On the MixMax Model and Cepstral Features for Noise-Robust Voice Recognition

Thilo Stadelmann and Bernd Freisleben*, *Member, IEEE*

***Abstract*—The MixMax model is a well-known technique to build and evaluate statistical models of signals in the presence of background noise. It has successfully been applied to noise-robust voice recognition. The major drawback of the MixMax model is that it can only be applied to (log-)filterbank features that have been shown to be inferior to Mel-Frequency Cepstrum Coefficients (MFCCs) in audio processing. Nevertheless, good results using MixMax models for MFCC features have been reported in the literature. This paper proves that the MixMax model cannot work with MFCC features. Furthermore, it is shown that the good results reported in the literature have been obtained by a different method that has been used accidentally. The other model's formulation and its properties are discussed, (re-)opening new research perspectives for noise-robust voice modeling.**

***Index Terms*—noise masking, MixMax model, mel-frequency cepstral coefficients, singer identification, speaker identification**

I. Introduction

IN supervised speech- or voice recognition tasks, several existing approaches suffer from the mismatch between training and evaluation conditions caused by interfering background signals, called noise. A prominent technique to deal with such conditions in the modeling- or recognition stage is the MixMax model. Nádas et al. [1] have introduced it as a technique for speech recognition in the presence of noise. It provides a way to build a statistical mixture model, normally a Gaussian mixture model (GMM) [2], of a signal, while simultaneously keeping a model of the accompanying noise. Through the interaction of both models, noise compensation is achieved via a statistical variant of noise masking [3]: the noisy speech mixtures get ”masked“ by the background mixtures rather than cleaned. In the likelihood computation, the feature vectors are scored against the combined speaker-background model. The more a speaker mixture is masked by noise, the less it contributes to the final likelihood score. As a consequence, testing previously unseen signals against models built from training data under different noise conditions is possible as long as a model for the current noise exists.

Varga and Moore [4] have developed the same idea independently of Nádas et al. for the decomposition of speech and noise to facilitate speech recognition. Rose et al. [5] have used the MixMax model for robust speaker recognition and called it the Gaussian mixture model with integrated background (GMM-IB). They placed it in a framework of general signal–noise interaction and modeling. Burshtein and Gannot [6] have used the approach for speech enhancement on embedded devices, focusing on accelerating the necessary computations. Tsai et al. [7] have employed the MixMax model for singer's voice modeling within music information retrieval in several works [8]–[10]. Afify et al. [11] have derived upper and lower bounds on the mean of noise-corrupted speech signals using the MixMax' modeling assumptions. Furthermore, the MixMax equations have been used, extended and evaluated by Deoras and Hasegawa-Johnson [12] for simultaneous speech recognition (i.e., source separation) and Logan and Robinson [13], Erell and Weintraub [14] as well as Erell and Burshtein [15] for noisy speech recognition, enhancement and adaptation, among others.

This paper investigates the applicability of the MixMax model to cepstral features [16], inspired by contradicting views expressed in several recent publications. On the one hand, the MixMax' definition confines its use to linear transformations of the signal, such as, e.g., filterbank energies; on the other hand, features like mel frequency cepstral coefficients (MFCC) are generally deemed more powerful in voice recognition tasks, yielding improvements of up to 10%. Exploiting this advantage is clearly desirable, but not all authors agree on the feasibility of being successful in conjunction with the MixMax model. The contradictions in the literature are dissolved by experiments, arguments and proofs in this paper. This shows for the first time how noise compensation can be done on cepstral features directly. Additionally, small errors in the corpus of the MixMax model's training equations are corrected that have been repeated in the literature since their initial publication in 1994.

The paper is organized as follows: Section II introduces the idea behind the MixMax method, followed by the model's formal definition and an explication of its corpus of training- and evaluation equations in Section III. Section IV then introduces the problem of contradicting views about the MixMax model's suitability for cepstral features. They are investigated by providing an alternative explanation for publications claiming to use the MixMax model on MFCC features in Section V, and a proof that the other publications refraining from doing so are actually right in Section VI. Section VII concludes the paper and outlines areas for future research.

II. The MixMax Idea

The principal idea behind the MixMax model is as follows: given is an (unobserved) acoustic feature vector $\vec{z}'$ that is formed as the addition of independent pure signal and noise vectors $\vec{x}'$ and $\vec{y}'$, i.e., $\vec{z}' = \vec{x}' + \vec{y}'$, but the actual observations are logarithms of (possibly linear transformations of) these

The authors are with the Department of Mathematics & Computer Science, University of Marburg, Hans-Meerwein-Str. 3, D-35032 Marburg, Germany, e-mail: {stadelmann, freisleb}@informatik.uni-marburg.de; fon: +49 6421-2821 568; fax: +49 6421-2821 573). This work is supported by the Deutsche Forschungsgemeinschaft (SFB/FK 615, Teilprojekt MT).

vectors ($\vec{z} = \log t(\vec{z}')$, $\vec{x} = \log t(\vec{x}')$, $\vec{y} = \log t(\vec{y}')$, where $t()$ is some linear transformation or the identity). Then, the following approximation can be used to model the signal–noise interaction in the new (transformed, logarithmized) domain to simplify and speed up subsequent modeling computations:

$$\vec{z} = \log\left(t(\vec{x}') + t(\vec{y}')\right) = \log\left(e^{\vec{x}} + e^{\vec{y}}\right) \approx \max\left(\vec{x}, \vec{y}\right) \tag{1}$$

Note that both the $\log$-function and the $\max$-function are meant to operate component-wise if used with vector arguments, i.e., (1) is a shorthand notation for all components $\{z_d | 1 \leq d \leq D\}$, of $\vec{z} \in \mathbb{R}^D$.

Consider the following concrete situation: two frames of speech signal $\vec{x}'$ and noise $\vec{y}'$ are purely additive in the time-domain. This happens, for example, when two different sound recordings are mixed together after they have been recorded, as it is done within music (singing and diverse instruments) or movies (soundtrack or effects and possibly dubbed voices), or when different sound sources are recorded with a single microphone. Therefore, signal and noise are also additive in the FFT domain (i.e., $t() = FFT()$), because the FFT is linear with respect to addition. Thus, the signal is really additive in the frequency domain. But when the power-spectrum $|\ |^2$ is then computed of some Fourier-transformed signal $a = b + c$, it yields $|a|^2 = |b|^2 + |c|^2 + 2 \cdot |b| \cdot |c|$, which can be approximated by $|a|^2 \approx |b|^2 + |c|^2$. This (approximate) additivity in the power-spectral domain remains after passing the power spectrum through a bank of (probably mel-scaled) filters. But after taking the logarithm ($\vec{z} = \log FFT(\vec{z}')$, etc.) of these filterbank energies, the signal–noise interaction function becomes $\log(e^{\vec{x}} + e^{\vec{y}})$, which is approximated by $\max\left(\vec{x}, \vec{y}\right)$ for the sake of computational simplicity.

Thus, the MixMax model is appropriate, for example, if signal and noise are additive in the time domain, but the observations are log-filterbank energy (FBE) features. The $\max()$-approximation leads to manageable mathematical expressions and good results, explaining its application to numerous problems in the audio processing domain. It also explains the name: via GMMs, mixtures of maxima of signal and noise are modeled.

III. Definition of the MixMax Model

A MixMax model λ_{MM} consists of two separate GMMs λ^s and λ^b and specialized algorithms for training and testing. It is defined as follows [5]:

$$\lambda_{MM} = \{\lambda^s, \lambda^b\} \tag{2}$$

$$\lambda^s = \{(w_i^s, \vec{\mu_i}^s, \vec{\sigma_i^2}^s) | 1 \leq i \leq I\} \tag{3}$$

$$\lambda^b = \{(w_j^b, \vec{\mu_j}^b, \vec{\sigma_j^2}^b) | 1 \leq j \leq J\} \tag{4}$$

Here, λ^s is the signal model with I mixtures and λ^b is the background model with J mixtures, each having a weight w, a mean vector $\vec{\mu}$ and a diagonal covariance matrix $\vec{\sigma}^2$ per mixture.

A. Model Training

The background model has to be trained in advance using samples of the expected noise and a standard GMM training procedure [2]. Then, training the signal model via the EM algorithm [17] and the specialized equations derived in the literature [5] [6] can be accomplished independently for each dimension, taking into account the diagonal covariance matrix of the Gaussians:

$$\overline{w_i^s} = \frac{1}{T} \sum_{t=1}^{T} \sum_{j=1}^{J} \prod_{d=1}^{D} p(i, j | z_{t,d}, \lambda_{MM}) \tag{5}$$

$$\overline{\mu_{i,d}^s} = \frac{\sum_{t=1}^{T} \sum_{j=1}^{J} \frac{E\{x_{t,d} | z_{t,d}, i, j, \lambda_{MM}\}}{p(i,j|z_{t,d},\lambda_{MM})^{-1}}}{\sum_{t=1}^{T} \sum_{j=1}^{J} p(i, j | z_{t,d}, \lambda_{MM})} \tag{6}$$

$$\overline{\sigma_{i,d}^2}^{\,s} = \frac{\sum_{t=1}^{T} \sum_{j=1}^{J} \frac{E\{x_{t,d}^2 | z_{t,d}, i, j, \lambda_{MM}\}}{p(i,j|z_{t,d},\lambda_{MM})^{-1}}}{\sum_{t=1}^{T} \sum_{j=1}^{J} p(i, j | z_{t,d}, \lambda_{MM})} - \overline{\mu_{i,d}^2}^{\,s} \tag{7}$$

where $\overline{w_i^s}$, $\overline{\mu_{i,d}^s}$ and $\overline{\sigma_{i,d}^2}^{\,s}$ are the new (reestimated) parameters of the signal GMM λ^s for the next round of the EM algorithm. D is again the dimensionality of the feature vectors $\vec{z} \in \mathbb{R}^D$ and d the index for the dimension. Note the quotient and corresponding -1 exponent in (6)–(7) for layout reasons. To apply the formulas, several other terms must be defined:

$$p(i, j | z_d, \lambda_{MM}) = \frac{p(z_d | i, j, \lambda_{MM}) \cdot w_i^s \cdot w_j^b}{\sum_{i=1}^{I} \sum_{j=1}^{J} p(z_d | i, j, \lambda_{MM}) \cdot w_i^s \cdot w_j^b} \tag{8}$$

$$E\{x_d | z_d, i, j, \lambda_{MM}\} = \frac{z_d}{p(x_d = z_d | i, j, \lambda_{MM})^{-1}} + \frac{E\{x_d | x_d < z_d, i, j, \lambda_{MM}\}}{(1 - p(x_d = z_d | i, j, \lambda_{MM}))^{-1}} \tag{9}$$

$$E\{x_d^2 | z_d, i, j, \lambda_{MM}\} = \frac{z_d^2}{p(x_d = z_d | i, j, \lambda_{MM})^{-1}} + \frac{E\{x_d^2 | x_d < z_d, i, j, \lambda_{MM}\}}{(1 - p(x_d = z_d | i, j, \lambda_{MM}))^{-1}} \tag{10}$$

Here, $z_{t,d}$ is the d^{th} dimension of the t^{th} observation vector in the transformed, logarithmized domain, while x_d is its implicit clean signal estimate in the same domain. The meaning of (9) is as follows: the expected value $E\{x | z, i, j\}$ of a clean speech component, given the noisy observation and a specific foreground–background state combination, is the weighted mean of the noisy observation z and the signal's expected value given that its amplitude is below the noisy observation's amplitude ($E\{x | x < z\}$). The weights are defined by the probability that the current observation is already a clean signal ($p(x = z)$) and its complementary event. These equations already make use of the $\max$ assumption (in the formulation of the expected value for x, which needs its amplitude being smaller than the amplitude of z), which becomes evident in the following equations:

$$p(z_d | i, j, \lambda_{MM}) = b_j(z_d) \cdot S_i(z_d) + s_i(z_d) \cdot B_j(z_d) \tag{11}$$

$$p(x_d = z_d | i, j, \lambda_{MM}) = \frac{s_i(z_d) \cdot B_j(z_d)}{b_j(z_d) \cdot S_i(z_d) + s_i(z_d) \cdot B_j(z_d)} \tag{12}$$

$$E\{x_d|x_d < z_d, i, j, \lambda_{MM}\} = \mu^s_{i,d} - {\sigma^2_{i,d}}^s \cdot \frac{s_i(z_d)}{S_i(z_d)} \tag{13}$$

$$E\{x^2_d|x_d < z_d, i, j, \lambda_{MM}\} = \left({\mu^2_{i,d}}^s + {\sigma^2_{i,d}}^s\right) - {\sigma^2_{i,d}}^s \cdot \frac{s_i(z_d)\cdot(z_d + \mu^s_{i,d})}{S_i(z_d)} \tag{14}$$

Here, $b_j()$ and $s_i()$ are the univariate parametrized Gaussian probability density functions (PDFs) $\phi(..)$ for mixtures j and i of the background- and signal GMM, respectively. $B_j()$ and $S_i()$ are the corresponding cumulative density functions (CDFs) as defined below. Note the squared form ${\sigma^2_{i,d}}^s$ in (13): this has been incorrectly given un-squared in the original paper [5] and in the subsequent literature.

$$b_j(z_d) = \phi(z_d, \mu^b_{j,d}, {\sigma^2_{j,d}}^b) \tag{15}$$

$$\phi(x, \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{16}$$

$$B_j(z_d) = \Phi\left(\frac{z_d - \mu^b_{j,d}}{\sigma^b_{j,d}}\right) \tag{17}$$

$$s_i(z_d) = \phi(z_d, \mu^s_{i,d}, {\sigma^2_{i,d}}^s) \tag{18}$$

$$S_i(z_d) = \Phi\left(\frac{z_d - \mu^s_{i,d}}{\sigma^s_{j,d}}\right) \tag{19}$$

The Gaussian CDF $\Phi()$ is defined in terms of the error function erf as follows:

$$\Phi(x) = \frac{1}{2} \cdot \left[1 + erf\left(\frac{x}{\sqrt{2}}\right)\right] \tag{20}$$

$$= \frac{1}{\sqrt{2\pi}} \cdot \int_{-\infty}^{x} \phi(x, 0, 1)dt \tag{21}$$

B. Model Evaluation

During training, the mixtures of the signal model λ^s in the individual frequency bands (dimensions) get masked by the background mixtures at the points where both distributions overlap. During testing of the combined model against evaluation data, the probability of noise corruption for each feature vector, frequency band and state (mixture) in the combined signal–background mixture lattice is computed. The higher this probability is, the less does this component contribute to the final log-likelihood score l_{MM} in (22):

$$l_{MM} = \log p(Z|\lambda_{MM}) = \sum_{t=1}^{T} \log\left(\sum_{i=1}^{I}\sum_{j=1}^{J} w^s_i \cdot w^b_j \cdot \prod_{d=1}^{D} p(z_{t,d}|i, j, \lambda_{MM})\right) \tag{22}$$

where $Z = \{\vec{z_t} | 1 \leq t \leq T \wedge \vec{z_t} \in \mathbb{R}^D\}$ is the set of evaluation feature vectors.

IV. MixMax and MFCC Feature Vectors

The MixMax model has shown its effectiveness in reducing the influence of noise in the tasks mentioned above. Nevertheless, it suffers from not using the best possible input: by design, the MixMax assumption is not appropriate for cepstral features like MFCCs that have many advantages over conventional filterbank features. For example, they are more voice specific, have a lower susceptibility to noise, are completely decorrelated and more compact. These advantages can typically cause a drop in the final error rate as high as 5–10% absolute reduction and must be left unexploited in the case of the MixMax model.

Several researchers acknowledge this constraint, e.g., Nádas et al. [1], Varga and Moore [4] and Rose et al. [5]. Nevertheless, in a series of publications on singer identification in popular music databases, Tsai et al. have reported good results using MixMax models in conjunction with MFCC feature vectors [7]–[10].

As a motivating example, consider the power envelopes depicted in Fig. 1: the good concordance of FBEs with the $\max$-assumption can be seen as well as its violation within the MFCCs. Loosely speaking, the inappropriateness of the MixMax model for MFCC features is due to the MFCC vector being the discrete cosine transform (DCT) of a FBE vector. Thus, every single component of a MFCC vector is a weighted linear combination of all components of the FBE observation ($\vec{z} = DCT(\max(\vec{x}, \vec{y}))$), such that a highly non-linear coherence between $\vec{x}$ and $\vec{y}$ through the nested call to the $\max(.)$ function is created. No good results can be expected when this relationship is ignored.

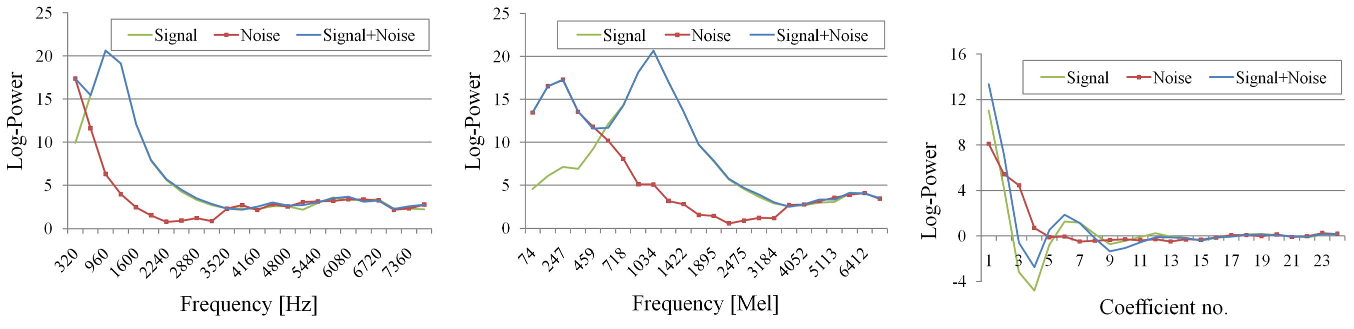
V. Explaining Good Results Using “MixMax” and MFCCs

In this section, one part of the mentioned contradiction is dissolved by explaining Tsai et al.’s good results. Our approach is to show that in fact a different model (“the actual model used”, AMU) has unawarely been applied by the authors, and to discover what this AMU looks like. Subsection V-A begins with the extraction of the AMU’s training and evaluation equations from the authors’ source code. The equations deviate strongly from the MixMax model’s formulation, and the implementation suggests that they might have evolved unintentionally. Then, Subsection V-B reports on extensive experiments comparing the results using these equations with the MixMax- and other models. The experiments allow us to draw the following conclusions:

a The actual model used by Tsai et al. in conjunction with MFCCs indeed performs significantly better than MixMax & FBE, GMM & MFCC and (of course) MixMax & MFCC on quite diverse data sets; this shows its *suitability* (to some extent) for noise compensation in the cepstral domain.

b The actual model used does not perform significantly different than a particular extension of the GMM baseline; this indicates that it is more related to this baseline extension rather than to the MixMax model.

Based on this analysis, we suggest to dissolve the contradiction in the literature by arguing that Tsai et al. seem to have used the model extracted here, but have described the MixMax model in their publications. Publishing the actual model used in the next section is meant to clarify which method actually produces good results in compensating noise



(a) FBE vectors on a linear frequency scale.
(b) FBE vectors on a Mel frequency scale.
(c) Corresponding MFCC vectors.

Fig. 1. Example of the power envelopes of FBE and MFCC vectors of some pure signal, pure noise and the corresponding combined observation.

and modeling voices in the cepstral domain. In straightens the body of literature regarding the MixMax model and its area of application: the MixMax model is not applicable in the cepstral domain.

A. The Actual Model Used

Tsai et al. thankfully provided the source code of their published singer recognition system in order to pursue the question why it shows good results in a context where it is not supposed to do so. Careful analysis revealed a set of equations for the actual model used that deviates from the MixMax equations given in (5)–(22).

Let λ_{AMU} denote the actual model used, defined as in (2)–(4). The following expressions are used to train its integrated signal model λ^s as revealed by reverse engineering:

$$\overline{\mu_{i,d}^s} = \mu_{i,d}^s \tag{23}$$

$$\overline{\sigma_{i,d}^s} = \sigma_{i,d}^s \tag{24}$$

$$\overline{w_i^s} = \frac{1}{T} \cdot \sum_{t=1}^{T} \frac{w_i^s \cdot \sum_{j=1}^{J} w_j^b \cdot p_{train}(\vec{z}_t|i,j,\lambda_{AMU})}{\sum_{u=1}^{I} w_u^s \cdot \sum_{v=1}^{J} w_v^b \cdot p_{train}(\vec{z}_t|u,v,\lambda_{AMU})} \tag{25}$$

$$\begin{aligned} p_{train}(\vec{z}_t|i,j,\lambda_{AMU}) &= \prod_{d=1}^{D} p(z_{t,d}|i,j,\lambda_{AMU}) \\ &= \prod_{d=1}^{D} \big(b_j(z_{t,d}) \cdot S_i(z_{t,d}) \\ &\qquad + s_i(z_{t,d}) \cdot B_j(z_{t,d})\big) \end{aligned} \tag{26}$$

Here, $p(z_{t,d}|i,j,\lambda_{AMU})$ is defined as in (11). The difference (apart from the domain of the observation vector, which is MFCC here) to the equations given by [5] and in Section III for the MixMax model is that the means and variances are not re-estimated, i.e., they remain as initialized prior to EM training. The expression for the log-likelihood function l_{AMU} has been determined to be

$$\begin{aligned} l_{AMU} &= \log p(Z|\lambda_{AMU}) \\ &= \sum_{t=1}^{T} \log \left(\sum_{i=1}^{I} \sum_{j=1}^{J} w_i^s \cdot w_j^b \cdot p_{eval}(\vec{z}_t|i,j,\lambda_{AMU}) \right) \end{aligned} \tag{27}$$

with

$$\begin{aligned} &p_{eval}(\vec{z}_t|i,j,\lambda_{AMU}) = \\ &\quad \left(\prod_{d=1}^{D} b_{j,d}(z_{t,d}) \right) \cdot \left(\frac{\sum_{d=1}^{D} S_{i,d}(z_{t,d})}{D} \right) \\ &\quad + \left(\prod_{d=1}^{D} s_{i,d}(z_{t,d}) \right) \cdot \left(\frac{\sum_{d=1}^{D} B_{j,d}(z_{t,d})}{D} \right) \end{aligned} \tag{28}$$

Note that different equations are used during training and evaluation to compute the "likelihood" $p_{train/eval}(\vec{z}|i,j,\lambda_{AMU})$ of the current vector to a given state of the model. The equation for $p_{eval}()$ differs from (26) in that it gives up the component-wise $\max()$ assumption. Instead, its meaning is "the probability that the components of the signal are *on the average* greater than the components of the noise or vice versa". Though this "average maximum" signal–noise interaction is also not generally true for the coherence of signal and noise in the MFCC domain, it might approximate the strongly non-linear behavior.

B. Experimentation

In this section, we

a show that the equations nevertheless point to an effective method for noise compensation in the cepstral domain as indicated by the positive results;
b give evidence about what this effective method might look like.

First, we report on the datasets used in our experiments that partly resemble those presented in Tsai et al.'s publications, but largely exceed them. Then, experiments are presented supporting the view that the AMU is in fact an effective model, before another set of experiments is performed that aims at revealing its "true" identity. All experiments follow the setup that Tsai et al. used, i.e., closed set singer/speaker identification experiments are conducted in the spirit of Reynolds [18].

1) Databases: Three different datasets are used to provide a broad basis for extensive computational simulations. Therefore, each dataset has a distinct focus: singing voice with music, spontaneous conversations or noisy telephone quality speech. In particular, the following datasets are utilized:

The DB-S-1 database introduced by Tsai et al. has been primarily designed for singer recognition experiments. It splits into the training set DB-S-1-T and the evaluation set DB-S-1-E, each consisting of a total of 100 Mandarin pop songs, 5 by each of 10 male and 10 female distinct solo singers. The data has been downsampled from CD quality to 22 kHz. Each song is between 2:15 and 6:30 minutes in length. However, in case of FBE features, only 7 male and 8 female distinct artists with 4 to 5 songs each are present in the database as provided by the authors, resulting in 72 songs per subset.

The Portuguese TV soap opera "Riscos SL" is part of the "MPEG-7 Content Set" [19]. All speech from speakers occurring more than once has been extracted, resulting in a population of 5 male and 6 female speakers in the set called MPEG7 in the rest of this section. It is further divided such that each speaker has an equal number of utterances in his/her training- and evaluation set, resulting in 3–47 seconds of training speech per speaker from 1–10 utterances (18.4 seconds in 4.3 utterances on the average) and 2–28 seconds of test data from 1–10 utterances (15.3 seconds in 4.2 utterances on the average). The speech within this database can be characterized as short, spontaneous and emotional in nature, accompanied by background noise such as speech babble and ambient sounds as well as music. This forms a challenging scenario for speaker identification experiments. The data is converted from an 44 kHz 192Kbps MPEG-1 layer II compressed audio stream to a 16 kHz waveform before further processing.

The NOIZEUS corpus has been introduced by Hu and Loizou [20] for the comparison of speech enhancement algorithms. It consists of read speech from 3 male and 3 female speakers. Each of them uttered 5 phonetically rich sentences that were later mixed with 5 different noise types from the AURORA database at 4 different SNRs from 15 dB to 0 dB. These studio-quality recordings were further processed to have telephone speech quality at 8 kHz sample rate. To use this data for voice recognition, it is split into a training and evaluation set as follows: the first two sentences of all speakers with accompanying restaurant ambient noise at 15 dB and 0 dB are used for model training, while the last 3 sentences with airport-/station-/train- and exhibition-noise at SNRs of 10 dB and 5 dB are used for testing. This way, there is no co-occurrence of sentences, SNRs or noise-types in both training and testing, making the task of speaker identification more difficult due to unforeseen circumstances.

The datasets are not proprietary and are also used by other works or are actually available to the public (in case of MPEG7 and NOIZEUS), so that the experiments are repeatable. For the purpose of noise model training, each set also contains samples of pure interfering noise, collected from the parts before, in between and after the speech in case of DB-S-1 and MPEG7, and from the pure noise samples in case of NOIZEUS.

2) Experiments Confirming the AMU's General Suitability: These experiments are designed to assess the performance of the MixMax model and the AMU on both log-filterbank energy- and cepstral features and to give evidence of their respective strengths and weaknesses. Following the setup of Tsai et al. the input data is first processed by HTK [21] to produce 20 MFCCs or 28 FBEs per frame. Each frame is preemphasized with a factor of $\alpha = 0.97$ and Hamming-windowed, with a frame length of 32 ms and a frame step of 10 ms. All voice (singer, speaker) models comprise 32 mixture components, while in case of noise models 8 mixtures are used. All models are initialized via 10 iterations of the k-means algorithm and trained using 20 iterations of the EM algorithm. As a baseline for comparison, scores for a standard GMM recognition system (EM-trained, without universal background model (UBM) score normalization) are also reported.

TABLE I
SINGER/SPEAKER IDENTIFICATION RATE ON ALL THREE DATABASES.

Features	Model	Recognition rate		
		DB-S-1 [%]	MPEG7 [%]	NOIZEUS [%]
FBE	GMM	88.89	54.35	45.83
	MixMax	91.67	56.52	64.58
	AMU	91.67	60.87	47.22
MFCC	GMM	93.00	63.04	70.13
	MixMax	75.00	39.13	68.05
	AMU	98.00	73.91	71.53

The first recognition rate column of Table I shows the results of voice recognition (in fact: closed set singer identification) on the DB-S-1 database. Several facts can be noted: looking at the performance of the GMM system with the different features, the superiority of MFCCs over FBEs can be seen. For the MixMax model, the predicted drop in recognition rate when using MFCCs is quite obvious. The AMU scores equal to the MixMax model when used with FBEs, but scores best in conjunction with MFCC features. This last result is comparable (except for small variations due to model initialization, score normalization etc.) to the one reported by Tsai et al. for the solo modeling case with automatic segmentation, validating our implementation as well as the experimental setup.

In the second recognition rate column of Table I, the results for the MPEG7 test set are reported. They are qualitatively equal to those on the DB-S-1 database, though the recognition rates are consistently shifted down by 20–30 percentage points. This may be due to very short training and evaluation utterance lengths as well as highly non-stationary noise, as reported earlier.

Finally, the results for the NOIZEUS corpus are shown in the last column of Table I. There are two differences to the previous results: The MixMax model, in combination with MFCC features, works better than with FBE features, though it is still the worst classifier on MFCCs. Also, the AMU is clearly outperformed on FBEs by the MixMax model. Again, note that the utterances here consist of only one short sentence.

In general, the new AMU & MFCC combination always performs best. Compared to the results of the formerly best combination, MixMax & FBE, an average relative improvement in identification rate of 16.14% is achieved (6.19% on DB-S-1, 30.77% on MPEG7 and 10.76% on NOIZEUS, respectively). This corresponds to an average increase of the scores as high as 10.22 percentage points. Table II gives the raw identification results for these two systems and all three databases.

These experiments support already expressed arguments: the MixMax model's inappropriateness in case of MFCCs is demonstrated by means of low recognition rates, and the general preference of cepstral features over log-filterbank energies

TABLE II
CONTINGENCY TABLE OF RAW IDENTIFICATION RESULTS ON ALL THREE DATABASES.

System	Correct ID	Wrong ID	Σ
	[#]	[#]	[#]
MixMax & FBE	185	77	262
AMU & MFCC	235	55	290
Σ	420	132	552

TABLE III
SINGER/SPEAKER IDENTIFICATION RATES FOR AMU VARIANTS AND BASELINES USING MFCC FEATURES ON ALL THREE DATABASES.

Model	Recognition rate		
	DB-S-1 [%]	MPEG7 [%]	NOIZEUS [%]
GMM (32)	93.00	63.04	70.14
GMM (40)	92.00	58.70	73.61
GMM (32/8, per dim.)	78.00	58.70	63.19
GMM (32/8, per frame)	*95.00*	*65.22*	*73.61*
MixMax (32/8)	75.00	39.13	68.05
AMU (32/8)	*98.00*	*73.91*	*71.53*
w/o eval. CDFs	94.00	69.57	71.53
w/o both CDFs	97.00	65.22	74.31
w/o both CDFs, $\vee$	97.00	65.22	74.31
w/o both CDFs, $\vee$, dim.	92.00	65.22	71.53
w/o both CDFs, $\vee$, frame	97.00	65.22	71.53

can be seen. The results are novel with regard to the AMU. Here, empirical evidence is given for a certain suitability of the specific model formulation in Section V-A in conjunction with MFCC features by means of high recognition rates in difficult voice recognition scenarios. A χ^2-test based on the values of Table II suggests that the H_0 hypothesis of these results being not significantly better than those of the MixMax & FBE approach has to be rejected with 99.5% confidence. This and the qualitative homogeneity of the results over all three highly different databases also gives evidence that the outcome is not data-dependent or random, but somewhat models the non-linear interaction of signal and noise in the transformed domain.

On the other hand, (23)–(28) or in fact the AMU's model formulation look too contrary to reason (i.e., too random) at some points. There seems to be another—yet hidden—model that still needs to be discovered, as described below.

3) Experiments Indicating the AMU's "True" Identity: A closer look at the AMU equations reveals that in the training part (23)–(26), only the weights are changed during subsequent EM iterations. Equation (25) uses (26), the probability that an observation vector at time t is reflected by the state (i, j) under the component-wise maximum assumption of signal–noise interaction, which is an unchanged adoption from the corresponding MixMax equation in (11). Two conclusions can be drawn:

a since this assumption is wrong for MFCC features, the meaning of (25) is questionable;

b since (26) is also used in the MixMax training equations for reestimating the mean- and variance-vectors, it is obvious that the omitted training of the means and variances in the AMU should be beneficial to the model's performance (in the sense of rather doing nothing than doing something wrong).

These findings directly suggest two changes in the AMU formulation with respect to training:

First, adjusting the means and variances of the model in a non noise-specific, standard GMM sense should further amplify the effect gained by leaving them as initialized by k-means (because initializing the parameters via k-means roughly clusters the training data by using a distance measure; EM training refines this clustering in a maximum likelihood sense, so reestimating $\vec{\mu_i}^s$ and $\vec{\sigma_i}^s$ via non noise-compensating equations should just improve the initialization). This direction has not yielded promising results in preliminary experiments, so it is excluded from further analysis.

Second, the use of p_{train} (26) in the reestimation of the weights (25) should be exchanged by a more suitable formulation. An option is to use (variants of) the adapted form p_{eval} (28) applied during the evaluation of an AMU. Results are reported in Table III for MFCC feature vectors and several reasonable baselines (the MixMax model, too, for comparison) and a couple of such variants, as described below:

"GMM (32)" indicates voice modeling with a 32-mixture GMM without regarding the background noise. "GMM (40)" describes the same system using 40 mixtures, thereby reaching the same number of used parameters as a model with 32 foreground- and 8 background mixtures. Thus, the statistical expressibility (in terms of number of parameters) is equal to all background-modeling techniques, and any difference in performance must be attributed to the expressive power (i.e., goodness of fit) of the specific model under consideration rather than to the model's size. "GMM (32/8, per dim.)" stands for a system comprising two separate standard GMMs, a 32-mixture GMM trained on the noisy speech samples, and an 8-mixture one trained on the pure noise. During recognition, for each dimension in each vector it is decided if it is better fitted by the noise- or the voice model, and only scores from a better fitting voice model contribute to the final likelihood; thus, it can be viewed as a non-probabilistic, on/off-like noise masking scheme per dimension. 'GMM (32/8, per frame)" makes the same decision based on a complete vector (all its dimensions). The MixMax model and AMU are already known from above.

The six variants of the AMU are all chosen with respect to finding "the original formulation" to (28), i.e., to find an improvement. Because of the mathematically questionable "average maximum" assumption expressed in the equation via the CDFs, the variants depict several approaches to reformulate the CDF part of the equation. First, "w/o eval. CDFs" describes the variant that eliminates all calls to CDFs in p_{eval}, i.e. they are replaced with a factor of 1. Second, "w/o both CDFs" stands for the variant without any CDFs in both p_{train} and p_{eval}. Third, "w/o both CDFs, $\vee$" describes the alternative that both in training and evaluation, the CDFs are omitted and the remaining two PDFs are joined not just via addition (meaning a probabilistic "or", denoted $\vee$, in the case of mutually exclusive events). Instead, the formulation models the probability that the vector $\vec{z}$ is speech or noise given that the two events are *not* mutually exclusive: $b(z) + s(z) - b(z) \cdot (z)$ (dropping all indices). The remaining difference between training and evaluation equations is now only the fact that during training, the equation correctly regards the multivariate nature of $\vec{z}$ by calculating the product over all dimensions of the diagonal-covariance Gaussians; during evaluation, however, the multivariate nature is oddly treated by building the product of the individual terms independently. This difference is resolved in the cases "w/o both CDFs, $\vee$, dim." and "w/o both CDFs, $\vee$, frame", where in the former case both training and evaluation equations work truly multivariate; in the latter case, both equations adopt the formulation of (28).

C. *Discussion*

From Table III, several conclusions can be drawn:

First, all variants explored to improve the AMU and to discover a hidden meaning fail, yielding worse results than the originally found equations. This suggests that both parts, the equations for model training and evaluation, interact in their specific form to create the good results: the training stage contributes mixture means and variances resulting from pure k-means clustering, and weights that are adjusted in a manner that tends to increase the impact of few mixtures while simultaneously dropping most others to have very low impact on the result. The likelihood computation stage is built on the assumption of signal and background interaction that has previously been called the "average maximum". It departs from the paradigm of component-wise likelihood computation by operating on the whole vectors at once (which can be implemented component-wise again, as suggested by the equations, through independence of the individual MFC coefficients). It appears that the approach of "optimizing" the AMU equations failed.

Second, Table III interestingly shows that the top-scoring original AMU formulation is not far away (in terms of identification rates) from the simple but effective "GMM (32/8, per frame)" approach (denoted as the "baseline" below). In fact, a detailed analysis of the two models' individual scores of all test utterances versus the enrolled speaker models reveals that the produced scores are very similar to each other. A simple value of concordance, c, reaches 94.70% agreement according to (29):

$$c = \frac{\sum_{t=1}^{T}\sum_{s_1=1}^{S}\sum_{s_2=s_1+1}^{S} r_{t,s_1,s_2}}{T \cdot S \cdot \frac{S-1}{2}} \qquad (29)$$

$$r_{t,s_1,s_2} = r(X_t, \lambda_{AMU}^{s_1}, \lambda_{AMU}^{s_2}, \lambda_{baseline}^{s_1}, \lambda_{baseline}^{s_2}) \qquad (30)$$

where X_t is the t^{th} feature vector set out of T test utterances and $\lambda_{AMU/baseline}^{s}$ the s^{th} enrolled speaker model out of a total of S trained models. The function $r(....)$ returns 1 if and only if the two models agree on the relative rank of two trained models (as produced by ordering them according to the achieved likelihood) for a specific test utterance.

$$r(X, \lambda_1^u, \lambda_1^v, \lambda_2^u, \lambda_2^v) = \begin{cases} 1 \quad \text{if} & \begin{array}{r} (l_1(X|\lambda_1^u) < l_1(X|\lambda_1^v) \wedge \\ l_2(X|\lambda_2^u) < l_2(X|\lambda_2^v)) \vee \\ (l_1(X|\lambda_1^u) > l_1(X|\lambda_1^v) \wedge \\ l_2(X|\lambda_2^u) > l_2(X|\lambda_2^v)) \end{array} \\ 0 \quad \text{otherwise} & \end{cases} \qquad (31)$$

Here, $l_{1/2}$ are the respective likelihood functions of the two speaker models $\lambda_{1/2}$.

TABLE IV
IDENTIFICATION SCATTER MATRIX FOR THE "GMM (32/8, PER FRAME)" MODEL ON MPEG7 DATA.

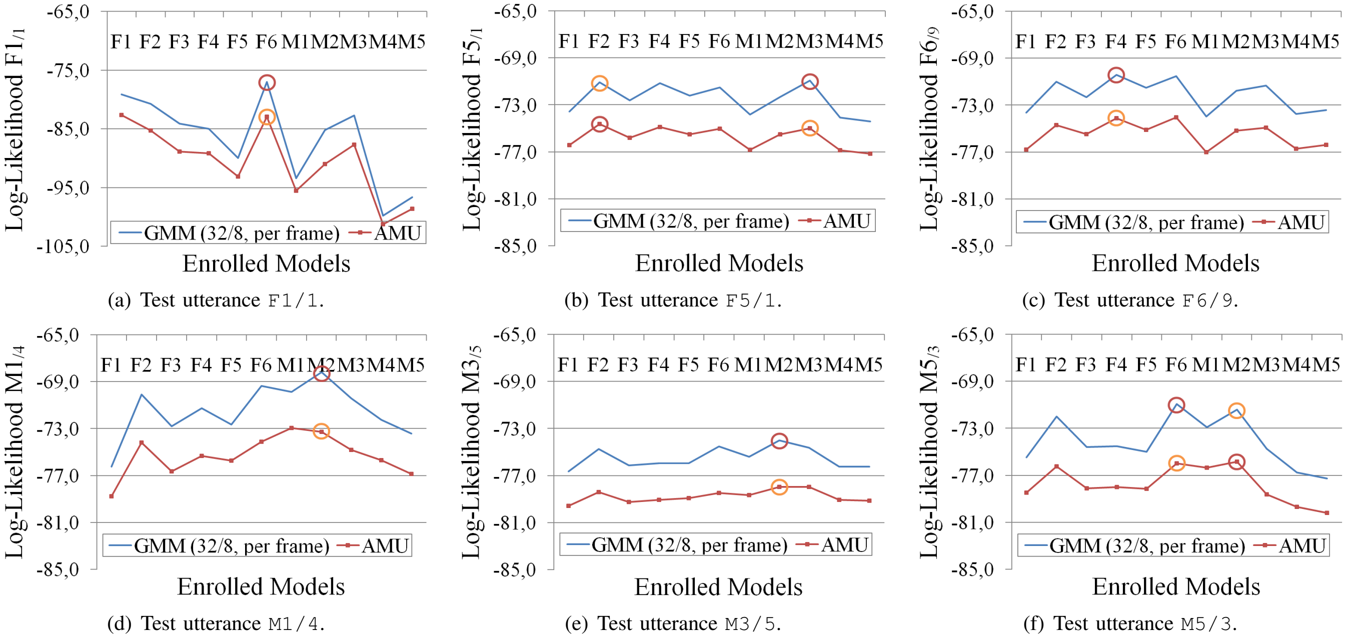
		F1	F2	F3	F4	F5	F6	M1	M2	M3	M4	M5
		[#]	[#]	[#]	[#]	[#]	[#]	[#]	[#]	[#]	[#]	[#]
F1	[#]	🟩										
F2	[#]		4							1		1
F3	[#]			2								
F4	[#]				3		2					
F5	[#]					🟩						
F6	[#]	1	3				8					
M1	[#]							3				
M2	[#]							1	7	2	1	1
M3	[#]					1				3	1	1
M4	[#]										🟩	
M5	[#]											🟩

TABLE V
IDENTIFICATION SCATTER MATRIX FOR THE AMU MODEL ON MPEG7 DATA.

		F1	F2	F3	F4	F5	F6	M1	M2	M3	M4	M5
		[#]	[#]	[#]	[#]	[#]	[#]	[#]	[#]	[#]	[#]	[#]
F1	[#]	1										
F2	[#]		4			1				1		
F3	[#]			2								
F4	[#]				3		1					
F5	[#]					🟩						
F6	[#]		3				9					
M1	[#]							4				
M2	[#]								7	1	1	2
M3	[#]									4	1	1
M4	[#]										🟩	
M5	[#]											🟩

The high agreement expressed by c as given above is further demonstrated in Tables IV–V and Fig. 2. The tables give scatter matrices for the baseline model and AMU. The numbers indicate how often utterances from specific speakers (indicated by the IDs in the column headers) are identified as coming form certain speaker models as indicated by the speaker ID in front of the rows. Correct identifications are found along the main diagonal, marked in green, while errors are individually marked in red. A coarse visual analysis of the graphical pattern created by the correct and incorrect identifications shows how similar both models work in terms of identification results and errors. This trend is further expressed in Fig. 2, where all six cases are depicted where the two models do not agree in their final identification decision on MPEG7 data. The envelopes of the likelihood scores of both models are very similar, letting the AMU's scores appear merely as scaled versions of the baseline's results. Furthermore, as indicated by the circles, in all six (out of 46 overall) cases of non-agreement, the second best score of one of the models always resembles the winner of the other model. Additionally, quite often the difference from the best to second best score is marginally small: nearly invisible departures decide over correct identification (in a nearest neighbor sense) in case of AMU and a false positive in case of the baseline several times.

Is the difference between the identification rates of the two models just random? Using a statistical χ^2 test (with and without Yates' correction for 1 degree of freedom [22]), no significant evidence speaks against the hypothesis H_0 that "the AMU does not perform differently than the baseline model".



(a) Test utterance F1/1.
(b) Test utterance F5/1.
(c) Test utterance F6/9.
(d) Test utterance M1/4.
(e) Test utterance M3/5.
(f) Test utterance M5/3.

Fig. 2. Log-likelihood scores for all misidentified test utterances from MPEG7 versus the enrolled speaker models, calculated using the AMU and baseline model. Data points encircled in red mark the highest overall score, in each case achieved by the wrong enrolled model; yellow circles mark the second-best score.

This test result is true for the combined identification results of all three databases (α-level of $\alpha = 0.610$), and also for each single database alone ($\alpha = 0.250$, $\alpha = 0.279$ and $\alpha = 0.639$ for DB-S-1, MPEG7 and NOIZEUS, respectively). Not rejecting H_0 is not in general an evidence in support of H_0, and the resulting α-levels of this particular test result are too low to be used as counter-arguments of the test's intention. But evaluating all the available facts carefully, we conclude that the AMU is best explained as being a distorted variant of the baseline approach "GMM (32/8, per frame)".

Thus, the explanation for the good results in Tsai et al.'s works is that not the MixMax, but a different model has been used by them; this different model appears deformed as extracted from their source code, but is best explained as resembling a non-probabilistic, multivariate noise masking scheme called "GMM (32/8, per frame)": for each feature vector of a test utterance, its likelihood to the voice model and to the noise model is computed; only those frames contribute to the final likelihood score of the integrated voice–noise model that are more likely to be modeled by the voice model.

VI. Proving the MixMax' Ineptness for Cepstral Features

In this section, we prove the following theorem:

Theorem 6.1: The MixMax model is inappropriate for modeling signal–noise interaction in the cepstral domain.

Proof: Let $\vec{x}$ and $\vec{y}$ be the FBE features of pure signal and pure noise as in Section II and let D be the dimensionality of these vectors, respectively. By reductio ad absurdum, it is shown that the following equation does *not* hold $\forall \vec{x}, \vec{y} \in \mathbb{R}^D$ and $\forall D \in \mathbb{N}\backslash\{0,1\}$ (for $D = 1$, it is easy to see):

$$DCT\left(\max\left(\vec{x}, \vec{y}\right)\right) = \max\left(DCT(\vec{x}), DCT(\vec{y})\right) \tag{32}$$

i.e., that the component-wise max-coherence between the FBEs does not remain after DCT computation, such that the MixMax-model is not applicable to MFCCs in general. Knowing that the k^{th} component $\overline{s_k}$, $1 \leq k \leq D$, of the DCT's resulting vector is computed as

$$\overline{s_k} = \alpha_k \cdot \sum_{d=1}^{D} s_d \cdot \cos\left[\frac{\pi}{D} \cdot k \cdot \left((d-1) + \frac{1}{2}\right)\right] \tag{33}$$

with $\vec{s}$ being the vector to be transformed and α_k a factor, the specific form (32) takes for the k^{th} coefficient (MFC coefficient, if $\vec{s}$ is a FBE vector) can now be considered:

$$\sum_{d=1}^{D} \max\left(x_d, y_d\right) \cdot c_{d,k} = \max\left(\sum_{d=1}^{D} x_d \cdot c_{d,k}, \sum_{d=1}^{D} y_d \cdot c_{d,k}\right) \tag{34}$$

where $c_{d,k} = \cos\left[\frac{\pi}{D} \cdot k \cdot \left((d-1) + \frac{1}{2}\right)\right]$ and the α_k is dropped for simplicity.

Let $D > 1$ be arbitrary and fixed, and let $x_l < y_l$ be for l arbitrary but fixed in $\{1, \ldots, D\}$ but $x_d \geq y_d \quad \forall d \in \{1, \ldots, D\}\backslash\{l\}$. With this setting, (34) becomes

$$\sum_{\substack{d=1\\ d\neq l}}^{D} x_d \cdot c_{d,k} + y_l \cdot c_{l,k} = \max\left(\sum_{d=1}^{D} x_d \cdot c_{d,k}, \sum_{d=1}^{D} y_d \cdot c_{d,k}\right) \tag{35}$$

Consider the case that $\sum_{d=1}^{D} x_d \cdot c_{d,k} > \sum_{d=1}^{D} y_d \cdot c_{d,k}$ to resolve the max()-function on the right hand side of (35).

Then, (35) becomes

$$\begin{aligned}
&\sum_{\substack{d=1\\d\neq l}}^{D} x_d \cdot c_{d,k} + y_l \cdot c_{l,k} = \sum_{d=1}^{D} x_d \cdot c_{d,k} \\
\Leftrightarrow &\sum_{\substack{d=1\\d\neq l}}^{D} x_d \cdot c_{d,k} + y_l \cdot c_{l,k} = \sum_{\substack{d=1\\d\neq l}}^{D} x_d \cdot c_{d,k} + x_l \cdot c_{l,k} \\
\Leftrightarrow & y_l \cdot c_{l,k} = x_l \cdot c_{l,k} \\
\Leftrightarrow & y_l = x_l \qquad (36)
\end{aligned}$$

This contradicts the previous postulation that $x_l < y_l$ and therefore proves that (32) does not hold in general. ■

In fact, the claim in (32) only holds in the following two unlikely cases: first, if *all* components of $\vec{x}$ are greater (smaller, equal) than *all* components of $\vec{y}$. Second, although each component x_d is related (possibly) differently to the corresponding y_d, equation (34) holds anyway. This means that summing up only components of $\vec{x}$ *or* $\vec{y}$ has always to equal the sum of a mixture of components of $\vec{x}$ *and* $\vec{y}$. Both cases are not existing in practice.

Thus, it is evident that the MixMax model is inapplicable to modeling the signal–noise interaction present in the MFCC domain.

Two attempts have been made in the past to overcome the MixMax' weakness of being confined to filterbank features: Gales and Young [23] have developed an approach where the parameters of the signal model in the MFCC domain are inversely transformed to the linear spectral domain. Here, noise masking is carried out using the noise model, and the parameters are transformed back to the MFCC domain. Mellor and Varga [24] have introduced a similar attempt, inversely transforming signal model parameters and observation vectors to the log-spectral domain for masking and back again. Both systems have the disadvantage of not directly operating on the MFCC vectors. Instead, computationally expensive bidirectional transformations or the maintenance of both MFCC- and FBE versions of the models and observations are necessary, resulting in higher memory and maintenance requirements. In the absence of a solution to these shortcomings, the method of Gales [25] is still applied today, for example, in the recent work of Tufekci et al. [26] on robust speech recognition.

VII. Conclusions

In this paper, the debate in the literature whether to use MFCC feature vectors in conjunction with the MixMax model or not has been enriched by new arguments: on the one hand, by providing a mathematical proof that shows its inappropriateness in the presented context from a theoretical point of view; on the other hand, by providing extensive experiments and a discussion explaining how published good results on MixMax & MFCC can be explained. The result of this explanation is also to explicitly report for the first time which methods really do work as part of one of the best current systems in automatic singer recognition. Additionally, a correction of the MixMax model's training equation (13) has been given.

Areas for further research lie within exploring more sophisticated methods for singing voice modeling within popular music, now that the effective baseline is explicitly known, obviously offering room for improvement.

Acknowledgments

The authors are very thankful for the discussions with Martin Schwalb, Christian Schridde, Anne Schwalb, Ralph Ewerth and Wei-Ho “Wesley” Tsai, who also provided the source code for his singer recognition experiments. The suggestions of Timothy J. Hazen, Alfred Ultsch and Eyke Hüllermeier are very much appreciated.

References

[1] A. Nádas, D. Nahamoo, and M. A. Picheny, “Speech Recognition Using Noise-Adaptive Prototypes,” *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37, pp. 1495–1503, 1989.

[2] D. A. Reynolds and R. C. Rose, “Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models,” *IEEE Trans. Speech Audio Process.*, vol. 3, pp. 72–83, 1995.

[3] D. H. Klatt, “A Digital Filterbank For Spectral Matching,” in *Proceedings of the 1^{st} IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'76).* Philadelphia, PA, USA: IEEE, April 1976, pp. 573–576.

[4] A. P. Varga and R. K. Moore, “Hidden Markov Model Decomposition of Speech and Noise,” in *Proceedings of the 15^{th} IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'90).* Albuquerque, NM, USA: IEEE, April 1990, pp. 845–848.

[5] R. C. Rose, E. M. Hofstetter, and D. A. Reynolds, “Integrated Models of Signal and Background with Application to Speaker Identification in Noise,” *IEEE Trans. Speech Audio Process.*, vol. 2, pp. 245–258, 1994.

[6] D. Burshtein and S. Gannot, “Speech Enhancement Using a Mixture-Maximum Model,” *IEEE Trans. Speech Audio Process.*, vol. 10, pp. 341–351, 2002.

[7] W.-H. Tsai, D. Rodgers, and H.-M. Wang, “Blind Clustering of Popular Music Recordings Based on Singer Voice Characteristics,” *Computer Music Journal*, vol. 28, no. 3, pp. 68–78, 2004.

[8] W.-H. Tsai and H.-M. Wang, “A Query-by-Example Framework to Retrieve Music Documents by Singer,” in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME'04).* Taipei, Taiwan: IEEE, June 2004, pp. 1863–1866.

[9] ——, “On the Extraction of Vocal-related Information to Facilitate the Management of Popular Music Collections,” in *Proceedings of the Joint Conference on Digital Libraries (JCDL'05)*, Denver, CO, USA, June 2005, pp. 197–206.

[10] ——, “Automatic Singer Recognition of Popular Music Recordings via Estimation and Modeling of Solo Voice Signals,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, pp. 330–331, 2006.

[11] M. Afify, O. Siohan, and C.-H. Lee, “Minimax Classification with Parametric Neighborhoods for Noisy Speech Recognition,” in *Proceedings of the 7^{th} European Conference on Speech Communication and Technology (Eurospeech'01).* Aalborg, Denmark: ISCA, September 2001, pp. 2355–2358.

[12] A. N. Deoras and M. Hasegawa-Johnson, “A Factorial HMM Approach to Simultaneous Recognition of Isolated Digits Spoken by Multiple Talkers on One Audio Channel,” in *Proceedings of the 29^{th} IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'04)*, vol. 1. Montreal, QC, Canada: IEEE, May 2004, pp. 861–864.

[13] B. T. Logan and A. J. Robinson, “Enhancement and Recognition of Noisy Speech Within an Autoregressive Hidden Markov Model Framework Using Noise Estimates from the Noisy Signal,” in *Proceedings of the 22^{nd} IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'97)*, vol. 2. Munich, Germany: IEEE, April 1997, pp. 843–846.

[14] A. Erell and M. Weintraub, “Filterbank-Energy Estimation Using Mixture and Markov Models for Recognition of Noisy Speech,” *IEEE Trans. Speech Audio Process.*, vol. 1, no. 1, pp. 68–76, January 1993.

[15] A. Erell and D. Burshtein, “Noise Adaptation of HMM Speech Recognition Systems Using Tied-Mixtures in the Spectral Domain,” *IEEE Trans. Speech Audio Process.*, vol. 5, no. 1, pp. 72–74, January 1997.

[16] S. B. Davis and P. Mermelstein, “Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences,” *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 28, pp. 357–366, 1980.

[17] A. Dempster, N. Laird, and D. Rubin, “Maximum Likelihood From Incomplete Data Via the EM Algorithm,” *Journal of the Royal Statistical Society, Series B*, vol. 39, pp. 1–38, 1977.

[18] D. A. Reynolds, “Speaker Identification and Verification using Gaussian Mixture Speaker Models,” *Speech Communication*, vol. 17, pp. 91–108, 1995.

[19] MPEG 7 Requirement Group, “Description of MPEG 7 Content Set,” *ISO/IEC JTC1/SC29/WG11/N2467*, 1998.

[20] Y. Hu and P. C. Loizou, “Subjective Comparison of Speech Enhancement Algorithms,” in *Proceedings of the* 31^{st} *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP’06)*, vol. 1. Toulouse, France: IEEE, May 2006, pp. 153–156.

[21] S. Young, G. Evermann, M. J. F. Gales, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK Book (for HTK Version 3.3)*. Cambridge, UK: Cambridge University Engineering Department, 2005, visited 18. March 2010. [Online]. Available: http://htk.eng.cam.ac.uk/

[22] F. Yates, “Contingency Table Involving Small Numbers and the χ^2 Test,” *Journal of the Royal Statistical Society*, vol. 1, no. 2, pp. 217–235, 1934.

[23] M. J. F. Gales and S. Young, “An Improved Approach to the Hidden Markov Model Decomposition of Speech and Noise,” in *Proceedings of the* 17^{th} *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP’92)*, vol. 1. San Francisco, CA, USA: IEEE, March 1992, pp. 233–236.

[24] B. A. Mellor and A. P. Varga, “Noise Masking in a Transformed Domain,” in *Proceedings of the* 18^{th} *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP’93)*, vol. 2. Minneapolis, MN, USA: IEEE, April 1993, pp. 87–90.

[25] M. J. F. Gales, “Model-Based Techniques for Noise Robust Speech Recognition,” Ph.D. dissertation, Cambridge University, UK, 1996.

[26] Z. Tufekci, J. N. Gowdy, S. Gurbuz, and E. Patterson, “Applied Mel-Frequency Wavelet Cofficients and Parallel Model Compensation for Noise-Robust Speech Recognition,” *Speech Communication*, vol. 48, pp. 1294–1307, 2006.