PAPER

A Time-Varying Adaptive IIR Filter for Robust Text-Independent Speaker Verification

Santi NURATCH[†], **Panuthat BOONPRAMUK**[†], *Nonmembers*, *and* **Chai WUTIWIWATCHAI**[††a)], *Member*

SUMMARY This paper presents a new technique to smooth speech feature vectors for text-independent speaker verification using an adaptive band-pass IIR filer. The filter is designed by considering the probability density of modulation-frequency components of an M-dimensional feature vector. Each dimension of the feature vector is processed and filtered separately. Initial filter parameters, low-cut-off and high-cut-off frequencies, are first determined by the global mean of the probability densities computed from all feature vectors of a given speech utterance. Then, the cut-off frequencies are adapted over time, i.e. every frame vector, in both low-frequency and high-frequency bands based also on the global mean and the standard deviation of feature vectors. The filtered feature vectors are used in a SVM-GMM Supervector speaker verification system. The NIST Speaker Recognition Evaluation 2006 (SRE06) core-test is used in evaluation. Experimental results show that the proposed technique clearly outperforms a baseline system using a conventional RelAtive SpecTrA (RASTA) filter.

key words: *speaker verification, feature smoothing, adaptive filter, Gaussian Mixture Model (GMM), Support Vector Machines (SVM)*

1. Introduction

Speaker verification aims to authenticate persons from their speech signals [1]. In the text-independent scheme, Gaussian Mixture Model (GMM) is one of the most widely used algorithms. A verification score is calculated based on the likelihood ratio of a GMM speaker model and a Universal Background Model (UBM) [2]. The UBM representing a speaker norm is trained from various speech data which cover a large set of speakers. The speaker model can be adapted from the UBM with an adaptation algorithm such as Maximum a Posteriori (MAP) [4]. Support Vector Machines (SVM) applied on GMM Supervectors [5] is also widely deployed in text-independent speaker recognition and verification.

Mel-Frequency Cepstral Coefficients (MFCCs) plus their first and second order derivatives ($MFCC+\Delta+\Delta\Delta$) [6], [7] are commonly used features. Often, the feature vector is post-processed by a RASTA filter to suppress unnecessary modulation-frequency components of the feature vector [8], followed by applying zero-mean and unit-variance normalization to compensate channel effects.

Manuscript received June 5, 2012.

Manuscript revised October 5, 2012.

[†]The authors are with the Department of Control System and Instrumentation Engineering, Faculty of Engineering, King Mungkut's University of Technology Thonburi, Thailand.

[††]The author is with National Electronics and Computer Technology Center, Thailand.

a) E-mail: chai.wutiwiwatchai@nectec.or.th

DOI: 10.1587/transinf.E96.D.699

The RASTA is an IIR band-pass filter that helps making the feature vector more robust to linear spectral distortion. Therefore, using RASTA, most of speaker verification and speaker recognition systems which operate in noisy environments can be improved [2], [8], [9]. It is known that the speech or speaker feature vector of m dimensions extracted from different speakers composes different modulation-frequency components. And most of feature vectors are extracted from the spectral magnitude and transformed by the Mel-scale or Mel-scale filter bank [18]. This results in different modulation-frequency components given by each filter in the filter bank. By the fact that a feature vector extracted from each speech frame changes over time, the RASTA or any other filters whose parameters are fixed over time may not be suitable.

In this paper, we propose a new technique to design an adaptive filter for producing robust feature vectors. The filter is IIR band-pass with cut-off frequencies adaptable based on the probability density of the modulation-frequency component of the feature vector. Furthermore, as the density of modulation-frequency components in each dimension of feature vector is different from each other, processing each dimension of the feature vector separately is expected to improve the filter performance.

The outline of this paper is as follows. In Sect. 2, we describe pre-processing and feature extraction procedures. Section 3 briefly reviews the detail of the RASTA filter and the SVM-GMM supervector for speaker verification. In Sect. 4, we explain the probability density of modulation-frequency components of the feature vector used to design a filter. Section 5 illustrates the proposed filter design algorithm, which is based on adaptive IIR. Section 6 describes experimental setup and results. A conclusion is given in Sect. 7.

2. Pre-Processing and Feature Extraction

Speech data used in this paper were provided by NIST [15]. Speech signals were recoded via several telephone channels and in many languages from both male and female speakers. To extract MFCCs, the speech signal $x[t]$ at 8-KHz sampling frequency is pre-emphasized. The pre-emphasized signal $y[t]$ is divided into overlapped frames of 25 ms long at 10 ms frame rate. The m-th speech frame $y[m,n]$ is then converted to a frequency domain signal $X[m,k]$. Its power spectral magnitude $|X[m,k]|^2$ is computed and weighted by a 27 Mel-scale filter bank to obtain $f_{mel}[m,i]$. The loga-

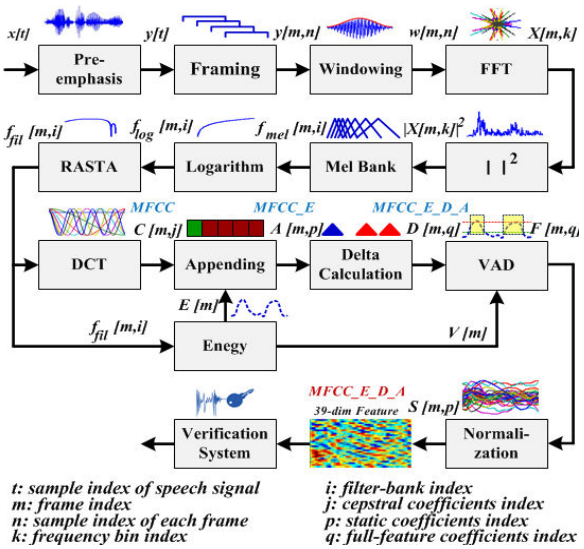

Fig. 1 A block diagram of pre-processing and MFCC feature extraction.

rithmic value $f_{\log}[m,i]$ is then filtered by a RASTA filter to suppress modulation-frequency components that spread below 0.26 Hz and above 12.8 Hz. The signal $f_{fil}[m,i]$ after RASTA is finally used to compute 12 MFCCs $C[m,j]$. A short-term energy $E[m]$ also computed from the filtered signal $f_{fil}[m,i]$ and appended to form a 13-dimensional static feature $A[m,p]$ or MFCC_E. Dynamic features, first and second derivatives, are computed from the static features over ±2 frames spanned, and attached to the static feature. Voice Activity Detection (VAD) could be applied at this stage to remove non-speech frames. The final feature vector $F[m,q]$ of the m-th frame containing 39 elements is normalized to be zero mean and unit variance. The resulting feature vector $S[m,p]$ is used in speaker verification. Figure 1 illustrates the whole pre-processing and feature extraction processes described above.

3. Review of Related Work

3.1 RASTA

In state-of-the-art signal processing for speaker recognition tasks, RASTA filtering is widely exploited with an aim to suppress spectral components that are likely out of the typical range of the human vocal tract [8], [22]. The RASTA is an IIR band-pass filter having cut-off frequencies at 0.26 Hz and 12.8 Hz. The RASTA transfer function described in Eq. (1).

$$H(z) = 0.1z^4 \times \frac{2 + z^{-1} - z^{-3} - 2z^{-4}}{1 - 0.98z^{-1}} \quad (1)$$

The spectrum of feature vector filtered by the RASTA is smoother than the non-filtered one. This phenomenon has been proven to improve the speaker verification performance. A preliminary experiment on SVM-GMM Supervector based speaker verification using male-speaker data

Table 1 Experimental results obtained from three types of feature vectors, MFCC, MFCC-RASTA and $f_{\log}$-RASTA.

Feature	*MFCC*	*MFCC–RASTA*	$f_{\log}$*–RASTA*
ERR	13.32%	10.61%	**9.47%**

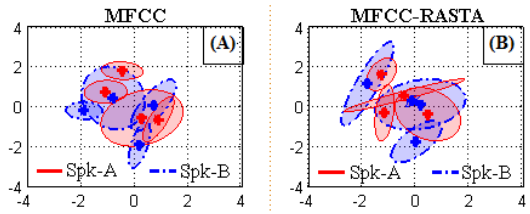

Fig. 2 A comparison of 4-mixture GMM speaker models. (A) is trained MFCC without filtering and (B) is trained by filtered MFCCs.

taken from the 2006 NIST Speaker Recognition task (SRE-06) was carried out to compare three basic features; MFCC, MFCC-RASTA and $f_{\log}$-RASTA. In MFCC-RASTA, the RASTA is applied on MFCC feature vectors whereas in $f_{\log}$-RASTA, the RASTA is applied on the log-Mel filter bank spectrum before MFCC extraction. Equal Error Rate (EER) results in Table 1 show that the RASTA applied on log-Mel filter bank spectrum is superior to that on MFCC.

A major reason why feature smoothing could improve the system performance is that standard deviations of features are decreased. In speaker recognition, reducing the feature standard deviation could make each speaker model more unique and distinguishable. Figure 2 shows a comparison between two speaker models, 4-mixture GMM, trained by MFCCs $C[m,i]$ with and without RASTA filtering. RASTA filtering obviously affects speaker models, both on their means and standard deviations. Speaker models passing RASTA filtering are more distinguishable as the standard deviations of Gaussian mixtures are reduced and the Gaussian means in each speaker model are shifted far away from each other.

Shifting means and reducing standard deviations may not always improve speaker differentiation. True speakers could be more rejected and false speakers could be more accepted by such parameter modification. Hence, the challenge of filter design hence is to suppress redundant modulation-frequency components given feature vectors with the minimal drawback on the overall system performance.

3.2 SVM-GMM Supervectors

A GMM Supervector is a vector of means of every GMM mixture arranged in one column. Given a speaker feature vector and Universal Background Model (UBM), the speaker GMM and GMM Supervectors are obtained using the MAP adaption algorithm [4]. The UBM is a GMM which is trained from a large speaker database [2].

Support Vector Machines (SVM) [19] is a two-class classifier widely applied for many classification tasks. In

GMM-SVM speaker verification, the SVM is used to classify whether the input GMM Supervector is from an underlined speaker or from an imposter.

4. Feature Vector Analysis for Filter Design

It is known that the speaker features extracted from different speakers are composed of different modulation-frequency components, and different modulation frequency ranges. Unfortunately, the RASTA has fixed cut-off frequencies (fixed pass-band) and it suppresses everything lying outside the pass-band. In practice, it is hard to identify whether each modulation-frequency component should be suppressed or reserved. In this section, we consider a modulation-frequency component density and the distribution of speaker feature vectors in order to improve the filter design.

To obtain the modulation-frequency components of the feature vector $f_{\log}[m,i]$, we apply FFT on the log Mel-scale filter bank value $f_{\log}[m,i]$ to obtain a modulation frequency distribution. The $f_{\log}[m,i]$ is then segmented to R frames at 128 samples per frame and 64 samples frame shift. Each frame is applied by a hamming window, followed by FFT and converted to a log-power spectrum. This procedure produces $f_{den}[r,i,k]$ representing R frames of the FFT power spectral density of speaker feature vector $f_{\log}[m,i]$, where r and k denotes the modulation frame index and FFT index respectively. To visualize the spectral density $f_{den}[r,i,k]$, we transform each dimension i-th of the matrix $f_{den}[r,i,k]$ to a $R\times K$ matrix, $f_{avg}[r,k]$, by averaging over all i-th dimensions as

$$f_{avg}[r,k] = \frac{1}{I}\sum_{i=1}^{I} f_{den}[r,i,k] \tag{2}$$

where, I is the number of filters in the Mel-scale filter bank. The average matrix $f_{avg}[r,k]$ is used to sketch a two dimensional spectral density as shown in Fig. 3, male speakers on the left side and female speakers on the right side. The horizontal dash line in each picture denotes the low-cut-off frequency, 12.8 Hz, of the RASTA.

Figure 3 shows that modulation-frequency components of male and female speakers are somewhat different. The low-frequency components of male speakers are observably stronger and wider than those of female. Also the frequency components of both genders can spread above and below the cut-off frequency of the RASTA and can change over time.

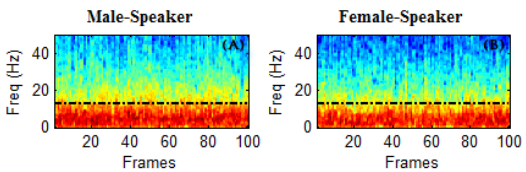

Fig. 3 The averaged modulation-frequency density of male (A) and female (B) speakers. The speakers were requested to speak (read) the same sentences, three minutes long approximately.

Therefore, RASTA and other existing filters which have a fixed pass-band may not be always suitable. The major idea of this work is that feature filtering could be more effective if the filter is specifically designed for each dimension of the feature vector of each particular speaker and is adapted properly at every speech frame.

For more clearly and confidently, we analyze the modulation-frequency component density by using 50 male speakers and 50 female speakers. Each gender is separately examined. To achieve this task, the log-power spectral density $f_{den}[r,i,k]$ is applied by the average method to form the average modulation-frequency density in each i-th dimension, represented by $f_{avg}[i,k]$:

$$f_{avg}[i,k] = \frac{1}{R}\sum_{r=1}^{R} f_{den}[r,i,k] \tag{3}$$

where R is a number of modulation frames, r is a modulation frame index, i is a dimension index, and k is FFT index.

The Eq. (3) looks like the Eq. (2), but it computes the average modulation-frequency density over R modulation frames instead of over I dimensions. The Eq. (2) explains how the modulation-frequency components change over time, while the Eq. (3) explains the variation of the modulation-frequency components in different dimensions. Figure 4 shows the average modulation-frequency density, all dimensions and all modulation-frequency components, of 50 male and 50 female speakers computed by the Eq. (3). In this research, the analysis frame rate is equal to 100 Hz, so that the maximum modulation frequency is equal to 50 Hz, a half of the analysis frame rate.

The modulation-frequency components of different genders and different dimensions of feature vectors are obviously different. The density could also change in different context. This implies that filters should be designed specifically for each feature dimension, speaker, and spoken context. Another observation is that the modulation-frequency components density of feature vectors, both averaged over all feature dimensions $f_{avg}[r,k]$ or over modulation frames

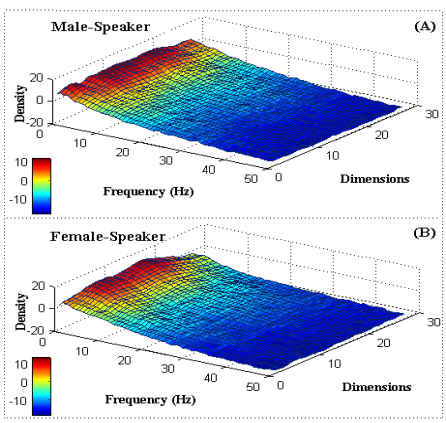

Fig. 4 The average modulation-frequency density of the 27-dimensional feature vector computed from 50 male speakers (A) and 50 female speakers (B).

$f_{avg}[i,k]$, that spread over 30 Hz are very low and hard to distinguish. Therefore, high modulation-frequency components could be suppressed with not much effect to the verification performance.

5. Adaptive IIR Filter Design

As explained in the previous section, modulation-frequency components of speech feature vectors change over time and thus it is difficult to design a fixed-parameter filter that works well on all speech frames. Therefore, an adaptive filter in which cut-off parameters could be changed optimally given a feature vector is motivated. The adaptive filter employs a statistical method to estimate useful modulation-frequency components of a given feature vector.

Normally, the pass-band of the RASTA filter is from 0.26 Hz to 12.8 Hz [8], [22]. As illustrated in the Fig. 3 and Fig. 4, modulation-frequency components of feature vector useful for speaker recognition may spread outside the pass-band of the RASTA. To ensure that useful modulation-frequency components are retained, we enlarge the frequency band being considered to 0.1 to 25 Hz. In the proposed adaptive filter, its low-pass cut-off frequency will be constrained to be within the frequency range 10 to 25 Hz and its high-pass cut-off frequency will be within 0.1 to 0.5 Hz. These constrained frequency ranges, called hereafter *Analysis-range-high* and *Analysis-range-low* respectively, could provide filter cut-off frequencies that are corresponding to that proposed by van Vuuren and Hermansky [13]. It is noted that the proposed adaptive filter is a band-pass filter constructed by cascading a low-pass filter (LPF_i) and a high-pass filter (HPF_i). The low-pass and high-pass cut-off frequencies will be separately designed.

To specify a proper adaptation band of the filter, cut-off frequencies are initialized by using a normal distribution of all frame vectors in a given speech utterance as being described in the following subsections.

5.1 Determining the Low-Pass Cut-Off Frequency of the LPF_i

Step 1: Normalizing the feature vector X_i using the following equation:

$$\hat{X}_i = \frac{X_i}{\max_i |X_i|} \tag{4}$$

where $\hat{X}_i$ is a normalized vector, and $|\cdot|$ is an absolute operator.

Step 2: Applying zero-mean normalization to the $\hat{X}_i$ in order to remove a DC-offset by using the following equation

$$\tilde{X}_i = \hat{X}_i - \left(\frac{1}{M}\sum_{i=1}^{M}\hat{X}_i\right) \tag{5}$$

where $\tilde{X}_i$ is the zero-mean normalized vector and M is the number of speech frames.

Step 3: Dividing the sequence of normalized vectors $\tilde{X}_i$ into Q blocks. Each block has N samples and $N/2$ samples block shift. In this research, the block size N is equal to 128. The i-th feature dimension of the q-th block is denoted by a sequence $Z_{i,q}$ as

$$Z_{i,q} = [\tilde{x}_{i,q,1},\ \tilde{x}_{i,q,2},\ \ldots,\ \tilde{x}_{i,q,N}] \tag{6}$$

Step 4: Computing a log power-spectral magnitude of the vector $Z_{i,q}$ by applying FFT:

$$P_{i,q,k} = 20.\log\left|\sum_{n=0}^{N-1} z_{i,q,n}.e^{\frac{-i2\pi kn}{N}}\right| \tag{7}$$

where $k = 0, 1, 2, \ldots, N-1$ is a modulation frequency index. The only half part of the $P_{i,q,k}$, $k = 0, 1, 2, \ldots, N/2-1$, is retained and normalized to $\hat{P}_{i,q,k}$ by the following equation:

$$\hat{P}_{i,q,k} = \frac{1}{K}P_{i,q,k} \tag{8}$$

where K is the number of modulation frequency elements, the half part of the $P_{i,q,k}$.

Step 5: Computing a mean vector $U_{i,k}$ of all Q blocks for each feature dimension i-th and frequency index k using the following equation:

$$U_{i,k} = \frac{1}{Q}\sum_{q=1}^{Q}\hat{P}_{i,q,k} \tag{9}$$

The Eq. (8) yields the mean vector of all blocks which

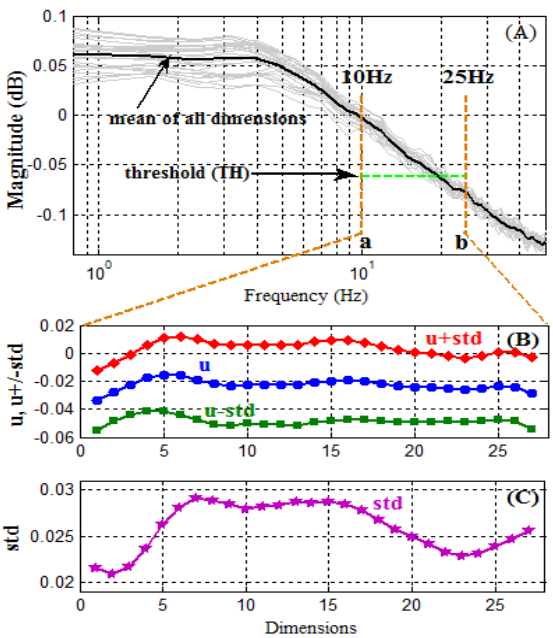

Fig. 5 (A) is the normalized mean magnitudes of modulation-frequency density of the 27-dimensional feature vectors, computed from a speech utterance. (B) the circle-doted line is the mean values of each dimension, the diamond-doted line is the mean values plus std, and the square-doted line is the mean values minus std. (C) shows the standard deviation (std) values of each dimension.

represent the modulation-frequency component density of the i-th feature. Figure 5 shows a plot of magnitudes of features over a modulation frequency axis. The mean vectors $U_{i,k}$ are illustrated as the gray lines. It is noted that the modulation frequency index k can be transformed to an actual modulation frequency f_k in Hz by

$$f_k = \frac{kf_s}{2K} \tag{10}$$

where f_s is the frame rate.

Step 6: Computing a decision threshold TH, which is used to classify between useful and useless samples of the $U_{i,k}$. First, a mean vector $\hat{U}_k$ over all i-th dimensions is computed by the following equation:

$$\hat{U}_k = \frac{1}{I}\sum_{i=1}^{I} U_{i,k} \tag{11}$$

where I is the vector dimension. Second, a decision threshold TH is computed within the *Analysis-range-high* area, i.e. between the a and b interval shown in the Fig. 5(A).

$$TH = \frac{1}{b-a+1}\sum_{k=a}^{b} \hat{U}_k \tag{12}$$

The mean vector $\hat{U}_k$ and the decision threshold TH are also shown in the Fig. 5(A). Finally, useful components are chosen by

$$\tilde{U}_{i,\vartheta} = U_{i,\tilde{k}}, \quad a \le \tilde{k} \le b \text{ and } U_{i,\tilde{k}} > TH \tag{13}$$

The $\tilde{U}_{i,\vartheta}$ is a mean vector of each feature order, containing only $U_{i,k}$ lying within *Analysis-range-high*, and having their magnitudes higher than the threshold TH. It is noted that if no element meets the condition, the maximum value of the vector $U_{i,\tilde{k}}$ will be chosen as

$$\tilde{U}_{i,\vartheta} = \max_{a\le\tilde{k}\le b} U_{i,\tilde{k}} \tag{14}$$

Step 7: Computing a mean μ_i and a standard deviation σ_i of the $\tilde{U}_{i,\vartheta}$ over all useful elements by:

$$\mu_i = \frac{1}{\zeta}\sum_{\vartheta=1}^{\xi} \tilde{U}_{i,\vartheta}, \quad \sigma_i = \sqrt{\frac{1}{\zeta}\sum_{\vartheta=1}^{\xi} (\tilde{U}_{i,\vartheta} - \mu_i)^2} \tag{15}$$

where ζ is the number of elements in the $\tilde{U}_{i,\vartheta}$. With the μ_i and σ_i, vectors $\mu_i - \sigma_i$ and $\mu_i + \sigma_i$ can be computed. These vectors are shown in Fig. 5 (B). The area between these two vectors is used to specify an adaptive region of the adaptive filter, which can be changed in every frame.

Step 8: Computing the modulation frequency index vector ψ_i using the following equation.

$$\Psi_i = a + \sum_{\tilde{k}=a}^{b} \phi_{i,\tilde{k}} \tag{16}$$

where $\phi_{i,\tilde{k}}$ is a vector containing binary values (0 or 1), which is obtained by an equation:

$$\phi_{i,\tilde{k}} = \begin{cases} 1\ ; & U_{i,\tilde{k}} > \mu_i \\ 0\ ; & otherwise \end{cases} \tag{17}$$

where $\tilde{k}$ indexes only from a to b.

Step 9: Transforming the mean vector ψ_i to an actual modulation frequency vector $f_i^{(\mu)}$ by:

$$f_i^{(\mu)} = \frac{Kf_s}{2\psi_i} \tag{18}$$

where K is a half of the FFT size ($N/2$) and f_s is a frame rate. Each element of the $f_i^{(\mu)}$ vector represents an actual modulation frequency in Hz, which is used as an initial cut-off frequency of the filter LPF_i. Two more actual modulation frequency vectors, $f_i^{(\sigma-)}$ and $f_i^{(\sigma+)}$, are computed in the same way as $f_i^{(\mu)}$ (Step 8 and Step 9) with substitution of μ_i by $\mu_i - \sigma_i$ and $\mu_i + \sigma_i$ respectively. The transformed modulation frequency vectors are shown in Fig. 6.

Each element in the vector $f_i^{(\mu)}$ is an initial low-pass cut-off frequency of each LPF_i filter. During filter adaptation, the low-pass cut-off frequency of each feature dimension can be changed within the adaptation band from $f_i^{(\sigma-)}$ to $f_i^{(\sigma+)}$.

Step 10: Designing LPF_i filters is based on a second order IIR low-pass filter. The numerator and denominator coefficients of the filters are obtained by the Butterworth algorithm [24]. Each element of the vector $f_i^{(\mu)}$ is use as an input parameter of the Butterworth algorithm to obtain a numerator $B_i^{(\mu)}$ and the denominator $A_i^{(\mu)}$. Similarly, the vectors $f_i^{(\sigma-)}$ and $f_i^{(\sigma+)}$ are taken by the algorithm to obtain $B_i^{(\sigma+)}$, $A_i^{(\sigma+)}$, $B_i^{(\sigma-)}$, and $A_i^{(\sigma-)}$.

5.2 Determining the High-Cut-Off Frequency of the HPF_i

As mentioned in the early of Sect. 4, the high-pass cut-off frequency is allowed to be adapted within the range 0.1 to 0.5 Hz. It is known that a FFT frequency resolution can be computed as:

$$f_{res} = f_s/N \tag{19}$$

where f_{res} is the frequency resolution of the FFT, f_s is the

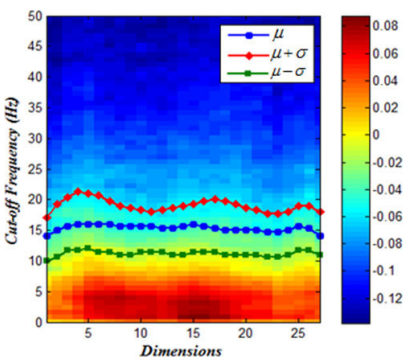

Fig. 6 Modulation-frequency component densities of the feature vectors $f_i^{(\mu)}$, $f_i^{(\sigma-)}$, $f_i^{(\sigma+)}$ and the vectors μ_i, $\mu_i - \sigma_i$, $\mu_i + \sigma_i$.

frame rate and N is the FFT size. In this work, $f_s = 100$ and $N = 128$ so the f_{res} is equal to 0.781 Hz, which is higher than the allowable adaptation band. Therefore, the high-pass cut-off frequency of the HPF_i can only be computed from the first bin of the FFT value, i.e. the first element of $U_{i,k}$ with $k = 1$. An initial high-pass cut-off frequency of the HPF_i is determined by the following steps.

Step 1: Computing a mean $\hat{\mu}$ and a standard deviation $\hat{\sigma}$ of the first element of $U_{i,k}$ over all feature dimensions.

$$\hat{\mu} = \frac{1}{I}\sum_{i=1}^{I} U_{i,1}, \quad \hat{\sigma} = \sqrt{\frac{1}{I}\sum_{i=1}^{I}(U_{i,1} - \hat{\mu})^2} \tag{20}$$

Step 2: Computing $\hat{\mu} - \hat{\sigma}$ and $\hat{\mu} + \hat{\sigma}$, which define an allowable adaptation band for the high-pass cut-off frequency as illustrated in Fig. 7.

Step 3: Computing an initial cut-off frequency of HPF_i filters by using linear interpolation as:

$$\hat{f}_i = \hat{f}_{\min} + \left[\frac{(\wp_2 - U_{i,1})(\hat{f}_{\max} - \hat{f}_{\min})}{(\wp_2 - \wp_1)}\right] \tag{21}$$

where $\hat{f}_i$ is a vector containing initial cut-off frequencies of the HPF_i, $\hat{f}_{\min}$ and $\hat{f}_{\max}$ are the lower-bound and upper-bound of the adaptation frequency, and the $\wp_1$ and $\wp_2$ represent the values $\hat{\mu} - \hat{\sigma}$ and $\hat{\mu} + \hat{\sigma}$ respectively. The Eq. (21) may produce some frequency values that lie outside the desired adaptation band if $U_{i,1}$ is much different from $\hat{\mu} \pm \hat{\sigma}$. To prevent this problem, each element in $\hat{f}_i$ having value lower than $\hat{f}_{\min}$ or higher than the $\hat{f}_{\max}$will be substituted by $\hat{f}_{\min}$ or $\hat{f}_{\max}$ respectively.

Step 4: Similar to the LPF_i, the HPF_i is designed based on second-order IIR high-pass filters using the Butterworth algorithm. Filter parameters including $\hat{B}_i^{(\mu)}$, $\hat{A}_i^{(\mu)}$, $\hat{B}_i^{(\sigma-)}$, $\hat{A}_i^{(\sigma-)}$, $\hat{B}_i^{(\sigma+)}$, and $\hat{A}_i^{(\sigma+)}$ are computed.

Finally, the initial cut-off frequencies of the LPF_i and HPF_i filters are used to form initial band-pass filters BPF_i.

5.3 Filter Adaptation over Time

Cut-off frequencies of LPF_i and HPF_i that form the proposed band-pass filter are updated at every m-th speech frame. To reduce computation, the normalized log power spectral magnitude $\hat{P}_{i,q,k}$ described in the Eq. (8) is reused in this step. Filter adaptation will be performed at every block q-th. At the q-th block, we repeat the Steps 5 to 9 described in Sect. 5.1, with replacing $U_{i,k}$ by $\hat{P}_{i,q,k}$. After the Step 9, we will get $f_i^{(\mu)}$which is denoted this time as $\aleph_{i,q}$. The $\aleph_{i,q}$ vector represents modulation-frequency components of feature vectors at the q-th block, which are used to adjust the low-pass cut-off frequency in the LPF_i. This task can be done by changing filter numerator and denominator coefficient vectors, $B_{i,q}$ and $A_{i,q}$, using linear interpolation. The numerator vector is updated by the following equation:

$$\tilde{B}_{i,q} = B_{i,q}^{(\sigma-)} + \left(\frac{\Delta f_{i,q} \times \Delta B_{i,q}}{f_i^{(\sigma+)} - f_i^{(\sigma-)}}\right) \tag{22}$$

where the $\Delta f_{i,q}$ and $\Delta B_{i,q}$ are defined as:

$$\Delta f_{i,q} = f_1^{(\sigma+)} - \aleph_{i,q}, \quad \Delta B_{i,q} = B_{i,q}^{(\sigma+)} - B_{i,q}^{(\sigma-)} \tag{23}$$

Similarly, a new denominator vector can be computed by the Eq. (22) with replacing $B_{i,q}$ by $A_{i,q}$.

$$\tilde{A}_{i,q} = A_{i,q}^{(\sigma-)} + \left(\frac{\Delta f_{i,q} \times \Delta A_{i,q}}{f_i^{(\sigma+)} - f_i^{(\sigma-)}}\right) \tag{24}$$

where $\Delta A_{i,q} = A_{i,q}^{(\sigma+)} - A_{i,q}^{(\sigma-)}$. Numerator and denominator coefficients of the HPF_i can be updated in the same way. During updating, the current frequency of the HPF_i can be computed by replacing the $U_{i,1}$ in the Eq. (21) by $U_{i,1,q}$.

$$\hat{f}_{i,q} = \hat{f}_{\min} + \left[\frac{(\wp_2 - U_{i,1,q})(\hat{f}_{\max} - \hat{f}_{\min})}{\wp_2 - \wp_1}\right] \tag{25}$$

Here, the modulation-frequency mean and standard deviation are assumed to be 20 Hz and 5 Hz for the LPF_i and 0.3 Hz and 0.2 Hz for the HPF_i. The adaptation ranges thus span from 15 to 25 Hz for the LPF_i and from 0.1 to 0.5 Hz for the HPF_i.

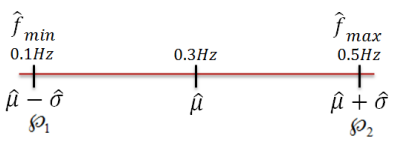

Fig. 7 Relationship of the adaptation band of the high-pass cut-off frequency and the $\hat{\mu} \pm \hat{\sigma}$ values.

6. Experimental Setup and Results

6.1 Experimental Setup

Experiments were performed on the core test of the NIST-SRE2006 evaluation (1-side training, 1-side testing, and all trials) [15]. Data from male and female speakers were separately evaluated. The UBM was trained by 2.5 minute speech utterances selected from the SWITCHBOARD corpus. This data set covered various kinds of telephone channel. Combined with imposter speaker utterances, this data set was also used to train the SVM. 2,091 speech utterances from the NIST-SRE2004 corpus were selected for the Nuisance Attribute Projection (NAP) [19] training data. The 1-side training data in the NIST-SRE2005 was used to train cohort models in the Tnorm approach [29]. An energy based Voice Activity Detection (VAD) [29] is used to detect and eliminate non-speech feature vectors. The 512 Gaussian mixture UBM was trained by using the EM algorithm. Speaker models obtained in this training phase were adapted from the UBM by MAP adaptation [4] provided in the Hidden Markov Toolkit (HTK) [16], with only mean vectors adapted. In SVM training and testing phases, GMM Supervectors composed of the means of speaker GMMs were used as input for SVM training. SVM processes were implemented by using the SVMTorch tool [30].

Table 2 Experimental results of male speaker data.

TYPE	EER			DCF		
	NoNorm	Tnorm	TZnorm	NoNorm	Tnorm	TZnorm
RASTA	10.150	8.408	**6.369**	0.099	0.079	**0.061**
AIIR-A	9.299	7.197	**5.287**	0.090	0.068	**0.047**
AIIR-B	8.089	6.167	**4.841**	0.079	0.061	**0.042**

Table 3 Experimental results of female speaker data.

TYPE	EER			DCF		
	NoNorm	Tnorm	TZnorm	NoNorm	Tnorm	TZnorm
RASTA	11.050	9.618	**7.193**	0.109	0.094	**0.068**
AIIR-A	9.333	7.543	**5.860**	0.090	0.073	**0.055**
AIIR-B	8.788	7.452	**5.159**	0.086	0.072	**0.046**

6.2 Experimental Results

The aim of evaluations was to compare the effectiveness of the RASTA filter and our proposed adaptive filter. Two types of the proposed adaptive filter were prepared: "AIIR-A" and "AIIR-B". The pass-band of AIIR-A is computed from a global modulation-frequency distribution of the whole feature vectors, while the pass-band of AIIR-B is computed and updated continuously at every speech frame. Comparing these two adaptive filters will show the benefit of filter adaptation over time as described in Sect. 5. Experimental results in term of Equal Error Rate (EER) and Detection Cost Function (DCF) [31] were obtained by the SVM-GMM Supervector based speaker verification system [23] with Tnorm and TZnorm score normalization [7]. The DCF was computed by using the following equation:

$$DCF = (C_{fr} \times p_{fr} \times P_{tar}) + (C_{fa} \times p_{fa} \times P_{imp}) \quad (26)$$

where p_{fr} and p_{fa} are a false rejection rate and a false acceptance rate at an operating point. C_{fr} and C_{fa} are costs for false rejection and false acceptance. P_{tar} and P_{imp} are the prior probabilities of target-speaker trials and impostor trials. P_{tar} and P_{imp} were set to 0.01 and 0.99 respectively [31]. The lower the DCF, the higher the system performance.

Table 2 and Table 3 conclude the experimental results evaluated separately on male and female data. The results indicate that both AIIR-A and AIIR-B filters outperform the conventional RASTA filter. The normalization techniques, Tnorm and TZnorm, are still important for improving the system accuracy.

The best results of each filter are obtained after TZnorm. Detection Error Trade-Off (DET) curves [27], only by TZnorm, are shown in Fig. 8 and Fig. 9 for male and female speakers respectively. The results clearly show that our proposed filters, both AIIR-A and AIIR-B, can improve the speaker verification performance over the conventional RASTA filter. As the cut-off frequencies of RASTA are defined by fixed-values without considering an actual modulation-frequency distributions of speaker feature vectors, some useful information may be suppressed while some useless information are not attenuated. The cut-off frequencies of AIIR-A designed by considering the global distribution of modulation-frequency components of feature

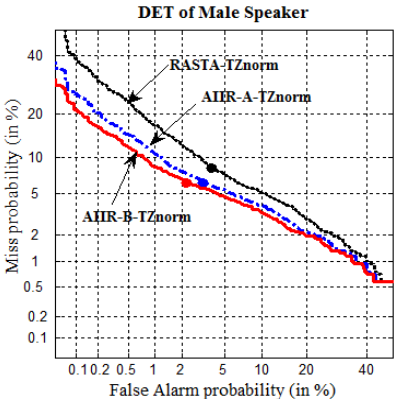

Fig. 8 DET curve of Male speakers.

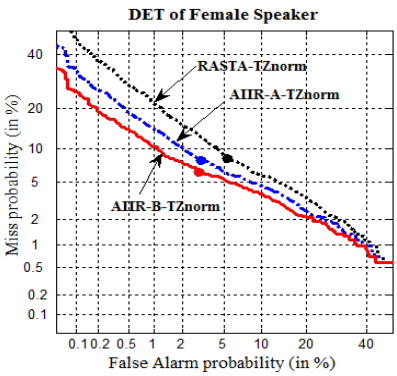

Fig. 9 DET curve of Female speakers.

vectors hence help improving the filtering capability. It is known that the speech features are time-varying. Applying the AIIR-B filter whose parameters are adapted properly at every speech frame can give a better result over the fixed pass-band AIIR-A.

Finally, as the objective of the proposed method is to relax filter cut-off frequencies regarding the feature vector being processed, dynamic or delta parameters with an appropriate window [13] could be able to provide a similar result. According to a preliminary experiment on the male speaker set, including delta parameters with a 3-frame window with no score normalization gained a 10.09 EER which was slightly lower than that of the baseline RASTA system, but still higher than those produced by the proposed AIIR-A and AIIR-B methods. One possible reason is that the proposed methods allow finer adjustment of the filter parameters. This issue will be extensively explored in our future work.

7. Conclusion

In this paper, a new filter design technique based on adaptive IIR filtering was proposed for improving the robustness of text-independent speaker verification. Each of modulation-frequency components of the speech feature vector extracted from each speaker was analyzed separately and the adaptive

IIR filter was designed based on the analysis results. Filter cut-off frequencies, both at low-pass and high-pass, were initialized based on the statistical distribution of features and were updated at every speech frame. By the proposed technique, important modulation-frequency components of the speech feature were optimally preserved at every time frame. According to the experimental results, the proposed technique clearly helped improving the overall verification performance, compared with the baseline system using the RASTA filter. Conventional score normalization techniques such as Tnorm and TZnorm were still applicable in the proposed system.

Acknowledgments

The authors would like to thank Prof. Dr. Li Haizhou, Dr. Ma Bin, and Dr. Zhu Donglai, Institute of Infocomm Research (I^2R), who provided useful tools and resources for this research. This work has been partly granted by NSTDA TGIST 2009-2011.

References

[1] D.A. Reynolds, T.F. Quatieri, and R.B. Dunn, "Speaker verification using adapted Gaussian mixture models," Digital Signal Processing, vol.10, pp.19–41, Jan. 2000.
[2] D.A. Reynolds, W. Campbell, T. Gleason, C. Quillen, D. Sturim, P. Torres-Carrasquillo, and A. Adami, "The 2004 MIT Lincoln laboratory speaker recognition system," Proc. ICASSP 2005, pp.122–131.
[3] B. Buser, I. Guyon, and V. Vapnik, "A training algorithm for optimal margin classifier," Proc. 5th Annual ACM Workshop on Computational Learning Theory, pp.144–152, 1992.
[4] V. Vapnik, The Nature of Statistical Learning Theory, Springer-Verlag, 1995.
[5] W.M. Campbell, D.E. Sturim, and D.A. Reynolds, "Support vector machine using GMM supervector for speaker identification," IEEE Signal Process. Lett., vol.23, no.5, pp.308–311, 2006.
[6] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," IEEE Trans. Speech Audio Process., vol.ASSP-28, no.4, pp.357–366, 1980.
[7] F. Bimbot, J. Bonastre, C. Fredouille, G. Gravier, I. Magrin-Chagnolleau, S. Meignier, T. Merlin, J. Ortega-Garcia, D. Petrovska-Delacrétaz, and D. A. Reynolds, "A tutorial on text-independent speaker verification," EURASIP J. Adpplied Signal Processing, pp.430–451, 2004.
[8] H. Hermanksy and N. Morganm "RASTA Processing of Speech," IEEE Trans. Speech Audio Process., vol.2, no.4, pp.578–589, 1994.
[9] J.L. Shen, W.L. Hwang, and L.S. Lee, "Robust speech recognition features based on temporal trajectory filtering of frequency band spectrum," Proc. ICSLP, pp.881–884, 1996.
[10] S. Furui, "Cepstral analysis techniques for automatic speaker verification," IEEE Trans. Acoust. Speech Signal Process., vol.ASSP-29, no.2, pp.254–272, 1981.
[11] D. Reynolds and R. Rose, "Robust text independent speaker identification using Gaussian mixture models," IEEE Trans. Speech Audio Process., vol.3, no.1, pp.72–83, Jan. 1995.
[12] T.F. Quatieri, Discrete-Time Speech Signal Processing: Principles and Practice, Prentice-Hall, Upper-Sadder River, NJ, 2002.
[13] S. van Vuuren and H. Hermansky, "On the importance of components of the modulation spectrum for speaker verification," Proc. ICSLP, vol.7, pp.3205–3208, Sydney, Australia, May 1998.
[14] L. Burget, "Analysis of feature extraction and channel compensation in a GMM speaker recognition system," IEEE Trans. Audio Speech Language Process., vol.15, no.7, pp.1979–1986, 2007.
[15] "The NIST speaker recognition evaluation," http://www.itl.nist.gov/iad/mig/tests/spk/, 2009.
[16] Hidden Markov Toolkit, http://htk.eng.cam.ac.uk/
[17] Ronan Collobert and Samy Bengio, "SVMTorch: Support vector machines for larg-scale regression problems," J. Mach. Learn. Res., vol 1, pp.143–160, 2001.
[18] L. Rabiner and B.-H. Juang, Fundamentals of Speech Recognition, Prentice-Hall, Englewood Cliffs, NJ, 1993.
[19] W.M. Campbell, D. Sturim, and D.A. Reynolds, "SVM based speaker verification using a GMM supervector Kernel and NAP variability compensation," Proc. ICASSP, pp.1969–1976, 2006.
[20] B. Buser, I. Guyon, and V. Vapnik, "A training algorithm for optimal margin classifier," Proc. 5th Annual ACM Workshop on Computational Learning Theory, pp.144–152, 1992.
[21] K. Chen, L. Wang, and H. Chi, "Methods of combining multiple classifiers with different features and their applications to text-independent speaker identification," Int. J. Pattern Recognit. Artif. Intell., vol.11, no.3, pp.417–445, 1997.
[22] H. Hermansky, N. Morgan, A. Bayya, and P. Kohn, "RASTA-PLP speech analysis technique," Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP'92), vol.1, pp.121–124, 1992.
[23] W.M. Campbell, D. Sturim, and D.A. Reynolds, "SVM based speaker Verification using a GMM supervector kernel and NAP variability compensation," Proc. ICASSP, 2006.
[24] P.P. Vaidyanathan, "Robust digital filter structures," in Handbook for Digital Signal Processing, ed. S.K. Mitra, and J.F. Kaiser, John Wiley & Sons, New York, 1993.
[25] L.B. Jackson, Digital Filters and Signal Processing, 3rd ed., Kluwer Academic Publishers, Boston, 1996.
[26] S.K. Mitra, Digital Signal Processing: A Computer-Based Approach, McGraw-Hill, New York, 1998.
[27] A. Martin, G. Doddington, T. Kamm, M. Odowski, and M. Przybocki, "The DET curve in assessment of detection task performance," Proc. Eurospeech, pp.1895–1898, 1997.
[28] R. Auckenthaler, M. Carey, and H. Lloyd-Thomas, "Score normalization for text-independent speaker verification system," Digital Signal Process., vol.10, pp.42–54, 2000.
[29] T. Kinnunen and H. Li, "An overview of text-independent speaker recognition: From feature to supervectors," Speech Commun., vol.52, pp.12–40, 2010.
[30] R. Collobert and S. Bengio, "SVMTorch: Support vector machines for large-scale regression problems," J. Mach. Learn. Res., vol 1, pp.143–160, 2001.
[31] "The NIST Year 2002 speaker recognition evaluation plan," 2002, http://www.nist.gov/speech/tests/spk/2002/doc/.

Santi Nuratch received the B.Eng. and M.Eng. degrees of Control System and Instrumentation Engineering from King Mongkut's Unniversity of Technology Thonburi (KMUTT), Thailand in 2007 and 2009 respectively. He is currently pursuing the Ph.D. degree in Electrical and Computer Engineering at King Mongkut's Unniversity of Technology Thonburi, Thailand. His research interests include Digital Signal Processing, Speech Recognition, Speaker Recognition, Speaker Verification, Machine Learning, Computer Vision and Real-Time Embedded System.

Panuthat Boonpramuk received the B.Eng. and M.Eng. degrees of Electrical Engineering from King Mongkut's Unniversity of Technology Thonburi (KMUTT), Thailand in 1992 and 1995 respectively. He received Ph.D. from Kanazawa University, Japan in 2004. He is an Assistant Professor at Department of Control System and Instrumentation Engineering, King Mongkut's Unniversity of Technology Thonburi, Thailand. His research interests include Speech Signal Processing, Pattern Recognition, Neural Network and Adaptive Filter.

Chai Wutiwiwatchai received the B.Eng. (the first honor) and M.Eng. degrees of Electrical Engineering from Thammasat and Chulalongkorn University, Thailand in 1994 and 1997 respectively. He received Ph.D. from Tokyo Institute of Technology in 2004 under the Japanese Governmental scholarship. He is now the Head of Speech and Audio Technology Laboratory, Intelligent Informatics Research Unit, National Electronics and Computer Technology Center (NECTEC), Thailand. His research interests include Speech Processing, Natural Language Processing, and Human-Machine Interaction. His research work includes several international collaborative projects in a wide area of speech and language processing. He is now a member of International Speech Communication Association (ISCA).