Discriminative Auditory-based Features for Robust Speech Recognition

Brian Mak, Yik-Cheung Tam, and Qi Li

Abstract

Recently, a new auditory-based feature extraction algorithm for robust speech recognition in noisy environments was proposed [1], [2]. The new features are derived by mimicking closely the human peripheral auditory process and the filters in the outer ear, middle ear, and inner ear are obtained from psychoacoustics literature with some manual adjustments. In this paper, we extend the auditory-based feature extraction algorithm and propose to further train the auditory-based filters through discriminative training. Using the data-driven approach, we optimize the filters by minimizing the subsequent recognition errors on a task. One significant contribution over similar efforts in the past (generally under the name of "discriminative feature extraction") is that we make no assumption on the parametric form of the auditory-based filters. Instead, we only require the filters to be triangular-like: the filter weights have a maximum value in the middle and then monotonically decrease to both ends. Discriminative training of these constrained auditory-based filters leads to improved performance. Furthermore, we study the combined discriminative training procedure for both feature and acoustic model parameters. Our experiments show that the best performance can be obtained in a sequential procedure under the unified framework of MCE/GPD.

Keywords

auditory-based filter, discriminative feature extraction, minimum classification error, generalized probabilistic descent

EDICS Category: 1-RECO

Corresponding Author: Dr. Brian Kan-Wing Mak.

Dr. Brian Mak is with the Department of Computer Science, the Hong Kong University of Science and Technology (HKUST), Clear Water Bay, Hong Kong. E-mail: mak@cs.ust.hk.

Mr. Yik-Cheung Tam is now a student of the Carnegie Mellon University. This work was finished when he was an MPhil student of HKUST. E-mail: yct@andrew.cmu.edu

Dr. Qi Li is now with the Li Creative Technologies, Inc., New Providence, NJ, which he founded. This work was finished when he was with the Department of Dialogue System Research, Multimedia Communication Research Laboratory, Bell Labs, Lucent Technologies, 600 Mountain Avenue, Murray Hill, NJ 07974. E-mail: qili@ieee.org.

I. Introduction

In automatic speech recognition (ASR), the design of acoustic models involves two main tasks: feature extraction and data modeling. Feature extraction aims at finding succinct, relevant, and discriminative features from acoustic data for later modeling, and data modeling tries to create mathematical models for each acoustic entity with high discriminability. Acoustic features such as linear predictive cepstral coefficients (LPCC), mel-frequency cepstral coefficients (MFCC), perceptual linear predictive coefficients (PLP) are commonly used, and the most popular data modeling techniques in current ASR are based on hidden Markov modeling (HMM). Recently, a new auditory-based feature extraction algorithm for robust speech recognition was proposed [1]. It attempts to mimic more closely the human auditory process from the outer ear, through the middle ear and to the inner ear. The filters in various parts of the feature extraction module were obtained from psychoacoustics literature with some manual adjustments and the resulting auditory-based features were shown more robust in noisy environments [2]. However, not unlike the extraction of other more commonly used features like MFCCs, some parameters in the feature extraction process are still set by experience, heuristics, or simple psychoacoustics results. For example, the shape of the auditory-based filters are manually modified from psychoacoustic results that were derived using simple or mixed tones. There are reasons to believe that these auditory-based filters may not be optimal for continuous speech in the sense that they may not result in minimum recognition errors, especially under the context of a particular recognition task. In this paper, we attempt to design the auditory-based filters (AF) used in the feature extraction algorithm discriminatively in a data-driven approach so as to minimize the final recognition errors. We will call the resulting filters "*discriminative auditory-based filters*" (DAF). Furthermore, the training of DAF is followed by the discriminative training of acoustic models under the unified framework of MCE/GPD(minimum classification error and generalized probabilistic descent) [3], [4].

The past approaches of discriminative feature extraction (DFE) may be divided into two major categories:

(1) Most DFE-related works are based on common features such as log power spectra [5], mel-filterbank log power spectra [6], and LPCC [7]. In these works, a transformation network is discriminatively trained to derive new discriminative features for the following data modeling process. For instance, in [6], Chengalvarayan and Deng generalized the use of the discrete cosine transform (DCT) in the generation of MFCCs and replaced it by state-dependent linear transformations that were optimally and discriminatively trained in conjunction with the dis-

criminative training of the HMM parameters. In the meantime, Rahim used neural networks to transform LPCCs before they were input into digit HMMs [7]. Discriminative feature weighting by Torre *et al.* [8] is a special form of feature transformation in which the transformation matrix is diagonal. Notice that all these work do not touch the front-end signal processing module that derives inputs to their transformation networks.

(2) In contrast, Alain Biem [9] applied joint discriminative training on both HMM parameters and filters in the front-end. The two kinds of parameters, HMM parameters and filter parameters, were assumed independent. Two kinds of filter parameterization were tried: Gaussian filters or free-formed filters. The acoustic models and the evaluation tasks were relatively simple to today's standard, and the improvement was small. Furthermore, the free-formed filters performed worse than the Gaussian filters.

This paper is *not* about discriminative design of feature transformation networks. Instead, optimization of the auditory-based filters is studied under the framework of the auditory-based feature extraction algorithm. We postulate that the shape of human auditory filters is not arbitrary, but based on psychoacoustic evidence, it should be "*triangular-like*" as defined in Section II. Our non-parametric formulation of the triangular-like constraint is general enough that it covers the triangular filters commonly used in computing MFCCs as well as the Gaussian filters used in Biem's work [9]. One of the challenges in this paper is to derive a mathematical expression for such constrained filters. We achieve this through two parameter-space transformations.

This paper is organized as follows: In the next Section, we first review the auditory-based feature extraction algorithm and explain the suggested triangular-like constraint in details. In Section III, we derive the formulas for discriminative training of various auditory-based filter parameters. This is followed by experiments and results on the Aurora2 corpus in Section IV. Finally in Section V, we conclude with remarks on future investigations.

II. Review of the Auditory-based Feature and Its Psychoacoustic Constraints

Based on an analysis of humans' peripheral auditory system, the auditory system was divided into several modules. The function of each module was analyzed from a signal processing point of view and was implemented by appropriate signal processing models. The final algorithm also took into consideration the computational complexity. A diagram of the feature extraction algorithm is presented in Fig. 1. There are three modules which differ from other auditory-based approaches: outer-middle ear transfer function (OME-TF), linear interpolation from linear frequency to Bark frequency, and auditory-based filtering. The frequency responses of the outer-middle ear and the auditory-based filters are obtained from psychoacoustics literature with some manual adjustments and are plotted in Fig. 2 and Fig. 3 respectively. The OME-TF is used to replace the pre-emphasis filter used in other feature extraction (such as LPCC and MFCC), and the auditory-based filters are used to replace the triangular filters commonly used in computing MFCC. There are two reasons to select the shape of an auditory-based filter in Fig. 3:

1. it is similar to the shape of human auditory filter as reported in [10], and
2. it is also similar to the shape of speech formants. Through moving-average operation, the formants can be enhanced. Some studies showed that enhanced formants may improve speech intelligibility [11], [12].

We notice that the shape of an auditory filter in the psychoacoustic literature is determined experimentally using simple or mixed tones that are not real speech, and we have to manually adjust these filters to get better recognition performance [2]. It is clear that the filters may not be optimal for different speech recognition tasks. In this paper, we would like to extend the algorithm and design the auditory-based filters to achieve minimum classification errors.

In the human auditory system, the filtering function is actually performed when the traveling wave due to a sound runs along the basilar membrane. Each specific location exhibits the highest response to the traveling wave of a characteristic frequency. Therefore, the human auditory filters must satisfy the following constraints:

- their shape must be continuous and should not contain discontinuities as in the triangular filter commonly used to compute MFCCs;
- the peak of a filter must locate at somewhere in the middle; and
- the response must taper off to both ends monotonically.

We will call any filters that satisfy the above constraints "*triangular-like filters*" and the constraints collectively as the "*triangular-like constraint*". The triangular-like constraint can then be represented mathematically as follows: If the width of a filter is $(L_l + 1 + L_r)$, with weights $\{w_{-L_l}, w_{-L_l+1}, \ldots, w_{-1}, w_0, w_1, \ldots, w_{L_r-1}, w_{L_r}\}$, then

(a) $0.0 \leq w_{-L_l} \leq w_{-L_l+1} \leq \cdots \leq w_{-1} \leq w_0$;

(b) $0.0 \leq w_{L_r} \leq w_{L_r-1} \leq \cdots \leq w_1 \leq w_0$; and

(c) $w_0 = 1.0$.

In the next Section, we will derive a non-parametric mathematical expression for a triangular-like filter using two parameter-space transformations.

III. Discriminative Filter Design

In our acoustic modeling framework, there are two types of free parameters $\Theta = \{\Lambda, \Phi\}$: the HMM parameters Λ and the parameters Φ that control feature extraction (FE). Λ includes state transition probabilities a_{ij}, and observation probability distribution functions b_j, $i, j = 1, 2, \ldots, S$, where S is the number of states in a model. Φ consists of all parameters in the feature extraction process as described in Fig. 1. In this paper, we only deal with the auditory-based filters which are trained discriminatively by minimizing the classification errors.

A. Auditory-based Filter Design

The auditory-based features are extracted as described in Section II and [1]. In the design, a 128-point Bark spectrum from the outer-middle ear is fed to 32 auditory-based filters in the inner ear that are equally spaced at an interval of 4 points apart in the spectrum as shown in Fig. 4. Thus, after auditory-based filtering, the 128-point Bark spectrum is converted to 32 filter outputs (or channel energies) from which cepstral coefficients are computed using the discrete cosine transform. An auditory-based filter of our system may be treated as a two-layer perceptron without nonlinearity as depicted in Fig. 5. In the figure, the weight in the second-layer perceptron $w_{\beta k}$ is the gain and weights $\boldsymbol{w_{\alpha k}}$ in the first layer represent the filter. Although the two-layer perceptron can be represented by an equivalent one-layer perceptron, the structure allows us to examine the resulting filter shapes and gains separately [1].

B. Discriminative Training of Feature Extraction Parameters

Let's first denote various parameters in the auditory-based feature extraction as follows:

$\boldsymbol{e_t}$	: FFT inputs to auditory-based filters at time t
$\boldsymbol{u_t}$	: outputs from auditory-based filters at time t
$\boldsymbol{z_t}$	: channel outputs at time t
$\boldsymbol{x_t}$	: acoustic features at time t
$\boldsymbol{v_t}$	: static acoustic features at time t
$\boldsymbol{v'_t}$	: delta acoustic features at time t
$\boldsymbol{v''_t}$	: delta delta acoustic features at time t
$\boldsymbol{w_{\alpha k}}$	: filter weights of the k-th channel
$w_{\beta k}$	: gain of the k-th channel
$\boldsymbol{\delta_k}$	: supplementary deltas associated with $\boldsymbol{w_{\alpha k}}$

[1]Similar treatment appears in [9] which used Gaussian filters with separate trainable gains.

These parameter notations are also illustrated in Fig. 6. As usual, vectors or matrices are bold-faced and matrices are capitalized.

The empirical expected string-based misclassification error $\mathcal{L}$, is defined as

$$\mathcal{L}(\Theta) = \frac{1}{N_u} \sum_{u=1}^{N_u} \mathcal{L}_u(\Theta) = \frac{1}{N_u} \sum_{u=1}^{N_u} l(d(X_u)) \tag{1}$$

where X_u is one of the N_u training utterances; $d(.)$ is a distance measure for misclassifications; and $l(.)$ is a soft error counting function. We follow the common practice of using the sigmoid function for counting soft errors and using log-likelihood ratio between the correct string and its competing hypotheses as the distance function. That is,

$$l(d) = \frac{1}{1 + \exp(-\alpha d + \beta)} \tag{2}$$

where the parameter α controls the slope of the sigmoid function and β controls its offset from zero, and

$$d(X_i) = G_i(X_i) - g_i(X_i) \tag{3}$$

in which the discriminant function $g(.)$ is the log-likelihood of a decoding hypothesis of an utterance. Thus, if an utterance X_i of duration T is represented by the model sequence Λ_i, then its log-likelihood is denoted as $g_i(X_i)$ and

$$\begin{aligned} & g_i(X_i) \\ & \quad = \log P(X_i|\Lambda_i) \\ & \quad \approx \log P(X_i, Q_{max}|\Lambda_i) \\ & \quad = \log \pi_{q_1} + \sum_{t=1}^{T-1} \log a_{q_t q_{t+1}} + \sum_{t=1}^{T} \log b_{q_t}(\boldsymbol{x_t}) \end{aligned} \tag{4}$$

where $Q_{max}(= q_1, q_2, \ldots, q_t, q_{t+1}, \ldots, q_T)$ is the HMM state sequence obtained from Viterbi decoding. $G_i(X_i)$ is the log of the mean probabilities of its N_c competing strings and is defined as

$$G_i(X_i) = \log \left[\frac{1}{N_c} \sum_{j=1;j\neq i}^{N_c} \exp(\eta g_j(X_i)) \right]^{1/\eta} \tag{5}$$

where η controls the weightings of the various hypotheses. Usually the competing hypotheses are obtained through N-best decoding but as we will explain later, we also attempt to improve the convergence and solution of MCE/GPD by a new heuristic which we call "*N-nearest hypotheses*" [13].

To optimize any parameter $\theta \in \Theta$, one has to find the derivative of the loss function $\mathcal{L}_i$ with respect to θ for each training utterance X_i:

$$\frac{\partial \mathcal{L}(X_i)}{\partial \theta} = \frac{\partial \mathcal{L}_i}{\partial \theta} = \frac{\partial l}{\partial d}\left[\frac{\partial d}{\partial g_i}\cdot\frac{\partial g_i}{\partial \theta} + \frac{\partial d}{\partial G_i}\cdot\frac{\partial G_i}{\partial \theta}\right] . \tag{6}$$

$l(.)$ is the sigmoid function given in Eqn.(2), and its derivative is

$$\frac{\partial l}{\partial d} = \alpha l(1-l) . \tag{7}$$

From Eqn.(3), we obtain the derivatives of the distance function w.r.t. the discriminant functions as

$$\frac{\partial d}{\partial g_i} = -1 \tag{8}$$

and

$$\frac{\partial d}{\partial G_i} = +1 . \tag{9}$$

Also, from Eqn.(5) we get

$$\frac{\partial G_i}{\partial \theta} = \frac{\sum_{j\neq i}^{N_c} \exp(\eta g_j(X_i)) \frac{\partial g_j}{\partial \theta}}{\sum_{j\neq i}^{N_c} \exp(\eta g_j(X_i))} . \tag{10}$$

Hence, Eqn.(6) becomes

$$\frac{\partial \mathcal{L}_i}{\partial \theta} = \frac{\partial l}{\partial d}\left[\frac{\sum_{j\neq i}^{N_c} \exp(\eta g_j(X_i)) \left(\frac{\partial g_j}{\partial \theta} - \frac{\partial g_i}{\partial \theta}\right)}{\sum_{j\neq i}^{N_c} \exp(\eta g_j(X_i))}\right] . \tag{11}$$

To evaluate Eqn.(11), one has to find the partial derivative of g_i w.r.t. any trainable parameters. We will drop the utterance index i for clarity from now on. Also, since many works have been done on discriminative training of HMM parameters with MCE/GPD, one may refer to the tutorial paper [4] for the re-estimation formulas of HMM parameters and we will only present those of feature extraction parameters. To do that, we first assume that the trainable FE parameters in Φ are independent of HMM parameters (as in [9]). Secondly, it is helpful to see that the log-likelihood of an utterance is related to an FE parameter $\phi \in \Phi$ through the static features, and the dynamic features are related to ϕ also through the static features. Let's assume that the final feature vector $\boldsymbol{x_t}$ at time t consists of N static features $\boldsymbol{v_t}$ and N dynamic features $\boldsymbol{v'_t}$ which are computed from $\boldsymbol{v_t}$ by the following regression formula

$$\boldsymbol{v'_t} = \sum_{m=-L_1}^{L_1} c'_m \boldsymbol{v_{t+m}} \tag{12}$$

where $c'_m (m = -L_1, \ldots, L_1)$ are the regression coefficients. Hence, the derivative of the utterance log-likelihood g of Eqn.(4) w.r.t an FE parameter ϕ is given by

$$\begin{aligned}
\frac{\partial g}{\partial \phi} &= \sum_t \frac{1}{b_{q_t}(\boldsymbol{x_t})} \sum_{j=1}^{2N} \frac{\partial b_{q_t}}{\partial x_{tj}} \cdot \frac{\partial x_{tj}}{\partial \phi} \qquad (13) \\
&= \sum_t \frac{1}{b_{q_t}(\boldsymbol{x_t})} \sum_{j=1}^{N} \left[\frac{\partial b_{q_t}}{\partial v_{tj}} \cdot \frac{\partial v_{tj}}{\partial \phi} + \frac{\partial b_{q_t}}{\partial v'_{tj}} \cdot \frac{\partial v'_{tj}}{\partial \phi} \right] \\
&= \sum_t \frac{1}{b_{q_t}(\boldsymbol{x_t})} \sum_{j=1}^{N} \frac{\partial b_{q_t}}{\partial v_{tj}} \cdot \frac{\partial v_{tj}}{\partial \phi} + \\
&\quad \sum_t \frac{1}{b_{q_t}(\boldsymbol{x_t})} \sum_{j=1}^{N} \frac{\partial b_{q_t}}{\partial v'_{tj}} \left(\sum_{m=-L_1}^{L_1} c'_m \frac{\partial v_{(t+m)j}}{\partial \phi} \right). \qquad (14)
\end{aligned}$$

Finally, the derivative of an utterance log-likelihood g w.r.t. any (static or dynamic) feature parameters x_{tj} is given by

$$\frac{\partial b_{q_t}}{\partial x_{tj}} = -\sum_{m=1}^{M} c_{q_t,m} \, \mathcal{N}_{q_t}(\boldsymbol{x_t}) \left(\frac{x_{tj} - \mu_{q_t,mj}}{\sigma^2_{q_t,mj}} \right) . \qquad (15)$$

Using Eqn.(7), Eqn.(14), and Eqn.(15), Eqn.(11) may be computed if $\frac{\partial v_{tj}}{\partial \phi}$ is also known. The computation of $\frac{\partial v_{tj}}{\partial \phi}$ depends on the nature of the training parameter ϕ. In this paper, we are only interested in optimizing the filter parameters and the re-estimation formula of each type of filter parameters is presented below.

B.1 Re-estimation of Triangular-like Filter Parameters

As explained in Section II, we only require the auditory-based filters to be triangular-like. We further simplify the design by having the peak of a filter right at the middle. For a digital filter with $(2L+1)$ points, we associate the filter weights $\{w_{-L}, w_{-L+1}, \ldots, w_{-1}, w_0, w_1, \ldots, w_{L-1}, w_L\}$ with a set of *deltas* $\boldsymbol{\delta}$, $\{\delta_{-L}, \ldots, \delta_{-1}, \delta_1, \ldots, \delta_L\}$. (Notice that $\Delta w_i = w_{i-1} - w_i$ in Fig. 7 for $i = 1, 2, \ldots, L$, are related to δ_i by Eqn.(16) below.) Positively-indexed weights are related to the positively-indexed deltas mathematically as follows:

$$w_j = 1 - F\left(\sum_{i=1}^{j} H(\boldsymbol{\delta}_i)\right) \quad , \quad j = 1, \ldots, L \qquad (16)$$

where, $F(.)$ and $H(.)$ are any monotonically increasing functions such that

$$0.0 \le F(x) \le 1.0 \quad \text{and} \quad 0.0 \le \mathrm{H(x)} \ . \qquad (17)$$

Negatively-indexed weights are related to the negatively-indexed deltas in a similar manner. The motivation is that we want to subtract more positive quantities from the maximum weight

w_0 as we move towards the two ends of a filter. Eqn.(16) involves two functions: $H(.)$ is any monotonically increasing function which turns arbitrarily-valued deltas to positive quantities, and $F(.)$ is any monotonically increasing function that restricts the sum of transformed deltas to less than unity. In this paper, $H(x)$ is set to the exponential function $\exp(x)$, and $F(x)$ is a sigmoid function defined as follows:

$$F(x) = \frac{2}{1 + e^{-\gamma x}} - 1$$

where γ was set to 0.001 in the following experiments.

C. Re-estimation of Filter Gains

The gain of the k-th channel filter is represented by the weight $w_{\beta k}$ in the second layer of the filter shown in Fig. 5. Since the static feature $\boldsymbol{v_t}$ is related to the non-linearity function output $\boldsymbol{z_t}$ which in turn is related to the filter output $\boldsymbol{u_t}$, by applying the chain rule (see Fig. 5 and Fig. 6), one may obtain the derivative of each static feature v_{tj} w.r.t. the gain of the k-th channel as follows:

$$\frac{\partial v_{tj}}{\partial w_{\beta k}} = \frac{\partial v_{tj}}{\partial z_{tk}} \cdot \frac{\partial z_{tk}}{\partial u_{tk}} \cdot \frac{\partial u_{tk}}{\partial w_{\beta k}} \ . \tag{18}$$

The static feature $\boldsymbol{v_t}$ is the discrete cosine transform (DCT) of $\boldsymbol{z_t}$, and if we denote the DCT matrix by $\boldsymbol{W^{(D)}}$, then $\boldsymbol{v_t} = \boldsymbol{W^{(D)}} \cdot \boldsymbol{z_t}$. Therefore,

$$\frac{\partial v_{tj}}{\partial z_{tk}} = W^{(D)}_{jk} \ . \tag{19}$$

The derivative of the non-linearity function $r(.)$ depends on its exact functional form, and we will denote it by r'. Since $\boldsymbol{z_t} = r(\boldsymbol{u_t})$, the derivative is

$$r' = \frac{\partial z_{tk}}{\partial u_{tk}} \ . \tag{20}$$

For the k-th channel, the gain is related to the channel output u_{tk} (see Fig. 5) as

$$u_{tk} = w_{\beta k} \cdot y_{tk} \tag{21}$$

where y_{tk} is the output from the first layer of the filter. Thus,

$$\frac{\partial u_{tk}}{\partial w_{\beta k}} = y_{tk} \ . \tag{22}$$

Substituting the derivatives in Eqn.(19), Eqn.(20), and Eqn.(22) into Eqn.(18), we obtain

$$\frac{\partial v_{tj}}{\partial w_{\beta k}} = W^{(D)}_{jk} \cdot r' \cdot y_{tk} \ . \tag{23}$$

D. Re-estimation of Filter Weights

Filter weights of the k-th channel $\boldsymbol{w_{\alpha k}}$ are re-estimated indirectly through the associated deltas. Again using the chain rule in a similar fashion as in the re-estimation of filter gains (see Fig. 5 and Fig. 6), the derivative of the j-th static feature w.r.t. the h-th delta in the k-th channel filter is given by

$$\frac{\partial v_{tj}}{\partial \delta_{kh}} = \frac{\partial v_{tj}}{\partial z_{tk}} \cdot \frac{\partial z_{tk}}{\partial u_{tk}} \cdot \frac{\partial u_{tk}}{\partial y_{tk}} \cdot \frac{\partial y_{tk}}{\partial \delta_{kh}} \ . \tag{24}$$

Since $u_{tk} = w_{\beta k} \cdot y_{tk}$, therefore

$$\frac{\partial u_{tk}}{\partial y_{tk}} = w_{\beta k} \ . \tag{25}$$

The filter output y_{tk} is given by

$$\begin{aligned} y_{tk} &= \sum_{i=-L}^{L} w_{\alpha k i} \cdot e_{tki} \\ &= e_{tk0} \\ &\quad + \sum_{i=-L}^{-1} \left[1 - F\left(\sum_{m=i}^{-1} H(\delta_{km}) \right) \right] e_{tki} \\ &\quad + \sum_{i=1}^{L} \left[1 - F\left(\sum_{m=1}^{i} H(\delta_{km}) \right) \right] e_{tki} \ . \end{aligned} \tag{26}$$

Hence, for the positively-indexed deltas

$$\frac{\partial y_{tk}}{\partial \delta_{kh}} = -H'(\delta_{kh}) \left[\sum_{i=h}^{L} F' \cdot e_{tki} \right] \ . \tag{27}$$

Substituting Eqn.(19), Eqn.(20), Eqn.(25), and Eqn.(27) into Eqn.(24), we obtain

$$\frac{\partial v_{tj}}{\partial \delta_{kh}} = W_{jk}^{(D)} \cdot r' \cdot w_{\beta k} \left\{ -H'(\delta_{kh}) \left[\sum_{i=h}^{L} F' \cdot e_{tki} \right] \right\} \ . \tag{28}$$

A similar expression can be derived for the negatively-indexed deltas.

E. Updates

Finally, a (locally) optimal model or feature extraction parameter $\theta \in \Theta$ may be found through the iterative procedure of GPD using the following update rule:

$$\theta(n+1) = \theta(n) - \epsilon(n) \cdot \left. \frac{\partial \mathcal{L}}{\partial \theta} \right|_{\theta = \theta(n)} \ . \tag{29}$$

Notice that the convergence of GPD requires that $\sum_n \epsilon(n)^2 < \infty$. One common way to ensure the requirement is to have the learning rate decrease properly with time.

F. Miscellaneous

It is worth to mention that some of the filter parameters such as the filter gains must be positive quantities. (Weights of our triangular-like auditory-based filter are always positive by our definition though.) To ensure any parameter X to be positive, we apply the following parameter transformation [9], [4]

$$X \rightarrow \tilde{X} : X = e^{\tilde{X}} \; . \tag{30}$$

Optimization is performed over the transformed parameters which are converted back to the original parameter space after MCE training.

Moreover, most current systems employ Cepstral Mean Subtraction (CMS) to alleviate the adverse effect due to channel mismatches. To do that, the re-estimation formulas above have to be modified slightly. The changes are small and straight-forward, and we will leave them to the readers. However, when CMS is used together with log non-linearity, one may verify that the filter gains then will *not* be changed.

IV. Evaluation

Extensive experiments were performed to answer the following questions:

- May the auditory-based filters be further optimized for a task?
- As there have not been many works done on training both the front-end feature extraction parameters as well as the model parameters together, what may be a good procedure to do so?
- How sensitive is the training procedure to the initial configuration of the filters?

A. The Aurora2 Corpus

Discriminative training of our new auditory-based filters was evaluated on the Aurora2 corpus. It was created for research in robust distributed speech recognition under noisy environments. It was chosen because the auditory-based features were originally developed for noisy ASR.

The Aurora2 [14], [15] corpus consists of connected digits from the adult portion of the clean TIDIGITS database [16]. The utterances were pre-filtered using simulated telephone channel and GSM channel respectively, and various types of noises were added to the filtered TIDIGITS at 6 different SNR levels ranging from 20dB to -5dB at 5dB steps. There are totally 8440 utterances in the training data spoken by 55 male and 55 female adults. Two training methods are defined for evaluating recognition technologies on the corpus; but only the multi-condition training method was attempted in this work. The test data consist of 4004 utterances spoken by 52 male and 52 female speakers that are different from the training speakers. Three test sets are defined to evaluate recognition technologies under different matching conditions as summarized in Table I.

B. Experimental Setup

B.1 Feature Extraction

Auditory-based features were extracted from speech utterances every 10ms as described in Section II over a window of 25ms. Thirteen cepstral coefficients including c0 were computed from each frame of auditory-based features, which we will call them as *auditory-based feature cepstral coefficients* (AFCC). Here c0 was used to represent the frame energy. Each auditory-based filter had 11 weights and the middle (6-th) weight was assumed maximum with the value of 1.0. However, each of the 32 channels had its own filter and the triangular-like filters were not assumed symmetric. The logarithm function was used as the non-linearity function to compress channel outputs before DCT was performed. The final feature vector has 39 dimensions consisting of 13 static AFCCs and their first- and second-order derivatives, which are computed using linear regression. Cepstral mean subtraction was performed to alleviate channel mismatches.

B.2 Acoustic Modeling

We followed the baseline model configurations of the Aurora evaluation of ICSLP 2002: all the 11 digit models were whole-word models, and they were strictly left-to-right HMMs with no skips. Each digit HMM has 16 states with 3 Gaussian mixture components per state. The silence model was a 3-state HMM with 6 Gaussian mixture components. In addition, there was a 1-state HMM to represent short pauses which was tied to the middle state of the silence HMM. The HTK software was used for maximum-likelihood estimation (MLE) of all models as well as the subsequent decoding.

B.3 Discriminative Training

From the initial MLE digit models, discriminative training was performed to derive MCE estimates of the HMM parameters and/or MCE estimates of the filter parameters. The same training data were utilized to perform both MLE and discriminative training. In theory, it will be better to use another set of data that are independent of the MLE training data so as to avoid over-fitting the data. However, we would like to conform with Aurora's set-ups so that our results may be compared with other similar efforts; therefore, we did not use any additional data. Furthermore, corrective training was employed. Competing hypotheses were obtained using our new N-nearest decoding algorithm [13] instead of the commonly used N-best decoding method. A problem with N-best hypotheses is that when the correct hypothesis is too far from the N-best hypotheses, the training datum will fall into the *un-trainable region* of the sigmoid function. The use of the N nearest competing hypotheses tries to keep the training data as close to the trainable region as possible. Consequently, the amount of *effective* training data is increased, and since there is no need to use a flatter sigmoid and a large learning rate, the training seems to be more stable.

Since the HMM parameters and the feature extraction (FE) parameters were assumed independent, different learning rates might be employed as suggested by [9]. This can be important as the two types of parameters may have very different dynamic ranges. In our current investigation, the following starting learning rates were found empirically to give good results:

starting learning rate of FE parameters : 1.0

starting learning rate of HMM parameters : 442.0 .

These learning rates R decreased with iterations n as $R(n) = R(0) \cdot (1 - \frac{n}{I_{max}})$, and we limited the maximum number of iterations I_{max} to 50.

Other parameters in the MCE/GPD procedure were set as follows: α and β in the sigmoid function of Eqn.(2) were set to 0.1 and 0.0 respectively. β was set to zero so that our discriminative training will try to correct *all* mis-recognized utterances. The α value of 0.1 was empirically determined to give good performance. Finally, the value of η in Eqn.(5) was set to 1.0 to give equal weights to all competing hypotheses.

B.4 Various Training Methods

Since there are two sets of trainable parameters: HMM or FE parameters, we also studied the most effective procedure for their training. The following different training methods were explored:

Method 1. **MFCC Baseline**: ML estimation of the digit models using traditional MFCCs;

Method 2. **AFCC Baseline**: ML estimation of the digit models using AFCCs;

Method 3. **M-only**: discriminative training of HMM parameters *only*;

Method 4: **F-only**: discriminative training of FE parameters — filter gains and weights — *only*;

Method 5. **F+M**: joint or simultaneous discriminative training of HMM *and* FE parameters;

Method 6. **F + M-mle**: discriminative training of FE parameters followed by an ML re-estimation of the models under the new feature space;

Method 7. **F + M-mce**: discriminative training of FE parameters followed by discriminative training of the HMM parameters under the new feature space; and

Method 8. **F + M-mle + M-mce**: same as *F + M-mle* but followed by a subsequent discriminative training of HMM parameters.

The various training methods are also summarized in Table II.

C. Experiments and Results

The performance of various discriminative training methods of our triangular-like filters on each test subset as well as on the overall test set of the Aurora2 corpus is plotted in Fig. 8. Notice that the recognition accuracy of each test set is computed as the mean accuracy over speech of all SNRs except the clean portions and the speech at -5dB in conformity with the evaluation metric of Aurora2. Two baseline results are included for comparison purpose: the baseline results used by ICSLP 2002 for Aurora Evaluation [17] (labelled as "MFCC Baseline"), and the baseline results using the auditory-based features (labelled as "AFCC Baseline").

From Fig. 8, it is clear that the use of the auditory-based features (AFCC Baseline) outperforms the MFCC Baseline results and reduces the overall WER by 11.6%. Although the models were trained on noises and channels that create test set A, the auditory-based features give similar performance on test set B which contains mismatched noises, and even better performance on test set C which employs different channel characteristics and noises. On the other hand, the performance of the MFCC Baseline drops substantially on test set C. It is our general observation that the auditory-based features are more robust than MFCCs in mismatched testing environments, and the findings agrees with those reported in [1], [2]. Since the AFCC Baseline results are far better than the MFCC Baseline results, and we are investigating whether discriminative auditory-based filter (DAF) gives additional gains over the basic AFCC parameters, we compare the performance of various DAF training methods with respect to the AFCC Baseline, and the corresponding *relative* word error rate reductions (WERR) are computed in Table III.

Optimizing the auditory-based filters alone by discriminative training ("F-only" method) gives a WERR of 4.1% relative to the AFCC Baseline results. On the other hand, discriminative training of the HMM parameters alone is much more effective: the "M-only" training method achieves a WERR of 17.3%. It will be interesting to see if the two gains are additive. We first investigated joint optimization of the two kinds of parameters by discriminative training, but the result turned out to be very close to that of "M-only" training. The finding agrees with Torre's comments in [8] that joint optimization of model and FE parameters is only effective for small classification problem; and for large complex system, the evolution of the FE parameters is small compared with that of the model parameters during joint optimization. There are two plausible reasons:

• There are far more model parameters (11 models $\times$ 16 states $\times$ 3 mixtures $\times$ 39 features $\times$ 2 = 41184 Gaussian means and variances) than FE parameters (32 channels $\times$ 11 = 352 filter gains and weights). The larger model space allows more room for improvement than the smaller feature space, and the improvement due to HMM parameters over-shadows that due to FE parameters.

• The model parameters and feature extraction parameters are *not* truly independent. When the feature space is changed (due to changes in FE parameters) the HMM parameters should be re-estimated under the *new* feature space.

The "F+M-mce" training method tries to remedy the first problem by combining the two training methods, "F-only" and "M-only" training, sequentially rather than simultaneously. To partially address the second problem as well, in the "F+M-mle+M-mce" training method, FE parameters are trained discriminatively, then new HMMs are re-estimated using the ensuing features by the EM algorithm, from which discriminative models are derived. The "F+M-mle" training method stops after new ML models are re-estimated from the new features. The last two training methods are computationally expensive owing to the ML re-estimation step. Below are what we find:

• Although simultaneous discriminative training of HMM and FE parameters does not give additional improvement over the "M-only" training method, sequential application of discriminative training firstly on FE parameters and then HMM parameters gives a small but significant additional gain over the "M-only" training method: the "F+M-mce" training method achieves a WERR of 20.0% compared with a WERR of 17.3% by the "M-only" training method.

• However, the biggest performance improvement comes from the "F+M-mle+M-mce" training method when models were re-trained under the new feature space as demanded by the new DAFs using the ML criterion before discriminative training was used again to fine-tune the HMM parameters. The final WERR is 21.7%.

• We also had checked the importance of the final MCE training of HMM parameters by leaving out the step as in the "F+M-mle" training method. It can be seen that simply ML re-training of the models with the new DAFs alone does not give additional gain over DAFs.

Finally the performance difference between the AFCC Baseline system and the "F-only" system, and that between the "M-only" system and the "F+M-mle+M-mce" system are both statistically significant (at the 0.05 confidence level).

C.1 Shape of Trained Filters

Fig. 9 shows the shape of all the 32 DAFs (solid curves) after training that was initialized by Li's filters as in [1] (dotted curves). In general, the DAFs are not symmetric and the deltas δ_i or $\Delta w_i, i = -L, \ldots, L$ are not the same. When we compare the trained filters with the initialization filters, we observe that, with a few exceptions:

- the first 10 filters are asymmetric and their left halves are raised from their initial positions.
- the next 12 filters are more or less symmetric and both left and right halves are raised from their original positions.
- the changes to the remaining filters are mixed.

C.2 Sensitivity to Initial Filter Parameters

Like other numerical methods that is based on gradient descent to solve optimization problems, different initial parameters may lead to different local minima using the MCE/GPD algorithm. An experiment was conducted to check the sensitivity of our DAF training algorithm using a different type of initial filters, the triangular filters. The triangular filters were chosen for two reasons: (1) they are commonly used in the computation of MFCCs, and (2) it is better if our algorithm may work with initial filters of simple shapes than one particular set of filters — Li's filters which was determined from real telephone data with different kinds of background noise [1]. The previous experiments with various training methods were repeated using triangular filters as the initial filters and the results are shown in Table IV. We also plot out the envelopes of the overall frequency response due to DAFs using Li's filters and triangular filters as the initial filters in Fig. 10 (a) and (b) respectively. It can be seen that both their recognition results and filter response envelopes are very similar to each other. One reason for the similar results may be that Li's filters and triangular filters are close enough for our application. The performance of our DAF training algorithm using other very dissimilar initial filters — for example, rectangular filters or filters of non-simple shapes — is still an open question.

V. Conclusions

In this paper, we proposed a discriminative training algorithm to extract auditory-based features and found an effective procedure to estimate both feature and model parameters discriminatively. We make as few assumptions as possible on the shape of the auditory-based filters, and only require them to be triangular-like. It is found that *joint* discriminative training of both the model parameters and filter parameters cannot supersede discriminative training of the model parameters alone. On the other hand, if discriminative training is applied *sequentially* to the filter parameters and then the model parameters, an additional gain is observed. Although the gain is modest, we believe that there are rooms for improvement. For instance, we will try to improve the resolving power of the filters under different acoustic contexts and phonetic contexts.

In the current training procedure, the model and feature parameters are assumed independent of each other. Theoretically, the model parameters are trained for a given feature space; if the features are changed, the model parameters should follow suit. In one sense, our current work is our first step to merge the two processes: feature extraction and estimation of model parameters together. In this paper, the two processes are linked together via the recognition feedbacks. In the future, we will try to couple them together more closely and re-visit the problem of their joint optimization.

VI. Acknowledgements

This work is supported by the Hong Kong Research Grants Council under the grant numbers HKUST6201/02E and CA02/03.EG04.

References

[1] Qi Li, Frank Soong, and Olivier Siohan, "A High-Performance Auditory Feature for Robust Speech Recognition," in *Proceedings of the International Conference on Spoken Language Processing*, 2000.

[2] Qi Li, Frank Soong, and Olivier Siohan, "An Auditory System-based Feature for Robust Speech Recognition," in *Proceedings of the European Conference on Speech Communication and Technology*, 2001.

[3] B.H. Juang and S. Katagiri, "Discriminative Training for Minimum Error Classification," *IEEE Transaction on Signal Processing*, vol. 40, no. 12, pp. 3043–3054, Dec 1992.

[4] W. Chou, "Discriminant-Function-Based Minimum Recognition Error Rate Pattern-Recognition Approach to Speech Recognition," *Proceedings of the IEEE*, vol. 88, no. 8, pp. 1201–1223, August 2000.

[5] J.S. Bridle and L. Doddi, "An Alphanet Approach to Optimising Input Transformations for Continuous Speech Recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1991, vol. 1.

[6] R. Chengalvarayan and Li Deng, "HMM-Based Speech Recognition using State-Dependent, Discrimi-

natively Derived Transforms on Mel-Warped DFT Features," *IEEE Transactions on Speech and Audio Processing*, vol. 5, no. 3, pp. 243–256, May 1997.

[7] M. Rahim and C.H. Lee, "Simultaneous ANN Feature and HMM Recognizer Design Using String-based Minimum Classification Error (MCE) Training," in *Proceedings of the International Conference on Spoken Language Processing*, 1996.

[8] A. Torre, A. M. Peinado, A. J. Rubio, J. C. Segura, and C. Benitez, "Discriminative Feature Weighting for HMM-based continuous Speech Recognizers," *Speech Communication*, vol. 38, no. 3–4, pp. 267–286, Nov. 2002.

[9] A. Biem, S. Katagiri, E. McDermott, and B.H. Juang, "An Application of Discriminative Feature Extraction to Filter-Bank-Based Speech Recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 2, pp. 96–110, Feb 2001.

[10] Brian C. J. Moore, *An Introduction to the Psychology of Hearing*, Academic Press, 4th edition, 1997.

[11] A. M. Simpson, B. C. J. Moore, and B. R. Glasberg, "Spectral Enhancement to Improve the Intelligibility of Speech in Noise for Hearing-impaired Listeners," *Acta Otolaryngol*, pp. 101–107, 1990, Suppl. 469.

[12] B. C. J. Moore, "Perceptual Consequences of Cochlear Hearing Loss and Their Implications for Design of Hearing Aid," *Ear & Hearing*, pp. 133–161, 1990.

[13] Y. C. Tam and B. Mak, "An Alternative Approach of Finding Competing Hypotheses for Better Minimum Classification Error Training," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Orlando, Florida, USA, 2002, vol. 1, pp. 101–104.

[14] H. G. Hirsch and D. Pearce, "The AURORA Experimental Framework for the Performance Evaluations of Speech Recognition Systems under Noisy Conditions," in *ISCA ITRW ASR2000 "Automatic Speech Recognition: Challenges for the Next Millennium"*, September 2000.

[15] D. Pearce, "Enabling New Speech Driven Services for Mobile Devices: An Overview of the ETSI Standards Activities for Distributed Speech Recognition Front-ends," in *Proceedings of AVIOS*, May 22–24 2000.

[16] R.G. Leonard, "A Database for Speaker-Independent Digit Recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1984.

[17] Reference [10] in the webpage of the 4th Special Section: "Aurora: Speech Recognition in Noise" at the ICSLP 2002 official website. *http://icslp2002.colorado.edu/special_sessions/aurora*.

TABLE I

DESCRIPTION OF THE 3 TEST SETS OF AURORA2

Test Set	Testing Conditions		#Utterances
	Matched Channel	Matched Noises	
A	yes	yes	28,028
B	yes	no	28,028
C	no	yes + no	14,014

TABLE II

VARIOUS TRAINING METHODS ("JOINT" MEANS JOINT TRAINING; "SEQ" MEANS SEQUENTIAL TRAINING)

Method	Name	MCE of Feature	MLE of HMM	MCE of HMM
1	MFCC Baseline	X	$\surd$	X
2	AFCC Baseline	X	$\surd$	X
3	M-only	X	X	$\surd$
4	F-only	$\surd$	X	X
5	F+M	$\surd_{joint}$	X	$\surd_{joint}$
6	F+M-mle	$\surd_{seq}$	$\surd_{seq}$	X
7	F+M-mce	$\surd_{seq}$	X	$\surd_{seq}$
8	F+M-mle+M-mce	$\surd_{seq}$	$\surd_{seq}$	$\surd_{seq}$

TABLE III

WORD ERROR RATE REDUCTION OF VARIOUS TRAINING METHODS

Test Set	M-only	F-only	F +M-mle	F +M-mce	F +M-mle +M-mce
A	21.2%	5.5%	5.3%	23.9%	27.0%
B	13.4%	2.6%	4.3%	16.4%	16.2%
C	17.4%	4.2%	4.9%	19.1%	22.5%
Overall	17.3%	4.1%	4.8%	20.0%	21.7%

TABLE IV

SENSITIVITY TO THE INITIAL FILTER PARAMETERS

Initial Filters	AFCC-Baseline	M-only	F-only	F +M-mce	F+M-mle +M-mce
Li's	88.54%	90.52%	89.01%	90.83%	91.03%
Triangular	88.65%	90.33%	89.08%	90.60%	90.89%

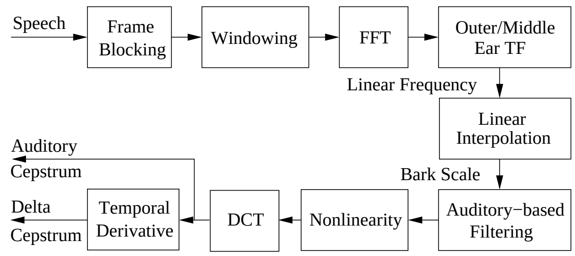

Fig. 1. Extraction of auditory-based features (after Li *et al.* [1])

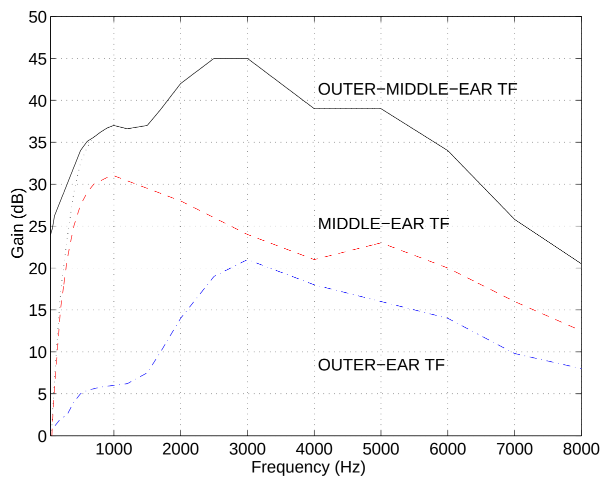

Fig. 2. Frequency response of outer-middle ear (after Li *et al.* [1])

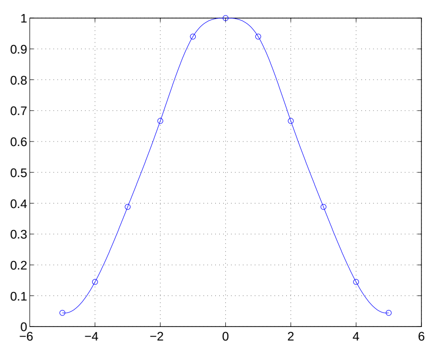

Fig. 3. Frequency response of an 11-point auditory-based filter (after Li *et al.* [1]. The difference between 2 filter points is 0.135 Barks.)

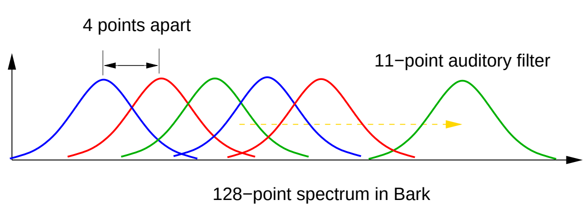

Fig. 4. Auditory-based filtering in our system

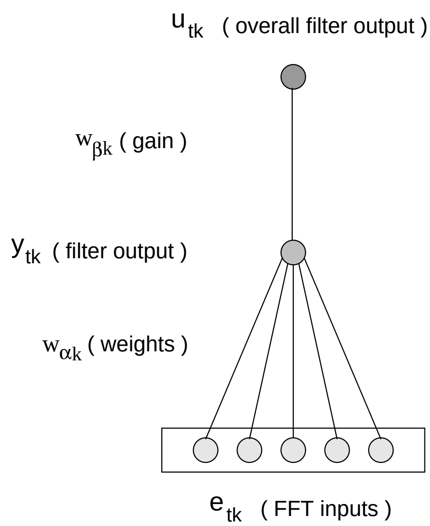

Fig. 5. The auditory-based filter of the k-th channel

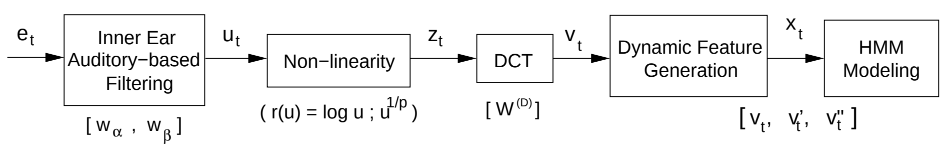

Fig. 6. Parameter notations in the extraction of the auditory-based features

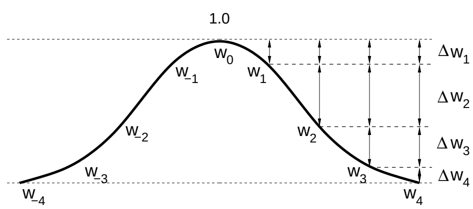

Fig. 7. Weights of our triangular-like auditory-based filter

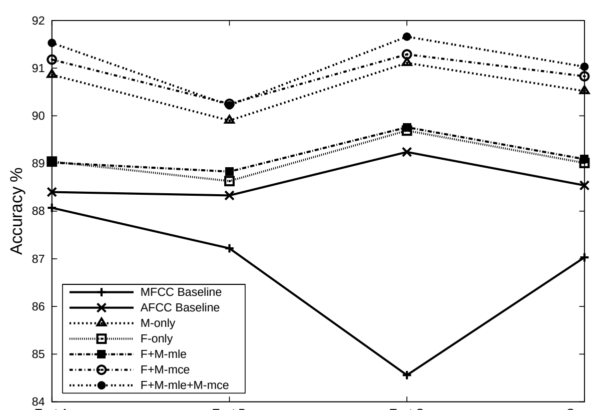

Fig. 8. Comparing various methods of training model or feature extraction parameters

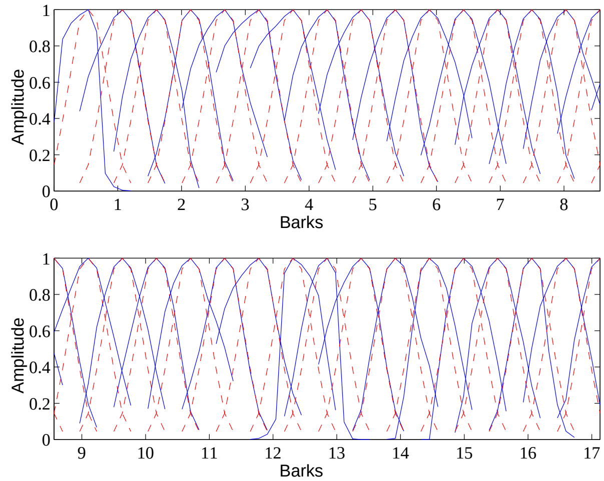

Fig. 9. Shape of the 32 MCE-trained auditory-based filters (solid curves). The training was initialized by Li's filters as in [1] (dotted curves).

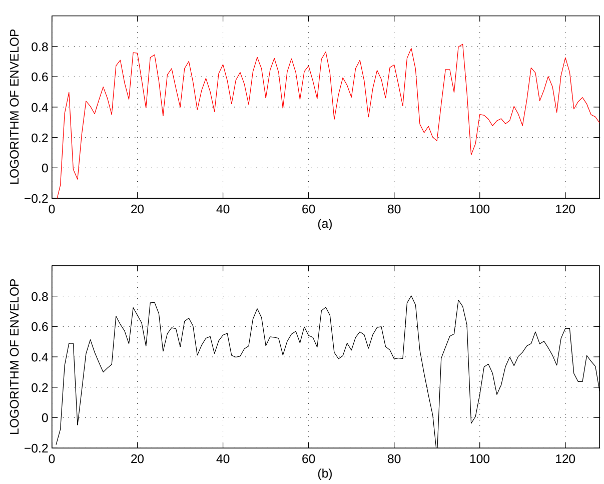

Fig. 10. Envelope of the overall frequency response of the outer-middle-inner ear using DAFs that trained from initial (a) triangular filters, and (b) Li's filters.