Hindawi Publishing Corporation
Advances in Acoustics and Vibration
Volume 2012, Article ID 172625, 10 pages
doi:10.1155/2012/172625

Research Article

Multiengine Speech Processing Using SNR Estimator in Variable Noisy Environments

Ahmad R. Abu-El-Quran,[1] Adrian D. C. Chan,[2] and Rafik A. Goubran[2]

[1] *Department of Enterprise and Higher Education, International Turnkey Systems, ITS Tower, Mubarak Al Kabeer Street, P.O. Box 26729, Safat, Kuwait City 13128, Kuwait*
[2] *Department of Systems and Computer Engineering, Carleton University, 1125 Colonel By Drive, Ottawa, ON, Canada K1S 5B6*

Correspondence should be addressed to Ahmad R. Abu-El-Quran, ahmad.rami@its.ws

Received 14 August 2011; Accepted 26 October 2011

Academic Editor: Akira Ikuta

We introduce a multiengine speech processing system that can detect the location and the type of audio signal in variable noisy environments. This system detects the location of the audio source using a microphone array; the system examines the audio first, determines if it is speech/nonspeech, then estimates the value of the signal to noise (SNR) using a Discrete-Valued SNR Estimator. Using this SNR value, instead of trying to adapt the speech signal to the speech processing system, we adapt the speech processing system to the surrounding environment of the captured speech signal. In this paper, we introduced the Discrete-Valued SNR Estimator and a multiengine classifier, using Multiengine Selection or Multiengine Weighted Fusion. Also we use the SI as example of the speech processing. The Discrete-Valued SNR Estimator achieves an accuracy of 98.4% in characterizing the environment's SNR. Compared to a conventional single engine SI system, the improvement in accuracy was as high as 9.0% and 10.0% for the Multiengine Selection and Multiengine Weighted Fusion, respectively.

1. Introduction

Speech processing systems, such as Speaker identification (SI) and Automatic Speech Recognition (ASR), have two operating modes: a training mode and a testing mode. The goal of the training mode is for the classifier to learn how to discern between different speech classes, using exemplars whose classes are known. The purpose of the testing mode is to classify speech data, from an unknown class, based on the training.

The performance of most speech processing systems is degraded severely when the training and the testing audio are captured in different acoustic environments (e.g., different levels of noise [1–4]). For example, training speech may be captured using a telephone handset in a noise-free environment, while the test speech is captured using a hands-free phone in a noisy environment. The conventional method to improve the performance of speech processing in noisy environments is to perform speech enhancement prior to speech processing [2–7]. In this method, training speech signals are obtained in a clean environment (i.e., no noise). The test speech signals, however, are obtained in environments that may contain noise. Denoising and/or speech enhancement methods may be applied to these signals to reduce the effects of the environment; however, often these methods enhances the speech signal audibility from the human perspective, but ignores the fact that the enhancements may remove or distort information in the speech signal, which may be essential to the speech processing.

In [2], the effect of five different speech enhancement techniques on a GMM-based ASR system was examined. These enhancement techniques were (1) perceptual wavelet adaptive denoising (PWAD) (TPS-PWAD), (2) VAD-based noise floor estimation (VAD-NFE), (3) minimum-mean-square-error-based two-dimensional spectral enhancement (MMSE-TDSE), (4) two-pass adaptive noise estimation (TPANE), and (5) combined two-pass spectral and PWAD. The performance of the speech enhancement techniques was evaluated using Perceptual Evaluation of Speech Quality-Mean Opinion Score (PESQ-MOS) [7, 8]. PESQ-MOS predicts the subjective quality of input speech by returning a score between 0 and 4.5; the higher the score, the cleaner

the input speech (i.e., higher SNR). These enhancement techniques were used as preprocessing blocks to the ASR engine. The overall performance was evaluated based on the ASR classification accuracy using each enhancement technique.

The author of [2] noted the following.

(i) The speech enhancement techniques of the corrupted speech signal with white noise generally improved the classification accuracy of the ASR; however, this improvement is inconsistent for different SNR values for a single enhancement technique.

(ii) The MMSE-TDSE has the greatest average improvement in the speech quality PESQ-MOS score, over SNR range from 0 to 20 dB, an average improvement of 0.7 points. This improvement, however, did not translate into the best ASR classification accuracy.

(iii) At SNR = 20 dB, the MMSE-TDSE speech enhancement resulted in a decrease in ASR classification accuracy of −0.71%, relative to ASR without speech enhancement techniques.

(iv) The VAD-NFE achieved the best SNR improvement of 0.5 dB and 3.7 dB at 20 and 15 dB, respectively; however, this did not translate into the best ASR classification accuracy at these SNR values.

The results of [2] suggest that optimizing the quality of speech perception or SNR, using speech quality enhancement techniques, does not optimize speech processing performance. Indeed, speech quality enhancement techniques may actually degrade the speech processing performance instead of improving it.

In [5], the combination of four speech enhancement techniques have been combined to enhance the ASR in noisy environment. These speech enhancement techniques were (1) spectral subtraction with smoothing of time direction, (2) temporal domain SVD-based speech enhancements, (3) GMM-based speech estimation, and (4) KLT-based compfiltering. In [5], two combination methods have been introduced to combine the speech enhancement techniques: selection of front-end processor and combination of results from multiple recognitions processors. The speech enhancement techniques are applied on the captured noisy speech. A combination of these speech enhancement techniques is combined using one of the two proposed combination methods. The overall performance was evaluated based on the ASR classification accuracy using the two combination techniques. The authors of [5] reported an ASR accuracy enhancement of 5.4% of the proposed combination methods; compared to best performing standalone speech enhancement technique. This combination has been achieved using a noise environment detector with a detection accuracy of 54%. Using an ideal noise environment detector with detection accuracy of 100% can enhance the ASR accuracy with only 1.9%, compared to the environment detector with detection accuracy of 54%. As indicated in [5] the main factor of enhancing the ASR accuracy is the proposed combination method using different speech enhancement techniques. Meanwhile the authors [5] did not discussed the fact indicated in [2] that speech enhancement techniques may actually degrade the speech processing performance instead of improving it.

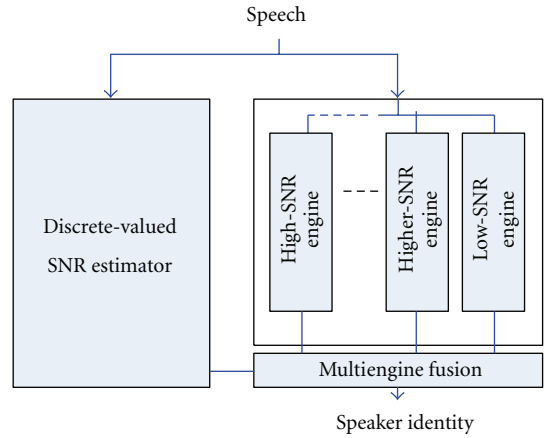

Figure 1: Multiengine SI system.

In this paper, we introduce a speech processing system that adapts to the surrounding environment of the speech signal, rather than trying to adapt the speech signal to the system; in this system, no speech enhancements are applied to the captured speech signal. This speech processing system characterizes its environment from the captured speech signal and adapts the speech processing to better suit the environment, enhancing the overall performance in variable environments. In this paper, speaker identification (SI) is used as the speech processing system.

The SI system proposed in this paper uses multiple SI engines, each trained in a noisy environment but with different SNR levels. Two multiengine fusion SI systems are proposed: Multiengine Selection and Multiengine Weighted Fusion (Figure 1). These proposed systems permit the intelligent fusion of the multiple SI engines using noise information of the surrounding environment. The noise information is determined by a Discrete-Valued SNR Estimator from the incoming speech signal. The proposed system enhances the SI performance, improving classification accuracy and robustness to different noise conditions.

The multiengine fusion system is being proposed for use in a novel security instrument using microphone array. This security instrument is capable of robustly determining the identity and location of a talker, even in a noisy environment. The instrument is also capable of identifying the audio type and location of nonspeech audio signals (e.g., footsteps, windows breaking, and cocktail noise) [9]. This instrument can be incorporated in a variety of applications, such as hands-free audio conferencing systems to perform voice-based security authentication for the conference users [10]. Also, this instrument can be utilized in smart homes and health monitoring for older adults that will support independent living and enable them to continue to age in their own homes [11]. Smart homes can benefit from this instrument by monitoring their occupants through analysis of the audio signals present in the home and recognition of

certain sound sources. For instance, respiration, falls to the ground, and water leaking sounds could be detected. The proposed instrument could perform further analyses and make decisions according to the captured audio type, such as extracting the respiration rate, calling emergency help, or alerting the occupant that a plumber may be needed. The analysis of the audio data preserves the privacy of the users better than video monitoring.

Previously mentioned applications require high performance to have successful functionality. In addition, the audio processing system needs to be robust even in variable conditions, such as environments with different levels of noise. Poor instrument robustness would limit utility of these systems.

The remainder of the paper is organized as follows. In Section 2 of this paper, we describe the overall implementation of the proposed approach. This section introduces and discusses the speech/nonspeech classifier, the Discrete-Valued SNR Estimator, the multiengine SI system, and the two methods of multiengine fusion: Multiengine Selection and Multiengine Weighted Fusion. Performance of the multiengine SI system is evaluated using a library of speech signals in noisy environments. In Section 3, results of this performance evaluation are discussed and compared against the baseline method of SI. Major conclusions are summarized in Section 4.

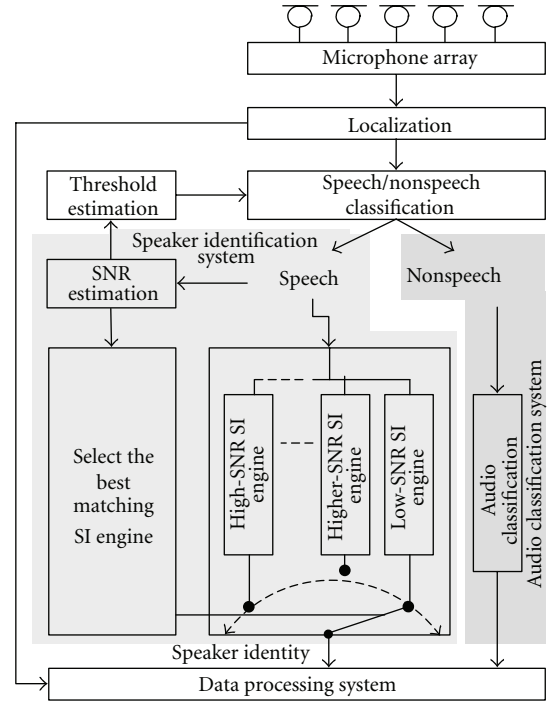

Figure 2: Block diagram of proposed system.

2. Proposed System

2.1. Instrument Overview. A block diagram of the overall instrument is shown in Figure 2. The microphone array captures the audio signal then sends the digitized audio signal to the localization block to determine the location of the sound source [12, 13] in addition to steering the microphone array to reduce undesired acoustic noise.

A speech/nonspeech classifier is used as a frontend to appropriately select the next stage in audio processing, which is either the SI system or the nonspeech audio classification system [9]. The speech/nonspeech classifier is also adapted to the environment, so that it is robust to varying levels of noise. In this paper, we utilize the Discrete-Valued SNR Estimator (Section 2.4) to adapt the speech/nonspeech classifier to the surrounding environment based on the estimated SNR value. If the detected signal is nonspeech, the audio classification system will determine the audio type of the detected signal (e.g., footsteps, widows breaking, and door opening sounds); otherwise, the speech signal will be sent to both the Discrete-Valued SNR Estimator and the multiengine SI, consisting of *E*SI engines to identify the talker. Each SI engine is trained in an environment with a specific noise condition. Using the SNR estimate value from the Discrete-Valued SNR Estimator, the system fuses the outputs of the *E*SI engines.

We are introduced to a new fusion method, Multiengine Weighted Fusion. Multiengine Selection selects the SI engine with training environments that best matches the surrounding environment; the multiengine SI system decision is equal to the decision associated with this single SI engine. The Multiengine Weighted Fusion system intelligently weights and combines the output decisions of the *E*SI engines; the higher the similarity of the testing environment to the engine's training environment, the higher the weighting that engine receives.

The resultant talker identity (or audio type for nonspeech audio), along with the location information, is sent to the data processing system for application specific tasks. For example, these data can be utilized in audio-video hands-free conferencing, where a video camera can be appropriately steered to talkers and not steered toward undesired audio sources.

2.2. Speech/Nonspeech Classification. In [9, 14], pitch ratio (PR) was used to perform speech/nonspeech classification. The PR algorithm calculates the presence of human pitch ($70\,Hz < human\ pitch < 280\,Hz$) in multiple small frames in certain audio segment duration as shown in Figure 3. The PR is calculated as follows:

$$\text{Pitch Ratio (PR)} = \frac{\text{NP}}{\text{NF}}, \tag{1}$$

where

$$\text{NF} = \left\lfloor \frac{(\text{SD} - \text{FS})/(1 - \text{OL})}{\text{FS}} \right\rfloor + 1, \tag{2}$$

where $\lfloor \bullet \rfloor$ is floor operator, NP is the number of frames that have human pitch, and NF is the total number of frames. This PR value is then compared to a certain threshold. The speech/nonspeech decision is made as follows:

$$\text{Decision} = \begin{cases} \text{Speech}, & \text{if PR} > \text{Threshold}, \\ \text{Nonspeech}, & \text{if PR} < \text{Threshold}. \end{cases} \tag{3}$$

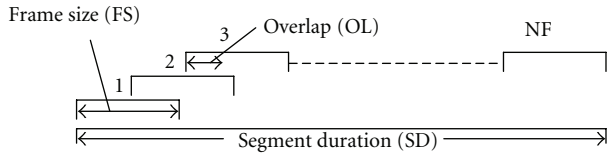

Figure 3: PR parameters.

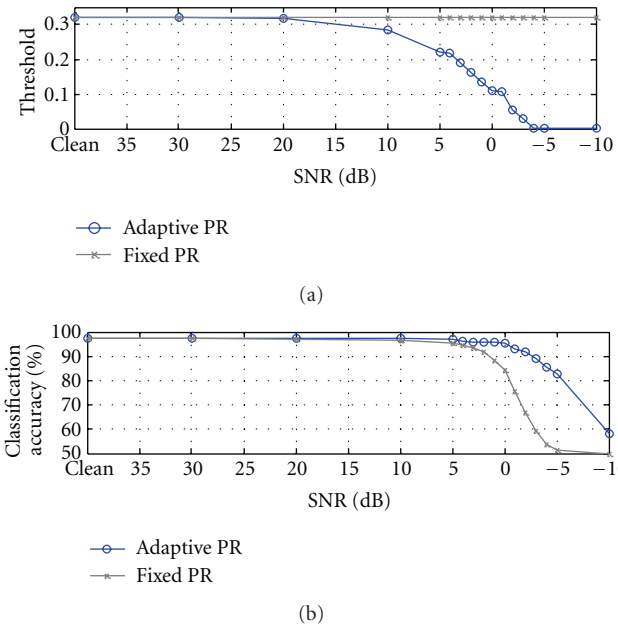

Figure 4: Relation between the PR threshold and SNR (sampling rate = 8 KHz, FS = 60 ms, SD = 450 ms, and OL = 0.85).

In [9], a threshold was empirically determined as a function of frame size (FS), segment duration (SD), and overlap (OL). Noise was shown to deteriorate the pitch period of sound. This phenomenon reduces the capability of the pitch detection algorithm to detect the presence of the pitch. To remedy this, the threshold of the PR algorithm is adapted to maximize the accuracy of the speech/nonspeech classifier. We have found that the optimal threshold (Figure 4(a)), the threshold corresponding to the maximum speech/nonspeech classification accuracy (Figure 4(b)), reduces as the SNR reduces. This adaptive speech/nonspeech algorithm was termed the adaptive pitch ratio (APR) algorithm [9]. Adaption is performed based on the SNR estimates from the Discrete-Valued SNR Estimator.

If the decision of the APR is nonspeech, further audio classification is performed to determine the audio type. Nonspeech audio classification will not be discussed here, as it is not the focus of this paper; however, readers can refer to [9] for more details. If the decision of the APR is speech, then an SI algorithm is used to identify the talker.

2.3. Discrete-Valued SNR Estimator. An SNR estimator that is computationally efficient with low complexity is desirable for this system, as it enables the selection of the SI engine in real time. Output of this estimator should be a discrete value that matches one of the training environments' SNRs. This estimator provides information about the noise in the surrounding environment by extracting features from the captured audio sounds emanating within this environment.

In this paper, the Discrete-Valued SNR Estimator is implemented using covariance matrix modeling [1, 15, 16]. The feature vector of the noisy environment is the mel-frequency spectrum coefficients (MFSCs) and the delta MFSC. The MFSCs are computed in a similar fashion as mel-frequency cepstrum coefficients (MFCCs) using a filter bank output but without the use of the discrete cosine transform (DCT) [17]. The DCT of the MFSC vector decorrelates the feature vectors' elements, which leads to a feature vector with low sensitivity to environment change in the audio signal and high sensitivity to the talker (i.e., dependent on the talker) [18]. In our case we desire a feature vector with high sensitivity to the noise but independent of the talker (i.e., provides information about the environments without being affected by who uttered the speech). Thus, the DCT stage has been removed from the feature vector generation process.

To implement the Discrete-Valued SNR Estimator, let X_e represent the covariance matrix of the M training feature vectors for environment $e\{\vec{x}_j, 1 \le j \le M\}$ and let Y represent the covariance matrix of the N test feature vectors $\{\vec{y}_i, 1 < i < N\}$ from an unknown environment. Feature vectors for both training and testing covariance matrices, have a length L; therefore, X_e and Y are matrices of size $L \times L$ and are computed as follows:

$$\begin{aligned} X_e &= \frac{1}{M}\sum_{i=1}^{M}(\vec{x}_i - \overline{x})^T(\vec{x}_i - \overline{x}), \\ Y &= \frac{1}{N}\sum_{j=1}^{N}\left(\vec{y}_j - \overline{y}\right)^T\left(\vec{y}_j - \overline{y}\right), \end{aligned} \tag{4}$$

where vector $\overline{x}$ is the mean of the M training vectors $\vec{x}_i$ and vector $\overline{y}$ is the mean of the N test vectors $\vec{y}_j$. Using the following property:

$$\frac{1}{N}\sum_{j=1}^{N}\left(\vec{y}_j - \overline{y}\right)^T X_e^{-1}\left(\vec{y}_j - \overline{y}\right) = \operatorname{tr}\left(YX_e^{-1}\right). \tag{5}$$

The distance between training environment e and test environment can be calculated using the arithmetic harmonic sphericity measure as follows [1, 16, 18]:

$$\begin{aligned} d(e) = \log&\left[\operatorname{tr}\left(YX_e^{-1}\right)\operatorname{tr}\left(X_eY^{-1}\right)\right] \\ &- 2\log(L), \quad 1 \le e \le E, \end{aligned} \tag{6}$$

where L is the feature vector length. The distance $d(e)$ represents the similarity between training environment e and the test environment; the smaller the distance, the more the similarity between the training and test environment. Then the distance vector is generated as follows:

$$D = [d(1)d(2)\cdots d(E)]^T, \quad d(e) \ge 0. \tag{7}$$

The proposed instrument only requires a rough estimation of the SNR value to set the threshold of the APR and to

calculate the distance between the surrounding environment and the training environments to fuse the output of the multiengines. Other SNR estimators, such as SNR estimation using high-order statistics [19], have higher resolution (i.e., estimates SNR as continuous scale); meanwhile, they have higher complexity. The increased precision is not needed in the proposed instrument implementation, so the increased complexity can be avoided.

2.4. Speaker Identification Engines. Gaussian mixed models (GMM) are used to implement the individual SI engines. GMMs are used commonly for speaker identification [20, 21]. During training, the parameters of each talker GMM is determined from a sequence of M training vectors. First, the vectors are grouped into clusters using Linde et al.'s (LBG) clustering algorithm [22]. Second, the mean and variance of each cluster are then computed. During testing, the probability that a given talker uttered a certain test speech is computed from the GMMs. Using extracted N feature vectors from the test utterance, the SI engine computes a log-likelihood probability vector as follows:

$$L = [l(1)l(2)\cdots l(S)]^T, \tag{8}$$

where S is the number of elements in this vector which is equal to the numbers of the talkers. The SI engine selects the talker that uttered the test speech as the talker with the highest log-likelihood value. In this work, a 32-GMM is used as a classifier, with a feature vector of 16 MFCCs.

The accuracy of the SI engine can be degraded by a mismatch between the training and test speech. In the context of this work, this mismatch occurs when different levels of noise are present in the training and testing sets. In the proposed multiengine SI system, a set of ESI engines are used, each trained with different levels of noise. For the multiengine SI, E log-likelihood probability vectors are produced, one vector from each engine. Using the SNR estimate from the Discrete-Valued SNR Estimator multiengine fusion is performed.

2.5. Multiengine Selection. Each environment e in a set of E training environments is modeled by a single covariance matrix X_e. A test noisy utterance from an unknown environment is modeled by a covariance matrix Y. The distance $d(e)$ is computed for each training environment. The SI engine associated with the minimum distance is selected, and the talker identity determined by this engine is used as the SI system output (Figure 5). The engine selection is calculated as follows:

$$\hat{e} = \arg\min D, \quad 1 \le e \le E. \tag{9}$$

The estimated SNR value is the SNR of the selected training environment $\{\text{SNR} = \text{SNR}_{\hat{e}},\ 1 \le \hat{e} \le E\}$.

2.6. Multiengine Weighted Fusion. Performance of SI engines varies in different environments. This performance variation depends on the similarity between the test and the training environment. Using this fact, SI system can utilize the output decision of each engine and combine the decisions of these engines (Figure 6).

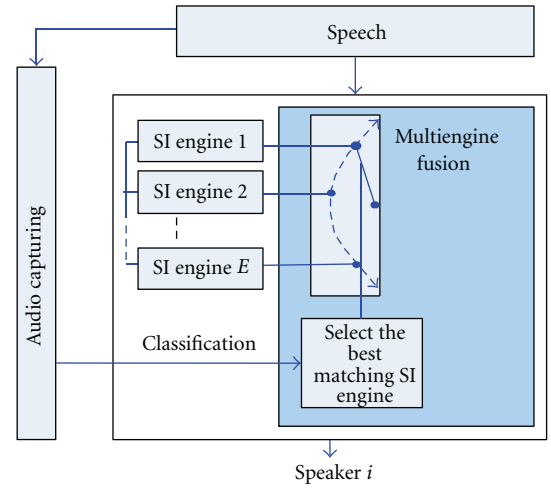

Figure 5: Multiengine speaker identification using the Engine Selection method.

Assume that we have S talkers t_i $(i \in (\Omega = \{1, 2, \ldots, S\}))$ and ESI engines trained in E environments. The conditional probability for each talker t_i generated from engine e is denoted as $p(t_i \mid \text{env}_e)$ and is computed using the following steps. Each SI engine that is trained in environment e generates a maximum log-likelihood vector L_e as follows:

$$L_e = [l_e(1)l_e(2)\cdots l_e(S)]^T, \tag{10}$$

where $l_e(i)$ is likelihood of the ith talker from the eth SI engine. These likelihoods are normalized as

$$p(t_i \text{env}_e) = \frac{l_e(i) - \min_j[l_e(j)]}{\sum_k \left(l_e(k) - \min_j[l_e(j)]\right)}. \tag{11}$$

This normalization ensures that the values of $p(t_i \mid \text{env}_e)$ have properties consistent with probabilities; that is,

$$\sum_{i=1}^{S} p(t_i \mid \text{env}_e) = 1, \quad p(t_i \mid \text{env}_e) \ge 0. \tag{12}$$

The probability that the surrounding environment is environment e is denoted as $p(\text{env}_e)$ and is computed as follows. First, the distance between the surrounding environment and the training environment is measured using the Discrete-Valued SNR Estimator. Then the distance vector is generated as in (7). The training environment e with the smallest distance $d(e)$ value has the highest similarity with the surrounding environment. The distances are normalized to the probability of the surrounding environments as follows:

$$p(\text{env}_e) = \frac{d(e) - \max_j[d(j)]}{\sum_k \left(d(k) - \max_j[d(j)]\right)}. \tag{13}$$

This normalization ensures the values of $p(\text{env}_e)$ have properties consistent with probabilities; that is:

$$\sum_{e=1}^{E} p(\text{env}_e) = 1, \quad p(\text{env}_e) \ge 0. \tag{14}$$

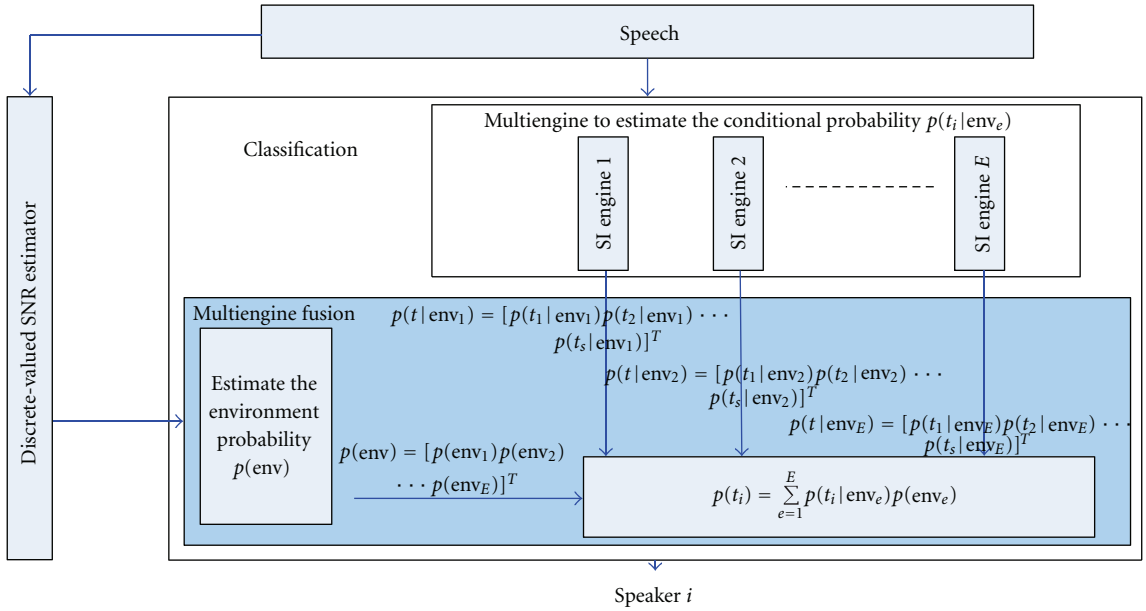

Figure 6: Multiengine speaker identification using the Weighted Fusion method.

The probability of talker t_i is computed for the SI system as follows:

$$p(t_i) = \sum_{e=1}^{E} p(t_i \mid \text{env}_e)p(\text{env}_e). \quad (15)$$

The decision of the Multiengine Weighted Fusion engines is the talker with the maximum probability. The Multiengine Weighted Fusion method generalizes the solution for the speaker identification systems in a variable environment.

3. Experiment and Results

3.1. Audio Database. A speech/nonspeech audio database containing a total of 1356 audio segments was used to train and test the speech/nonspeech classifier. Audio segments were 0.45 seconds in length, with a sampling rate of 8 kHz. The speech/nonspeech audio database contains nonspeech audio (e.g., windows breaking, wind, fan, and footsteps sounds) and speech audio. A full list of the audio types can be found in Table 1. Half of each audio type in the speech/nonspeech audio database was used in training the speech/nonspeech classifier, and the remaining half was used as a test set.

The KING speech database is used in training and testing the SI system. The speech/nonspeech portion of the system was trained using the previously mention speech/nonspeech audio database. The KING database consists of speech from 51 speakers; each speaker has 10 utterances with the exception of 2 speakers, which only have 5 utterances [23]. Data are sampled at a rate of 8 kHz, with 16-bit resolution, and utterance durations ranging from 30 to 60 seconds, which were all shortened to 15 seconds. The first 3 utterances of each speaker was used to form a training set (153 utterances), and the remaining utterances were used to form a test set (347 utterances). For the speech/nonspeech and KING databases, the training sets and test sets are mutually exclusive.

Table 1: Speech/nonspeech audio database.

Audio class	Number of Audio files
Speech	
Male	282
Female	282
Nonspeech	
Chair	20
Cocktail noise	20
Door	61
Drawer	20
Fan noise	169
Footsteps	41
Glass breaking	145
Rain	50
Thunder	40
Traffic	20
Water	20
Wind	186
Total	1356

A total of 6 levels of SNR (−10, 0, 10, 20, and 30 dB and a clean environment) were used for training, using additive white Gaussian noise to adjust the SNR. A total of 11 levels of SNR (−10, −5, 0, 5, 10, 15, 20, 25, 30, and 35 dB and a clean environment) were used for testing; 5 levels of SNR were different from the training SNR levels. Additive white Gaussian noise and pink noise were used to create the different levels of noise; therefore, while the SNR level may be the same in the training and test set, the type of noise may differ.

3.2. Speech/Nonspeech Classification Results. The speech/nonspeech classifier was tested over an SNR range of −10 dB to clean environment, using white and pink noise. The average performance using the pink and white noise of APR algorithm is shown in Table 2. The average classification performance with pink noise is similar to the performance with white noise. Also, the classification performance of KING database is similar to the speech in the speech/nonspeech audio database. Due to the deterioration of the pitch period, APR has a poor speech classification performance of 32.5% and 28.2% for white noise and pink noise at low SNR = −10 dB, respectively. APR maintains an overall accuracy above 93.0% for SNR above 0 dB.

3.3. Discrete-Valued SNR Estimator Results. The Discrete-Valued SNR Estimator was trained using one randomly selected talker from the KING database. Training involved 3 utterances in the training set, associated with this talker, with the 6 training SNR levels. Discrete-Valued SNR Estimator training and testing are conducted using a feature vector of 19 MFSCs and delta MFSCs. The minimum distance is used to classify test cases to one of the training environments as described in Section 2.4. Performance of the Discrete-Valued SNR Estimator is evaluated on the KING database test set, using the 11 testing SNR levels with white and pink noise.

Table 3 summarizes the results of the Discrete-Valued SNR Estimator classification. The total number of utterances for $j$thSNR value in the test set is equal to U_j, where $U_j = 347$ for $j = 1$ (clean environment) and $U_j = 2 \times 347 = 694$ for $j \neq 1$ (noisy environments). The elements of Table 3 Q_{ji} are the percentage of the total number of utterances U_j that were classified to the ith training SNR value.

The Discrete-Valued SNR Estimator output decision is assumed to be correct if the SNR of the selected training environment has the closest value to the SNR of the surrounding test environment or one of the adjacent environments in terms of SNR (these are set in bold in Table 3). For example if the surrounding environment has an SNR = 15 dB, then the output decision is assumed to be correct if the selected SNR = 10 or 20 dB (accuracy = 30.7% + 62.1% = 92.8%).

The SNR classification is not expected to require high precision, which justifies the inclusion of the adjacent SNR values in the definition of a correct decision. Overall, the average accuracy of the Discrete-Valued SNR Estimator is 98.4% overall accuracy.

3.4. Multiengine Selection Results. Six SI engines were trained using the SNR values of the training environments. Figure 7 plots the results of the 6 individual SI engines with two different noise types (white and pink noise) using the KING database. The maximum performance of a single SI occurs when the test data from a particular noisy environment is used with the SI engine trained in that environment. Also, the performance of SI engines trained in similar noise environments is better than those trained in environments that greatly differ.

The estimate of the SNR from the Discrete-Valued SNR Estimator is utilized to select the best performing SI engine for Multiengine Selection method. This method has average classification accuracy of 88.7% in noisy environments over an SNR range from 10 dB to clean environment. Compared to the accuracy using the SI engine trained in a clean environment, the improvement in accuracy was as high as 30.6%. Also the Multiengine Selection achieved an improvement in accuracy as high as 17% over an SNR range of −10 to 10 dB. Due to inaccuracies in the Discrete-Valued SNR estimator, the Multiengine Selection accuracy is slightly lower than the best individual SI engine at each SNR level. The Multiengine Selection maintains an accuracy above 75% for SNR above 10 dB, whereas the clean SI engine only does so for SNR above 30 dB.

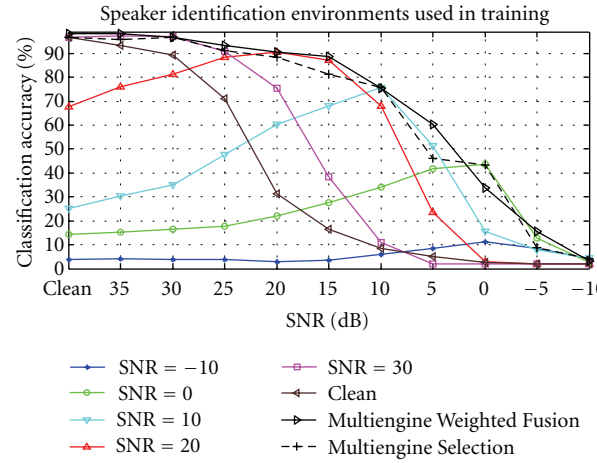

Figure 7: Speaker identification accuracy using the proposed multiengine methods.

3.5. Multiengine Weighted Fusion Results. Figure 7 plots the Multiengine Weighted Fusion SI engines' performance using the KING database. The estimate of the SNR from the Discrete-Valued SNR Estimator is utilized to weight and combine the SI decision of the 6 SI engines. Figure 7 shows that this Multiengine Weighted Fusion method has average classification accuracy of 91.3% in noisy environments over an SNR range of 10 dB to clean environment. The identification accuracy of the Multiengine Weighted Fusion method outperforms Multiengine Selection method (88.7%) and the identification accuracy using the SI engine trained in a clean environment (58.1%).

For SNR 5 ≥ dB, the classification accuracy of the Multiengine Weighted Fusion SI engines is superior to all individual SI engines. For example, at SNR = 15 dB, the engines trained at SNR = 10 and 20 dB have an accuracy of 68% and 87%, respectively, the Multiengine Selection classification accuracy has an accuracy of 81%, and the Multiengine Weighted Fusion has an accuracy of 89%. Naturally, the Multiengine Selection method cannot perform better than all individual SI engines and as previously mentioned is typically worse due to inaccuracies in the Discrete-Valued SNR estimator.

Table 2: Speech/nonspeech classification results (%).

Audio class	SNR (dB)					
	−10	0	10	20	30	Clean
White noise						
Nonspeech	84.0	95.2	98.1	98.3	98.6	98.6
Speech	32.0	93.2	96.8	97.3	97.3	97.3
KING	33.0	93.7	98.0	98.6	98.6	98.6
Average	58.2	94.3	97.7	98.1	98.3	98.3
Pink noise						
Nonspeech	87.0	94.0	97.5	97.8	98.4	98.6
Speech	27.1	90.0	96.0	95.9	95.9	96.7
KING	29.2	89.1	97.5	98.2	97.4	97.5
Average	57.5	91.7	97.1	97.8	97.9	97.9
Speech/nonspeech average	57.8	93.0	97.4	98	98.1	98.1

Table 3: Discrete-valued noise estimator classification results (%).

	Training env.					
Test env.	Clean	30	20	10	0	−10
Clean	**100.0**	0.0	0.0	0.0	0.0	0.0
35	**67.1**	**32.9**	0.0	0.0	0.0	0.0
30	**34.0**	**64.5**	**1.5**	0.0	0.0	0.0
25	10.1	**70.0**	**19.9**	0.0	0.0	0.0
20	0.0	**19.5**	**78**	**2.5**	0.0	0.0
15	0.0	7.2	**62.1**	**30.7**	0.0	0.0
10	0.0	0.0	**18.6**	**78.5**	**2.9**	0.0
5	0.0	0.0	0.0	**29.8**	**70.2**	0.0
0	0.0	0.0	0.0	**0.0**	**97.1**	**2.9**
−5	0.0	0.0	0.0	0.0	**2.9**	**97.1**
−10	0.0	0.0	0.0	0.0	**0.0**	**100**

The Multiengine Weighted Fusion method has higher complexity than the Multiengine Selection method, albeit only slightly higher complexity associated with (11), (13), and (15). Meanwhile, the Multiengine Weighted Fusion method has higher classification accuracy.

For variable environments with a known SNR set, the SI training environments can be matched with the testing environments. In cases, where the training and testing environments are the same (SNR = 30, 20, 10, 0, and −10 dB and clean environment), the Multiengine Selection method has almost similar accuracy as the Multiengine Weighted Fusion method. This makes the Multiengine Selection method perhaps a better solution because of its lower complexity; however, when the testing environments are different from the training environments (SNR = 35, 25, 15, 5, −5 dB), the classification performance enhancement of the Multiengine Weighted Fusion is 7.2% compared to the Multiengine Selection. This suggests that the Multiengine Weight Fusion is a more generalizable solution.

For high-SNR environments (SNR ≥ 30 dB), it seems there is no benefit for a multiengine system. Indeed, the engine trained at SNR = 30 dB has similar accuracy as the multiengine systems (Figure 7); however, it is clearly not robust as there is severe degradation in performance for SNR < 30 dB.

3.6. Speech/Nonspeech Classification with Multiengine SI. The overall adaptive system accuracy (with SNR estimator and APR) is compared with a nonadaptive system (clean SI engine and PR threshold = 0.32) (Figure 8) using the KING database. The main contributor of the accuracy enhancement of the introduced instrument at SNR above 10 dB is the Multiengine SI system (from Table 2 the performance of the APR is nearly perfect for SNR above 10 dB). The per formance curve of the Multiengine Selection, shown in Figure 8, has a large dip at SNR = 5 dB. For SNR below 10 dB the performance curve could be smoothed by decreasing the SNR spacing between SI engines.

The classification accuracy performance using a perfect SNR estimator with the APR and Multiengine Selection is shown in Figure 8. Using this SNR value, the best-performing SI engine and the APR threshold are selected. Compared to APR with Multiengine Weighted Fusion using the Discrete-Valued SNR Estimator, the improvement in accuracy is only 1.6% over SNR range clean to −10 dB. This

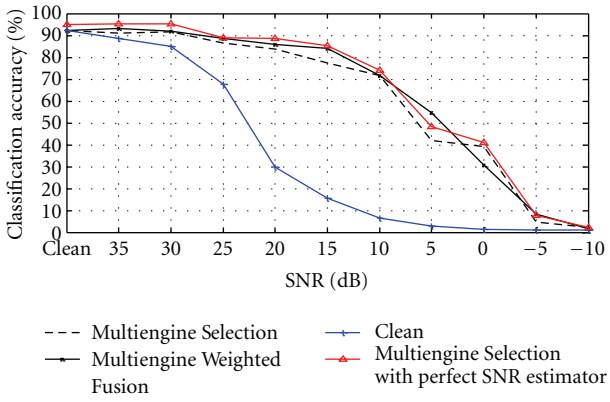

Figure 8: Overall accuracy of the introduced system.

result justifies the utility of the rough estimation of the SNR value.

4. Conclusion

This paper confirms that the closer the match between the training and testing environment the higher the SI accuracy. This paper proposed a security instrument that can detect location and identity of talker in noisy environment. To achieve this instrument, an adaptive multiengine speaker identification system and Discrete-Valued SNR Estimator are proposed. The Discrete-Valued SNR Estimator is also utilized to adapt the speech/nonspeech classifier. The proposed instrument is reliable and robust even in variable noise environments.

In the multiengine SI system, six speaker identification engines were utilized to accommodate five noisy environments in addition to clean environment. The SI system using the Multiengine Selection and Multiengine Weighted Fusion methods achieved average identification accuracies of 88.7% and 91.3%, respectively; this represents an enhancement of 30.6% and 33.2% over the SI engine trained in a clean environment, respectively, for SNR $\geq$ 10 dB. Also the Multiengine Selection and the Multiengine Weighted Fusion achieved an improvement in accuracy as high as 17% and 18.5% over an SNR range of -10 to 10 dB. While the Multiengine Weighted Fusion has high accuracy and reliability than the Multiengine Selection, the tradeoff is a slightly higher complexity.

In future work, this framework of this system (i.e., the combination of the Discrete-Value SNR Estimator and a Multiengine Fusion method) can be deployed in other applications, such as biomedical applications. An example of a biomedical application area that can benefit from the proposed instrument is hearing aid devices. Hearing aids could utilize the proposed instrument to customize their parameters to achieve an optimal hearing performance in variable environments. Modern hearing aids allow the user to set their internal equalizer or beam former variables to predefined settings by selecting the surrounding environment. The deployment of the proposed instrument in hearing aids can automate the selection of the predefined settings according to the surrounding environment.

Acknowledgment

The authors would like to acknowledge the funding from CITO, Mitel, NSERC, and OGS.

References

[1] H. Gish, "Robust discrimination in automatic speaker identification," in *IEEE International Conference on Acoustics, Speech, and Signal Processing: Speech Processing*, pp. 289–292, Albuquerque, NM, USA, April 1990.

[2] A. El-Solh, *Evaluation of speech enhancement techniques for speaker recognition in noisy environments*, M.S. thesis, Carleton University, 2006.

[3] A. El-Solh, A. Cuhadar, and R. A. Goubran, "Evaluation of speech enhancement techniques for speaker identification in noisy environments," in *9th IEEE International Symposium on Multimedia-Workshops*, pp. 235–239, Taichung, Taiwan, December 2007.

[4] J. Ming, T. J. Hazen, J. R. Glass, and D. A. Reynolds, "Robust speaker recognition in noisy conditions," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, no. 5, pp. 1711–1723, 2007.

[5] N. Kitaoka, S. Hamaguchi, and S. Nakagawa, "Noisy speech recognition based on integration/selection of multiple noise suppression methods using noise GMMs," *IEICE Transactions on Information and Systems*, vol. E91-D, no. 3, pp. 411–421, 2008.

[6] R. Gomez and T. Kawahara, "Optimizing spectral subtraction and Wiener filtering for robust speech recognition in reverberant and noisy conditions," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '10)*, pp. 4566–4569, Dallas, Tex, USA, March 2010.

[7] International Telecommunications Union (ITU), *Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs*, Telecom Standardization Section, ITU-T, 2001.

[8] T. Yamada, M. Kumakura, and N. Kitawaki, "Performance estimation of speech recognition system under noise conditions using objective quality measures and artificial voice," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 6, pp. 2006–2013, 2006.

[9] A. R. Abu-El-Quran, R. A. Goubran, and A. D. C. Chan, "Security monitoring using microphone arrays and audio classification," *IEEE Transactions on Instrumentation and Measurement*, vol. 55, no. 4, pp. 1025–1032, 2006.

[10] S. Ntalampiras, I. Potamitis, and N. Fakotakis, "Probabilistic novelty detection for acoustic surveillance under real-world conditions," *IEEE Transactions on Multimedia*, vol. 13, no. 4, pp. 713–719, 2011.

[11] A. M. Vainio, M. Valtonen, and J. Vanhala, "Proactive fuzzy control and adaptation methods for smart homes," *IEEE Intelligent Systems*, vol. 23, no. 2, pp. 42–49, 2008.

[12] L. Wang, N. Kitaoka, and S. Nakagawa, "Robust distant speech recognition by combining multiple microphone-array

processing with position-dependent CMN," *EURASIP Journal on Applied Signal Processing*, vol. 2006, Article ID 95491, 11 pages, 2006.

[13] E. Mabande, A. Schad, and W. Kellermann, "Design of robust superdirective beamformers as a convex optimization problem," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '09)*, pp. 77–80, Taipei, Taiwan, April 2009.

[14] A. R. Abu-El-Quran and R. A. Goubran, "Adaptive pitch-based speech detection for hands-free applications," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)*, pp. 305–308, March 2005.

[15] F. Bimbot, I. Magrin-Chagnolleau, and L. Mathan, "Second-order statistical measures for text-independent speaker identification," *Speech Communication*, vol. 17, no. 1-2, pp. 177–192, 1995.

[16] R. D. Zilca, "Text-independent speaker verification using covariance modeling," *IEEE Signal Processing Letters*, vol. 8, no. 4, pp. 97–99, 2001.

[17] J. Pinquier, J. L. Rouas, and R. André-Obrecht, "A fusion study in speech/music classification," in *IEEE International Conference on Accoustics, Speech, and Signal Processing*, pp. 17–20, Hong Kong, April 2003.

[18] J. S. Gammal and R. A. Goubran, "Combating reverberation in speaker verification," in *Proceedings of the IEEE Instrumentation and Measurement Technology Conference (IMTC '05)*, pp. 687–690, Ottawa, Canada, May 2005.

[19] E. Nemer, R. Goubran, and S. Mahmoud, "SNR estimation of speech signals using subbands and fourth-order statistics," *IEEE Signal Processing Letters*, vol. 6, no. 7, pp. 171–174, 1999.

[20] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 1, pp. 72–83, 1995.

[21] P. J. Barger and S. Sridharan, "On the performance and use of speaker recognition systems for surveillance," in *IEEE International Conference on Video and Signal Based Surveillance (AVSS '06)*, Sydney, Australia, November 2006.

[22] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Transactions on Communications Systems*, vol. 28, no. 1, pp. 84–95, 1980.

[23] National Institute of Standards and Technology (NIST), "Brief Description of the KING Speech Data Base," 1992.