=1}^{N} \log p_{\lambda}(y_n).
\]

Local maximization of (16) can be achieved by the Baum reestimation algorithm. This algorithm generates a sequence of HMM's with non-decreasing likelihood values (16). Each iteration of the Baum algorithm starts with an old set of parameters, say \( \lambda_t \), and estimates a new set of parameters, say \( \lambda_t \) by maximizing the following auxiliary function,

\[
\Phi(\lambda_t) = N \sum_{n=1}^{N} \sum_{y_n} p_{\lambda_t}(x_n, h_n, y_n) \log p_{\lambda_t}(x_n, h_n, y_n)
\]

over \( \lambda_t \) subject to the constraints \( p_{\beta, n} \geq 0 \), \( \sum_{n=1}^{N} p_{\beta, n} = 1 \), \( a_{\beta} \geq 0 \), \( \sum_{\beta=1}^{a_{\beta}} \sum_{n=1}^{N} p_{\lambda,n} = 1 \), and AR covariance matrices

\[
S_{\gamma_T}(\alpha, \beta) = \sum_{n=1}^{N} \sum_{r=x_0}^{x_T} \sum_{r=x_0}^{x_T} q_{\lambda,n}(x_n, h_n, y_n)
\]

and the parameters of the AR output PD's are obtained from

\[
\min_{\beta} \{ \log R_{\gamma_T}(\beta) \}
\]

where

\[
q_{\lambda,n}(x_n, h_n, y_n) = \sum_{x_{n-1}}^{x_n} \sum_{h_{n-1}}^{h_n} p_{\lambda,n}(x_n, h_n, y_n)
\]

is the conditional probability, under \( p_{\lambda,n} \), of being in state \( \alpha \) at time \( t-1 \), in state \( \beta \) at time \( t \), and choosing mixture component \( y \) while in state \( \beta \), given the \( n \)-th utterance of the speech training sequence, and

\[
q_{\lambda,n}(x_n, h_n, y_n) = \sum_{x_{n-1}}^{x_n} \sum_{h_{n-1}}^{h_n} p_{\lambda,n}(x_n, h_n, y_n)
\]

is the conditional probability, under \( p_{\lambda,n} \), of being in state \( \beta \) at time \( t-1 \), in state \( \beta \) at time \( t \), and choosing mixture component \( y \) while in state \( \beta \), given the \( n \)-th utterance of the speech training sequence.
r and choosing mixture component γ while in state β, given the nth utterance of the speech training sequence. The reestimation formulas (19)-(21) are valid provided that the terms in the denominators of these expressions are greater than zero. If any of these conditions is not satisfied, then the affected reestimated parameter can be arbitrarily chosen up to the constraints associated with the problem (17), without affecting the likelihood value. For example, if the denominator of (19) equals zero for a particular α, then any α[b], b=1,..., M, which satisfy \( \sum_{b=1}^{M} \alpha[b] = 1 \) can be chosen.

The probability measures \( q_{n}(\alpha, \beta, \gamma) \) and \( q_{n}(\beta, \gamma) \) can be efficiently calculated using the forward-backward formulas as follows:

\[
q_{n}(\alpha, \beta, \gamma) = \frac{F_{n}(\alpha, \gamma) B_{n}(\beta, \gamma)}{\sum_{\alpha, \beta = 1}^{M} F_{n}(\alpha, \gamma) B_{n}(\beta, \gamma)} \quad 0 \leq t \leq T_{n+1},
\]

\[
q_{n}(\beta, \gamma) = \frac{F_{n}(\beta, \gamma) B_{n}(\beta, \gamma)}{\sum_{\beta = 1}^{M} F_{n}(\beta, \gamma) B_{n}(\beta, \gamma)} \quad 0 \leq t \leq T_{n},
\]

where

\[
F_{n}(\alpha, \gamma) = \pi_{\alpha} c_{\alpha} \delta(y(0), \gamma, \alpha),
\]

\[
B_{n}(\beta, \gamma) = \sum_{\alpha = 1}^{M} \sum_{n=1}^{N_{s}} B_{n}(\alpha, \gamma) \delta(y(n), \gamma, \alpha),
\]

The minimization problem in (21) has a unique solution provided that \( R_{T+1} \) is positive definite. The minimizing AR parameter set can be found by AR modeling of \( R_{T+1} \) using a variant of the covariance method of linear prediction. An approximate solution can be obtained by the autocorrelation method of linear prediction if end-block effects are neglected. More specifically, if the K-th order vector \( A_{\gamma}(k) \) is considered as the convolution result of \( y_{n}(k) \) with itself. The two vectors are identical in their first K elements, but the vector which results from convolution of \( y_{n}(k) \) with \( y_{n}(k) \) has \( N_{s} - 1 \) additional elements. This approach was found reasonable for the values of \( K = 128 \) and \( N_{s} = 10 \) used here. In this case, it can be shown by substituting \( R_{T+1} \) and \( \delta(\gamma, \beta) \) into (21) that the AR parameter set is obtained from AR modeling of the autocorrelation function given by

\[
r_{T+1}(\gamma) = \frac{1}{T+1} \sum_{t=0}^{T} y_{t}^{\gamma}(k),
\]

where

\[
r_{n}(\gamma) = \frac{1}{K} \sum_{k=0}^{K-1} y_{n}(k) y_{n}(k+1) \delta(y(n), \gamma, \alpha),
\]

The likelihood function (16), which has to be evaluated in checking convergence of the Baum algorithm, can be efficiently calculated similarly to the calculation of the denominator of (23) using the forward-backward formulas as is shown in (25).

B. Noise model estimation

The estimation problem of the parameter set of the AR model for the noise process results from (8) into (9). This problem is equivalent to

\[
\min_{\gamma} \left( \text{tr } R_{T} V^{-1} \text{ - } \ln \det V^{-1} \right),
\]

where

\[
R_{T} = \frac{1}{T} \sum_{t=0}^{T} y_{t}^{\gamma}(k),
\]

This problem is similar to that associated with the estimation of the parameter set of each AR output process of the HMM. Approximate solution is obtained from AR modeling of the autocorrelation function given by

\[
r_{n}(\gamma) = \frac{1}{K} \sum_{k=0}^{K-1} y_{t}^{\gamma}(k+1) + m \}
\]

IV. EM Speech Enhancement Algorithm

In this section we apply the EM algorithm for MAP estimation of the clean speech signal given the noisy speech. Let \( z \) be a given sequence of \( T+1 \) \( K \)-dimensional vectors of noisy speech. Let \( \lambda_{3}(\hat{\lambda}, \hat{\mu}) \) have that the estimation of the clean speech signal. Similarly, let \( y(k+1) \) be a current estimate of the speech signal. Using Jensen's inequality and the fact that given \( y, (x, h) \) are statistically independent, we have that

\[
\ln p_{x}(y(k+1) \mid z) \leq \ln p_{x}(y(k)) + \ln p_{x}(y(k+1) \mid z) \leq \ln p_{x}(y(k))
\]

where

\[
\Phi(y(k+1)) = \Phi(y(k)),
\]

where

\[
\Phi(y(k+1)) = \sum_{x \in A} p_{x}(x, y(k)) \ln p_{x}(x, y(k+1) \mid z).
\]

Hence, maximization of \( \Phi(y(k+1)) \) over \( y(k+1) \) results in

\[
\ln p_{x}(y(k+1) \mid z) \geq \ln p_{x}(y(k)) \quad \text{for } y(k+1) \rightarrow \gamma(\gamma(k)) \text{ almost everywhere}
\]

This standard argument of the EM algorithm implies that a MAP estimate of the speech waveform can be achieved by reestimation of the speech waveform using the maximization of the auxiliary function \( \Phi(y(k+1)) \).

On substituting (1)-(5), (8), and (12) into (32), and setting the gradient of \( \Phi(y(k+1)) \) with respect to \( y_{n}(k+1) \) to zero, we obtain the following reestimation formula for the clean speech signal.

\[
y_{n}(k+1) = \left[ \sum_{\beta, \gamma} q_{n}(\beta, \gamma) y_{n}(k) H_{n+1}^{\beta \gamma} \right]^{-1} \sum_{\beta, \gamma} q_{n}(\beta, \gamma) y_{n}(k) H_{n+1}^{\beta \gamma},
\]

where \( q_{n}(\beta, \gamma) \) is defined similarly to (32) with \( \gamma_{T+1} \) being replaced by \( \gamma(k) \), and \( H_{T+1}^{\beta \gamma} \) is a Wiener filter for the output Gaussian process from state \( \beta \) and mixture \( \gamma \) and the Gaussian noise process (8).
speakers and recorded in a manner similar to that of the training set. The speech material and the speakers used for training were different from those used for testing. The model was trained to estimate directly from the noisy speech, using an initial interval whose length was about 10% of the length of the utterance to be enhanced, and in which speech was not present.

In all of our experiments, the dimension of the speech vectors was $K=128$ at a sampling rate of 8kHz. Training was done using non-overlapped frames, while enhancement was performed using overlapped frames. The speech was processed either as a short time Fourier transform based model or using the model designed by the segmental k-means algorithm. A Hanning analysis window was applied to the speech frames during training and enhancement. The synthesis of the enhanced signal from the individually processed frames was done using the standard short time Fourier transform overlap and add technique. The order of each AR output process of the HMM was set to $N_p=10$, which is a compromise between linear and nonlinear analysis of speech signals. The order of the AR model for the noise process was set to $N_w=4$, since the noise examined here has a theoretically flat power spectral density. The iterative algorithms for designing the models, and for performing the enhancement, were terminated whenever the difference in likelihood values at two consecutive iterations, normalized by the older likelihood value, was less than or equal to $10^{-6}$.

The number of states $M$, and mixture components for each state $L$, were experimentally determined by examining the enhancement results obtained using different values of $(M,L)$ at input SNR of 10 dB. Table I shows the minimum and maximum SNR values achievable in this experiment. The case VQ-AMAP represents enhancement results obtained using HMM's designed by the standard AR model vector quantization approach, and the approximate MAP enhancement approach described in [1]. The VQ model was used for initializing the segmental $k$-means algorithm, and the segmental $k$-means model was used to initialize the Baum algorithm. The case SEG-AMAP represents enhancement results obtained using segmental $k$-means training and approximate MAP enhancement. The case ML-MAP represents enhancement results obtained using ML Baum training and MAP enhancement (33) approaches. Finally, the cases VQ-CLN and SEG-CLN represent some theoretical performance bounds within the proposed framework for speech enhancement. Here, the clean speech was used for estimating the most likely sequence of states and mixture components, and the noisy speech was filtered by a time varying Wiener filter. At each time instant, the filter was constructed from the spectrum of the AR process associated with the estimated state and mixture component, and the spectrum of the AR model of the noise process. In the VQ-CLN case, AR model vector quantization was applied to the clean speech on a frame-by-frame basis using the ML VQ designed for initializing the segmental $k$-means algorithm. In the SEG-CLN case, Viterbi decoding was applied to the clean speech using the model designed by the segmental $k$-means algorithm. The major difference between the two cases is that the VQ-CLN case is memoryless while the SEG-CLN version incorporates the Markovian memory.

Table I shows that the three proposed speech enhancement schemes, VQ-AMAP, SEG-AMAP, and ML-MAP, provide very similar SNR improvement for all of the examined values of $(M,L)$. Furthermore, this SNR improvement is about 0.5dB lower than that obtained in the VQ-CLN and SEG-CLN cases which use the clean speech for performing the decoding. The SNR improvement obtained in the latter two cases is essentially identical. Careful informal listening tests indicate that for a given $(M,L)$, the three enhancement schemes, VQ-AMAP, SEG-AMAP, and ML-MAP, provide very similar enhanced speech quality. In some cases, however, the ML-MAP approach provided slightly better results than the other two procedures. The best enhancement results were obtained using the five-state five-mixture model. For this case, the enhanced speech has almost no residual noise, it is reasonably intelligible, and it contains fewer gross estimation errors than the enhanced speech obtained using $M=8, L=4$ or $M=16, L=8$. Those gross estimation errors are due to decoding errors which result in an incorrect filter selection. The enhanced speech corresponding to VQ-AMAP and SEG-AMAP sounds identical, a fact which implies the unimportance of the Markovian memory in decoding the speech signal, given the clean speech, in this application. The differences between the best enhanced speech signals and the speech signals obtained in the VQ-CLN or SEG-CLN cases, are generally small. In both cases, the input noise was effectively removed. The speech obtained using VQ-CLN is somewhat crisper and somewhat noisier than the enhanced signals obtained using either of VQ-AMAP, SEG-AMAP, or ML-MAP.

Table II focuses on the ML-MAP approach with the five-state five-mixture model, and shows minimum and maximum values of SNR of the enhanced speech obtained at values of different input SNR. The minimum and maximum number of iterations used in each case here using long training sequences from the two processes. Given the estimated statistics of the speech and the noise processes, a MAP estimation approach was developed and implemented using the EM algorithm. This approach proved especially useful for enhancing noisy speech with SNR greater than or equal to 10 dB.

We opt for HMM's due to their general acceptability as reliable models for speech signals in the speech recognition community. The most natural way to use these models in speech enhancement applications, is for simultaneous enhancement of the entire utterance of noisy speech. This, however, is not the only way the proposed approach can be implemented, and a frame-by-frame enhancement implementation is possible, for example, by considering $P(x_{t} | P_{t})$, or $P(x_{t} | P_{t-1})$. This frame-by-frame version of the MAP enhancement algorithm can also be efficiently implemented using slightly different forward-backward formulas than those used here. The estimate obtained in simultaneous enhancement of all the frames in the noisy input utterance is usually more accurate than that obtained on a frame-by-frame basis, since the number of noisy speech samples upon which the estimation is based is larger. Since this is the first paper on the subject, our goal was to establish a benchmark on the performance of the proposed approach. Hence, we focused on the simultaneous enhancement of all the frames in each given noisy input utterance.

Table I: Enhancement results for different number of states and mixture components at 10 dB input SNR.

<table>
<thead>
<tr>
<th>$M/L$</th>
<th>VQ-AMAP</th>
<th>SEG-AMAP</th>
<th>ML-MAP</th>
<th>VQ-CLN</th>
<th>SEG-CLN</th>
</tr>
</thead>
</table>

Informal listening to the enhanced speech signals indicates that at 5 dB input SNR the enhancement was effective only for some of the sentences, while for the other sentences it introduced some noticeable distortions. At the higher input SNR values of 15 and 20 dB, very good enhanced speech quality was obtained. The noise was completely removed and the speech was minimally distorted. The crispness and naturalness of the original speech were well preserved.

VI. Comments

We proposed a new approach for enhancing speech signals which have been degraded by statistically independent additive noise. The approach capitalizes on statistical modeling of the clean speech and the noise processes using long training sequences from the two processes. Given the estimated statistics of the speech and the noise processes, a MAP estimation approach was developed and implemented using the EM algorithm. This approach proved especially useful for enhancing noisy speech with SNR greater than or equal to 10 dB.

We opt for HMM's due to their general acceptability as reliable models for speech signals in the speech recognition community. The most natural way to use these models in speech enhancement applications, is for simultaneous enhancement of the entire utterance of noisy speech. This, however, is not the only way the proposed approach can be implemented, and a frame-by-frame enhancement implementation is possible, for example, by considering $P(x_{t} | P_{t})$, or $P(x_{t} | P_{t-1})$. This frame-by-frame version of the MAP enhancement algorithm can also be efficiently implemented using slightly different forward-backward formulas than those used here. The estimate obtained in simultaneous enhancement of all the frames in the noisy input utterance is usually more accurate than that obtained on a frame-by-frame basis, since the number of noisy speech samples upon which the estimation is based is larger. Since this is the first paper on the subject, our goal was to establish a benchmark on the performance of the proposed approach. Hence, we focused on the simultaneous enhancement of all the frames in each given noisy input utterance.

References

