A Noise-Robust FFT-Based Auditory Spectrum With Application in Audio Classification

Wei Chu and Benoît Champagne

Abstract—**In this paper, we investigate the noise robustness of Wang and Shamma's early auditory (EA) model for the calculation of an auditory spectrum in audio classification applications. First, a stochastic analysis is conducted wherein an approximate expression of the auditory spectrum is derived to justify the noise-suppression property of the EA model. Second, we present an efficient fast Fourier transform (FFT)-based implementation for the calculation of a noise-robust auditory spectrum, which allows flexibility in the extraction of audio features. To evaluate the performance of the proposed FFT-based auditory spectrum, a set of speech/music/noise classification tasks is carried out wherein a support vector machine (SVM) algorithm and a decision tree learning algorithm (C4.5) are used as the classifiers. Features used for classification include conventional Mel-frequency cepstral coefficients (MFCCs), MFCC-like features obtained from the original auditory spectrum (i.e., based on the EA model) and the proposed FFT-based auditory spectrum, as well as spectral features (spectral centroid, bandwidth, etc.) computed from the latter. Compared to the conventional MFCC features, both the MFCC-like and spectral features derived from the proposed FFT-based auditory spectrum show more robust performance in noisy test cases. Test results also indicate that, using the new MFCC-like features, the performance of the proposed FFT-based auditory spectrum is slightly better than that of the original auditory spectrum, while its computational complexity is reduced by an order of magnitude.**

Index Terms—**Audio classification, C4.5, early auditory (EA) model, noise suppression, self-normalization, support vector machine (SVM).**

I. Introduction

Recent years have seen extensive research on audio classification algorithms which provide useful information for both audio and video content understanding. Among many different audio classes in the field of audio classification, the generic classes of speech and music have attracted much attention. Saunders [1] has used a measure of energy contour and the distribution of zero-crossing rate (ZCR) in discriminating speech from music on broadcast FM radio. Scheirer and Slaney [2] proposed to use as many as 13 features, such as 4-Hz modulation energy, spectral centroid, etc., to classify speech and music, where a correct classification rate of 94.2% has been reported for 20-ms segments and 98.6% for 2.4-s segments. Low bit-rate audio coding is another application that can benefit from distinguishing speech from music [3], [4].

Besides speech and music, many other audio classes, including environmental sounds and background noise, have been investigated. In [5], a system for content-based classification, search, and retrieval of audio signals is presented wherein a wide variety of sounds are selected from animals, machines, musical instruments, speech, and the nature. Zhang and Kuo [6] have proposed a hierarchical system for audio classification and retrieval where audio clips are first classified and segmented into speech, music, environmental sounds, and silence; the environmental sounds are then further classified into ten classes using a hidden Markov model (HMM). Lu *et al.* [7] also proposed a two-stage robust approach that is capable of classifying and segmenting an audio stream into speech, music, environment sound, and silence. In [8] and [9], special sound effects which are related to entertainment or sport events, such as laughter, scream, etc., have been investigated. Mixed or hybrid sounds have also been studied, for example, speech with noise or music background, environmental sound with music background, etc. [10], [11]. Recently, a fuzzy approach was proposed where a fuzzy class is reserved for input audio that cannot be classified as pure speech, music, or silence [12].

Despite the growing interest in audio classification algorithms, as seen from the above references, the effect of background noise on the classification performance has not been investigated widely. In fact, a classification algorithm trained using clean sequences may fail to work properly when the actual testing sequences contain background noise with certain signal-to-noise ratio (SNR) levels (see test results in [13]–[16]). For example, results from [13] show that, using a set of Mel-frequency cepstral coefficients (MFCCs) as features, the error rate of speech/music classification increases significantly from 0% in a clean test to 41% in a test where SNR $= 10$ dB. For practical applications wherein environmental sounds are involved in audio classification tasks, noise robustness is an essential characteristic of the processing system.

Recently, the early auditory (EA) model presented by Wang and Shamma [17] has been employed in a two-class audio classification task (i.e., speech/music using a Gaussian mixture model as the classifier), and robust performance in noisy environments has been reported [13]. For example, at SNR $= 15$ dB, the error rate of the auditory based features is 17.7% compared to 40.3% for the conventional MFCC features. The EA model calculates a so-called auditory spectrum based on a series of linear and nonlinear processing steps including filtering with a set of constant-Q bandpass filters. According to the analysis in [17], the noise-robustness of the EA model can be attributed in part to its self-normalization property which causes spectral enhancement

Manuscript received February 16, 2007; revised July 25, 2007. This work was supported by a grant from the Natural Sciences and Engineering Research Council of Canada (NSERC). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Hiroshi Sawada.

The authors are with the Department of Electrical and Computer Engineering, McGill University, Montréal, QC H3A 2A7, Canada (e-mail: wei.chu@mail.mcgill.ca; benoit.champagne@mcgill.ca).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASL.2007.907569

or noise suppression. These conclusions on the self-normalization property are obtained using a qualitative analysis first, followed by a quantitative analysis wherein a closed-form expression of the auditory spectrum is derived. Due to the nonlinearity of the EA model, for the quantitative analysis, only a special simplified case has been studied wherein a step function is used to replace the original nonlinear sigmoid compression function. With respect to the limitation of the quantitative analysis in [17], it is of interest to investigate the noise-suppression property from a broader perspective, i.e., to derive a closed-form expression for auditory spectrum using a more general sigmoid-like function, and to conduct relevant analysis.

The noise-robustness of the original EA model has been demonstrated in different applications [13], [16], [17]. However, this model is characterized by high computational requirements and the use of nonlinear processing. It is therefore desirable to derive an approximated version of the EA model in the frequency domain, where efficient fast Fourier transform (FFT) algorithms are available. In an earlier study [16], we proposed such a simplified FFT-based spectrum wherein a local self-normalization scheme is implemented. Results from a speech/music/noise classification task show that the performance of the proposed FFT-based spectrum is comparable to that of the original EA model while its computational complexity is much lower. This FFT-based spectrum employs a simple grouping scheme to reduce the dimension of the power spectrum vector. However, this scheme fails to give a clear interpretation of the meaning of the frequency index. In applications where frequency-dependent audio features need to be extracted (e.g., spectral centroid, bandwidth), it would be more appropriate, instead of this simple grouping scheme, to group or select power spectrum components based on the original constant-Q bandpass filters.

In this paper, first, we extend the analysis in [17] by investigating the noise-suppression property of the EA model from a more general perspective wherein a closed-form expression of the auditory spectrum is derived by using Gaussian cumulative distribution function (CDF) as an approximation to the original sigmoid compression function. Second, an improved implementation is presented for the calculation of an FFT-based auditory spectrum which extends our previous work in [16]. The introduced improvements include the use of characteristic frequency (CF) values of the cochlear filters in the EA model for power spectrum selection, and the use of a pair of fast and slow running averages over the frequency axis for the implementation of the self-normalization. With these improvements, the proposed FFT-based auditory spectrum allows flexibility in the extraction of noise-robust audio features.

To evaluate the noise-robustness of the proposed FFT-based auditory spectrum, a three-class (i.e., speech/music/noise) audio classification task is carried out wherein a support vector machine (SVM) algorithm and a decision tree learning algorithm (C4.5 [18]) are employed for classification. Audio features used in this work include: conventional Mel-frequency cepstral coefficients (MFCCs), MFCC-like features obtained from the original auditory spectrum (i.e., based on the EA model), and the proposed FFT-based auditory spectrum, as well as spectral features (spectral centroid, bandwidth, etc.) computed from the latter. Compared to the conventional MFCC features, the MFCC-like features and the spectral features derived from the original or the proposed FFT-based auditory spectra show more robust performance in noisy test cases. It is also noted that, using the new MFCC-like features, the performance of the proposed FFT-based auditory spectrum is slightly better than that of the original auditory spectrum, while its computational complexity is reduced by an order of magnitude, i.e., a factor of 10 or more. The robustness of the MFCC-like features derived from the proposed FFT-based auditory spectrum is further confirmed by test results of noise/non-noise classification experiments.

The paper is organized as follows. The EA model presented by Wang and Shamma [17], together with the original analysis of its noise-robustness property, are summarized in Section II. As an extension to this original analysis, the proposed analysis of self-normalization based on Gaussian cumulative distribution function is presented in Section III. The improved implementation for the calculation of the FFT-based auditory spectrum is detailed in Section IV. Section V discusses the extraction of audio features and the setup of the classification tests. Test results are presented in Section VI, while Section VII concludes this work.

II. EA Model

A. Structure of the EA Model

In [17] and [19], a computational auditory model is described based on neurophysiological, biophysical, and psychoacoustical investigations at various stages of the auditory system. It consists of two basic stages, i.e., an early stage and a central stage. The former, called the EA model, describes the transformation of the audio signal into an internal neural representation referred to as auditory spectrogram, whereas the latter analyzes the spectrogram to estimate the content of its spectral and temporal modulations. In this paper, we focus on an EA model which can be simplified as a three-stage process as shown in Fig. 1 [17]. An audio signal entering the ear first produces a complex spatio–temporal pattern of vibrations along the basilar membrane (BM). A simple way to describe the characteristic response of the BM is to model it as a bank of constant-Q highly asymmetric bandpass filters with impulse responses $h(t, s)$, where t is the time index and s denotes a specific location on the BM (or equivalently, a channel index).

At the next stage, the motion on the BM is transformed into neural spikes in the auditory nerves. The process at this stage can be modeled by the following three steps: a temporal derivative which converts instantaneous membrane displacement into velocity, a sigmoid compression function $g(\cdot)$ which models the nonlinear channel through the hair cells, and a low-pass filter $w(t)$ accounting for the leakage of the cell membranes.

At the last stage, a lateral inhibitory network (LIN) detects discontinuities along the cochlear axis s. The operations can be divided into the following four steps: a derivative with respect to the tonotopic axis s which describes the lateral interaction among LIN neurons, a local smoothing $v(s)$ which accounts for the finite spatial extent of the lateral interactions, a half-wave rectifier (HWR) which models the nonlinearity of the LIN

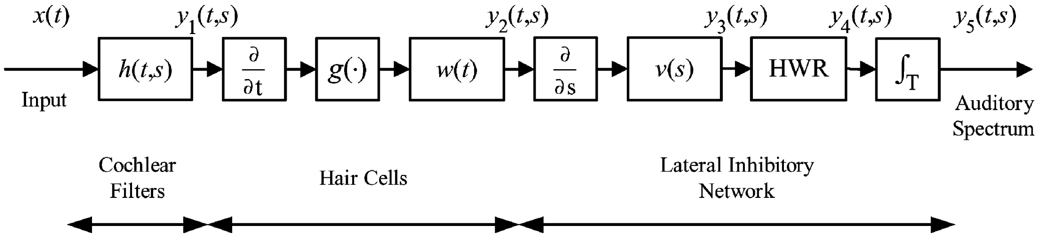

Fig. 1. Schematic description of the early auditory model [17].

neurons, and a temporal integration which reflects the fact that the central auditory neurons are unable to follow rapid temporal modulations.

These operations effectively compute a spectrogram of an acoustic signal. At a specific time index t, the output $y_5(t,s)$ is referred to as an auditory spectrum.

B. *Noise Robustness of the EA Model*

In [17], through a stochastic analysis, this EA model is proved to be noise robust due to an inherent self-normalization property. The main results of this analysis are summarized next.

1) Qualitative Analysis of the Self-Normalization Property: Suppose the input signal $x(t)$ can be modeled as a random process with zero mean. If the bandwidth of the temporal integrator in Fig. 1 is narrow enough, the output auditory spectrum $y_5(t,s)$ can be approximated by $E[y_4(t,s)]$ [17], where $E[\cdot]$ denotes statistical expectation; $E[y_4(t,s)]$ is referred to as an auditory spectrum in [17].

For the sake of simplicity, the temporal and spatial smoothing filters $w(t)$ and $v(s)$ are ignored in the analysis [17]. Define quantities U and V as[1]

$$U \equiv U(t,s) = \frac{\partial}{\partial t} y_1(t,s) = \left(\frac{\partial}{\partial t} x(t)\right) *_t h(t,s) \qquad (1)$$

$$V \equiv V(t,s) = \frac{\partial^2}{\partial s \partial t} y_1(t,s) = \left(\frac{\partial}{\partial t} x(t)\right) *_t \left(\frac{\partial}{\partial s} h(t,s)\right) \qquad (2)$$

where $*_t$ denotes the time-domain convolution. It can be shown that

$$E\left[y_4(t,s)\right] = \int_{-\infty}^{\infty} g'(u) E\left[\max(V,0) | U = u\right] f_U(u) du \qquad (3)$$

where $f_U(u)$ denotes the probability density function (pdf) of U at given (t,s) and the derivative function $g'(u)$ is assumed nonnegative. Based on (3), the following qualitative conclusions are reached in [17].

1) The auditory spectrum $E[y_4(t,s)]$ is proportional to the energy of V [due to the quantity $E[\max(V,0)|U]$ in (3)], and inversely proportional to the energy of U (due to function $g'(\cdot)$), where U and V are defined in (1) and (2).
2) Considering that the cochlear filters $h(t,s)$ are broad while the differential filters $(\partial/\partial s)h(t,s)$ are narrow and centered around the same frequencies, U can be viewed as a smoothed version of V.
3) Combining 1 and 2, the auditory spectrum is a self-normalized spectral profile. Specifically, a spectral peak receives a relatively small self-normalization factor (i.e., the energy of U is relatively small), whereas a spectral valley receives a relatively large self-normalization factor.
4) The above difference in the self-normalization further enlarges the ratio of spectral peak to valley, a phenomenon referred to as spectral enhancement or noise suppression.

2) Quantitative Analysis of a Special Case: It is desirable that the above qualitative analysis on the self-normalization property be verified by some results of a quantitative nature. However, due to the nonlinearity of the EA model, it is difficult to find a simple closed-form expression for the integral in (3).

In [17], a special case has been studied wherein the hair cell nonlinear sigmoid compression function $g(u)$ is replaced by a step function; in this case, $g'(u)$ becomes a delta function $\delta(u)$. Assuming the input signal $x(t)$ is a zero mean Gaussian process, (3) can be expressed in closed form as

$$E\left[y_4(t,s)\right] = \frac{\sigma_v}{2\pi\sigma_u}\sqrt{1-r^2} \qquad (4)$$

where r, σ_u and σ_v denote the correlation coefficient between U and V, the standard deviation of U, and the standard deviation of V, respectively. This expression demonstrates the self-normalization nature of the auditory spectrum as analyzed above, i.e., $E[y_4(t,s)]$ is proportional to the standard deviation[2] of V and inversely proportional to that of U.

III. Analysis of the Self-Normalization Property

Although a step function can be treated as a very special case of the sigmoid compression function $g(\cdot)$ in Fig. 1, it is desirable to obtain the closed-form expression of $E[y_4(t,s)]$ using a better, yet mathematically tractable, approximation. In particular, it is of interest to determine whether the resulting expression still supports the original analysis on self-normalization based on a step function. Having noticed the general nonlinear compression nature of the Gaussian cumulative distribution function (CDF), and the resemblance between the graph of the sigmoid function and that of the Gaussian CDF, below, we use Gaussian CDF as an approximation to the sigmoid compression function to derive a closed-form expression of $E[y_4(t,s)]$ and conduct relevant analysis.

[1] The dependence of U and V on indices (t,s) is dropped in the main text for notational simplicity.

[2] In [17], considering the one-to-one correspondence between the standard deviation σ and the variance σ^2, the former is referred to as energy.

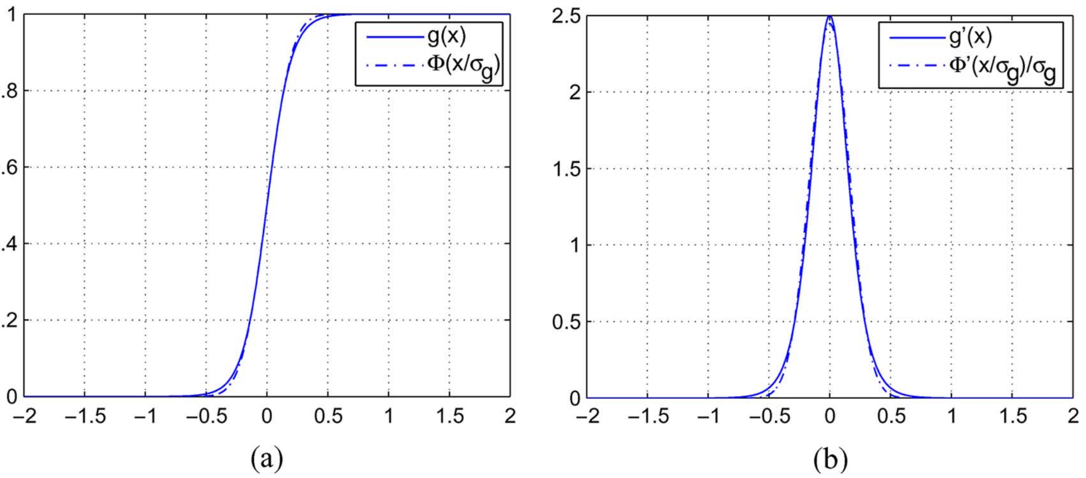

Fig. 2. Sigmoid function ($\alpha = 0.1$) and Gaussian distribution function ($\sigma_g = 0.163$). (a) $g(x)$ and $\Phi(x/\sigma_g)$. (b) $g'(x)$ and $(1/\sigma_g)\Phi'(x/\sigma_g)$.

A. Approximation to the Sigmoid Compression Function

Referring to Fig. 1, the sigmoid compression function at the hair cells stage takes the form of [20]

$$g(x) = \frac{1}{1 + e^{-x/\alpha}} \tag{5}$$

where the coefficient α is set to 0.1.

Fig. 2(a) shows the sigmoid function $g(x)$ with $\alpha = 0.1$. By inspecting (5) and Fig. 2(a), it is noted that $g(x)$ resembles the CDF of a Gaussian random variable with zero mean. In particular, with $\alpha = 0.1$ in (5), $g(x)$ is close to the CDF of a Gaussian variable with zero mean and standard deviation $\sigma_g = 0.163$, i.e., $\Phi(x/\sigma_g)$, where $\Phi(x)$ is the CDF of a standard normal random variable as defined as follows:

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-t^2/2} dt. \tag{6}$$

The function $\Phi(x/0.163)$ is also shown in Fig. 2(a). The derivatives of the function $g(x)$ and $\Phi(x/\sigma_g)$, respectively, $g'(x)$ and $(1/\sigma_g)\Phi'(x/\sigma_g)$, are shown in Fig. 2(b). The relative difference between the two curves in Fig. 2(b) over a practical range of values of U, as determined from experimental measurements, is of the order of 2% or less for the different processing channels.[3]

In the following analysis, based on the above considerations, $g'(x)$ with $\alpha = 0.1$ is approximated as

$$g'(x) \approx \frac{\Phi'(x/\sigma_g)}{\sigma_g} = \frac{1}{\sqrt{2\pi}\sigma_g} e^{-x^2/2\sigma_g^2} \tag{7}$$

where $\sigma_g = 0.163$.

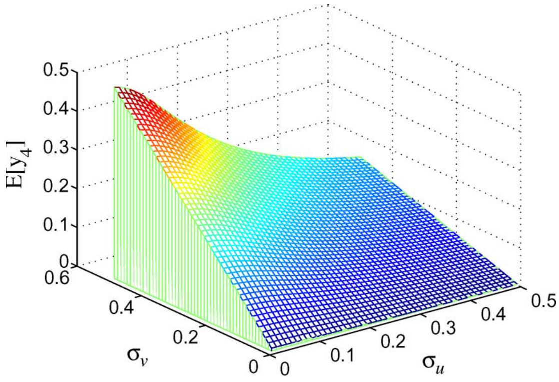

Fig. 3. $E[y_4(t,s)]$ as a function of σ_u and σ_v.

B. Closed-Form Expression of $E[y_4(t,s)]$

As in [17], assume the input signal $x(t)$ is a zero mean stationary Gaussian process. For given values of (t,s), $U(t,s)$ and $V(t,s)$ are obtained by linear filtering of $x(t)$ and are thus zero mean Gaussian random variables, i.e., $U(t,s) \sim \mathcal{N}(0,\sigma_u^2)$, and $V(t,s) \sim \mathcal{N}(0,\sigma_v^2)$, where $\sigma_u \equiv \sigma_u(s)$ and $\sigma_v \equiv \sigma_v(s)$ denote the standard deviations of U and V, respectively. According to [21] and [22], the conditional pdf of V given $U = u$, denoted $f_{V|U}(v|u)$, is also Gaussian with mean $\mu_{v|u} = ru\sigma_v/\sigma_u$ and variance $\sigma_{v|u}^2 = \sigma_v^2(1-r^2)$, where r represents the correlation coefficient between U and V.

With the assumptions made above, the result of (3) is (see Appendix I for details)

$$E\left[y_4(t,s)\right] = \frac{\sigma_v\sqrt{\sigma_g^2 + \sigma_u^2(1-r^2)}}{2\pi\left(\sigma_g^2 + \sigma_u^2\right)}. \tag{8}$$

From (8), it is noted that $E[y_4(t,s)]$ is a linear function of σ_v. Furthermore, given that σ_u, σ_v and σ_g are all positive values, and $|r| \leq 1$, it is found that $\partial E[y_4(t,s)]/\partial\sigma_v > 0$ and $\partial E[y_4(t,s)]/\partial\sigma_u < 0$, which means that $E[y_4(t,s)]$ is an increasing function of σ_v and a decreasing function of σ_u.

[3] There are 129 channels, corresponding to a set of 129 bandpass filters. See Section IV.

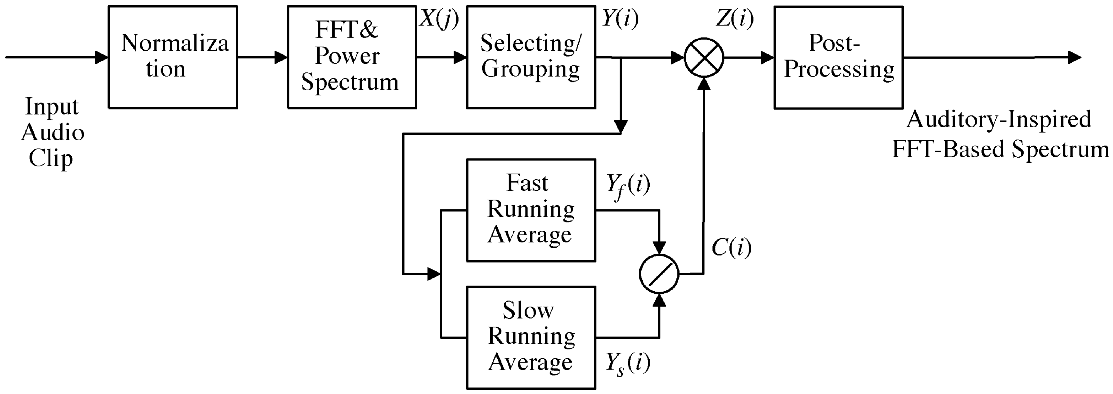

Fig. 4. Schematic description of the proposed FFT-based implementation.

Fig. 3 gives a three-dimensional view of $E[y_4(t,s)]$ as a function of σ_u and σ_v, where $\sigma_g = 0.163$ and r^2 is set to a fixed value of 0.1 to facilitate the analysis.[4]

The results given in (8) and Fig. 3 indicate that, $E[y_4(t,s)]$ is proportional to σ_v (or, the "energy" of V according to [17]) and inversely proportional to σ_u (or, the "energy" of U according to [17]). Therefore, using a Gaussian CDF to approximate the original sigmoid function, the derived results support the original analysis on self-normalization which is summarized in Section II-B1.

C. Local Spectral Enhancement

With respect to the conclusions on self-normalization summarized in Section II-B1, statement 4 refers to a desirable situation where spectral enhancement is achieved. It seems to be a natural result from statement 3, but it may not be necessarily the case.

To facilitate the following analysis on local spectral enhancement due to the self-normalization property, we assume that $E[y_4(t,s)] \propto V(t,s)/U(t,s)$ where V and U are treated as positive quantities. Suppose that $V(t,s_p)$ corresponds to a power spectral peak, and $U(t,s_p)$ is a smoothed version of $V(t,s_p)$. Similarly, $V(t,s_v)$ and $U(t,s_v)$ are assumed to be a power spectral valley and its smoothed version, respectively.

In statement 3, the word "relatively" indicates a comparison between the power spectrum component and the corresponding smoothed version, i.e., we have the following:

$$V(t,s_p)/U(t,s_p) > 1 \tag{9}$$

$$V(t,s_v)/U(t,s_v) < 1. \tag{10}$$

However, to have the ratio of spectral peak to valley enlarged, the following should be satisfied:

$$\frac{V(t,s_p)/U(t,s_p)}{V(t,s_v)/U(t,s_v)} > \frac{V(t,s_p)}{V(t,s_v)} \tag{11}$$

i.e., it is required that $U(t,s_p) < U(t,s_v)$, which may not be necessarily so. In the case with the above simplified assumptions, (9) and (10) do not necessarily ensure that we have (11). Thus, the statement 4 is not guaranteed, although it refers to a property that is desirable for noise suppression.

Although the enlargement of the ratio of spectral peak to valley is not guaranteed from the above analysis, conditions given in (9) and (10) do provide a basis for spectral enhancement. Given (9) and (10), a simple way to enlarge the ratio of spectral peak to valley is to multiply the spectral components $V(t,s_p)$ and $V(t,s_v)$ with the corresponding ratios given in (9) and (10), i.e.,

$$\frac{V(t,s_p)/U(t,s_p)}{V(t,s_v)/U(t,s_v)} \cdot \frac{V(t,s_p)}{V(t,s_v)} > \frac{V(t,s_p)}{V(t,s_v)}. \tag{12}$$

Next, we will propose a simple FFT-based system wherein the idea presented in (12) is implemented.

IV. New Auditory-Inspired FFT-Based Spectrum

The EA model [17] is characterized by a complicated computation procedure and the use of nonlinear processing. It would be desirable that the model be simplified, or approximated in the frequency domain where efficient FFT algorithms are available. In our earlier study [16], such a simplified implementation has been proposed to calculate a self-normalized FFT-based spectrum which is proved to be noise-robust in audio classification tests.

The FFT-based implementation we proposed in [16] employs a simple grouping scheme to reduce the dimension of the power spectrum vector. However, this scheme fails to give a clear interpretation of the meaning of the frequency index. In applications where frequency-dependent audio features need to be extracted (e.g., spectral centroid, bandwidth), it would be more appropriate, instead of the simple grouping scheme we have proposed, to group or select power spectrum components based on the original constant-Q bandpass filters $h(t,s)$ (see Section II).

In this paper, by making use of the characteristic frequency (CF) values of the bandpass filter set of the EA model [17], and by integrating the self-normalization property through a pair of running averages, we present a new implementation for the calculation of the FFT-based spectrum proposed in [16], as illustrated in Fig. 4. The details of the proposed implementation are presented next.

[4] According to our tests based on the implementation [20], the mean values of r^2 for the three audio classes (i.e., speech, music, and noise) in different noise environments are around 0.1.

TABLE I
FREQUENCY INDEX VALUES OF N_k AND ϕ_i

k	N_k	i	ϕ_i
1	8	1	8
2	9	-	-
3	9	2	9
4	9	-	-
5	9	-	-
6	10	-	-
7	10	3	10
8	10	-	-
9	11	-	-
10	11	4	11
11	11	-	-
⋮	⋮	⋮	⋮
142	491	119	491
143	506	120	506

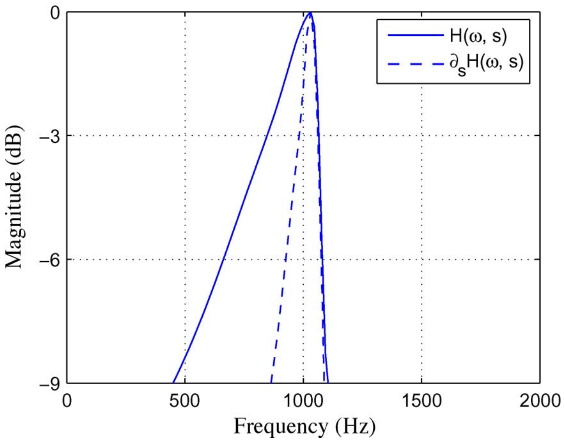

Fig. 5. Cochlear filter $H(\omega, s)$ centered at 1017 Hz and the corresponding differential filter $\partial_s H(\omega, s)$. (The 3-dB bandwidth of the cochlear filter is about 220 Hz, while the 3-dB bandwidth of the differential filter is 80 Hz.)

A. Normalization of the Input Signal

To make the algorithm adaptable to input signals with different energy levels, each input audio clip (with a length of 1 s) is normalized with respect to the square-root value of its average energy.

B. Calculation of a Short-Time Power Spectrum

Using the normalized audio signal, a short-time power spectrum is calculated through an M-point FFT algorithm. To determine an appropriate value for M, we have to trade performance against complexity.

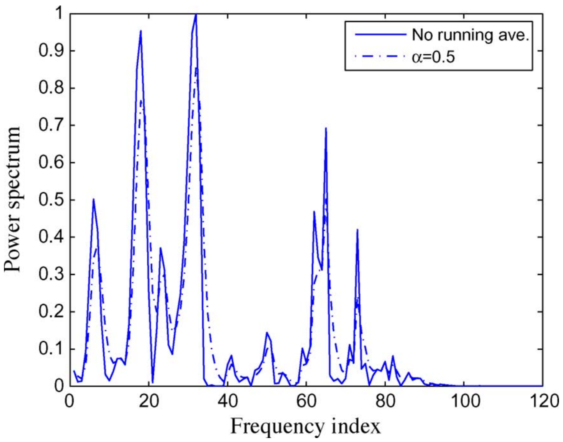

Fig. 6. Running average scheme.

The cochlear filters in the EA model are modeled as a set of constant-Q bandpass filters [17], [23]. In implementation [20], the 129 CF values of the corresponding constant-Q bandpass filters F_k are determined by[5]

$$F_k = 2^{l_k} F_0, \quad k = 1, 2, \ldots, 129 \tag{13}$$

where $F_0 = 440$ Hz, and $l_k = (k - 32)/24$.

According to (13), the CF values cover a range from 180 to 7246 Hz. The difference between two neighboring CF values is as low as about 5.27 Hz for $k = 1$ and 2. For a signal sampled at 16 kHz, which is used in this study, even with a 2048-point FFT, such a small frequency interval cannot be resolved. Meanwhile, since the CF values are logarithmically located, the frequency resolution achieved from a 2048-point or even higher order FFT algorithm is more than necessary for the high-frequency bands. In this paper, we use an $M = 1024$-point FFT to achieve a tradeoff between frequency resolution and computational complexity. The length of the analysis window is 30 ms and the overlap is 20 ms.

C. Power Spectrum Selection

To reduce the dimension of the obtained power spectrum vector, a simple selection scheme is proposed as follows. First, we extend the values of k in (13), i.e., from -10 to 132. Or equivalently, (13) is modified as[6]

$$F_k = 2^{\bar{l}_k} F_0, \quad k = 1, 2, \ldots, 143 \tag{14}$$

where $\bar{l}_k = (k - 43)/24$. For each F_k, the corresponding frequency index N_k is determined by

$$N_k = \text{int}\left(\frac{F_k M}{F_s}\right), \quad k = 1, 2, \ldots, 143 \tag{15}$$

where function $\text{int}(x)$ returns the nearest integer value of x, and F_s is the sampling frequency. After discarding the repeated N_k values and renumbering the remaining values, we obtain a set of

[5] Instead of 129, the actual size of the output auditory spectrum vector is 128 due to the derivative with respect to the channel (see Fig. 1).

[6] One purpose of extending the values of k is to include more low-frequency components for power spectrum selection. The second purpose is to make the size of the proposed FFT-based auditory spectrum vector [i.e., 120, see (16)] comparable to that of the original auditory spectrum vector (i.e., 128).

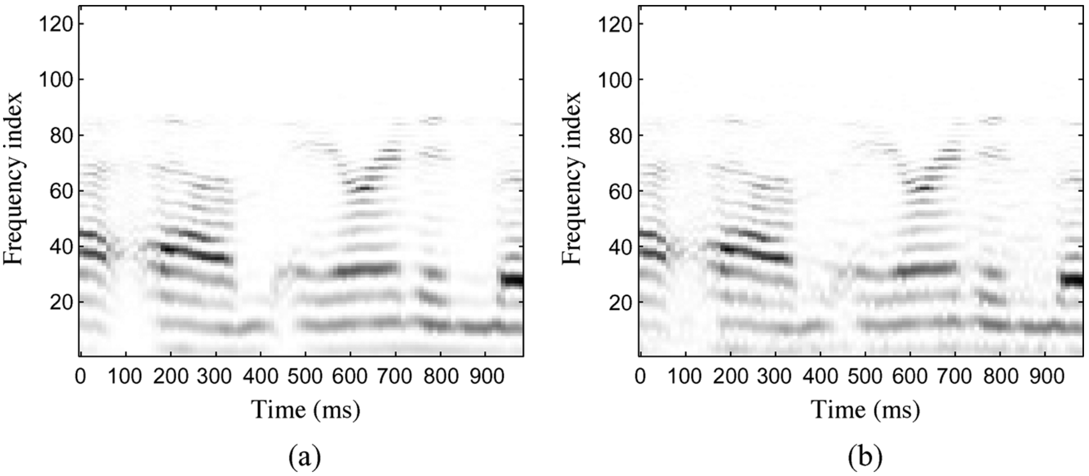

Fig. 7. Proposed FFT-based spectrograms of a 1-s speech clip. (a) Clean case. (b) SNR = 15 dB.

120 characteristic frequency index values $\phi_i, i = 1, 2, \ldots, 120$, as illustrated in Table I.

Using frequency index values ϕ_i, the power spectrum selection (see Fig. 4) is as follows:

$$Y(i) = X(\phi_i), \quad i = 1, 2, \ldots, 120. \tag{16}$$

Based on (16), a set of $M/2$, i.e., 512, power spectrum components is transformed into a 120-dimensional vector, with each frequency index value corresponding to a specific CF value of the original cochlear filters.

D. Spectral Self-Normalization

As discussed in Section III-C, the ratio of spectral peak to valley can be enlarged through the scheme given by (12). In [16], such a local self-normalization is implemented through the use of a pair of wide and narrow windows defined in the frequency domain. Below, we propose an improved implementation for self-normalization which is simpler and easier to use than the one in [16].

According to [17], the cochlear filters are broad and highly asymmetric, and the differential filters are narrowly tuned and centered around the same frequencies. Fig. 5 shows the magnitude responses of a cochlear filter $H(\omega, s)$, which is centered at 1017 Hz, and the corresponding differential filter $(\partial/\partial s)H(\omega, s)$ [20]. Based on the magnitude responses shown in Fig. 5, an iterative running average is defined over the frequency index i as follows:

$$Y_r(i) = (1-\alpha)Y_r(i-1) + \alpha Y(i) \tag{17}$$

where $0 \leq \alpha \leq 1$, and $Y(i)$ and $Y_r(i)$ are the input and averaged output, respectively. A relatively large α corresponds to a "fast" running average, while a relatively small α results in a "slow" running average. A slow and fast running averages are employed here to simulate a cochlear filter and a differential filter, respectively.

An example of the running average defined in (17) is illustrated in Fig. 6, which shows a power spectrum vector in relative values and its running averaged version with $\alpha = 0.5$. In general, for a spectral peak, the corresponding smoothed value is smaller, while for a spectral valley, the corresponding smoothed value is larger.

Let $Y_f(i)$ and $Y_s(i)$ represent the outputs from a fast and a slow running averages, respectively. $Y_s(i)$ may be viewed as a smoothed version of $Y_f(i)$. Based on $Y_f(i)$ and $Y_s(i)$, a self-normalization coefficient at frequency index i, $C(i)$, is defined as

$$C(i) = \frac{Y_f(i)}{Y_s(i)}, \quad i = 1, 2, \ldots, 120. \tag{18}$$

Therefore, in general $C(i)$ is larger than 1 for a spectral peak and smaller than 1 for a spectral valley, which coincides with the conditions given in (9) and (10).

To implement the self-normalization, the selected power spectrum at frequency index i, i.e., $Y(i)$, is multiplied by the corresponding self-normalization coefficient $C(i)$, generating a set of self-normalized power spectrum data $Z(i)$. By using different parameters for the two running averages, the effect of self-normalization varies, leading to variable classification performance (see Section VI-C).

E. Postprocessing

The square-root values of the self-normalized spectrum data $Z(i)$ are further calculated. Finally, the proposed auditory-inspired FFT-based spectrum is obtained by applying a smoothing operation on the square-root spectrum data. The smoothing can be implemented using a fast running average as defined in (17). For the sake of simplicity, the smoothing process is not considered in this paper. Fig. 7 gives an example of the proposed FFT-based spectrograms of a 1-s speech clip in a clean case and in a noisy case where SNR = 15 dB. From Fig. 7, we can see that the two spectrograms are fairly close to each other.

Compared to the self-normalization scheme we proposed in [16], the new implementation presented here is simpler and easier to use since it only involves two parameters to adjust, i.e., a fast and a slow running average coefficients. Besides, by making use of the CF values of the original bandpass filters, a relationship is created between the frequency index of the proposed FFT-based auditory spectrum vector and the physical frequency value. Therefore, the proposed FFT-based auditory spectrum allows more flexibility in the extraction of different audio features.

V. Features Extraction and Classification Tests

A. Audio Features

In this paper, six sets of frame-level audio features are calculated, specifically: conventional MFCC features with and without cepstral mean subtraction (CMS); MFCC-like features computed from the original auditory spectrum (i.e., the output of the EA model), from the FFT-based spectrum of [16], and from the proposed FFT-based auditory spectrum; as well as spectral features obtained from the FFT-based auditory spectrum. The corresponding clip-level features are the statistical mean and variance values of these frame-level features calculated over a 1-s time window. The clip-level features are used for the training and testing of the classification algorithm. The details of the frame-level features are given below.

1) Conventional MFCC Features: Being widely used in speech/speaker recognition, MFCCs [24] are also useful in audio classification. For the purpose of performance comparison, the conventional MFCCs are used in this paper. A Matlab toolbox developed by Slaney [25] is used to calculate a set of 13 conventional MFCCs.

As for the conventional MFCC features, a so-called CMS technique may improve the robustness of frame-level MFCCs by removing the time averages from the cepstrum data [26]. In this paper, we have used a 10-s window to calculate the time averages of the MFCCs data needed in the application of the CMS operation for frame-based MFCCs data.

2) MFCC-Like Features: These are obtained by applying the discrete cosine transform (DCT) to the original auditory spectrum, the FFT-based spectrum of [16], and the proposed FFT-based auditory spectrum. Specifically, a set of 13 coefficients is calculated as follows:[7]

$$F_n[l] = \begin{cases} \frac{1}{\sqrt{K}} \sum\limits_{k=0}^{K-1} A_n[k], & l = 0 \\ \sqrt{\frac{2}{K}} \sum\limits_{k=0}^{K-1} A_n[k] \cos \frac{l(2k+1)\pi}{2K}, & 1 \le l \le 12 \end{cases} \tag{19}$$

where $A_n[k]$ is the kth component of the magnitude spectrum vector (either the auditory spectrum vector or the FFT-based spectrum vector) for the nth frame signal, K is the size of the magnitude spectrum vector A_n, and $F_n[l]$ is the lth component of the corresponding MFCC-like feature vector.

3) Spectral Features: To show the flexibility of the proposed FFT-based auditory spectrum in the extraction of different audio features, a set of spectral features is calculated using the corresponding FFT-based auditory spectrum. These features include energy, spectral flux, spectral rolloff point, spectral centroid, and bandwidth.

Energy: The energy is a simple yet reliable feature for audio classification. In this paper, we calculate for each frame the total energy and the energy of three subbands covering frequency ranges of 0–1 kHz, 1–2 kHz, and 2–4 kHz, respectively.

Spectral flux: The spectral flux is a measure of spectral change which comes in different forms. The first-order spectral flux is defined as the 2-norm of the frame-to-frame magnitude spectrum difference vector [2], [27]

$$SF1_n = \sqrt{\sum_{k=1}^{K} (A_{n+1}[k] - A_n[k])^2}. \tag{20}$$

The second-order spectral flux $SF2_n$ is calculated similarly as follows:

$$SF2_n = \sqrt{\sum_{k=1}^{K} (\Delta A_{n+1}[k] - \Delta A_n[k])^2} \tag{21}$$

where $\Delta A_n[k] = A_{n+1}[k] - A_n[k]$.

Spectral rolloff point: Scheirer and Slaney defined the spectral rolloff point as the 95th percentile of the power spectrum distribution [2]. It is a measure of the skewness of the spectral shape. In this paper, two spectral rolloff points are calculated which correspond to the 50th and 90th percentiles of the power spectrum distribution, respectively.

Spectral centroid: As a measure of the centroid of the magnitude spectrum, the spectral centroid, or brightness, can be defined as [5], [28]

$$SC_n = \left(\sum_{k=1}^{K} k A_n[k] \right) \Bigg/ \sum_{k=1}^{K} A_n[k] \tag{22}$$

where SC_n denotes the spectral centroid.

Bandwidth: Here, the bandwidth is obtained as the magnitude-weighted average of the differences between the frequency indices and the centroid [5], [28]. The bandwidth can be expressed as follows:

$$BW_n = \sqrt{\left(\sum_{k=1}^{K} (k - SC_n)^2 A_n[k] \right) \Bigg/ \sum_{k=1}^{K} A_n[k]} \tag{23}$$

where BW_n denotes the bandwidth, and SC_n is the spectral centroid as defined in (22).

In the calculation of spectral rolloff points, spectral centroid, and bandwidth, instead of using the frequency indices i in Table I, the corresponding physical frequency values are used. Finally, all these features are grouped together to form a ten-dimensional spectral feature vector for audio classification application.

B. Setup of the Classification Test

1) Audio Sample Database: To carry out audio classification test, two generic audio databases are built which includes speech, music, and noise clips. The sampling rate of the first audio sample database is 16 kHz. This database is created for the performance comparison of all audio features introduced above. The detailed information of the 16-kHz database is as follows.

- Speech: Speech clips are captured from several English web radio stations. These samples are spoken by different male and female speakers and at different speaking rates. These clips are treated as clean speech samples.

[7] According to our tests, a better classification performance is achieved in noisy environments without the use of logarithmic operation, which is employed in the calculation of conventional MFCCs.

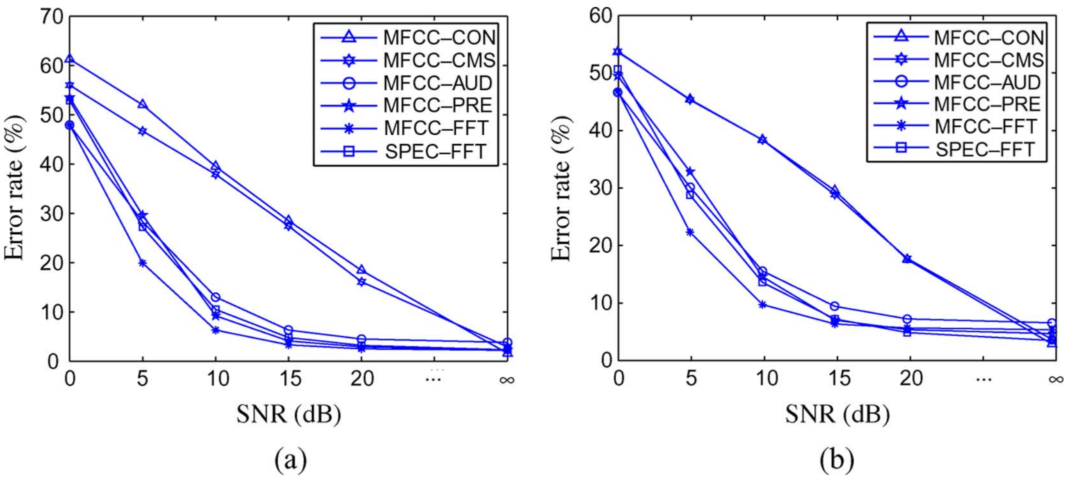

Fig. 8. Speech/music/noise classification error rates of different audio features. (a) SVM. (b) C4.5.

- Music: Music clips include five common types, namely, blues, classical, country, jazz, and rock. The music clips also contain segments that are played by some Chinese traditional instruments (either alone or together with some other instruments). These music samples include both instrumental music and vocal music with instrumental accompaniment. These clips are treated as clean music samples.
- Noise: Noise samples are selected from the NOISEX database which contains recordings of various noises [29]. A total of 15 different noise samples are used, including speech babble, factory floor noises recorded in two places, buccaneer noises recorded at two traveling speeds, destroyer engine room noise, destroyer operation room noise, F16 cockpit noise, noises of two military vehicles, machine gun noise, vehicle interior noise, noise from high-frequency (HF) radio channels, pink noise, and white noise.

The total length of all the audio samples is 200 min, including 70-min speech, 76-min music, and 54-min noise. These samples are divided equally into two parts for training and testing, respectively. Three-class (speech, music, and noise) classification tests are conducted using this database to compare the performance of different audio features. The audio classification decision is made on a 1-s basis.

The second database is created with 8-kHz sampling frequency and used to further evaluate the performance of the MFCC-like features calculated using the proposed FFT-based auditory spectrum, as compared to the conventional MFCC features, in a narrow-band case. The 50-min speech samples and 42-min music samples are selected from the first database and resampled at 8-kHz. The 48-min noise samples are selected from a database provided by [30]. These noise samples are recorded in four different environments, i.e., a moving car with different speeds and with windows up and down, parking garage, urban street and shopping mall, and commuter train. Noise and nonnoise (i.e., speech plus music) classification tests are conducted using this database. The audio classification decision is made using both 1- and 5-s clip lengths.

In the following, a clean test refers to a test wherein both the training set and testing set contain clean speech, clean music, and noise. A test with a specific SNR value refers to a test wherein the training set contains clean speech, clean music, and noise while the testing set contains noisy speech and noisy music[8] (both with that specific SNR value), and noise.

2) Implementation: We use a Matlab toolbox developed by the Neural Systems Laboratory, University of Maryland [20], to calculate the original auditory spectrum. Relevant modifications are introduced to this toolbox to meet the needs of our study.

As for the classification, in this paper, we use a support vector machine algorithm SVM$^{\text{struct}}$ [31] and a decision tree learning algorithm C4.5 [18] as two classifiers. The support vector machine, which is a statistical machine learning technique often used in pattern recognition, has been recently applied to the audio classification task [14], [32]. A SVM first transforms input vectors into a high-dimensional feature space using a linear or nonlinear transformation, and then conducts a linear separation in feature space. In this paper, we use radial basis function (RBF) as the kernel function, and the model is tuned to achieve the best training performance.

C4.5 is a widely used decision tree learning algorithm. Its classification rules are in the form of a decision tree, which is generated by recursively partitioning the training data into smaller subsets based on the value of a selected attribute.

VI. Performance Analysis

A. Performance Comparison for Different Feature Sets With 16-kHz Database

Using a 16-kHz audio database and with SVM and C4.5 as the classifiers, error rates of speech/music/noise classification for different audio features are shown in Fig. 8, where MFCC-CON, MFCC-CMS, MFCC-AUD, MFCC-PRE, MFCC-FFT, and SPEC-FFT represent the conventional MFCC features, conventional MFCC features with CMS operation,

[8] These noisy samples are generated by adding noise segments which are randomly selected from the noise database to clean speech/music segments based on long-term average energy measurement.

TABLE II
SPEECH/MUSIC/NOISE CLASSIFICATION ERROR RATES OF DIFFERENT AUDIO FEATURES (%)

		MFCC-CON	MFCC-CMS	MFCC-AUD	MFCC-PRE	MFCC-FFT	SPEC-FFT
SVM	Clean	1.6	2.9	3.8	2.3	2.2	2.3
	Average-Noisy	39.9	36.8	20.0	19.9	16.0	19.7
	Average-Overall	20.8	19.9	11.9	11.1	9.1	11.0
C4.5	Clean	2.7	3.5	6.4	4.5	5.2	3.3
	Average-Noisy	37.0	36.9	21.8	21.8	18.1	21.0
	Average-Overall	19.9	20.2	14.1	13.2	11.7	12.2

MFCC-like features computed from the original auditory spectrum, MFCC-like features computed from the FFT-based spectrum of [16], MFCC-like features obtained from the proposed FFT-based auditory spectrum, and spectral features obtained from the FFT-based auditory spectrum, respectively. To calculate the proposed FFT-based auditory spectrum, the fast and slow running average coefficients are set to 1 and 0.5, respectively.

To compare the classification performance in noisy test cases, an average error rate is calculated based on the results of five noisy test cases (i.e., SNR $=$ 20, 15, 10, 5 and 0 dB). Table II gives error classification rates for different audio features, where "Clean," "Average-Noisy," and "Average-Overall" denote the error rate in the clean test case, the average error rate in noisy test cases, and the overall average over these two cases.

Test results presented in Fig. 8 and Table II indicate that, although the conventional MFCC features provide an excellent performance in the clean case, its performance degrades rapidly as the SNR decreases, leading to a relatively poor overall performance. On the other hand, the MFCC-like features obtained from the original auditory spectrum, FFT-based spectrum of [16] and the proposed FFT-based auditory spectrum, as well as the spectral features derived from the FFT-based auditory spectrum are more robust in noisy test cases, especially when SNR is between 5 and 20 dB.

Test results given in Table II also indicate that the MFCC-like features computed from the proposed FFT-based auditory spectrum slightly outperform those computed from the FFT-based spectrum of [16] and those computed from the original auditory spectrum.

With SVM as the classifier, the use of MFCC-CMS features resulted in a small improvement in noisy cases as compared to the conventional MFCC features. However, the slight improvements are obtained at the price of performance loss in clean test which may be a problem in some applications. Indeed, in this paper, based on the frame-level conventional MFCCs, statistical mean and variance values are further calculated over a 1-s time window. The resulting mean and variance values are grouped together to form the corresponding clip-level features which are used for the training and testing of the classification algorithm. Hence, CMS operation has been already implicitly implemented in a different way in the proposed clip-level MFCC features, and thus the use of CMS may not significantly improve the robustness of MFCC features as observed in our experiments.

As for the two classification approaches, results from Table II indicate that in most cases SVM outperforms C4.5.

TABLE III
CONFUSION MATRICES FOR DIFFERENT AUDIO FEATURE SETS AT SNR $=$ 10 dB.
(a) MFCC-CON (CCR $=$ 60.5%). (b) MFCC-CMS (CCR $=$ 62.1%).
(c) MFCC-AUD (CCR $=$ 87.0%). (d) MFCC-PRE (CCR $=$ 90.8%).
(e) MFCC-FFT (CCR $=$ 93.7%). (f) SPEC-FFT (CCR $=$ 89.5%)

(a)

Output Input	Music	Noise	Speech
Music	1690	590	0
Noise	38	1582	0
Speech	1673	71	356

(b)

Output Input	Music	Noise	Speech
Music	1752	526	2
Noise	39	1581	0
Speech	1452	256	392

(c)

Output Input	Music	Noise	Speech
Music	2067	197	16
Noise	33	1587	0
Speech	528	5	1567

(d)

Output Input	Music	Noise	Speech
Music	1981	276	23
Noise	15	1605	0
Speech	152	88	1860

(e)

Output Input	Music	Noise	Speech
Music	2033	206	41
Noise	4	1616	0
Speech	95	32	1973

(f)

Output Input	Music	Noise	Speech
Music	1979	265	36
Noise	12	1608	0
Speech	305	10	1785

Table III presents confusion matrices for a noisy test case with 10-dB SNR and with SVM as the classifier where "Input" and

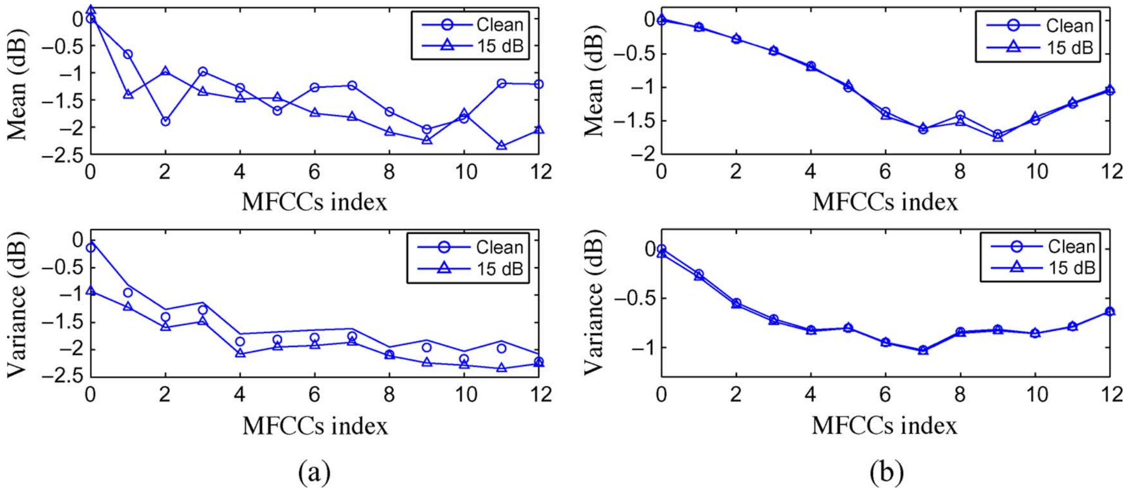

Fig. 9. MFCC/MFCC-like features for a 1-s speech clip in clean case and in noisy case with 15-dB SNR. (a) Conventional MFCC features. (b) MFCC-like features computed from the proposed FFT-based auditory spectrum.

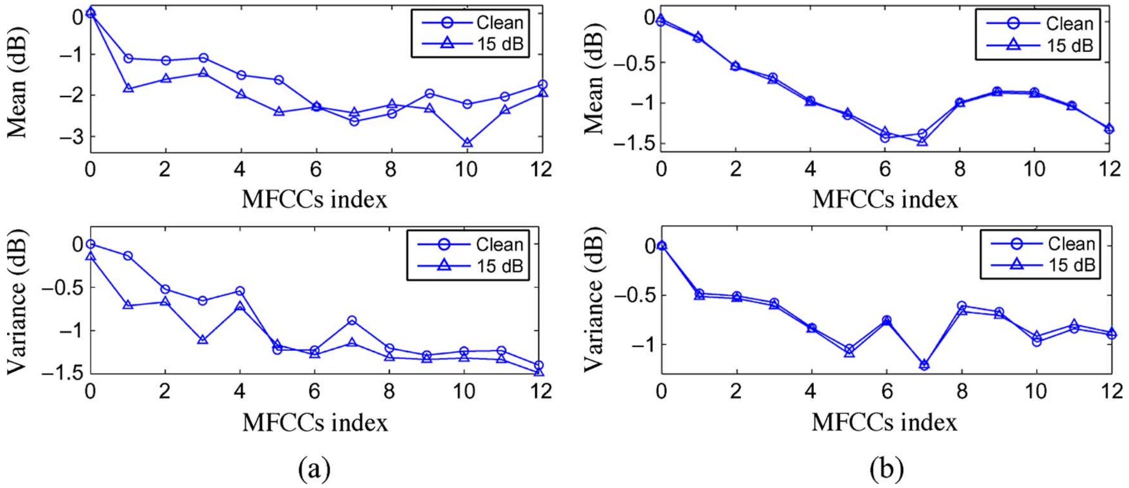

Fig. 10. MFCC/MFCC-like features for a 1-s music clip in clean case and in noisy case with 15-dB SNR. (a) Conventional MFCC features. (b) MFCC-like features computed from the proposed FFT-based auditory spectrum.

"Output" represent the input audio types and the output classification decisions, respectively. Shown in Table III are the numbers of decisions on a 1-s basis. The correct classification rate (CCR) for each feature set is given in the table caption. It is seen that features computed from the original auditory spectrum, the FFT-based spectrum of [16], and the proposed FFT-based auditory spectrum generally lead to a better classification performance of the three audio categories. The low overall classification rates of the conventional MFCC features and MFCC-CMS features [60.5% from Table III(a) and 62.1% from Table III(b), respectively] are due in a large part to the low proportion of speech samples correctly identified. In contrast, for the MFCC-like features computed from the proposed FFT-based auditory spectrum [Table III(e)], the proportion of correctly identified samples is high for all three classes, i.e., speech, music, and noise.

Two examples of clip-level MFCC/MFCC-like features, i.e., absolute mean and variance values over a 1-s window, are given in Figs. 9 and 10 (in relative values). Fig. 9 shows the conventional MFCC features and the MFCC-like features computed from the proposed FFT-based auditory spectrum for a 1-s speech clip in clean test case and in noisy test case with 15-dB SNR. At SNR = 15 dB, the MFCC-like features computed from the proposed FFT-based auditory spectrum are close to that in the clean test case. However, this is not so for the conventional MFCC features wherein the change is relatively large. A similar situation can be found in Fig. 10 which shows results for a 1-s music clip. The results shown in Figs. 9 and 10 demonstrate the noise robustness of the proposed FFT-based auditory spectrum.

B. *Experiments Using 8-kHz Audio Database*

To further evaluate the performance of the proposed FFT-based auditory spectrum in a narrow-band application where the main focus is on the identification of noise, noise/nonnoise classification tests are conducted using 8-kHz audio database and with SVM as the classifier wherein nonnoise samples include speech and music clips. Error classification rates of the conventional MFCC features and the MFCC-like features derived from the proposed FFT-based auditory spectrum are listed

TABLE IV
NOISE/NONNOISE CLASSIFICATION ERROR RATES WITH SVM AS THE CLASSIFIER (%)

SNR (dB)	MFCC-CON		MFCC-FFT	
	1 sec	5 sec	1 sec	5 sec
∞	0.1	0.0	0.1	0.0
15	4.1	3.8	0.9	0.4
10	12.8	13.0	2.6	1.3
5	30.0	30.0	7.3	6.3

TABLE V
ERROR CLASSIFICATION RATES OF MFCC-LIKE FEATURES COMPUTED FROM THE PROPOSED FFT-BASED AUDITORY SPECTRUM WITH DIFFERENT RUNNING AVERAGE COEFFICIENTS (%)

SNR (dB)	$\alpha = 0.5$	$\alpha = 0.1$	$\alpha = 0.05$
∞	2.2	2.7	3.2
10	6.3	5.7	5.6
0	48.0	40.2	37.1

in Table IV, where the decisions are made using both 1-s and 5-s clip lengths. As for the calculation of the proposed FFT-based auditory spectrum, a 512-point FFT is now used for 8-kHz samples. Hence, the outputs from (15) are same as those with 16-kHz sampling frequency and using a 1024-point FFT. Therefore, power spectrum selection can be conducted using Table I as before except that we now only consider frequency components within 0–4 kHz range instead of 0–8 kHz. Accordingly, the dimension of the proposed self-normalized FFT-based auditory spectrum vector is now 96 as compared to 120 in case of 16-kHz sampling frequency.

Results in Table IV shows the ability of the two sets of features in discriminating noise from nonnoise samples. These results also confirm the noise robustness of the proposed FFT spectrum-based MFCC-like features as compared to the conventional MFCC features. Meanwhile, as the length of audio clip increases from one second to five seconds, the proposed FFT spectrum-based MFCC-like features achieve a relatively large improvement in performance as compared to the conventional MFCC features.

C. Effect of Running Average Coefficients

As mentioned in Section IV, the proposed running average scheme is easier to use than the implementation in [16] since there are only two parameters to adjust. To see how different running average coefficients in (17) affect the performance of the proposed FFT-based auditory spectrum, we carry out tests using different coefficients wherein the fast running average coefficients are simply set to 1, and the slow running average coefficients are set to 0.5, 0.1, and 0.05, respectively. Using MFCC-like features and SVM algorithm, test results from speech/music/noise classification using 16-kHz database are listed in Table V. Results in Table V indicate that, as the slow running average coefficient increases, the corresponding performance in clean test case is improved, while the performance in noisy test cases (with 10- or 0-dB SNR) degrades. Here, the use of a relatively small coefficient for the slow running average leads to a relatively large increase in the ratio of spectral peak to valley, which on the one hand improves the robustness, but on the other hand may reduce the interclass difference to some extent, and thus degrades the performance in the clean test.

D. Computational Complexity

Besides the robustness to noise, an additional advantage of the proposed auditory-inspired FFT-based spectrum lies in its low computational complexity. An estimation of the computational load for the original auditory spectrum and the proposed FFT-based auditory spectrum is obtained by measuring the corresponding running time.

The implementation platform is a general PC with CPU Intel P4 (3.2 GHz). The EA model and the proposed FFT-based auditory spectrum are implemented using Matlab. Results are obtained using the 16-kHz database. Corresponding to a 1-s audio input clip, the time used for the calculation of the original auditory spectrum and that of the proposed FFT-based auditory spectrum are around 1.07 and 0.08 s, respectively. Instead of the actual processing time, the comparative performance may make more sense in this case, i.e., compared to the original auditory spectrum, the reduction in the processing time of the proposed FFT-based auditory spectrum is more than a factor of 10.

VII. CONCLUSION

In this paper, a stochastic analysis on the noise-suppression property of an EA model [17] has been presented. We have derived a closed-form expression for the auditory spectrum by using Gaussian CDF as an approximation to the original sigmoid compression function. Inspired by the EA model, we have presented an improved implementation for the calculation of an FFT-based auditory spectrum which allows flexibility in the extraction of noise-robust audio features. To evaluate the performance of the proposed FFT-based auditory spectrum, a speech/music/noise classification task was conducted wherein SVM$^{\mathrm{struct}}$ and C4.5 algorithms are used as the classifiers. Compared to the conventional MFCC features, the MFCC-like features computed from the original auditory spectrum, and both the MFCC-like and spectral features computed from the proposed FFT-based auditory spectrum show more robust performance in noisy test cases. Test results also indicate that, using the new MFCC-like features, the performance of the proposed FFT-based auditory spectrum is slightly better than that of the original auditory spectrum, while the computational complexity is reduced by an order of magnitude. The robustness of the MFCC-like features computed from the proposed FFT-based auditory spectrum was further confirmed by test results of noise/nonnoise classification experiments.

APPENDIX I
CLOSED-FORM EXPRESSION OF $E[y_4(t,s)]$

Assume random variables U and V are jointly normal with zero mean and standard deviation σ_u and σ_v, respectively. Accordingly, the conditional distribution function of V given $U = u$, $f_{V|U}(v|u)$, is also normal with mean $\mu_{v|u} = ru\sigma_v/\sigma_u$ and variance $\sigma_{v|u}^2 = \sigma_v^2(1-r^2)$, where r represents the correlation coefficient between U and V [21], [22].

To facilitate the analysis, we first define the following quantities:

$$\beta = \frac{\sigma_v}{\sigma_u} r \tag{24}$$

$$\sigma'_{v|u} = \frac{\sigma_{v|u}}{\beta}. \tag{25}$$

With the above assumptions about the distributions of U and V, and the conditional distribution of V given $U = u$, $E[\max(V,0)|U = u]$ in (3) is calculated as follows:

$$\begin{aligned} E\left[\max(V,0)|U=u\right] &= \int_0^\infty v \frac{1}{\sqrt{2\pi}\sigma_{v|u}} e^{-\frac{(v-\beta u)^2}{2\sigma^2_{v|u}}} dv \\ &= \frac{\sigma_{v|u}}{\sqrt{2\pi}} e^{-\frac{-u^2}{2\sigma'^2_{v|u}}} + \beta u \Phi\left(\frac{u}{\sigma'_{v|u}}\right). \end{aligned} \tag{26}$$

Therefore, (3) can be rewritten as

$$\begin{aligned} E\left[y_4(t,s)\right] = &\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma_g} e^{-\frac{u^2}{2\sigma_g^2}} \\ &\times \left[\frac{\sigma_{v|u}}{\sqrt{2\pi}} e^{-\frac{u^2}{2\sigma'^2_{v|u}}} + \beta u \Phi\left(\frac{u}{\sigma'_{v|u}}\right)\right] \\ &\times \frac{1}{\sqrt{2\pi}\sigma_u} e^{-\frac{u^2}{2\sigma_u^2}} du \\ = &C + D \end{aligned} \tag{27}$$

where C and D are calculated as follows:

$$\begin{aligned} C &\equiv \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma_g} e^{-\frac{u^2}{2\sigma_g^2}} \frac{\sigma_{v|u}}{\sqrt{2\pi}} e^{-\frac{u^2}{2\sigma'^2_{v|u}}} \frac{1}{\sqrt{2\pi}\sigma_u} e^{-\frac{u^2}{2\sigma_u^2}} du \\ &= \frac{\sigma_{v|u}}{(\sqrt{2\pi})^3 \sigma_g \sigma_u} \int_{-\infty}^{\infty} e^{-\left(\frac{u^2}{2\sigma_g^2} + \frac{u^2}{2\sigma'^2_{v|u}} + \frac{u^2}{2\sigma_u^2}\right)} du. \end{aligned} \tag{28}$$

Define

$$\frac{1}{\sigma_1^2} = \frac{1}{\sigma_g^2} + \frac{1}{\sigma'^2_{v|u}} + \frac{1}{\sigma_u^2}. \tag{29}$$

Then

$$C = \frac{\sigma_1 \sigma_{v|u}}{2\pi\sigma_g\sigma_u}. \tag{30}$$

As for D, we have

$$\begin{aligned} D &\equiv \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma_g} e^{-\frac{u^2}{2\sigma_g^2}} \beta u \Phi\left(\frac{u}{\sigma'_{v|u}}\right) \frac{1}{\sqrt{2\pi}\sigma_u} e^{-\frac{u^2}{2\sigma_u^2}} du \\ &= \frac{\beta}{2\pi\sigma_g\sigma_u} \int_{-\infty}^{\infty} u \Phi\left(\frac{u}{\sigma'_{v|u}}\right) e^{-\left(\frac{u^2}{2\sigma_g^2} + \frac{u^2}{2\sigma_u^2}\right)} du. \end{aligned} \tag{31}$$

Define

$$\frac{1}{\sigma_2^2} = \frac{1}{\sigma_g^2} + \frac{1}{\sigma_u^2}. \tag{32}$$

By using partial integration, we have the following result for D:

$$D = \frac{\beta\sigma_1\sigma_2^2}{2\pi\sigma_g\sigma_u\sigma'_{v|u}}. \tag{33}$$

Therefore, the closed-form expression of (3) is

$$\begin{aligned} E\left[y_4(t,s)\right] &= \frac{\sigma_1\sigma_{v|u}}{2\pi\sigma_g\sigma_u} + \frac{\beta\sigma_1\sigma_2^2}{2\pi\sigma_g\sigma_u\sigma'_{v|u}} \\ &= \frac{\sigma_v\sqrt{\sigma_g^2 + \sigma_u^2(1-r^2)}}{2\pi\left(\sigma_g^2 + \sigma_u^2\right)}. \end{aligned} \tag{34}$$

REFERENCES

[1] J. Saunders, "Real-time discrimination of broadcast speech/music," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 1996, vol. 2, pp. 993–996.

[2] E. Scheirer and M. Slaney, "Construction and evaluation of a robust multifeature speech/music discriminator," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Apr. 1997, vol. 2, pp. 1331–1334.

[3] R.-Y. Qiao, "Mixed wideband speech and music coding using a speech/music discriminator," in *Proc. IEEE Region 10 Annu. Conf. Speech Image Technol. for Comput. Telecomm.*, Dec. 1997, vol. 2, pp. 605–608.

[4] L. Tancerel, S. Ragot, V. T. Ruoppila, and R. Lefebvre, "Combined speech and audio coding by discrimination," in *Proc. IEEE Workshop Speech Coding*, Sep. 2000, pp. 154–156.

[5] E. Wold, T. Blum, D. Keislar, and J. Wheaten, "Content-based classification, search, and retrieval of audio," *IEEE Multimedia*, vol. 3, no. 3, pp. 27–36, Fall, 1996.

[6] T. Zhang and C.-C. J. Kuo, "Hierarchical classification of audio data for archiving and retrieving," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Mar. 1999, vol. 6, pp. 3001–3004.

[7] L. Lu, H.-J. Zhang, and H. Jiang, "Content analysis for audio classification and segmentation," *IEEE Trans. Speech Audio Process.*, vol. 10, no. 7, pp. 504–516, Oct. 2002.

[8] R. Cai, L. Lu, H.-J. Zhang, and L.-H. Cai, "Highlight sound effects detection in audio stream," in *Proc. Int. Conf. Multimedia Expo*, Jul. 2003, vol. 3, pp. 37–40.

[9] M. Zhao, J. Bu, and C. Chen, "Audio and video combined for home video abstraction," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Apr. 2003, vol. 5, pp. 620–623.

[10] T. Zhang and C.-C. J. Kuo, "Audio content analysis for online audiovisual data segmentation and classification," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 4, pp. 441–457, May 2001.

[11] G. Lu and T. Hankinson, "An investigation of automatic audio classification and segmentation," in *Proc. IEEE Int. Conf. Signal Process*, Aug. 2000, vol. 2, pp. 776–781.

[12] S. Kiranyaz, A. F. Qureshi, and M. Gabbouj, "A generic audio classification and segmentation approach for multimedia indexing and retrieval," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 3, pp. 1062–1081, May 2006.

[13] S. Ravindran and D. Anderson, "Low-power audio classification for ubiquitous sensor networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 2004, vol. 4, pp. 337–340.

[14] N. Mesgarani, S. Shamma, and M. Slaney, "Speech discrimination based on multiscale spectro–temporal modulations," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 2004, vol. 1, pp. 601–604.

[15] W. Chu and B. Champagne, "A simplified early auditory model with application in speech/music classification," in *Proc. IEEE Can. Conf. Elect. Comput. Eng.*, May 2006, pp. 578–581.

[16] W. Chu and B. Champagne, "A noise-robust FFT-based spectrum for audio classification," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 2006, vol. 5, pp. 213–216.

[17] K. Wang and S. Shamma, "Self-normalization and noise-robustness in early auditory representations," *IEEE Trans. Speech Audio Process.*, vol. 2, no. 3, pp. 421–435, Jul. 1994.

[18] J. R. Quinlan, *C4.5:Programs for Machine Learning*. San Mateo, CA: Morgan Kaufmann, 1993.

[19] M. Elhilali, T. Chi, and S. Shamma, "A spectro–temporal modulation index (STMI) for assessment of speech intelligibility," *Speech Commun.*, vol. 41, pp. 331–348, Oct. 2003.

[20] Neural Syst. Lab., Univ. Maryland, “NSL Matlab Toolbox.” [Online]. Available: http://www.isr.umd.edu/Labs/NSL/nsl.html.

[21] A. Papoulis and S. U. Pillai, *Probability, Random Variables, and Stochastic Processes*, 4th ed. New York: McGraw-Hill, 2002.

[22] P. L. Meyer, *Introductory Probability and Statistical Applications*, 2nd ed. Reading, MA: Addison-Wesley, 1970.

[23] P.-W. Ru, “Perception-based multi-resolution auditory processing of acoustic signals,” Ph.D. dissertation, Univ. Maryland, College Park, 2000.

[24] D. O’Shaughnessy, *Speech Communications-Human and Machines*, 2nd ed. New York: IEEE Press, 2000.

[25] M. Slaney, “Auditory Toolbox: A Matlab Toolbox for Auditory Modeling Work (Version 2),” Interval Research Corp., Tech. Rep. 1998-010, 1998 [Online]. Available: http://www.slaney.org/malcolm/pubs.html.

[26] B. S. Atal, “Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification,” *J. Acoust. Soc. Amer.*, vol. 55, no. 6, pp. 1304–1312, Jun. 1974.

[27] C. Xu, N. C. Maddage, and X. Shao, “Automatic music classification and summarization,” *IEEE Trans. Speech Audio Process.*, vol. 13, no. 3, pp. 441–450, May 2005.

[28] S. Z. Li, “Content-based audio classification and retrieval using the nearest feature line method,” *IEEE Trans. Speech Audio Process.*, vol. 8, no. 5, pp. 619–625, Sep. 2000.

[29] A. Varga, H. J. M. Steeneken, M. Tomlinson, and D. Jones, “The NOISEX-92 study on the effect of additive noise on automatic speech recognition,” 1992, documentation included in the NOISEX-92 CD-ROMs.

[30] *TDMA Cellular/PCS-Radio Interface-Minimum Performance Standards for Discontinuous Transmission Operation of Mobile Stations*, TIA/EIA/IS-727, Jun. 1998.

[31] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun, “Large margin methods for structured and interdependent output variables,” *J. Mach. Learn. Res.* vol. 6, pp. 1453–1484, Sep. 2005 [Online]. Available: http://svmlight.joachims.org/svm_struct.html.

[32] Y. Li and C. Dorai, “SVM-based audio classification for instructional video analysis,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 2004, vol. 5, pp. 897–900.

Wei Chu received the B.E. and M.E. degrees from Hefei University of Technology, Hefei, China, and the M.A.Sc. degree from Concordia University, Montréal, QC, Canada. He is currently pursuing the Ph.D. degree in the Department of Electrical and Computer Engineering, McGill University, Montréal.

His research interests are in the area of speech and audio signal processing, including content-based audio analysis, noise suppression, speech detection, speech/audio compression, and real-time DSP implementation of speech/audio algorithms.

Benoît Champagne was born in Joliette, QC, Canada, in 1961. He received the B.Ing. degree in engineering physics from École Polytechnique, Montréal, QC, Canada, in 1983, the M.Sc. degree in physics from Université de Montréal in 1985, and the Ph.D. degree in electrical engineering from the University of Toronto, Toronto, ON, Canada, in 1990.

From 1990 to 1999, he was an Assistant and then Associate Professor at INRS-Télécommunications, Université du Québec, Montréal, where he is currently a Visiting Professor. In September 1999, he joined McGill University, Montréal, as an Associate Professor within the Department of Electrical and Computer Engineering, where he is currently acting as Associate Chairman of Graduate Studies. His research interests lie in the area of statistical signal processing, including signal/parameter estimation, sensor array processing, and adaptive filtering and applications thereof to communications systems.