Published as: *IEEE Trans Audio Speech Lang Processing*. 2010 August 11; 18(6): 1127–1136.

Speech Enhancement Using Gaussian Scale Mixture Models

Jiucang Hao,
Computational Neurobiology Laboratory, Salk Institute, La Jolla, CA 92037 USA, and also with the Institute for Neural Computation, University of California, San Diego, CA 92093 USA

Te-Won Lee[Senior Member, IEEE], and
Qualcomm, Inc., San Diego, CA 92121 USA

Terrence J. Sejnowski[Fellow, IEEE]
Howard Hughes Medical Institute and Computational Neurobiology Laboratory, Salk Institute, La Jolla, CA 92037 USA, and also with the Division of Biological Sciences, University of California, San Diego, CA 92093 USA

Abstract

This paper presents a novel probabilistic approach to speech enhancement. Instead of a deterministic logarithmic relationship, we assume a probabilistic relationship between the frequency coefficients and the log-spectra. The speech model in the log-spectral domain is a Gaussian mixture model (GMM). The frequency coefficients obey a zero-mean Gaussian whose covariance equals to the exponential of the log-spectra. This results in a Gaussian scale mixture model (GSMM) for the speech signal in the frequency domain, since the log-spectra can be regarded as scaling factors. The probabilistic relation between frequency coefficients and log-spectra allows these to be treated as two random variables, both to be estimated from the noisy signals. Expectation-maximization (EM) was used to train the GSMM and Bayesian inference was used to compute the posterior signal distribution. Because exact inference of this full probabilistic model is computationally intractable, we developed two approaches to enhance the efficiency: the Laplace method and a variational approximation. The proposed methods were applied to enhance speech corrupted by Gaussian noise and speech-shaped noise (SSN). For both approximations, signals reconstructed from the estimated frequency coefficients provided higher signal-to-noise ratio (SNR) and those reconstructed from the estimated log-spectra produced lower word recognition error rate because the log-spectra fit the inputs to the recognizer better. Our algorithms effectively reduced the SSN, which algorithms based on spectral analysis were not able to suppress.

Index Terms

Gaussian scale mixture model (GSMM); Laplace method; speech enhancement; variational approximation

I. Introduction

Speech enhancement improves the quality of signals corrupted by the adverse noise, channel distortion such as competing speakers, background noise, car noise, room reverberations, and low-quality microphones. A broad range of applications includes mobile communications, robust speech recognition, low-quality audio devices, and aids for the hearing impaired.

Although speech enhancement has attracted intensive research [1] and algorithms motivated from different aspects have been developed, it is still an open problem [2] because there are

no precise models for both speech and noise [1]. Algorithms based on multiple microphones [2]–[4] and single microphone have also been successful in achieving some measure of speech enhancement [5]–[13].

In spectral subtraction [5], the noise spectrum is subtracted to estimate the spectral magnitude which is believed to be more important than phase for speech quality. Signal subspace methods [6] attempt to find a projection that maps the signal and noise onto disjoint subspaces. The ideal projection splits the signal and noise, and the enhanced signal is constructed from the components that lie in the signal subspace. This approach has been applied to single microphone source separation [14]. Other speech enhancement algorithms have been based on audio coding [15], independent component analysis (ICA) [16] and perceptual models [17].

Statistical-model-based speech enhancement systems [7] have proven to be successful. Both the speech and noise are assumed to obey random processes and treated as random variables. The random processes are specified by the probability density function (pdf) and the dependency among the random variables is described by the conditional probabilities. Because the exact models for speech and noise are unknown [1], speech enhancement algorithms based on various models have been developed. The short-time spectral amplitude (STSA) estimator [8] and the log-spectral amplitude estimator (LSAE) [9] use a Gaussian pdf for both speech and noise in the frequency domain, but differ in signal estimation. The STSA minimizes the minimum mean square error (MMSE) of the spectral amplitude, while the LSAE minimizes the MMSE of the log-spectrum, which is believed to be more suitable for speech processing. Hidden Markov models (HMMs) that include the temporal structure has been developed for clean speech. An HMM with gain adaptation has been applied to the speech enhancement [18] and to the recognition of clean and noisy speech [19]. Super-Gaussian priors, including Gaussian, Laplacian, and Gamma densities, have been used to model the real part and imaginary part of the frequency components [10], and the MMSE estimator used for signal estimation. The log-spectra of speech has often been explicitly and accurately modeled by the Gaussian mixture model (GMM) [11]–[13]. The GMM clusters similar log-spectra together and represents them by a mixture component. The family of GMM has the ability to model any distribution given a sufficient number of mixtures [20], although a small number of mixtures is often enough. However, because signal estimation is intractable, MIXMAX [11] and Taylor expansion [12], [13] are used. Speech enhancement using the log-spectral domain models offers better spectral estimation and is more suitable for speech recognition.

Previous models have estimated either the frequency coefficients or the log-spectra, but not both. The estimated frequency coefficients usually produced better signal quality measured by the signal-to-noise ratio (SNR), but the estimated log-spectra usually provided lower recognition error rate, because higher SNR may not necessarily give a lower error rate. In this paper, we propose a novel approach to estimating both features at the same time. The idea is to specify the relation between the log-spectra and frequency coefficients stochastically. We modeled the log-spectra using a GMM following [11]–[13], where each mixture captures the spectra of similar phonemes. The frequency coefficients obey a Gaussian density whose covariances are the exponentials of the log-spectra. This results in a Gaussian scale mixture model (GSMM) [21], which has been applied to the time-frequency surface estimation [22], separation of of the sparse sources [23], and musical audio coding [24]. In a probabilistic setting, both features can be estimated. An approximate EM algorithm was developed to train the model and two approaches, the Laplace method [25] and the variational approximation [26], were used for signal estimation. The enhanced signals can be constructed from either the estimated frequency coefficients or the estimated log-spectra, depending on the applications.

This paper is organized as the follows. Section II introduces the GSMM for the speech and the Gaussian for the noise. In Section III, an EM algorithm for parameter estimation is derived. Section IV presents the Laplace method and a variational approximation for the signal estimation. Section V shows the experimental results and the comparisons to other algorithms applied to enhance the speeches corrupted by speech shaped noise (SSN) and Gaussian noise. Section VI concludes the paper.

Notation

We use $x[t]$, $y[t]$, and $n[t]$ to denote the time domain signal for clean speech, noisy speech, and noise, respectively. The upper cases X_{kt}, Y_{kt}, and N_{kt} denote the frequency coefficients for frequency bin k at frame t. The ξ_{kt} is the log-spectrum. The $\mathcal{N}(\xi_k|\mu_{ks}, v_{ks})$ is a Gaussian density for ξ_{kt}. with mean μ_{ks} and precision v_{ks} which is defined as the inverse of the variance $1/v_{ks} = E\{|\xi_k - \mu_{ks}|^2|s\}$, where s is the mixture.

II. Gaussian Scale Mixture Model

A. Acoustic Model

Assuming additive noise, the time domain acoustic model is $y[t] = x[t] + n[i]$. After fast Fourier transform (FFT) it becomes

$$Y_k=X_k+N_k \tag{1}$$

where k denotes the frequency bin.

The noise is modeled by a Gaussian

$$p(Y_k|X_k)=\mathcal{N}(Y_k|X_k,\gamma_k)=\frac{\gamma_k}{\pi}e^{-\gamma_k|Y_k-X_k|^2} \tag{2}$$

with zero mean and precision $1/\gamma_k = E\{|Y_k - X_k|^2\}$. Note this Gaussian is of a complex variable, because the FFT coefficients are complex.

B. Improperness of the Log-Normal Distribution for X_k

If the log-spectra $x_k = \log(|X_k|^2)$ are modeled by a GMM, for each mixture s,

$$p(x_k|s)=\sqrt{\frac{\nu_{ks}}{2\pi}}e^{-(\nu_{ks}/2)(x_k-\mu_{ks})^2} \tag{3}$$

is a Gaussian with mean μ_{ks} and precision v_{ks}. Express $X_k=X_k^{'}+iX_k^{''}$ by its real and imaginary parts. Then $X_k^{'}=e^{x_k/2}\cos\theta_k$ and $X_k^{''}=e^{x_k/2}\sin\theta_k$, where θ_k is the phase. If the phase is uniformly distributed, $p(\theta_k) = (1/2\pi)$, the pdf for X_k is $p(X_k|s)=p(X_k^{'},X_k^{''}|s)=(1/J_k)p(x_k|s)p(\theta_k)$, where J_k is the Jacobian $J_k=(\partial(X_k^{'},X_k^{''})/\partial(x_k,\theta_k))=|X_k|^2/2$. We have

$$p(X_k|s)=\frac{1}{\pi|X_k|^2}\sqrt{\frac{\nu_{ks}}{2\pi}}e^{-(\nu_{ks}/2)(\log(|X_k|^2)-\mu_{ks})^2} \tag{4}$$

as plotted in Fig. 1. This is a log-normal pdf because log ($|X_k|^2$) is normally distributed. Note that it has a saddle shape around zero. In contrast, for real speech, the pdf of the FFT coefficients is super-Gaussian and has a peak at zero.

C. Gaussian Scale Mixture Model for Speech Prior

Instead of assuming x_k = log ($|X_k|^2$), we model this relation stochastically. To avoid confusion, we denote the random variable for the log-spectra as ξ_k. The conditional probability is

$$p(X_k|\xi_k)=\frac{e^{-\xi_k}}{\pi}e^{-e^{-\xi_k}|X_k|^2}. \tag{5}$$

This is a Gaussian pdf with mean zero and precision $e^{-\xi_k}$. Note that ξ_k controls the scaling of X_k. Consider *log* $p(X_k|\xi_k) = -\xi_k - e^{-\xi_k}|X|^2 - \log \pi$, and its maximum is given by

$$\widehat{\xi_k}=\arg\max_{\xi_k} p(X_k|\xi_k)=\log|X_k|^2. \tag{6}$$

Thus, we term ξ_k the log-spectrum.

The phonemes of speech have particular spectra across frequency. To group phonemes of similar spectra together and represent them efficiently, we model the log-spectra by a GMM

$$p(\xi_k|s)=\mathcal{N}(\xi_k|\mu_{ks},\nu_{ks})=\sqrt{\frac{\nu_{ks}}{2\pi}}e^{-(\nu_{ks}/2)(\xi_k-\mu_{ks})^2} \tag{7}$$

$$p(\xi_1,\ldots,\xi_K)=\sum_s p(s)\prod_k p(\xi_k|s) \tag{8}$$

where s is the mixture index. Each mixture presents a template of log-spectra, with a corresponding variability allowed for each template via the Gaussian mixture component variances. The mixture may correspond to particular phonemes with similar spectra. Though the precision for ξ is diagonal, $p(\xi_1,\ldots,\xi_K)$ does not factorize over k, i.e., the frequency bins are dependent. The pdf for X_k is

$$p(X_1,\ldots,X_K)=\sum_s p(s)\prod_k \int d\xi_k p(X_k|\xi_k)p(\xi_k|s) \tag{9}$$

which is the GSMM because ξ_k controls the scaling of X_k and obeys a GMM [21]. Note that $\{X_1,-, X_K\}$ are statistically dependent because of the dependency among $\{\xi_1, -, \xi_K\}$.

The GSMM has a peak at zero and is super-Gaussian [21]. It is more peaky and has heavier tails than Gaussian, as shown in Fig. 1. The GSMM, which is unimodal and super Gaussian, is a proper model for speech and has been used in audio processing [22]–[24].

III. EM Algorithm for Training the GSMM

The parameters of the GSMM, $\theta = \{\mu_{ks}, v_{ks}, p(s)\}$, are estimated from the training samples by maximum likelihood (ML) using EM algorithm [27]. The log-likelihood is

$$\begin{aligned}\mathcal{L}(\theta)&=\sum_t \log p(X_{1t},\ldots,X_{kt})\\&=\sum_t \log\left(\sum_{s_t} p(s_t)\prod_k \int p(X_{kt}|\xi_{kt})p(\xi_{kt}|s_t)d\xi_{kt}\right)\\&\geq \sum_{t,s_t}\int q(s_t)\prod_k q(\xi_{kt}|s_t)\times\log\frac{p(s_t)\prod_k p(X_{kt}|\xi_{kt})p(\xi_{kt}|s_t)}{q(s_t)\prod_k q(\xi_{kt}|s_t)}d\xi_{1t}\ldots d\xi_{Kt}\\&=\mathcal{F}(q,\theta).\end{aligned} \tag{10}$$

The inequality holds for any choice of distribution q due to Jensen's inequality [28]. The EM algorithm iteratively optimizes $\mathcal{F}(q, \theta)$ over q and θ. When q equals the posterior distribution $q(\xi_{1t}, \ldots \xi_{Kt}, s_t) = p(\xi_{1t}, \ldots, \xi_{Kt}, s_t|X_{1t}, \ldots, X_{kt})$, the lower bound is tight, $\mathcal{F}\{q, \theta) = \mathcal{L}(\theta)$. The details of the EM algorithm are given in the Appendix.

IV. Two Signal Estimation Approaches

To recover the signal, we need the posterior pdf of the speech. However, for sophisticated models, the closed-form solutions for the posterior pdf are difficult to obtain. To enhance the tractability, we use the Laplace method [25] and a variational approximation [26].

Each frame is independent and processed sequentially. The frame index t is omitted for simplicity. We rewrite the full model as

$$\prod_k p(Y_k|X_k)p(X_k|\xi_k)p(\xi_k|s)p(s) \tag{11}$$

where $p(Y_k|X_k)$ is given by (2), $p(X_k|\xi_k)$ is given by (5), $p(\xi_k|s)$ is a GMM given in (8) and $p(s)$ is the mixture probability.

A. Laplace Method for Signal Estimation

The Laplace method [25] computes maximum *a posteriori* (MAP) estimator for each s. We estimate X_k and ξ_k by maximizing

$$\begin{aligned}\log p(X_k,\xi_k|Y_k,s)&=\log p(Y_k|X_k)+\log p(X_k|\xi_k)+\log p(\xi_k|s)+c\\&=-\gamma_k|Y_k-X_k|^2-\xi_k-e^{-\xi_k}|X_k|^2-\frac{v_{ks}}{2}(\xi_k-\mu_{ks})^2+c\\&=h_s(X_k,\xi_k).\end{aligned} \tag{12}$$

For fixed ξ_k, the MAP estimator for X_k is

$$X_k = \frac{\gamma_k Y_k}{\gamma_k + e^{-\xi_k}}. \tag{13}$$

For fixed X_k, the optimization over ξ_k can be performed using Newton's method.

$$\xi_{ks} \leftarrow \xi_{ks} - \frac{\frac{\partial h_s(X_k,\xi_k)}{\partial \xi_k}|_{\xi_k=\xi_{ks}}}{\frac{\partial^2 h_s(X_k,\xi_k)}{\partial \xi_k^2}|_{\xi_k=\xi_{ks}}} \tag{14}$$

where $(\partial h_s(X_k, \xi_k)/\partial \xi_k) = -1 + e^{-\xi_k}|X_k|^2 - \nu_{ks}\{\xi_k - \mu_{ks}\}$ and $(\partial^2 h_s(X_k, \xi_k)/\partial \xi_k^2) = -e^{-\xi_k}|X_k|^2 - \nu_{ks}$. This update rule is initialized by both $\xi_{ks} = \mu_{ks}$, the means of GSMM and $\xi_{ks} = \log|Y_k|^2$, the noisy log-spectra. After iterating to convergence, the ξ_{ks} that gives higher value of $h_s(X_k, \xi_k)$ is selected. Note that because $h_s(X_k, \xi_k)$ is a concave function in ξ_k, $(\partial^2 h_s(X_k, \xi_k)/\partial \xi_k^2) < 0$, Newton's method works efficiently.

Denote the convergent value for ξ_{ks} from (14) as $\bar{\xi}_{ks}$ and compute $\bar{X}_{ks}$ using (13). We obtain the MAP estimators

$$(\bar{X}_{ks}, \bar{\xi}_{ks}) = \arg\max_{X_k,\xi_k} \log p(X_k, \xi_k | Y_k, s). \tag{15}$$

Because the true s is unknown, the estimators are averaged over all mixtures. The posterior mixture probability is

$$p(s|Y_1, \ldots, Y_K) \propto p(s) \prod_k \int p(Y_k|X_k) p(X_k|s) dX_k \tag{16}$$

where $p(X_k|s) = \int p(X_k|\xi_k) p(\xi_k|s) d\xi_k$. This integral is intractable. The $p(X_k|s)$ has zero mean and variance $\beta_{ks} = \int |X_k|^2 p(X_k|s) dX_k = e^{\mu_{ks} + 1/(2\nu_{ks})}$, and is approximated by $p(X_k|s) \approx \mathcal{N}(X_k|0, 1/\beta_{ks})$. Under this approximation, we have

$$p(s|Y_1, \ldots, Y_K) \propto p(s) \prod_k \mathcal{N}\left(Y_k|0, \frac{1}{\frac{1}{\gamma_k} + e^{\mu_{ks} + 1/(2\nu_{ks})}}\right). \tag{17}$$

The estimated signal can be constructed from the average of either $\bar{X}_{ks}$ or $\bar{\xi}_{ks}$, weighted by the posterior mixture probability

$$\hat{X}_k = \sum_s \bar{X}_{ks} p(s|Y_1, \ldots, Y_K) \tag{18}$$

$$\widehat{\xi}_k = \sum_s \bar{\xi}_{ks} p(s|Y_1, \ldots, Y_K) \tag{19}$$

$$\widehat{X}_k^{ls} = e^{\widehat{\xi}_k/2} e^{i\angle Y_k} \tag{20}$$

where the phase of the noisy signal $\angle Y_k$ is used. The time domain signal is synthesized by applying inverse fast Fourier transform (IFFT).

B. Variational Approximation for Signal Estimation

Variational approximation [26] employs a factorized posterior pdf. Here, we assume the posterior pdf over X_k and ξ_k conditioned on s factorizes

$$p(X_k, \xi_k, |Y_k, s) \approx q(X_k|s) q(\xi_k|s). \tag{21}$$

The difference between q and the true posterior is measured by the Kullback–Leibler (KL)-divergence [28], D, defined as

$$D(q \,\|\, p) = -E^q \left\{ \log \frac{p(s|Y_1, \ldots, Y_K) \prod_k p(X_k, \xi_k|Y_k, s)}{q(s) \prod_k q(X_k|s) q(\xi_k|s)} \right\} \tag{22}$$

where E^q is the expectation over q. Choose the optimal q that is closest to the true posterior in the sense of the KL -divergence, $q = \arg\min_q D(q\|p)$.

Following the derivation in [26], the optimal $q(X_k|s)$ satisfies

$$\log q(X_k|s) \propto \log p(Y_k|X_k) + \int d\xi_k q(\xi_k|s) \log p(X_k|\xi_k) \propto -\gamma_k |Y_k - X_k|^2 - \int e^{-\xi_k} q(\xi_k|s) d\xi_k |X_k|^2. \tag{23}$$

As shown later in (28), we can use $q(\xi_k|s) = \mathcal{N}(\xi_k|\bar{\xi}_{ks}, \psi_{ks})$. Because the above equation is quadratic in X_k, $q(X_k|s)$ is Gaussian

$$q(X_k|s) = \mathcal{N}(X_k|\overline{X}_{ks}, \phi_{ks}) \tag{24}$$

$$\overline{X}_{ks} = \frac{\gamma_k}{\phi_{ks}} Y_k \tag{25}$$

$$\phi_{ks} = \gamma_k + e^{-\bar{\xi}_{ks} + 1/(2\psi_{ks})}. \tag{26}$$

The optimal $q(\xi_k|s)$ that minimizes $D(q\|p)$ is

$$\log q(\xi_k|s) \propto \int dX_k q(X_k|s) \log p(X_k|\xi_k) + \log p(\xi_k|s) \propto -\xi_k - e^{-\xi_k} \int |X_k|^2 q(X_k|s) dX_k - \frac{\nu_{ks}}{2}(\xi_k - \mu_{ks})^2. \quad (27)$$

Because this pdf is hard to work with, we use the Laplace method to approximate it by a Gaussian

$$q(\xi_k|s) = \mathcal{N}(\xi_k|\bar{\xi}_{ks}, \psi_{ks}) \quad (28)$$

$$\bar{\xi}_{ks} = \rho_{ks} + \frac{1}{\psi_{ks}} \left(e^{-\rho_{ks}} \left(|\bar{X}_{ks}|^2 + \frac{1}{\phi_{ks}} \right) - \nu_{ks}(\rho_{ks} - \mu_{ks}) - 1 \right) \quad (29)$$

$$\psi_{ks} = e^{-\rho_{ks}} \left(|\bar{X}_{ks}|^2 + \frac{1}{\phi_{ks}} \right) + \nu_{ks}. \quad (30)$$

The ρ_{ks} is chosen to be the posterior mode, $\rho_{ks} = \bar{\xi}_{ks}$, the update rule is

$$\bar{\xi}_{ks} \leftarrow \bar{\xi}_{ks} + \frac{1}{\psi_{ks}} \left(e^{-\bar{\xi}_{ks}} \left(|\bar{X}_{ks}|^2 + \frac{1}{\phi_{ks}} \right) - \nu_{ks}(\bar{\xi}_{ks} - \mu_{ks}) - 1 \right) \quad (31)$$

$$\psi_{ks} \leftarrow e^{-\bar{\xi}_{ks}} \left(|\bar{X}_{ks}|^2 + \frac{1}{\phi_{ks}} \right) + \nu_{ks}. \quad (32)$$

The $\psi_{ks} > 0$ indicates $\log q(\xi_k|s)$ is a concave function in ξ_k, thus Newton's method is efficient.

The variational algorithm is initialized with $\xi_{ks} = \log(|Y_k|^2)$ and $\boldsymbol{\phi}_{ks} = \gamma_k + \exp(-\xi_{ks})$. Note that X_{ks} in (25) can be substituted into (31) and (32) to avoid redundant computation. Then the updates over ψ_{ks}, ξ_{ks} and $\boldsymbol{\phi}_{ks}$ iterate until convergence.

To compute the posterior mixture probability, we define

$$g_{ks} = \int q(X_k|s) q(\xi_k|s) \log \frac{p(Y_k|X_k) p(X_k|\xi_k) p(\xi_k|s)}{q(X_k|s) q(\xi_k|s)}$$
$$= \log \frac{\gamma_k \sqrt{\nu_{ks}}}{\pi \phi_{ks} \sqrt{\psi_{ks}}} - \gamma_k |Y_k|^2 + \phi_{ks} |\bar{X}_{ks}|^2 - \bar{\xi}_{ks} - \frac{\nu_{ks}}{2} \left[(\bar{\xi}_{ks} - \mu_{ks})^2 + \frac{1}{\psi_{ks}} \right] + \frac{1}{2}. \quad (33)$$

The posterior mixture probability is

$$q(s)=\frac{\exp\left(\sum_k g_{ks}\right)p(s)}{Z} \tag{34}$$

$$Z=\sum_s \exp\left(\sum_k g_{ks}\right)p(s). \tag{35}$$

The function $\log(Z) = \log p(Y_1, \ldots, Y_K) - D(q||p)$ increases when $D(q||p)$ decreases. Because we use a Gaussian for $q(\xi_k|s)$, $\log(Z)$ is not theoretically guaranteed to increase, but it is used empirically to monitor the convergence.

With the estimated log-spectra $\bar{\xi}_{ks}$, FFT coefficients $\bar{X}_{ks}$, and posterior mixture probability $q(s)$, signals are constructed in two ways given by (18) and (20). Time domain signal is synthesized by applying IFFT.

V. Experiments

The performances of the algorithms were evaluated using the materials provided by the speech separation challenge [29].

A. Dataset Description

The data set contained six-word segments of 34 speakers. Each segment was 1–2 seconds long sampled at 25 kHz. The acoustic signal followed the grammar, ⟨$command⟩ ⟨$color⟩ ⟨$preposition⟩ ⟨$letter⟩ ⟨$number⟩ ⟨$adverb⟩. There were 25 choices for letter (A–Z except W), ten choices for number and four choices for others. The training set contained segments of clean signals for each speaker, and the test set contained speeches corrupted by noise. The spectra of speech and noise averaged over one segment are shown in Fig. 2. In the plot, the speech and noise have the same power, i.e, 0-dB SNR. Because the spectrum of noise has the similar shape to that of speech, it is called speech shape noise (SSN). The test data consisted of noisy signals at four different SNRs, −12 dB, −6 dB, 0 dB, and 6 dB. There were 600 utterances for each SNR condition from all 34 speakers who contributed roughly equally. The task is to recover the speech signals corrupted by SSN. The performances of the algorithms were compared by the word recognition error rate using the provided speech recognition engine [29].

To evaluate our algorithm under different types of noise, we added the white Gaussian noise to the clean signals at SNR levels of −12 dB, −6 dB, 0 dB, 6 dB, 12 dB, to generate noisy signals.

The signal is divided into frames of length 800 with half over-lapping, and a Hanning window of size 800 is applied to each frame. Then a 1024-point FFT is performed on the zero-padded frames to extract the frequency components. The log-spectral coefficients were obtained by taking the log magnitude of the FFT coefficients. Due to the symmetry of FFT, only first 513 components were kept.

B. Training the Gaussian Scale Mixture Model

The GSMM with 30 mixtures was trained using 2 min of signal concatenated from the training set for each speaker. We applied the *k*-mean algorithm to partition the log-spectra

into $k = 30$ clusters. They were used to initialize the GMM which was further trained by standard EM algorithm. Initialized by the GMM, we ran the derived EM algorithm in Section III to train the GSMM. After training, the speech model was fixed and served as signal prior. It was not updated when processing the noisy signals.

C. Benchmarks for Comparison

The benchmark algorithms included the Wiener filter, STSA [8], the perceptual model [17], the linear approximation [12], [13], and the super-Gaussian model [10]. The spectrum of noise was assumed to be known and estimated from the noise.

1) Wiener Filter—The time varying Wiener filter makes use of the power of the signal and noise, and assumes they are stationary for a short period of time. In the experiment, we first divided the signals into frames of 800 samples long with half overlapping. The power of speech and noise was constant within each frame. To estimate them, we further divided each frame into sub-frames of 200-sample long with half overlapping. The sub-frames are zero-padded to 256 points, Hanning windows were applied and a 256-points FFT was performed. The average power of FFT coefficients over all sub-frames belong to frame t gave the estimation of the signal power, denoted by P^x_{tk}. The same method was used to compute the noise power denoted by P^n_{tk}. The signal was estimated as $X_{tjk}=(p^x_{tk}/(P^x_{tk}+P^n_{tk}))Y_{tjk}$ where j is the sub-frame index and k denotes the frequency bin. Applying IFFT on X_{tjk}, each frame can be synthesized by overlap-adding the sub-frames, and the estimated speech signal was obtained by overlap-adding the frames.

The performance of the Wiener filter can be regarded as an experimental upper bound. The signal and noise power was derived locally for each frame from the clean speech and noise. So the Wiener filter contained strong detailed speech priors.

2) STSA—After performing the 1024-point FFT on the zero-padded frames of length 800. The STSA models the FFT coefficients of the speech and noise by a single Gaussian, respectively, whose variances are estimated from clean signal and noise. The amplitude estimator is given by [8, Eq. (7)].

3) Perceptual Model—Because we consider the SSN, it is interesting to test the performance of the perceptually motivated noise reduction technique. The spectral similarity may pose difficulty to such models. For this purpose, we included the method described in [17]. The algorithm estimated the spectral amplitude by minimizing the cost function

$$C(\widehat{a}_k, a_k)=\begin{cases} (\widehat{a}_k - a_k - \frac{m_k}{2})^2 - (\frac{m_k}{2})^2, & \text{if} |\widehat{a}_k - a_k - \frac{m_k}{2}| > \frac{m_k}{2} \\ 0, & \text{otherwise.} \end{cases} \tag{36}$$

where $\hat{a}_k$ is the estimated spectral amplitude and a_k is the true spectral amplitude. This cost function penalizes the positive and negative errors differently, because positive estimation errors are perceived as additive noise and negative errors are perceived as signal attenuation [17]. Because of the stochastic property of speech, $\hat{a}_k$ minimizes the expected cost function

$$\widehat{a}_k = \arg\min_{\widehat{a}_k} \int\int C(\widehat{a}_k, a_k) p(\alpha_k, a_k | Y_k) d\alpha_k da_k \tag{37}$$

where α_k is the phase and $p(\alpha_k, a_k|Y_k)$ is the posterior signal distribution. Details of the algorithm can be found in [17]. The MATLAB code is available online [30]. The original code adds synthetic white noise to the clean signal, we modified it to add SSN to corrupt a speech at different SNR levels.

4) Linear Approximation—This approach was developed in [12], [13] and worked in the log-spectral domain. It assumed a GMM for the signal log-spectra and a Gaussian for the noise log-spectra. So the noise had a log-normal density, in contrast to Gaussian noise. The relationship among the log-spectra of the signal x, the noisy signal y and the noise n is given by

$$y_k = x_k + \log(1+\exp(n_k - x_k)) + \varepsilon_k \tag{38}$$

where ε_k is an error term.

However, this nonlinear relationship causes intractability. A linear approximation was used in [12], [13] by expanding (38) around $\tilde{z}_{ks} = (\tilde{x}_{ks}, \tilde{n}_{ks})^T$ linearly. This approximation provided efficient speech enhancement. The choice for $\tilde{z}_{ks}$ can be iteratively optimized.

5) Super-Gaussian Prior—This method was developed in [10]. Let $X_R = \mathrm{Re}\{X\}$ and $X_I = \mathrm{Im}\{X\}$ denote the real and the imaginary part of the signal FFT coefficients. They were processed separately and symmetrically. We consider the real part and assume X_R obey double-sided exponential distribution

$$p(X_R) = \frac{1}{\sigma_x} e^{-(2|X_R|/\sigma_x)}. \tag{39}$$

Assume the Gaussian noise N with density $p(N) = \mathcal{N}(0, 1/\sigma_n^2)$. Here, σ_x^2 and σ_n^2 are the means of $|X|^2$ and $|N|^2$, respectively. Let $\xi = \sigma_x^2/\sigma_n^2$ be the *a priori* SNR, $Y_R = \mathrm{Re}\{Y\}$ be the real part of the noisy signal FFT coefficient. Define $L_{R+} = 1/\sqrt{\xi} + Y_R/\sigma_n$, and $L_{R-} = 1/\sqrt{\xi} - Y_R/\sigma_n$. It was shown in [10, Eq. (11)] that the optimal estimator for the real part is

$$\widehat{X}_R = Y_R + \frac{\sigma_n}{\sqrt{\xi}} \frac{e^{2Y_R/\sigma_x}\mathrm{erfc}(L_{R+}) - e^{-(2Y_R/\sigma_x)}\mathrm{erfc}(L_{R-})}{e^{2Y_R/\sigma_x}\mathrm{erfc}(L_{R+}) + e^{-(2Y_R/\sigma_x)}\mathrm{erfc}(L_{R-})} \tag{40}$$

where $\mathrm{erfc}(x)$ denotes the complementary error function. The optimal estimator for the imaginary part $\hat{X}_I$ was derived analogously in the same manner. The FFT coefficient estimator was given by $\hat{X} = \hat{X}_R + i\hat{X}_I$.

D. Comparison Criteria

We employed two criteria to evaluate performance of all algorithms: SNR and word recognition error rate. In all experiments, the estimated time domain signals $\hat{x}[t]$ were normalized such that they have the same power as the clean signals.

1) Signal-to-Noise Ratio (SNR)—SNR is defined in the time domain as

$$\mathrm{SNR}=10\log_{10}\frac{\sum_t |x(t)|^2}{\sum_t |\hat{x}(t)-x(t)|^2} \tag{41}$$

where $x[t]$ is the clean signal and $\hat{x}[t]$ is the estimated signal.

2) Word Recognition Error Rate—The speech recognition engine based on the HTK package was provided on the ICSLP website [29]. It extracts 39 features from the acoustic waveforms, including 12 Mel-frequency cepstral coefficients (MFCC) and the logarithmic frame energy, their velocities (Δ MFCC) and accelerations (ΔΔ MFCC). The HMM with no skipover states and two states for each phoneme was used to model each word. The emission probability for each state was a GMM of 32 mixtures, of which the covariance matrices are diagonal. The grammar used in the recognizer is the same as the one shown in Section V-A. More details about the recognition engine are provided at [29].

To compute the recognition error rate, a score of {0, 1, 2, 3} was assigned to each utterance depending on how many key words (*color*, *letter*, *digit*) were incorrectly recognized. The average word recognition error rate was the average of the scores of all 600 testing utterances divided by 3, i.e., the percentage of wrongly recognized key words. This was carried out for each SNR condition.

E. Results

1) Speech Shaped Noise—We applied the algorithms to enhance the speech corrupted by SSN at four SNR levels and compared them by SNR and word recognition error rate. The Wiener filer was regarded as an experimental upper bound, because it incorporates detailed signal prior from the clean speech.

The spectrograms of female speech and male speech are shown in Figs. 3 and 4, respectively. Fig. 5 shows the output SNR as a function of input SNR for all algorithms. The output SNR is averaged over the 600 test segments. Fig. 6 plots the word recognition error rate.

The Wiener filter outperformed other methods in low SNR conditions. This is because the power of noise and speech was calculated locally, and it incorporated detailed prior information. The perceptual model and STSA failed to suppress the SSN because of the spectral similarity between the speech and the noise. The linear approximation gave very low word recognition error rate, but not superior SNR. The reason is that, using a GMM in the log-spectral domain as speech model, it reliably estimated the log-spectrum which is a good fit to the recognizer input (MFCC). Because the super-Gaussian prior model treated the real and imaginary parts of the FFT coefficients separately, it provided less accurate spectral amplitude estimation and was inferior to the linear approximation. Both the Laplace method and variational approximation, based on GSMM for the speech signal, gave superior SNR for signals constructed from the estimated FFT coefficients and lower word recognition error rate for signals constructed from the estimated log-spectra. This agreed with the expectation that frequency domain approach gave higher SNR, while log-spectral domain method was more suitable for speech recognition. In comparing the two methods, the variational approximation performed better than the Laplace method in the high SNR range. It is hard to compare them in the low SNR range, because speech enhancement was minimal.

Perceptually, the Wiener filter gave smooth and natural speeches. The signals enhanced by STSA, perceptual model, and supper-Gausian prior model, contained obvious noise, because such techniques are based on spectral analysis and failed to remove the SSN. The linear approximation removed the noise, but the output signals were discontinuous. For the algorithms based on Gaussian scale mixture models, the signals constructed from the estimated FFT coefficients were smoother than those constructed from the log-spectra. The reason was that the perceptual quality of signals was sensitive to the log-spectra, because the amplitudes were obtained by taking the exponential of the log-spectra. The discontinuity in the log-spectra was more noticeable than that in the FFT coefficients. Because the phase of the noisy signals was used to synthesize the estimated signals, the enhanced signals contained reverberation. Among all the algorithms, we found GSMM with Laplace method gave the most satisfactory results, the noise was removed and the signals were smooth. The examples are available at http://chord.ucsd.edu/~jiucang/gsmm.

2) White Gaussian Noise—We also applied the algorithms to enhance the speeches corrupted by the white Gaussian noise. For this experiment, we tested them under five SNR levels: −12 dB, −6 dB, 0 dB, 6 dB, and 12 dB. The algorithms were the same as the previous section. Fig. 7 shows the output SNRs and Fig. 8 plotts the word recognition error rate.

We noticed that all the algorithms were able to improve the SNR. The signals constructed from the FFT coefficients estimated from the GSMM with Laplace method gave the best output SNR for all SNR inputs. The spectral analysis models, like STSA and perceptual models, were able to improve the SNR too, because of the spectral difference between the signal and noise. The algorithms that estimated the log-spectra (Linear, GSMM Lap LS, and GSMM VarLS) gave the lower word recognition error rate, because the log-spectra estimation was a good fit to the recognizer. For the GSMM, the FFT coefficients estimation offered better SNR and log-spectra estimation offered lower recognition error rate, as expected.

Although STSA, perceptual model and super-Gaussian prior all increased SNR, the residual noise was perceptually noticeable. Signals constructed from the estimated log-spectra sounded less continuous than signals constructed from the estimated FFT coefficients. However, the signals sounded like being synthesized, because the phase of the noisy signal was used. The examples are available at http://chord.ucsd.edu/~ji-ucang/gsmm.

VI. Conclusion

We have presented a novel Gaussian scale mixture model for speech signal and derived two methods for speech enhancement: the Laplace method and a variational approximation. The GSMM treats the FFT coefficients and log-spectra as two random variables, and models their relationship probabilistically. This enables us to estimate both the FFT coefficients, which produce better signal quality in the time domain, and the log-spectra, which are more suitable for speech recognition. The performances of the proposed algorithms were demonstrated by applying them to enhance speech corrupted by SSN and the white noise. The FFT coefficients estimation gave higher SNR, while the log-spectra estimation produced lower word recognition error rate.

Acknowledgments

The authors would like to thank H. Attias for suggesting the model and helpful advice on the inference algorithms. They would also like to thank the anonymous reviewers for valuable suggestions.

References

1. Ephraim, Y.; Cohen, I. The Electrical Engineering Handbook. Boca Raton, FL: CRC; 2006. Recent advancements in speech enhancement.
2. Attias H, Platt JC, Acero A, Deng L. Speech denoising and dereverberation using probabilistic models. in Proc NIPS 2000:758–764.
3. Gannot S, Burshtein D, Weinstein E. Signal enhancement using beamforming and nonstationarity with applications to speech. IEEE Trans Signal Process Aug;2001 49(8):1614–1626.
4. Cohen I, Gannot S, Berdugo B. An integrated real-time beamforming and postfiltering system for nonstationary noise environments. EURASIP J Appl Signal Process 2003;11:1064–1073.
5. Boll SF. Suppression of acoustic noise in speech using spectral subtraction. IEEE Trans Acoust, Speech, Signal Process Apr;1979 ASSP-27(2):113–120.
6. Ephraim Y, Trees HLV. A signal subspace approach for speech enhancement. IEEE Trans Speech Audio Process Jul;1995 3(4):251–266.
7. Ephraim Y. Statistical-model-based speech enhancement systems. Proc IEEE Oct;1992 80(10): 1526–1555.
8. Ephraim Y, Malah D. Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator. IEEE Trans Acoust, Speech, Signal Process 1984;ASSP-32(6):1109–1121.
9. Ephraim Y, Malah D. Speech enhancement using a minimum mean-square error log-spectral amplitude estimator. IEEE Trans Acoust, Speech, Signal Process Apr;1985 33(2):443–445.
10. Martin R. Speech enhancement based on minimum mean-square error estimation and supergaussian priors. IEEE Trans Speech Audio Process Sep;2005 13(5):845–856.
11. Burshtein D, Gannot S. Speech enhancement using a mixture-maximum model. IEEE Trans Speech Audio Process Sep;2002 10(6):341–351.
12. Frey B, Kristjansson T, Deng L, Acero A. Learning dynamic noise models from noisy speech for robust speech recognition. Proc NIPS 2001:1165–1171.
13. Kristjansson T, Hershey J. High resolution signal reconstruction. Proc IEEE Workshop ASRU 2003:291–296.
14. Hopgood JR, Rayner PJ. Single channel nonstationary stochastic signal separation using linear time-varying filters. IEEE Trans Signal Process Jul;2003 51(7):1739–1752.
15. Czyzewski A, Krolikowski R. Noise reduction in audio signals based on the perceptual coding approach. Proc IEEE WASPAA 1999:147–150.
16. Lee J-H, Jung H-J, Lee T-W, Lee S-Y. Speech coding and noise reduction using ica-based speech features. Proc Workshop ICA 2000:417–422.
17. Wolfe P, Godsill S. Towards a perceptually optimal spectral amplitude estimator for audio signal enhancement. Proc ICASSP 2000;2:821–824.
18. Ephraim Y. A Bayesian estimation approach for speech enhancement using hidden Markov models. IEEE Trans Signal Process Apr;1992 40(4):725–735.
19. Ephraim Y. Gain-adapted hidden Markov models for recognition of clean and noisy speech. IEEE Trans Signal Process Jun;1992 40(6):1303–1316.
20. Bishop, CM. Neural Networks for Pattern Recognition. New York: Oxford Univ. Press; 1995.
21. Andrews D, Mallows C. Scale mixture of normal distributions. J R Statist Soc 1974;36(1):99–102.
22. Wolfe P, Godsill S, Ng W. Bayesian variable selection and regularization for time-frequency surface estimation. J R Statist Soc 2004;66(3):575–589.
23. Fevotte C, Godsill S. A Bayesian approach for blind separation of sparse sources. IEEE Trans Audio, Speech, Lang Process Dec;2006 14(6):2174–2188.
24. Vincent E, Plumbley M. Low bit-rate object coding of musical audio using Bayesian harmonic models. IEEE Trans Audio, Speech, Lang Process May;2007 15(4):1273–1282.
25. Azevedo-Filho A, Shachter RD. Laplace's method approximations for probabilistic inference in belief networks with continuous variables. Proc UAI 1994:28–36.
26. Attias H. A variational Bayesian framework for graphical models. Proc NIPS 2000;12:209–215.
27. Dempster A, Laird N, Rubin D. Maximum likelihood from incomplete data via the em algorithm. J R Statist Soc 1977;39(1):1–38.

28. Cover, TM.; Thomas, JA. Elements of Information Theory. New York: Wiley-Interscience; 1991.
29. Cooke, M.; Lee, T-W. Speech Separation Challenge. [Online]. Available: http://www.dcs.shef.ac.uk/~martin/SpeechSeparationChallenge.html
30. Wolfe, P. Example of Short-Time Spectral Attenuation. [Online]. Available: http://www.eecs.harvard.edu/~patrick/research/stsa.html
31. Cohen I. Noise spectrum estimation in adverse environments: Improved minima controlled recursive averaging. IEEE Trans Speech Audio Process Sep;2003 11(5):466–475.
32. Cohen I, Berdugo B. Noise estimation by minima controlled recursive averaging for robust speech enhancement. IEEE Signal Process Lett Jan;2002 9(1):12–15.
33. McAulay R, Malpass M. Speech enhancement using a soft-decision noise suppression filter. IEEE Trans Acoust, Speech, Signal Process Apr;1980 ASSP-28(2):137–145.
34. Martin R. Noise power spectral density estimation based on optimal smoothing and minimum statistics. IEEE Trans Speech Audio Process Jul;2001 9(5):504–512.
35. Wang D, Lim J. The unimportance of phase in speech enhancement. IEEE Trans Acoust, Speech, Signal Process Aug;1982 ASSP-30(4):679–681.
36. Attias H, Deng L, Acero A, Platt J. A new method for speech denoising and robust speech recognition using probabilistic models for clean speech and for noise. Proc Eurospeech 2001:1903–1906.
37. Brandstein MS. On the use of explicit speech modeling in microphone array applications. Proc ICASSP 1998:3613–3616.
38. Hong L, Rosca J, Balan R. Independent component analysis based single channel speech enhancement. Proc ISSPIT 2003:522–525.
39. Beaugeant C, Scalart P. Speech enhancement using a minimum least-squares amplitude estimator. Proc IWAENC 2001:191–194.
40. Lotter T, Vary P. Noise reduction by maximum a posterior spectral amplitude estimation with supergaussian speech modeling. Proc IWAENC 2003:83–86.
41. Breithaupt C, Martin R. Mmse estimation of magnitude-squared dft coefficoents with supergaussian priors. Proc ICASSP 2003:848–851.
42. Benesty, J.; Chen, J.; Huang, Y.; Doclo, S. Study of the wiener filter for noise reduction. In: Benesty, J.; Makino, S.; Chen, J., editors. Speech Enhancement. New York: Springer; 2005. p. 9-42.

Appendix EM Algorithm for Training the GSMM

We present the details for the EM algorithm here. The parameters are estimated by maximizing the log-likelihood which is given by (10).

Expectation Step

When $q(\xi_{kt}|s_t)q(s_t)$ equals to the posterior distribution, the cost $\mathcal{F}(q, \theta)$ equals to $\mathcal{L}(\theta)$ and is maximized. The $q(\xi_{kt}|s_t)$ is computed as

$$\begin{aligned}\log q(\xi_{kt}|s_t) &= \log p(X_{kt}|\xi_{kt}) + \log p(\xi_{kt}|s_t) + c \\ &= -\xi_{kt} - e^{-\xi_{kt}}|X_{kt}|^2 - \frac{\nu_{ks}}{2}(\xi_{kt} - \mu_{ks})^2 + c\end{aligned} \tag{42}$$

where c is a constant. There is no closed-form density, we use Laplace method [25] approximate q by a Gaussian

$$q(\xi_{kt}|s_t) = \mathcal{N}(\xi_{kt}|\bar{\xi}_{kts_t}, \varphi_{kts_t}) \tag{43}$$

$$\bar{\xi}_{kts_t}=\hat{\xi}_{kts_t}+\frac{1}{\varphi_{kts_t}}\left(e^{-\hat{\xi}_{kts_t}}|X_{kt}|^2-\nu_{ks_t}\hat{\xi}_{kts_t}+\nu_{ks_t}\mu_{ks_t}-1\right) \tag{44}$$

$$\varphi_{kts_t}=e^{-\hat{\xi}_{kts_t}}|X_{kt}|^2+\nu_{ks_t}. \tag{45}$$

where $\hat{\xi}_{kts_t}$ is chosen to be the mode of the posterior and is iteratively updated by

$$\bar{\xi}_{kts_t}\leftarrow\bar{\xi}_{kts_t}+\frac{1}{\varphi_{kts_t}}\left(e^{-\bar{\xi}_{kts_t}}|X_{kt}|^2-\nu_{ks_t}\bar{\xi}_{kts_t}+\nu_{ks_t}\mu_{ks_t}-1\right). \tag{46}$$

This update rule is equivalent to maximizing $\log q(\xi_{kt}|s_t)$ using the Newton's method

$$\bar{\xi}_{kts_t}\leftarrow\bar{\xi}_{kts_t}-\frac{[\log q(\xi_{kt}|s_t)]'_{\xi_{kt}=\bar{\xi}_{kts_t}}}{[\log q(\xi_{kt}|s_t)]''_{\xi_{kt}=\bar{\xi}_{kts_t}}}. \tag{47}$$

Take the derivative of $\mathcal{F}(q,\theta)$ with respect to $q(s_t)$ and set it to zero, we can obtain the optimal $q(s_t)$. Define

$$\begin{aligned} f_{kts_t}&=\int q(\xi_{kt}|s_t)(\log p(X_{kt},\xi_{kt}|s_t)-\log q(\xi_{kt}|s_t))\\ &=\log\frac{\sqrt{\nu_{ks_t}}}{\pi\sqrt{\varphi_{kts_t}}}-e^{-\bar{\xi}_{kts_t}+1/(2\varphi_{kts_t})}|X_{kt}|^2-\bar{\xi}_{kts_t}-\frac{\nu_{ks_t}}{2}\left(\frac{1}{\varphi_{kts_t}}+(\bar{\xi}_{kts_t}-\mu_{ks_t})^2\right)+\frac{1}{2}. \end{aligned} \tag{48}$$

Then $q(s_t)$ can be obtained as

$$q(s_t)=\frac{\exp\left(\sum_k f_{kts_t}\right)p(s_t)}{Z_t} \tag{49}$$

$$Z_t=\sum_{s_t}\exp\left(\sum_k f_{kts_t}\right)p(s_t). \tag{50}$$

Maximization Step

The M-step optimizes $\mathcal{F}(q,\theta)$ over the model parameters θ

$$\mu_{ks} = \frac{\sum_t q(s_t = s)\xi_{kts_t}}{\sum_t q(s_t = s)} \tag{51}$$

$$\frac{1}{\nu_{ks}} = \frac{\sum_t q(s_t = s)\left[(\bar{\xi}_{kts_t} - \mu_{ks})^2 + \frac{1}{\varphi_{kts_t}}\right]}{\sum_t q(s_t = s)} \tag{52}$$

$$p(s) = \frac{\sum_t q(s_t = s)}{\sum_{ts} q(s_t = s)}. \tag{53}$$

The cost $\mathcal{F}$ is computed as $\mathcal{F} = \Sigma_t \log(Z_t)$ which can be used empirically to monitor the convergence, because the $\mathcal{F}$ is not guaranteed to increase due to the approximation in the E-step.

The parameters of a GMM trained in the log-spectral domain are used to initialize the EM algorithm. The E-step and M-step are iterated until convergence, which is very quick because ξ_k simulates the log-spectra.

Biographies

Jiucang Hao received the B.S. degree from the University of Science and Technology of China (USTC), Hefei, and the M.S. degree from University of California at San Diego (UCSD), both in physics. He is currently pursuing the Ph.D. degree at UCSD.

His research interests are to develop new machine learning algorithms and apply them to areas such as speech enhancement, source separation, biomedical data analysis, etc.

Te-Won Lee (M'03–SM'06) received the M.S. degree and the Ph.D. degree (*summa cum laude*) in electrical engineering from the University of Technology Berlin, Berlin, Germany, in 1995 and 1997, respectively.

He was Chief Executive Officer and co-Founder of SoftMax, Inc., a start-up company in San Diego developing software for mobile devices. In December 2007, SoftMax was acquired by Qualcomm, Inc., the world leader in wireless communications where he is now a Senior Director of Technology leading the development of advanced voice signal processing technologies. Prior to Qualcomm and SoftMax, he was a Research Professor at the Institute for Neural Computation, University of California, San Diego, and a Collaborating Professor in the Biosystems Department, Korea Advanced Institute of Science and Technology (KAIST). He was a Max-Planck Institute Fellow (1995–1997) and a Research Associate at the Salk Institute for Biological Studies (1997–1999).

Dr. Lee received the Erwin-Stephan Prize for excellent studies (1994) from the University of Technology Berlin, the Carl-Ramhauser prize (1998) for excellent dissertations from the DaimlerChrysler Corporation and the ICA Unsupervised Learning Pioneer Award (2007). In 2007, he received the SPIE Conference Pioneer Award for work on independent component analysis and unsupervised learning algorithms.

Terrence J. Sejnowski (SM'91–F'06) is the Francis Crick Professor at The Salk Institute for Biological Studies where he directs the Computational Neurobiology Laboratory, an Investigator with the Howard Hughes Medical Institute, and a Professor of Biology and Computer Science and Engineering at the University of California, San Diego, where he is Director of the Institute for Neural Computation. The long-range goal of his laboratory is to understand the computational resources of brains and to build linking principles from brain to behavior using computational models. This goal is being pursued with a combination of theoretical and experimental approaches at several levels of investigation ranging from the biophysical level to the systems level. His laboratory has developed new methods for analyzing the sources for electrical and magnetic signals recorded from the scalp and

hemodynamic signals from functional brain imaging by blind separation using independent components analysis (ICA). He has published over 300 scientific papers and 12 books, including *The Computational Brain* (MIT Press, 1994) with Patricia Churchland.

Dr. Sejnowski received the Wright Prize for Interdisciplinary Research in 1996, the Hebb Prize from the International Neural Network Society in 1999, and the IEEE Neural Network Pioneer Awardin 2002. His was elected an AAAS Fellow in 2006 and to the Institute of Medicine of the National Academies in 2008.

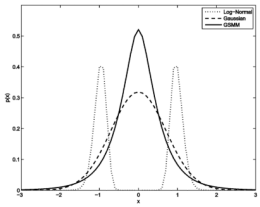

Fig. 1.

Distributions for the real part of X_k, with its imaginary part fixed at 0. The log-normal (dotted) has two modes. The GSMM (solid) is more peaky than Gaussian (dashed).

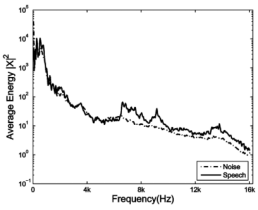

Fig. 2.

Plot of spectra of noise (dotted line) and clean speech (solid line) averaged over one segments under 0-dB SNR. Note the similar spectral shape.

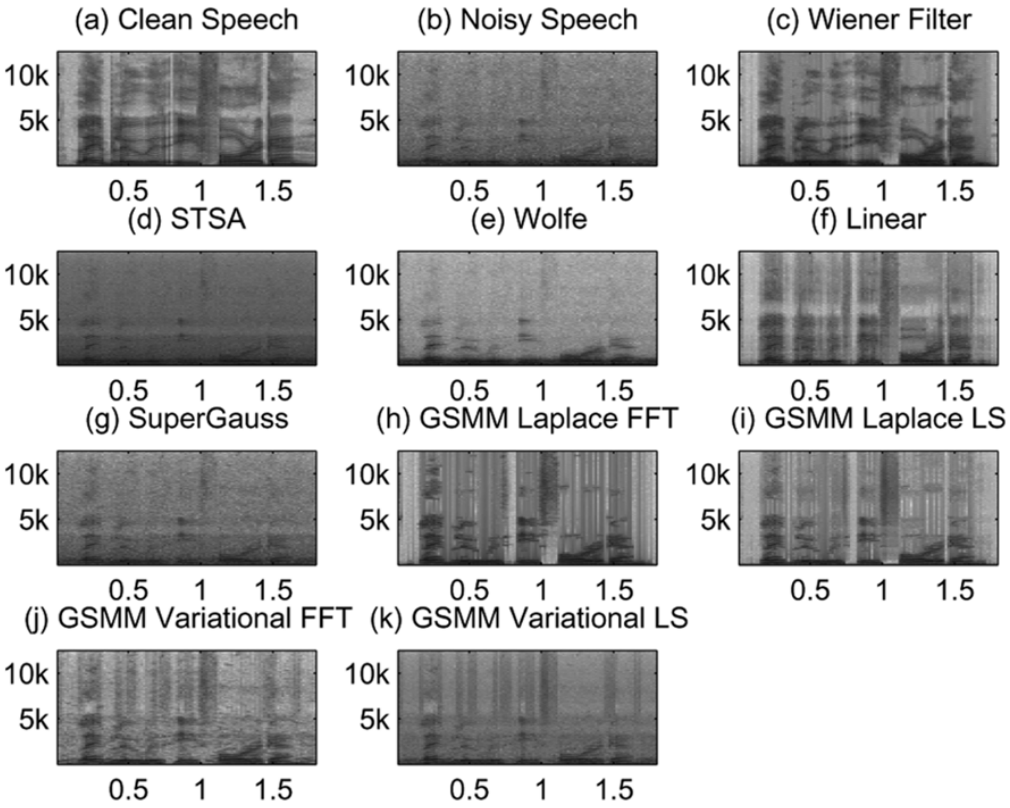

Fig. 3.

Spectrogram of a female speech “lay blue with e four again.” (a) Clean speech; (b) noisy speech of 6-dB SNR; (c–j) enhanced signals by (c) Wiener filter, (d) STSA, (e) perceptual model (Wolfe), (f) linear approximation (Linear), (g) super Gaussian prior (SuperGauss), (h) FFT coefficients estimation by GSMM using Laplace method, see (18), (i) log-spectra estimation by GSMM using Laplace method, see (20), (j) FFT coefficients estimation by GSMM using variational approximation, (k) log-spectra estimation by GSMM using variational approximation.

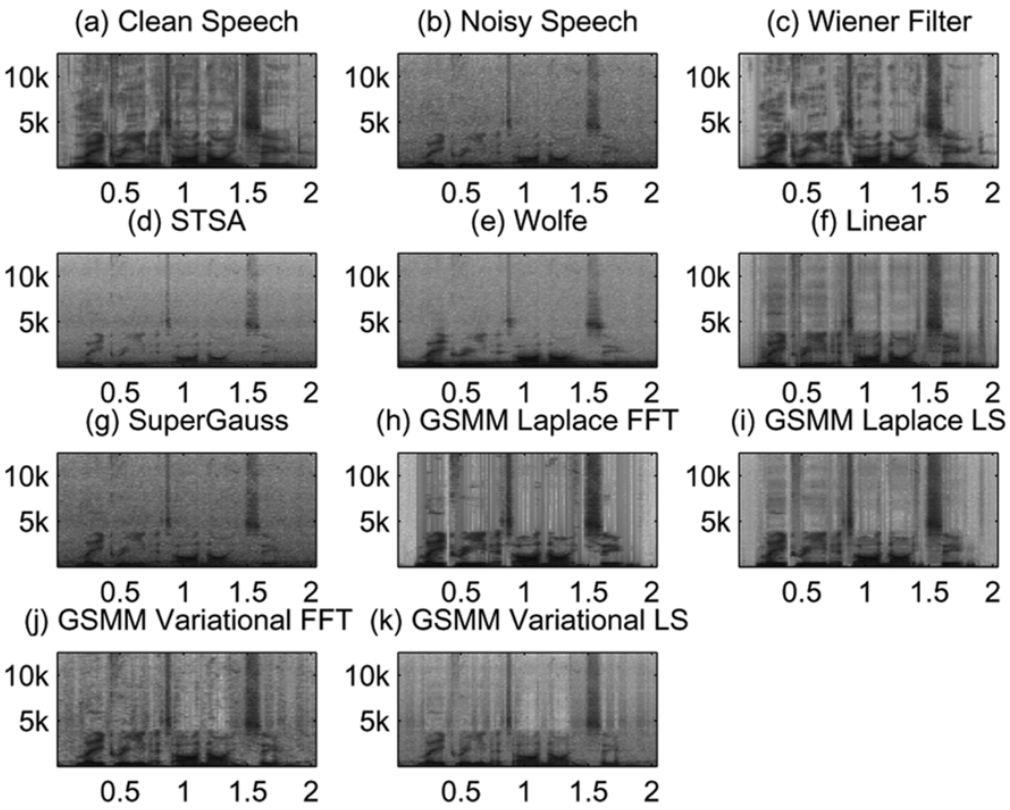

Fig. 4.

Spectrogram of a male speech “lay green at r nine soon.” (a) Clean speech; (b) noisy speech of 6-dB SNR; (c–i) enhanced signal by various algorithms. See Fig. 3.

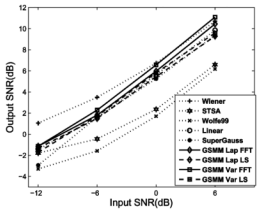

Fig. 5.

Output SNRs as a function of the input SNR for nine models (inset) for the case that the speeches are corrupted by SSN. See Fig. 3 for description of algorithms.

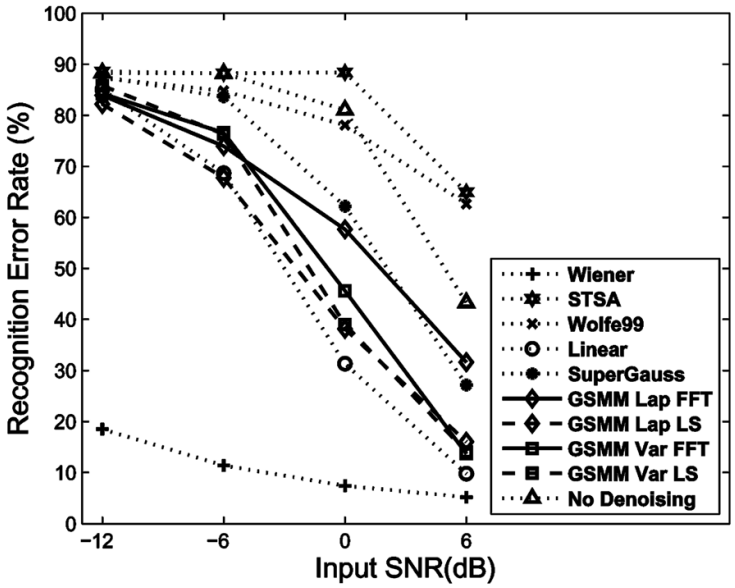

Fig. 6.

Word recognition error rate as a function of the input SNR for nine models (inset) for the case that the speeches are corrupted by SSN. See Fig. 3 for description of algorithms.

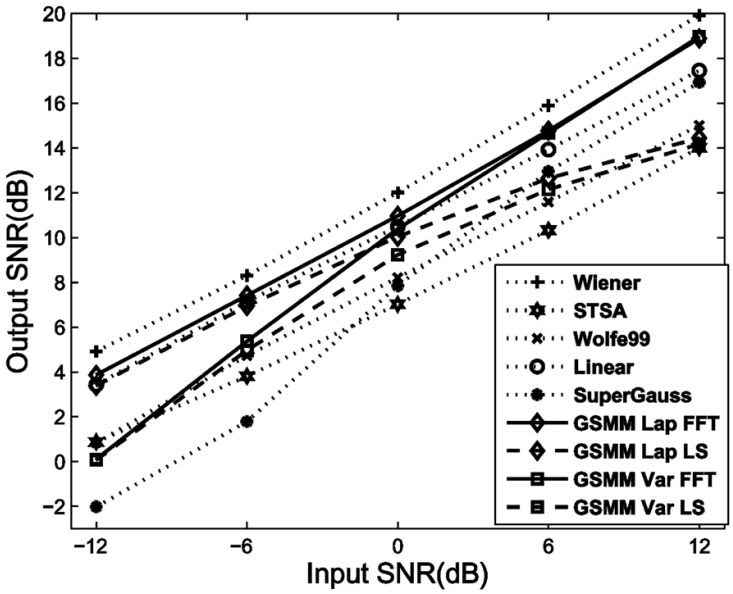

Fig. 7.

Output SNRs as a function of the input SNR for nine models (inset) for the case that the speeches are corrupted by white Gaussian noise. See Fig. 3 for description of algorithms.

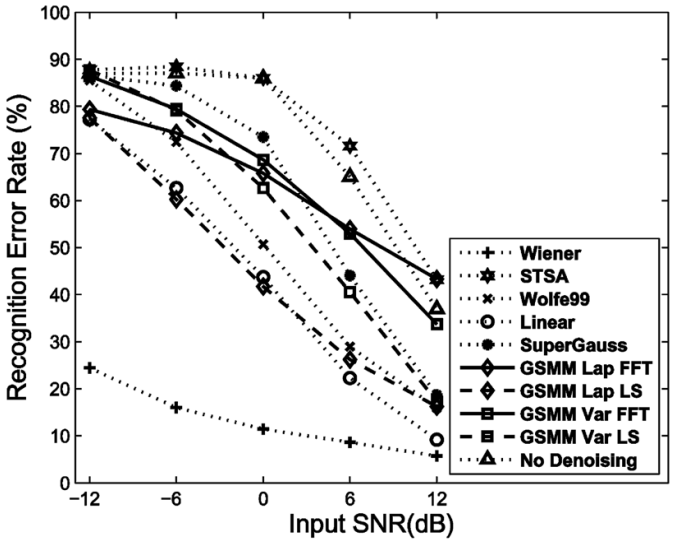

Fig. 8.

Word recognition error rate as a function of the input SNR for nine models (inset) for the case that the speeches are corrupted by white Gaussian noise. See Fig. 3 for description of algorithms.