EURASIP Journal on Audio, Speech, and Music Processing

RESEARCH Open Access

Feature compensation based on independent noise estimation for robust speech recognition

Yong Lü[1], Han Lin[2], Pingping Wu[2*] and Yitao Chen[2]

Abstract

In this paper, we propose a novel feature compensation algorithm based on independent noise estimation, which employs a Gaussian mixture model (GMM) with fewer Gaussian components to rapidly estimate the noise parameters from the noisy speech and monitor the noise variation. The estimated noise model is combined with a GMM with sufficient Gaussian mixtures to produce the noisy GMM for the clean speech estimation so that parameters are updated if and only if the noise variation occurs. Experimental results show that the proposed algorithm can achieve the recognition accuracy similar to that of the traditional GMM-based feature compensation, but significantly reduces the computational cost, and thereby is more useful for resource-limited mobile devices.

Keywords: Feature compensation, Independent noise estimation, Robust speech recognition, Vector Taylor series

1 Introduction

The automatic speech recognition (ASR) technology can provide convenient input interfaces for electronics devices, such as mobile phone, tablet computer, and navigation instrument. However, the performance of speech recognition systems is often severely degraded by the environmental noise. Therefore, the noise reduction technology is necessary for the embedded ASR systems.

The typical ASR system is composed of the front-end feature extraction and back-end pattern classification. In the front-end processing, the Mel frequency cepstral coefficient (MFCC) is widely used to represent the speech signal [1]. Besides, the perceptual linear predictive (PLP) features [2], spectro-temporal features [3], and cochlear filter cepstral coefficients (CFCC) features [4] have also been successfully used for speech recognition. In the back-end classification, the statistical acoustic models are commonly used, such as hidden Markov model (HMM) [5], artificial neural network (ANN) [6], and dynamic Bayesian network (DBN) [7].

In real-world applications, the environmental noise and other speech variations usually cause the serious mismatch between the present speech feature and pre-trained acoustic model. In order to reduce the mismatch, much research has been made and a large number of robust speech recognition techniques have been proposed [8, 9]. These methods can be mainly divided into two categories: the feature-domain and model-domain methods. The purpose of the front-end feature-domain approaches is to make the speech feature more robust to noise or to compensate the testing speech features to make the input testing data closer to the training condition. In general, the feature-domain methods can be further divided into three sub-categories: robust feature extraction [10, 11], feature normalization [12, 13], and feature compensation [14, 15]. Compared to the model compensation, feature-space methods are not related to the back-end acoustic models and have low computational cost.

In the back-end, model-domain methods modify the parameters of the prior trained acoustic model, which

*Correspondence: wupingping@nau.edu.cn
[2]School of Engineering Auditing, Jiangsu Key Laboratory of Public Project Audit, Nanjing Audit University, Nanjing, China
Full list of author information is available at the end of the article

makes the acoustic model match the noisy testing environment as well as possible. Maximum a posteriori (MAP) adaptation [16], maximum likelihood linear regression (MLLR) [17, 18], maximum a posteriori linear regression (MAPLR) [19], and parallel model combination (PMC) [20, 21] are representative examples of model compensation. Generally speaking, model compensation can achieve higher recognition accuracy than feature-domain methods. However, it usually leads to significantly larger computational expense and therefore may be not suitable for real-time applications.

This work focuses on the model-based feature compensation [22] and measurement science [23]. In the model-based feature compensation, the Gaussian mixture model (GMM) is typically employed to represent the distribution of speech features and it is assumed that the noisy speech GMM can be obtained by modifying the mean vectors and covariance matrices of the pre-trained clean speech GMM according to the noise parameters. The environmental noise is modeled by a single Gaussian distribution, whose mean and variance are estimated from the silence duration of the testing speech [24] or from the noisy speech [25] by the expectation-maximization (EM) algorithm [26]. To obtain the closed-form solution of the noise parameters, the vector Taylor series (VTS) technique [27, 28] is used to approximate the nonlinear relationship between the clean and noisy speech cepstral features. Finally, the clean speech feature is restored from the noisy speech feature by the minimum mean squared error (MMSE) method according to the estimated noisy speech GMM.

In this paper, we propose a novel feature compensation algorithm based on fast noise estimation technique using an independent Gaussian mixture model (IGMM) for resource-limited ASR devices. In this method, the noise estimation is separated from the feature compensation and is performed by an independent GMM with fewer Gaussian components. In other words, the feature compensation algorithm employs two GMM: GMM1 and GMM2. GMM1 is composed of fewer Gaussian components and used to rapidly estimate the noise parameters from the noisy testing speech. GMM2 consists of more Gaussian components and is used for the clean speech estimation. The proposed algorithm can achieve the recognition accuracy similar to that of the original GMM-based feature compensation and significantly reduces the computational complexity of the noise estimation. It can make a good balance between the computational complexity and recognition accuracy and thus is more suitable for resource-limited embedded systems.

The rest of this paper is organized as follows. In the next section, we describe the noise estimation method using independent Gaussian mixture model. The model combination and clean speech estimation are given in Section 3. The experimental procedures and results are presented and discussed in Section 4. Conclusions are drawn in Section 5.

2 Noise estimation using the independent Gaussian mixture model

In the traditional feature compensation, the noise estimation and clean speech estimation share the same GMM, which is trained by clean speech features during the training phase and is converted to noisy GMM in the testing condition. In order to guarantee the accuracy of clean speech estimation, the speech model usually consists of a large number of Gaussian components, which leads to a high computational cost. To improve computational efficiency without reducing the recognition accuracy, this paper employs two GMMs to estimate the noise parameters and restore the clean speech feature respectively, which is illustrated in Fig. 1. GMM1 composed of fewer Gaussian components is used to represent approximately the distribution of the speech feature and estimate rapidly the noise parameters from noisy speech features by the EM algorithm. Moreover, the average log-likelihood difference of GMM1 is considered as a sign of noise variation, which is used to decide whether or not to perform model combination. If the auxiliary function of the EM algorithm converges, the noise information which is composed of the noise variation sign and noise parameters is sent to model combination module, where the estimated single Gaussian noise model is combined with GMM2 to obtain noisy GMM for clean speech estimation. GMM2 has sufficient Gaussian components and can accurately characterize the distribution of speech in the cepstral domain. The noise distribution is independent of the speech distribution and thus it can be considered that the estimated noise parameters are weakly related to the Gaussian number of the speech model, which is used for the noise estimation. Therefore, in the model combination, the noise model estimated by GMM1 is closer to that estimated by the traditional GMM which consists of a large number of Gaussian components and is employed for both the noise estimation and clean speech restoration. On the other hand, the Gaussian number of GMM2 is similar to that of the traditional GMM-based feature compensation and the noisy GMM obtained by combining the GMM2 and estimated noise model can accurately restore the clean speech. Therefore, the proposed algorithm can achieve the recognition accuracy similar to that of the traditional GMM-based feature compensation.

This paper only considers the additive noise and ignores the channel distortion. According to the MFCC extraction method, we can obtain the relationship between the noisy speech cepstral feature y and clean speech cepstral feature x as:

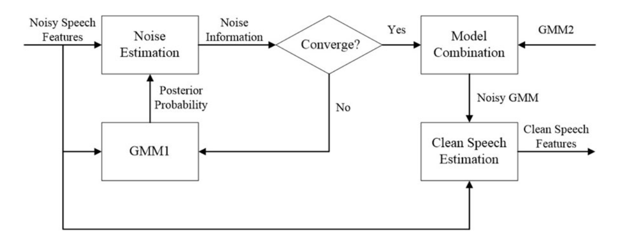


Fig. 1 The proposed feature compensation scheme

$$y = Clog\left(exp\left(C^{-1}x\right) + exp\left(C^{-1}n\right)\right) \tag{1}$$

where n denotes the cepstral features of the additive noise; C and C^{-1} denote the discrete cosine transform (DCT) matrix and its inverse transform matrix, respectively. By taking the first-order VTS expansion at point $(\mu_x,\ \mu_{n0})$ both sides of (1), we can obtain the following linear approximation:

$$y = (I-U)(x-\mu_x) + U(n-\mu_{n0}) + Clog\left(exp\left(C^{-1}\mu_x\right) + exp\left(C^{-1}\mu_{n0}\right)\right) \tag{2}$$

where μ_x and μ_{n0} are the mean of x and the initial mean of n, respectively; I denotes the identity matrix; U is given by,

$$U = Cdiag\left(\frac{exp\left(C^{-1}\mu_{n0}\right)}{exp\left(C^{-1}\mu_x\right) + exp\left(C^{-1}\mu_{n0}\right)}\right)C^{-1} \tag{3}$$

where $diag()$ denotes the diagonal matrix whose diagonal elements are equal to those of the vector in the parentheses. Taking the expectation on both sides of (2), the mean vector of the noisy speech μ_y can be expressed as:

$$\mu_y = U\mu_n - U\mu_{n0} + Clog\left(exp\left(C^{-1}\mu_x\right) + exp\left(C^{-1}\mu_{n0}\right)\right) \tag{4}$$

where μ_n is the mean of n. Similarly, we can obtain the variance of the noisy speech $\sum_y$ by taking the variance operation on both sides of (2):

$$\Sigma_y = (I-U)\Sigma_x(I-U)^T + U\Sigma_n U^T \tag{5}$$

where Σ_n denotes the variances of n.

In the noise estimation, the probability density function (PDF) of the speech signal is represented by GMM1:

$$b(x_t) = \sum_{m=1}^{M} c_m \bigg\{(2\pi)^{-\frac{D}{2}}|\Sigma_{x,m}|^{-\frac{1}{2}} \times exp\bigg[-\frac{1}{2}(x_t-\mu_{x,m})^T\Sigma_{x,m}^{-1}(x_t-\mu_{x,m})\bigg]\bigg]\bigg\} \tag{6}$$

where x_t denotes the tth static cepstral feature vector; $c_m, \mu_{x,m}$, and $\Sigma_{x,m}$ are the mixture coefficient, mean vector, and covariance matrix of the mth Gaussian component, respectively; and M and D denote the Gaussian number of GMM1 and the dimension of the static feature x_t, respectively. GMM1 is trained from clean speech in the training phase and used to estimate the noise parameters from noisy testing speech. The noise parameters, μ_n and Σ_n, are estimated using the EM algorithm under the maximum likelihood criterion and the auxiliary function is defined as:

$$Q(\bar{\lambda}|\lambda) = -\frac{1}{2}\sum_{m=1}^{M}\sum_{t=1}^{T}\gamma_m(t) \times \left[(y_t-\mu_{y,m})^T\Sigma_{y,m}^{-1}(y_t-\mu_{y,m}) - log\left|\Sigma_{y,m}^{-1}\right|\right] \tag{7}$$

where $\gamma_m(t) = P(k_t = m|y_t, \lambda)$ is the posterior probability of being in mixture component m at time t given the observation y_t and the prior parameter set λ; $\bar{\lambda}$ denotes the new GMM parameter set.

For the mth Gaussian component of GMM1, (4) can be rewritten as:

$$\mu_{y,m} = U_m\mu_n - U_m\mu_{n0} + Clog\left(exp\left(C^{-1}\mu_{x,m}\right) + exp\left(C^{-1}\mu_{n0}\right)\right) \tag{8}$$

Substituting (8) into (7) and setting the derivative of $Q(\bar{\lambda}|\lambda)$ with respect to μ_n to zero, the noise mean μ_n can be estimated by,

$$\mu_n = \left[\sum_{m=1}^{M}\sum_{t=1}^{T}\gamma_m(t)U_m^T\Sigma_{y,m}^{-1}U_m\right]^{-1} \times \left[\sum_{m=1}^{M}\sum_{t=1}^{T}\gamma_m(t)U_m^T\Sigma_{y,m}^{-1} \times \left(y_t - Clog\left(exp\left(C^{-1}\mu_{x,m}\right) + exp\left(C^{-1}\mu_{n0}\right)\right) + U_m\mu_{n0}\right)\right] \tag{9}$$

In the cepstral space, there are weak correlations among the different components of the cepstral vector, and thereby $\Sigma_{x,m}$, Σ_n, and $\Sigma_{y,m}$ can be simplified into the diagonal matrices. Equation (5) can be rewritten as:

$$\sigma_{y,m} = (V_m \cdot {*}V_m)\sigma_{x,m} + (U_m \cdot {*}U_m)\sigma_n \tag{10}$$

where $V_m = I - U_m$; $\sigma_{y,m}$, $\sigma_{x,m}$, and σ_n denote the variance vectors which are composed of the diagonal elements of $\Sigma_{y,m}$, $\Sigma_{x,m}$, and Σ_n, respectively; the operation symbol $\cdot*$ denotes the element-wise product for two vectors whose dimensions are the same. By substituting (10) into (7) and taking the derivative of $Q(\bar{\lambda}|\lambda)$ with respect to σ_n, we can obtain:

$$\begin{aligned}\frac{\partial Q(\bar{\lambda}|\lambda)}{\partial \sigma_n} &= \sum_{m=1}^{\mathrm{M}} \frac{\partial \eta_{y,m}}{\partial \sigma_n}\frac{\partial Q(\bar{\lambda}|\lambda)}{\partial \eta_{y,m}} \\ &= \sum_{m=1}^{\mathrm{M}} \frac{\partial \eta_{y,m}}{\partial \sigma_n}\sum_{t=1}^{\mathrm{T}} \gamma_m(t)\left[(y_t-\mu_{y,m})\cdot*(y_t-\mu_{y,m})\right. \\ &\quad \left. -(V_m\cdot*V_m)\sigma_{x,m}-(U_m\cdot*U_m)\sigma_n\right]\end{aligned} \tag{11}$$

where $\eta_{y,m}=(\sigma_{y,m})^{-1}=\left[(V_m\cdot*V_m)\sigma_{x,m}+(U_m\cdot*U_m)\sigma_n)\right]^{-1}$ and each element of $\eta_{y,m}$ is the reciprocal of the corresponding element of $\sigma_{y,m}$. The D×D matrix $\frac{\partial\eta_{y,m}}{\partial\sigma_n}$ can be regarded as the weighting factor of the mth Gaussian component and is written as:

$$\begin{aligned}G_m=-\frac{\partial\eta_{y,m}}{\partial\sigma_n}&=(U_m^T\cdot*U_m^T)\times diag\left[((V_m\cdot*V_m)\sigma_{x,m}\right. \\ &\quad \left.+(U_m\cdot*U_m)\sigma_n)^{-2}\right]\end{aligned} \tag{12}$$

To obtain the closed-form solution of the noise variance, the weighting factor G_m is approximated as a constant matrix:

$$\begin{aligned}G_m=(U_m^T\cdot*U_m^T)\times diag&\left[((V_m\cdot*V_m)\sigma_{x,m}\right. \\ &\left.+(U_m\cdot*U_m)\sigma_{n0})^{-2}\right]\end{aligned} \tag{13}$$

where σ_{n0} is the initial value of σ_n and is estimated from previous EM iteration. By setting the derivatives of $Q(\bar{\lambda}|\lambda)$ with respect to σ_n to zero, the noise variance σ_n can be computed as:

$$\begin{aligned}\sigma_n=&\left[\sum_{m=1}^{\mathrm{M}}\sum_{t=1}^{\mathrm{T}}\gamma_m(t)G_m(U_m\cdot*U_m)\right]^{-1}\times \\ &\left[\sum_{m=1}^{\mathrm{M}}\sum_{t=1}^{\mathrm{T}}\gamma_m(t)G_m((y_t-\mu_{y,m})\cdot*(y_t-\mu_{y,m})\right. \\ &\left.-(V_m\cdot*V_m)\sigma_{x,m})\right]\end{aligned} \tag{14}$$

In addition to noise estimation, another function of GMM1 is to monitor time variations of the environmental noise. When the recognizer is under the stationary condition, the parameters of GMM2 used for clean speech estimation are not updated and the noisy GMM2 estimated from the previous time interval is directly employed for the clean speech feature estimation of the current time interval, which can save energy and improve battery runtime for mobile devices. When the environmental noise varies, the clean GMM2 is combined with the estimated noise parameters μ_n and σ_n to produce the noisy GMM2 for computing clean speech features. It is difficult to determine whether the noise variation occurs by comparing the noise parameters of two time intervals directly. Therefore, this work employs the average log-likelihood difference over all the frames of the current time interval as the sign of noise variation. Besides the adapted noisy GMM1 estimated from the current testing speech, the noise parameters of the previous time interval are saved in memory and used to produce another noisy GMM1 by model combination with the clean GMM1. If the average log-likelihood difference of the two noisy GMM1 is more than the threshold, we can believe that the noise variation occurs. The noise variation sign and noise parameters compose the noise information, which is sent to model combination module to decide whether or not to update the parameters of the noisy GMM2.

As shown in Fig. 1, the complete noise estimation process is summarized below.

1. Initialize the initial mean μ_{n0} and initial variance σ_{n0} using the vector of all zeros and the vector of all ones, respectively.
2. Initialize the mean $\mu_{y,m}$ and variance $\sigma_{y,m}$ of GMM1 with $\mu_{y,m}$= $\mu_{x,m}$, $\sigma_{y,m}$=$\sigma_{x,m}$.
3. Compute the posterior probability of the noisy speech using GMM1.
4. Compute the auxiliary function of the EM algorithm by Eq. (7).
5. Estimate the noise parameters μ_n and σ_n using Eqs. (9) and (14), respectively.
6. Update the mean $\mu_{y,m}$ and variance $\sigma_{y,m}$ of GMM1 using Eqs. (8) and (10), respectively.
7. Update the initial mean μ_{n0} and initial variance σ_{n0} with μ_{n0} = μ_n, σ_{n0}=σ_n.
8. If the convergence criterion is not met, go to step 3.

3 Model combination and clean speech estimation

3.1 Model combination

The Gaussian number of GMM2 is much greater than that of GMM1 and thus it can accurately represent the distribution of cepstral speech features. GMM2 is trained by the clean speech during the training phase and its PDF can be written as:

$$\begin{aligned}b(x_t)=\sum_{i=1}^{\mathrm{N}}c_n&\left\{(2\pi)^{-\frac{\mathrm{D}}{2}}|\Sigma_{x,i}|^{-\frac{1}{2}}\right. \\ &\left.\times exp\left[-\frac{1}{2}(x_t-\mu_{x,i})^T\Sigma_{x,i}^{-1}(x_t-\mu_{x,i})\right]\right\}\end{aligned} \tag{15}$$

where x_t denotes the tth static cepstral feature vector; c_i, $\mu_{x,i}$, and $\Sigma_{x,i}$ are the mixture coefficient, mean vector, and covariance matrix of the ith Gaussian component, respectively; and N is the Gaussian number of GMM2. If the noise variation occurs, the means and variances of GMM2 will be updated using the following equations:

$$\mu_{y,i} = Clog\left(exp\left(C^{-1}\mu_{x,i}\right) + exp\left(C^{-1}\mu_n\right)\right) \tag{16}$$

$$\sigma_{y,i} = ((I - U_i)\cdot *(I - U_i))\sigma_{x,i} + (U_i \cdot *U_i)\sigma_n \tag{17}$$

where $\mu_{y,i}$ is the noisy mean vector of the ith Gaussian component; $\sigma_{y,i}$ is the noisy variance vector, which is composed of the diagonal elements of the noisy covariance matrix $\Sigma_{y,}$; and U_i is given by,

$$U_i = Cdiag\left(\frac{exp\left(C^{-1}\mu_n\right)}{exp\left(C^{-1}\mu_{x,i}\right) + exp\left(C^{-1}\mu_n\right)}\right)C^{-1} \tag{18}$$

In order to improve computational efficiency, Eq. (18) is implemented by the fast DCT algorithm and can be rewritten as:

$$U_i = C\left[diag\left(\frac{exp\left(C^{-1}\mu_n\right)}{exp\left(C^{-1}\mu_{x,i}\right) + exp\left(C^{-1}\mu_n\right)}\right)C^{-1}\right] \tag{19}$$

Equation (19) can be performed by D DCT calculations and thus its number of multiplications is approximately equal to $\mathrm{D}^2 log_2\mathrm{D} + \mathrm{D}^2$.

3.2 Clean speech estimation

The static coefficient $\hat{x}_t$ of the clean speech feature is estimated from the noisy speech feature y_t by the noisy GMM2 and the MMSE estimate of $\hat{x}_t$ is given by,

$$\hat{x}_t = E(x_t|y_t) \approx y_t - \sum_{i=1}^{\mathrm{N}} \hat{\gamma}_i(t) \times \left(Clog\left(1 + exp\left(C^{-1}\left(\mu_n - \mu_{x,i}\right)\right)\right)\right) \tag{20}$$

where $\hat{\gamma}_i(t) = P(k_t = i|y_t, \hat{\lambda})$ is the posterior probability of being in the ith Gaussian mixture at time t given the observation y_t and the noisy GMM2 parameter set $\hat{\lambda}$. The first-order coefficient of the clean speech feature $\Delta\hat{x}_t$ is obtained by differentiating the estimated clean static coefficients and the computing formula is written as:

$$\Delta\hat{x}_t = \frac{\sum_{\tau=-\mathrm{H}}^{\mathrm{H}} \tau\hat{x}_{t+\tau}}{\sum_{\tau=-\mathrm{H}}^{\mathrm{H}} \tau^2} \tag{21}$$

where H denotes the first-order differential constant. Similarly, the second-order coefficient of the clean speech feature $\Delta\Delta\hat{x}_t$ is computed by the following formula:

$$\Delta\Delta\hat{x}_t = \frac{\sum_{\tau=-\Gamma}^{\Gamma} \tau\,\Delta\hat{x}_{t+\tau}}{\sum_{\tau=-\Gamma}^{\Gamma} \tau^2} \tag{22}$$

where Γ denotes the second-order differential constant.

Since the covariance matrices of all the Gaussian components are diagonal in GMM2, we ignore the computational cost of obtaining the posterior probability $\hat{\gamma}_i(t)$. Thus, the computational complexity of the clean speech estimation mainly depends on (20). For the N values of $\hat{\gamma}_i(t)$, only a few probability values are non-zero and the most values are close to zero. Therefore, the following equation is used instead of (20):

$$\hat{x} = E(x_t|y_t) \approx y_t - \sum_{\hat{\gamma}_i(t)\in\mathrm{N}^*} \hat{\gamma}_i(t)\left(Clog\left(1 + exp\left(C^{-1}\left(\mu_n - \mu_{x,i}\right)\right)\right)\right) \tag{23}$$

where N^* denotes the set which is composed of the top 10% posterior probability. By taking (23) to restore the clean speech feature, the computational expense of the clean speech estimation can be ignored in the proposed feature compensation algorithm.

3.3 Computational complexity analysis

When the recognizer works in the stationary or slow time-varying noise condition, the model combination is seldom performed in the proposed algorithm and thus the computational cost is mainly dependent on the noise estimation. Assuming the GMM used in the traditional algorithm has the same Gaussian components as GMM2, the computational complexity of the proposed algorithm is reduced to about $\frac{\mathrm{M}}{\mathrm{N}}$ of that of the traditional GMM-based feature compensation, where M and N are the Gaussian numbers of GMM1 and GMM2, respectively.

In the case of fast time-varying noise, the proposed feature compensation employs (9) and (14) to estimate the noise parameters, which requires about $2\mathrm{D}^3$ multiplications. Thus, the noise estimation requires $2\mathrm{KMD}^3$ multiplications, where K and M denote the iteration number and the Gaussian number of GMM1, respectively; D is the channel number of the Mel filter bank. In the model combination, (16) can be performed by fast DCT technique and thereby its computational complexity is much lower than those of (17) and (19). For all the N Gaussian components of GMM2, the number of multiplications of (17) is approximately equal to $4\mathrm{ND}^2$ and that of (19) is about $\mathrm{N}\left(\mathrm{D}^2 log_2\mathrm{D} + \mathrm{D}^2\right)$. Therefore, the total number of multiplications of the proposed algorithm is approximately $2\mathrm{KMD}^3 + \mathrm{N}\left(5\mathrm{D}^2 + \mathrm{D}^2 log_2\mathrm{D}\right)$. At each EM iteration, the traditional GMM-based feature compensation firstly computes the noise parameters by (9) and (14), where GMM1 is replaced by the GMM with N Gaussian mixtures, and the two equations require about $2D^3$ multiplications. Then, it modifies the parameters of GMM using (16), (17), and (19), which take $\left(5D^2 + D^2 log_2 D\right)$ multiplications. For all the N Gaussian components of GMM and K EM iterations, the traditional algorithm performs approximately $KN\left(2D^3 + 5D^2 + D^2 log_2 D\right)$ multiplications. For example, when $D = 32$, $M = 20$, $N = 400$, and $K = 4$, the proposed and traditional GMM-based feature compensation methods require approximately 9,338,880 and 121,241,600 multiplications, respectively. The computational complexity of the proposed algorithm is reduced to about $\frac{1}{13}$ of that of the traditional algorithm.

4 Performance evaluation

4.1 Experimental conditions

To evaluate the proposed algorithm, the TIMIT speech database [29] and NOISEX-92 noise database [30] are employed to produce the training and testing speech in this paper. The two dialect sentences spoken by each speaker in the TIMIT database are segmented into 21 words for establishing the isolated word recognition system. The 6300 utterances spoken by 300 speakers are used to train GMM1 and GMM2. For each word, the 300 utterances spoken by the 300 speakers are employed to train the HMM of the word. The acoustic model is composed of the HMMs of all words and used for speech recognition in the back-end. The 2100 utterances spoken by 100 speakers are mixed with noise at different signal-to-noise ratio (SNR) values to obtain the noisy testing speech.

The original speech is down-sampled from 16 to 8 kHz and then the down-sampled speech is segmented into 16 ms frames with a frame shift of 8 ms. The feature vector of each frame is composed of 13 Mel frequency cepstral coefficients including the 0th coefficient, and their first-order time differential coefficients. Each word is modeled by a left-to-right HMM, which is composed of 6 states with 4 Gaussian components per state. The Gaussian number of GMM1 varies from 10 to 400 for the noise estimation and GMM2 consists of 400 Gaussian mixtures. For all the GMMs and HMMs, the covariance matrix of each Gaussian mixture is diagonal. The initial noise mean μ_{n0} are set to the vector of all zeros and the initial noise variance σ_{n0} is set to the vector of all ones for the first EM iteration.

The system configuration of the computer used for experiments is as follows: Intel Core i5-6400 Processor (2.70 GHz), 8.00 GB Random Access Memory (RAM), and Microsoft Windows 10 Operating System. The speech recognition system is constructed using the GNU Octave 4.4.1, and the computation time is the running time of the Octave software in the computational complexity measurement experiment.

4.2 Average log-likelihood difference

This experiment validates the effectiveness of the average log-likelihood difference as the sign of noise variation and the average log-likelihoods of the clean, combined, and adapted GMMs are illustrated in Fig. 2. The adapted GMM is obtained by modifying the parameters of GMM1 (clean GMM) according to the noise parameters estimated from the current noisy testing speech and the combined GMM is produced by combining GMM1 and the single Gaussian noise model obtained from the previous time interval. The white noise is used to produce the testing speech and the noise parameters are updated once per second using (9) and (14). The initial SNR is about 5 dB and then it is improved to 10 dB at the third second. The SNR is also approximately constant during 4∼7 s.

As demonstrated in Fig. 2, the average log-likelihood of the combined GMM is very close to that of the adapted GMM in the approximately stationary conditions. The SNR is improved from 5 to 10 dB at the third second and thus there exists an environmental mismatch between the combined GMM and testing condition. Therefore, the average log-likelihood of the combined GMM degrades drastically and is far less than that of the adapted GMM when the average log-likelihood is updated at the fourth second. The results show that the average log-likelihood difference of the adapted and combined GMMs can be used as the sign of noise variation. If the average log-likelihood difference is less than or equal to the threshold, it can be assumed that the noise does not vary. Thus, it is not necessary to perform the model combination and the noisy GMM2 of the previous time interval is used for the clean speech estimation of the current time interval. If the average log-likelihood difference is more than the threshold, we consider that the noise variation occurs and the parameters of the noisy GMM2 should be updated

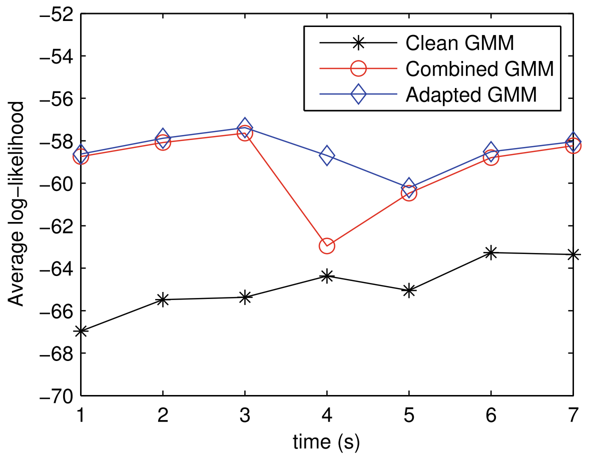


Fig. 2 Average log-likelihoods of the clean, combined, and adapted GMMs

by model combination. The threshold of the average log-likelihood difference is set to 0.5 in our experiments.

4.3 Number of Gaussian components

This experiment shows how to select the Gaussian number of the GMM1. Figure 3 illustrates the word error rates of the proposed algorithm in white noise environments, where the Gaussian number of the GMM1 varies from 10 to 400. The results indicate that the recognition performance of the proposed feature compensation is less affected by the Gaussian number of GMM1. Fewer Gaussian components mean less computational expense and more Gaussian mixtures can improve the accuracy of the noise estimation to a certain extent. Comprehensively considering the recognition rate and computational cost, the Gaussian number of GMM1 is set to 20 in the following experiments.

4.4 Comparison of recognition results

In this experiment, the proposed algorithm (IGMM20) is compared with the original GMM-based feature compensation (GMM400, GMM20) [25, 27], where GMM400 and GMM20 employ the 400-Gaussian GMM and 20-Gaussian GMM for feature compensation, respectively. Figure 4 shows the word error rates with different SNR levels for the three types of testing noise: (a) white noise, (b) pink noise, (c) factory noise, and (d) average over the three types of testing noise.

As shown in Fig. 4, the proposed algorithm can achieve similar performance with the traditional GMM-based feature compensation (GMM400). For example, at 0 dB SNR, the word error rates of GMM400 are 41.1%, 31.2%, and 36.3% for white, pink, and factory noise, respectively, while the corresponding results of IGMM20 are 42.8%, 32.1%, and 37.2%. This shows that the GMM-based noise

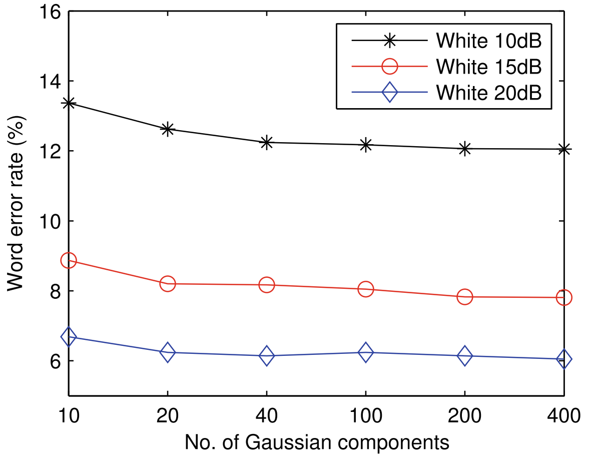
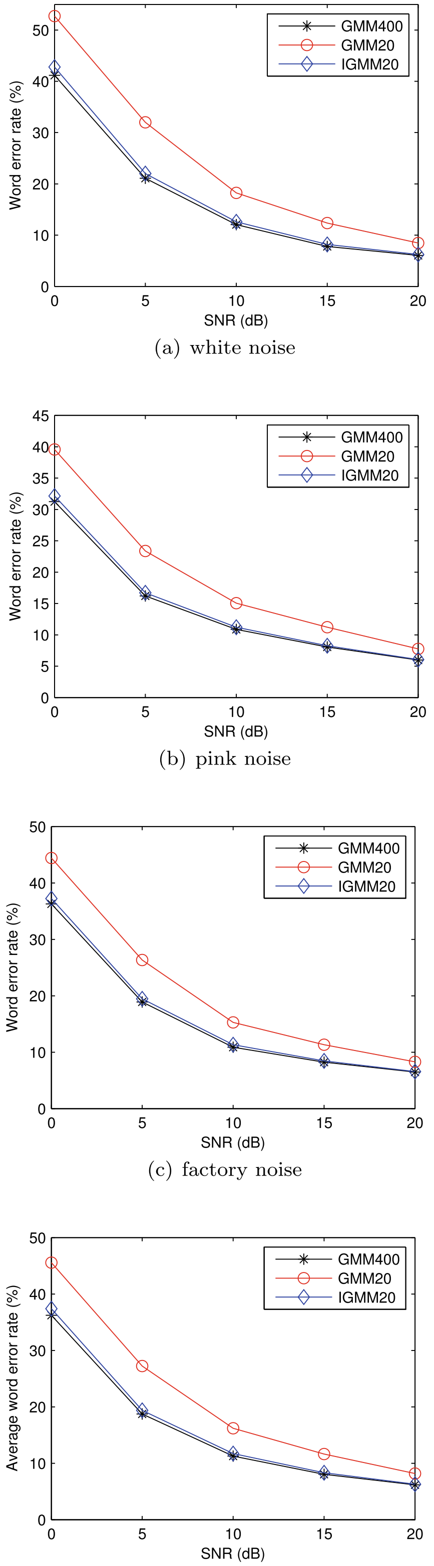


Fig. 3 Word error rates of the proposed algorithm with different numbers of Gaussian components



(a) white noise

(b) pink noise

(c) factory noise

(d) average over the three types of testing noise

Fig. 4 Performance comparison of the proposed algorithm (IGMM20) and original GMM-based feature compensation (GMM400 and GMM20) with different SNRs for the three types of testing noise

estimation is less affected by the Gaussian number of GMM and thus the noise parameters can be estimated by a GMM with fewer Gaussian components, which can significantly reduce the computational complexity of the noise estimation in real applications. When the Gaussian number of the traditional algorithm is reduced to 20, its recognition performance degrades drastically, which demonstrates that the clean speech estimation requires sufficient Gaussian mixtures and the GMM for restoring the clean speech feature should represent the PDF of the speech feature more accurately. In summary, using different GMMs to estimate the noise and reconstruct the clean speech respectively, we can reduce the computational cost without performance degradation.

4.5 Comparison of computational complexity

Finally, we discuss the computational cost of the proposed algorithm. Figure 5 illustrates the average computation time per frame of the proposed noise estimation with different Gaussian numbers at 10 dB SNR in white noise environment. The result of 400 Gaussian mixtures is equivalent to that of the traditional GMM-based feature compensation (GMM400).

From Fig. 5, it can be seen that when the Gaussian number decreases, the computational cost of the proposed algorithm is further reduced and the computation time is roughly proportional to the Gaussian number. The average computation time of 20 Gaussian components is 4.29 ms, which is only about one seventeenth of that of GMM400. This shows that the proposed algorithm can make a good balance between the computational complexity and recognition accuracy, and is more suitable for resource-limited embedded systems.

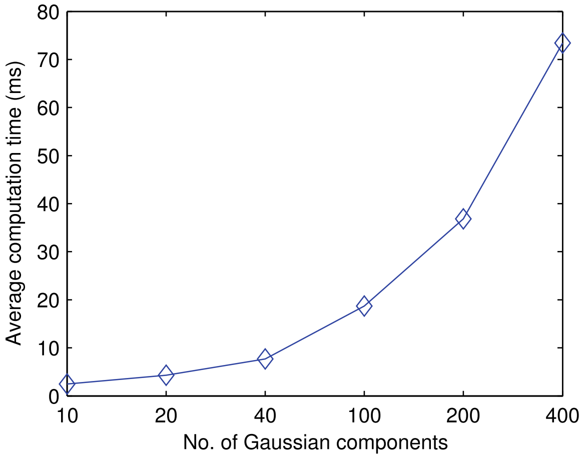


Fig. 5 Average computation time of the proposed noise estimation with different Gaussian numbers at 10 dB SNR in white noise environment

5 Conclusions

In this paper, we propose a novel feature compensation algorithm based on the independent noise estimation for robust speech recognition, which separates the noise estimation from the feature compensation and performs it using an independent GMM with fewer Gaussian components. Moreover, the GMM is used to monitor the time variations of the environmental noise according to the average log-likelihoods of the combined and adapted noisy GMMs. In order to guarantee the accuracy of the feature compensation, another GMM with sufficient Gaussian components is employed to estimate the clean speech feature. Only when the noise variation occurs, the parameters of noisy GMM for the clean speech estimation are updated by model combination with the estimated the single Gaussian noise model, which can save energy and improve battery runtime for mobile devices. The experimental results show that the proposed algorithm can achieve the recognition accuracy similar to that of the traditional GMM-based feature compensation, but significantly reduces the computational complexity. It can make a good balance between the computational complexity and recognition accuracy and thus is more suitable for resource-limited devices.

Abbreviations

GMM: Gaussian mixture model; ASR: Automatic speech recognition; MFCC: Mel frequency cepstral coefficient; PLP: Perceptual linear predictive; HMM: Hidden Markov model; ANN: Artificial neural network; CFCC: Cochlear filter cepstral coefficients; DBN: Dynamic Bayesian network; MAP: Maximum a posteriori; MLLR: Maximum likelihood linear regression; MAPLR: Maximum a posteriori linear regression; PMC: Parallel model combination; EM: Expectation-maximization; VTS: Vector Taylor series; MMSE: Minimum mean squared error; IGMM: Independent Gaussian mixture model; DCT: Discrete cosine transform; SNR: Signal-to-noise ratio

Acknowledgements

Not applicable.

Authors' contributions

YL performed the entire research and its writing. HL and PW supported the experimental part and supervised the research. All authors read and approved the final manuscript.

Funding

This work is supported by the National Natural Science Foundation of China (NSFC, No. 61701243, 71771125, 61871174) and the Major Project of Natural Science Foundation of Jiangsu Education Department (19KJA180002).

Availability of data and materials

Not applicable.

Declarations

Competing interests

The authors declare that they have no competing interests.

Author details

[1]College of Computer and Information Engineering, Hohai University, Nanjing, China. [2]School of Engineering Auditing, Jiangsu Key Laboratory of Public Project Audit, Nanjing Audit University, Nanjing, China.

Received: 7 January 2021 Accepted: 2 June 2021
Published online: 16 June 2021

References

1. B. S. Paul S, A. X. Glittas, L. Gopalakrishnan, A low latency modular-level deeply integrated MFCC feature extraction architecture for speech recognition. Integration. **76**, 69–75 (2021)
2. M. Malik, M. K. Malik, K. Mehmood, I. Makhdoom, Automatic speech recognition: a survey. Multimed. Tools Appl. **80**(6), 9411–9457 (2021)
3. N. Esfandian, F. Razzazi, A. Behrad, A clustering based feature selection method in spectro-temporal domain for speech recognition. Eng. Appl. Artif. Intell. **25**(6), 1194–1202 (2012)
4. Y. Shi, J. Bai, P. Xue, D. Shi, Fusion feature extraction based on auditory and energy for noise-robust speech recognition. IEEE Access. **7**, 81911–81922 (2019)
5. L. R. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition. Proc. IEEE. **77**(2), 257–286 (1989)
6. M. S. Yakoub, S.-a. Selouani, B.-F. Zaidi, A. Bouchair, Improving dysarthric speech recognition using empirical mode decomposition and convolutional neural network. EURASIP J. Audio Speech Music Process. **2020**(1), 1–7 (2020)
7. K. Daoudi, D. Fohr, C. Antoine, Dynamic Bayesian networks for multi-band automatic speech recognition. Comput. Speech Lang. **17**(2-3), 263–285 (2003)
8. M. Benzeghiba, R. De Mori, O. Deroo, S. Dupont, T. Erbes, D. Jouvet, L. Fissore, P. Laface, A. Mertins, C. Ris, *et al*, Automatic speech recognition and speech variability: a review. Speech Comm. **49**(10-11), 763–786 (2007)
9. J. Li, L. Deng, Y. Gong, R. Haeb-Umbach, An overview of noise-robust automatic speech recognition. IEEE/ACM Trans. Audio Speech Lang. Process. **22**(4), 745–777 (2014)
10. T. Hori, Z. Chen, H. Erdogan, J. R. Hershey, J. Le Roux, V. Mitra, S. Watanabe, Multi-microphone speech recognition integrating beamforming, robust feature extraction, and advanced DNN/RNN backend. Comput. Speech Lang. **46**, 401–418 (2017)
11. N. Moritz, K. Adiloğlu, J. Anemüller, S. Goetze, B. Kollmeier, Multi-channel speech enhancement and amplitude modulation analysis for noise robust automatic speech recognition. Comput. Speech Lang. **46**, 558–573 (2017)
12. H. F. Pardede, K. Iwano, K. Shinoda, Feature normalization based on non-extensive statistics for speech recognition. Speech Comm. **55**(5), 587–599 (2013)
13. V. Joshi, R. Bilgi, S. Umesh, L. Garcia, C. Benitez, Sub-band based histogram equalization in cepstral domain for speech recognition. Speech Comm. **69**, 46–65 (2015)
14. T. Kleinschmidt, S. Sridharan, M. Mason, The use of phase in complex spectrum subtraction for robust speech recognition. Comput. Speech Lang. **25**(3), 585–600 (2011)
15. J. Du, Q. Huo, An improved VTS feature compensation using mixture models of distortion and IVN training for noisy speech recognition. IEEE/ACM Trans. Audio Speech Lang. Process. **22**(11), 1601–1611 (2014)
16. J.-L. Gauvain, C.-H. Lee, Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains. IEEE Trans. Speech Audio Process. **2**(2), 291–298 (1994)
17. C. J. Leggetter, P. C. Woodland, Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models. Comput. Speech Lang. **9**(2), 171–185 (1995)
18. M. J. Gales, P. C. Woodland, Mean and variance adaptation within the MLLR framework. Comput. Speech Lang. **10**(4), 249–264 (1996)
19. C. Chesta, O. Siohan, C.-H. Lee, in *Sixth European Conference on Speech Communication and Technology*. Maximum a posteriori linear regression for hidden Markov model adaptation (ISCA, Budapest, 1999)
20. M. Gales, S. J. Young, Robust speech recognition in additive and convolutional noise using parallel model combination. Comput. Speech Lang. **9**(4), 289–307 (1995)
21. H. Veisi, H. Sameti, The integration of principal component analysis and cepstral mean subtraction in parallel model combination for robust speech recognition. Dig. Signal Proc. **21**(1), 36–53 (2011)
22. A. Erell, M. Weintraub, Filterbank-energy estimation using mixture and Markov models for recognition of noisy speech. IEEE Trans. Speech Audio Process. **1**(1), 68–76 (1993)
23. V. Witkovskỳ, I. Frollo, Measurement science is the science of sciences - there is no science without measurement. Meas. Sci. Rev. **20**(1), 1–5 (2020)
24. W. Kim, J. H. Hansen, Feature compensation in the cepstral domain employing model combination. Speech Comm. **51**(2), 83–96 (2009)
25. M. Korenevsky, Phase term modeling for enhanced feature-space VTS. Speech Commun. **89**, 84–91 (2017)
26. A. Dempster, N. Laird, D. Rubin, Maximum-likelihood from incomplete data via EM algorithm. J. R. Stat. Soc. **39**(1), 1–38 (1977)
27. Y. Lu, H. Wu, Z. Wu, Robust speech recognition using improved vector Taylor series algorithm for embedded systems. IEEE Trans. Consum. Electron. **56**(2), 764–769 (2010)
28. J. Li, L. Deng, D. Yu, Y. Gong, A. Acero, A unified framework of HMM adaptation with joint compensation of additive and convolutive distortions. Comput. Speech Lang. **23**(3), 389–405 (2009)
29. V. Zue, S. Seneff, J. Glass. Speech Database Development: TIMIT and Beyond, vol. 9, (1990), pp. 351–356
30. A. Varga, H. J. Steeneken, Assessment for automatic speech recognition: II, NOISEX-92: a database and an experiment to study the effect of additive noise on speech recognition systems. Speech Commun. **12**(3), 247–251 (1993)

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.