Technical Report OSU-CISRC-11/11-TR36
Department of Computer Science and Engineering
The Ohio State University
Columbus, OH 43210-1277

Ftpsite: **ftp.cse.ohio-state.edu**
Login: **anonymous**
Directory: **pub/tech-report/2011**
File: **TR36.pdf**
Website: **http://www.cse.ohio-state.edu/research/techReport.shtml**

A CASA based system for SNR estimation

Arun Narayanan
Department of Computer Science and Engineering
The Ohio State University, Columbus, OH 43210, USA
narayaar@cse.ohio-state.edu

DeLiang Wang
Department of Computer Science and Engineering & Center for Cognitive Science
The Ohio State University, Columbus, OH 43210, USA
dwang@cse.ohio-state.edu

Abstract – We present a system for robust signal-to-noise ratio (SNR) estimation based on computational auditory scene analysis (CASA). The proposed algorithm uses an estimate of the Ideal Binary Mask to segregate a time-frequency representation of the noisy signal into speech dominated and noise dominated regions. Energy within each of these regions is summated to derive the *filtered* global SNR. An SNR transform is introduced to convert the estimated filtered SNR to the true broadband SNR of the noisy signal. The algorithm is further extended to estimate subband SNRs. Evaluations are done using the TIMIT speech corpus and the NOISEX-92 noise database. Results indicate that both global and subband SNR estimates are superior to those of existing methods, especially at low SNR conditions.

Index Terms – Signal-to-noise ratio, broadband SNR, subband SNR, computational auditory scene analysis (CASA), ideal binary mask (IBM).

1 Introduction

Estimation of the signal-to-noise ratio has been studied for decades, mostly in the context of noise estimation, speech enhancement and robust automatic speech recognition (ASR). Typical algorithms estimate local or instantaneous SNR, i.e. the SNR at a particular time-frequency (T-F) unit (also referred to as short-time subband SNR) [18], which can then be directly used by speech enhancement algorithms [2] (see [20] for a comprehensive review). Two assumptions made by these algorithms are: 1) the background noise is stationary, at least between speech pauses and during the time interval when the noise energy is estimated (or updated) and 2) regular speech pauses occur in speech. For the estimation to be effective, the interval size should be chosen wisely. Longer intervals are suited for tracking stationary background noises. When noise statistics change quickly, a shorter interval is preferred. But using a shorter interval reduces the chance of seeing noise-only frames. In realistic noise conditions, such as the so called *cocktail-party* condition, most estimation techniques falter [20].

While most algorithms perform short-time subband SNR estimation, knowledge of the SNR at other levels is also useful. Global SNR of an utterance, for instance, can be used to devise SNR specific speech and speaker recognition strategies [6, 26]. In many applications, speech processing algorithms are optimized to function in certain specific SNR conditions. An SNR estimator can be used in such applications during the model selection process at runtime. Similarly, subband SNR estimates are useful in many speech processing tasks.

The main theme of this paper is to estimate broadband and subband global SNRs, i.e. SNRs at the utterance level. Typical utterance length is between 2–5 seconds (e.g., the utterances in the TIMIT core test set [7]). Traditional SNR estimation algorithms have difficulties dealing with such long intervals of speech when the underlying noise is non-stationary. Algorithms have been proposed for global broadband SNR estimation. They are based on identifying the noise and speech energy distributions [1, 4], or signal statistics [14].

We take a CASA-based approach for SNR estimation. A main goal of CASA is to estimate the ideal binary mask (IBM) [24], which identifies speech dominated and noise dominated units in a T-F representation of noisy speech. The IBM has been shown to be effective in improving speech intelligibility and robust automatic speech and speaker recognition in noise [25]. Motivated by this line of research, we propose to use the IBM to calculate both broadband and subband SNRs. Although IBM estimation algorithms are commonly based on short-time SNR estimation [13, 17], few have used the IBM to estimate the SNR of mixture signals. The proposed algorithm works under the assumption that at the utterance level, the total speech and noise energy can be well approximated using only the speech dominant and the noise dominant T-F units, respectively.

The remainder of the paper is organized as follows. In Section 2 we discuss existing SNR estimation strategies from the literature. A detailed description of our system is provided in

Section 3. Evaluation results are described in Section 4. We conclude with a discussion of our results in Section 5.

2 Prior Work

We first discuss short-time subband SNR estimation algorithms. In [9], histograms of the noisy spectral magnitudes are analyzed to estimate the noise energy in each frequency band. Since the histogram of the spectral magnitudes of clean speech typically has a peak close to 0, any shift in this peak is attributed to the stationary background noise. The author suggests using a time interval of 500 msec to best follow the changes in non-stationary noise conditions. The system is further developed in [10] to include a dynamic threshold that helps avoid overestimation of the noise level in situations when the above assumption is not met (typically in low frequency bands where the speech energy is relatively high). In the same paper, they also introduce a weighted average method in which the noise energy is calculated as the weighted sum of the past spectral energy values, along with the use of the dynamic threshold to prevent overestimation. An alternative approach based on minimum statistics was proposed in [18], where noise energy is estimated by tracking the low-energy envelope of the signal with the assumption that the energy minima occur during speech pauses. While estimating the SNR, the estimated noise energy is scaled to account for the fact that only the energy minima are considered in the first stage. Ris and Dupont [20] introduced a harmonic filtering method that makes use of low-energy valleys between formants in voiced intervals to update the noise estimate in the absence of regular speech pauses. Other strategies include energy clustering to distinguish speech and noise portions of the mixture [3,4,22], and explicit speech pause or voice-activity detection (VAD) [16]. Nemer et al. [19] make use of higher order statistics of speech and noise, assuming a sinusoidal model for band restricted speech and a Gaussian model for noise. Supervised classification based methods have also been applied to this task. For example, features inspired from psychoacoustics and an MLP based classifier are used in [15, 21] to estimate broadband and subband SNRs in short intervals of noisy speech.

Global SNR estimation has also been studied, although not as widely as short-time subband SNR estimation. A commonly used algorithm from NIST [1] builds a histogram of short-time signal power using the noisy utterance which is then used to infer noise and noisy speech distributions. With the assumption that the noisy signal has bimodal distribution, it fits a raised cosine function to the left hand peak and uses its mean as the mean noise power. The learned cosine function is then subtracted from the histogram distribution and the remainder is assumed to be the distribution of speech power. From these distributions, the peak signal-to-noise ratio is calculated rather than the mean SNR. The peak SNR is clearly an overestimate of the true SNR. Dat et al. [4] use a similar approach, but instead of fitting the histogram, they fit a 2-component Gaussian to the data using the expectation maximization (EM) algorithm. A similar approach was also used in [22] to model speech. Dat et al. extend the idea by using

the learned Gaussians in a principled way to derive the SNR of the signal. Similar to [1], their approach would have problems when the bimodal Gaussian assumption fails. It is also very sensitive to initialization and the stopping criterion used by the EM algorithm. In cases when the noise variance is estimated to be larger than that of clean speech, the algorithm can be numerically unstable. The method by Kim and Stern [14] is based on waveform amplitude distribution. It assumes that clean and noisy speech have Gamma distributions, and noise a Gaussian distribution. It infers the global SNR based on the parameter of the distribution estimated from noisy speech. Their algorithm works well when these assumptions are met. Performance degradation occurs at low SNR conditions and when the background noise has non-Gaussian characteristics. An alternative, relatively straightforward approach would be to use speech enhancement algorithms to estimate the noise power spectral density (PSD) [8] and the speech magnitude [5]. Assuming that the noise PSD approximates noise energy, which is reasonable, both global broadband and subband SNRs can be directly calculated from these estimates.

Long-term subband SNR estimation is not much studied, but global SNR estimation strategies can be extended to perform subband SNR estimation. The technique in [1] has a subband SNR estimation algorithm that is based on the same principle as broadband SNR estimation. It is fairly easy to extend the methods in [14] and [4], and speech enhancement based strategies to perform subband SNR estimation. A supervised learning approach to estimate long-term subband SNRs was proposed by Kleinschmidt and Hohmann [15]. Being supervised, it is very likely that the algorithm is dependent on training conditions. The authors note that their algorithm works better for stationary noise types compared to non-stationary noise types.

A system related to ours is the one described in [12] (referred to as the Hu-Wang'10 system). It estimates the SNR using a binary mask for only the voiced speech frames, by making the following assumptions: 1) the total voiced speech energy is approximately equal to the total noisy signal energy under the unmasked, speech dominant (1s in the voiced IBM) T-F units, 2) the total signal energy can be inferred from the total voiced signal energy, and 3) the per-frame noise energy in both voiced and unvoiced frames remains unchanged. Their system produces reasonable results at SNRs close to 0 dB but biased estimates at other conditions. Since only the voiced IBM is used, estimating subband SNRs will be challenging, especially at high frequencies. In addition to providing a novel framework for SNR estimation, our algorithm differs from the Hu-Wang'10 system since we use an estimate of the IBM in both voiced and unvoiced time frames.

3 System Description

The architecture of the proposed system is shown in Figure 1. The input to the system is a noisy speech signal, which is first processed using a 128-channel gammatone filterbank to perform T-F decomposition. The center frequencies of the filterbank are uniformly spaced in

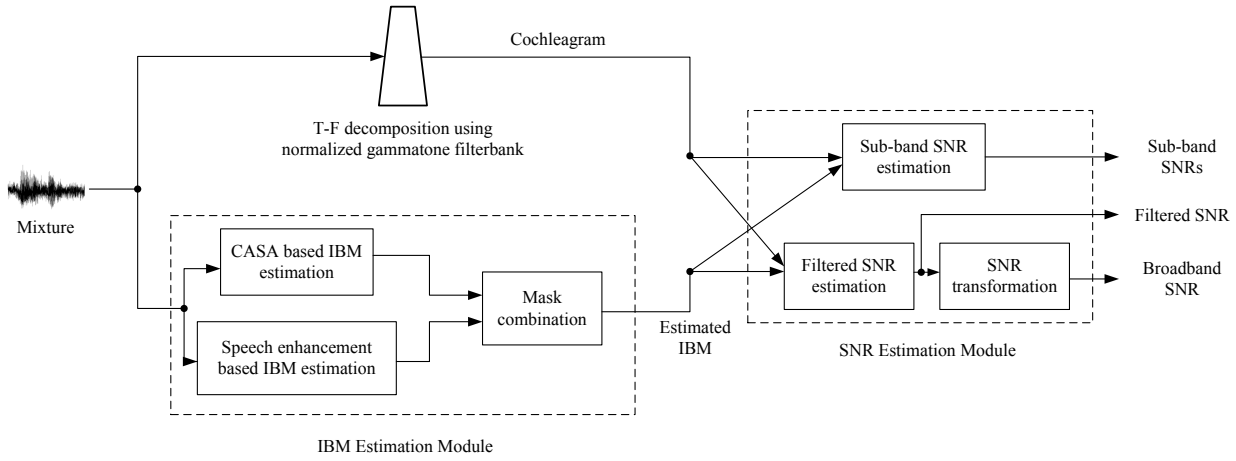

Figure 1: Schematic diagram of the proposed system. The input to the system is a noisy mixture. The outputs are the broadband SNR, filtered SNR and subband SNRs. The system includes an IBM estimation module and an SNR estimation module.

the ERB (Equivalent Rectangular Bandwidth) rate scale from 50 Hz to 8000 Hz [25]. The signals are sampled at 16 kHz in our experiments, and the chosen frequency range ensures that almost all useful speech information is retained in the filtered signal. A typical gammatone filterbank performs loudness equalization across frequencies to match cochlear filtering. As a result, different frequency components are scaled differently. This may alter the SNR of the filtered signal compared to the original signal in the time domain, even if the signal is band limited to 50–8000 Hz. In order to prevent this undesired effect, we normalize the gammatone filterbank. The normalized gammatone filterbank scales most of the frequency components covered by the filterbank so as to ensure that for speech signals, the filtered signal energy approximately equals its total time-domain energy. This may not be the case for noise and noisy speech signals if the underlying noise type has significant energy in the low-frequency range (e.g., the car interior noise from the NOISEX-92 corpus [23]). We will make use of the normalized filterbank in the subsequent SNR transformation step to estimate the true broadband SNR of a noisy signal, given its filtered SNR. Figure 2 compares the aggregated magnitude responses of the conventional gammatone filterbank and the normalized gammatone filterbank.

After T-F decomposition, the filtered signal is windowed using a 20 msec rectangular frame with a 10 msec frame shift. A *cochleagram* [25] of the signal is then created by calculating the signal energy within each of these windows. Because of the 50% overlap between adjacent frames, the total energy within the cochleagram will roughly be twice the energy of the speech signal in the time domain.

Let $y(t)$, $x(t)$ and $n(t)$ represent the noisy signal, clean signal and noise signal, respectively, and $\mathbf{Y}$, $\mathbf{X}$ and $\mathbf{N}$ their corresponding cochleagrams. Since noise is assumed to be additive and

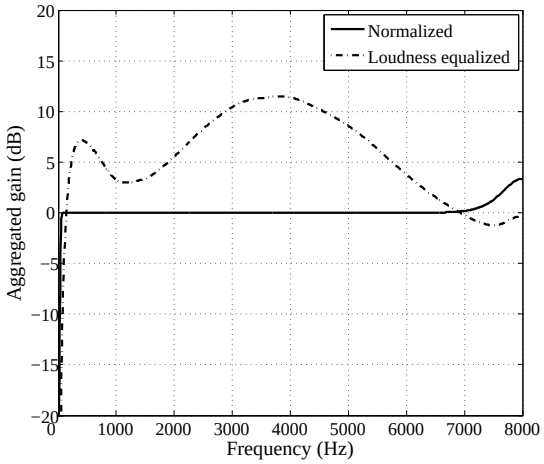

Figure 2: Aggregated magnitude response of the normalized and loudness equalized gammatone filterbank. The gain for a specific frequency is calculated by aggregating the gains across the 128 filters of the filterbank. Notice that most frequency components undergo no attenuation/amplification when processed using the normalized gammatone filterbank.

independent of speech, the following relationships hold:

$$\begin{aligned} y(t) &= x(t) + n(t), \\ \mathbf{Y}(m,c) &\approx \mathbf{X}(m,c) + \mathbf{N}(m,c). \end{aligned}$$

Here, t denotes a time sample, m indexes a time frame and c a frequency channel. We define the following SNRs of the signal:

$$SNR_b = 10\log_{10}\left(\frac{\sum_t (x(t))^2}{\sum_t (n(t))^2}\right), \quad (1)$$

$$SNR_f = 10\log_{10}\left(\frac{\sum_{m,c} \mathbf{X}(m,c)}{\sum_{m,c} \mathbf{N}(m,c)}\right), \quad (2)$$

$$SNR_c = 10\log_{10}\left(\frac{\sum_{m} \mathbf{X}(m,c)}{\sum_{m} \mathbf{N}(m,c)}\right), \quad (3)$$

where SNR_b, SNR_f and SNR_c denote the broadband SNR, the filtered SNR and the subband SNR, respectively.

Since we only have access to $y(t)$ and $\mathbf{Y}$ in practice, to calculate these SNRs, we approximate the total target speech and noise energy using $\mathbf{Y}$ and an estimated IBM. The IBM is a two-dimensional binary matrix, with the same dimensionality as $\mathbf{Y}$. An element in the matrix takes the value 1 if the speech energy within the corresponding T-F unit is greater than the

noise energy. Formally, the IBM is defined as:

$$IBM(m,c) = \begin{cases} 1 & \text{if } \mathbf{X}(m,c) > \mathbf{N}(m,c) \\ 0 & \text{otherwise} \end{cases} . \quad (4)$$

Note that the IBM can also be defined in terms of a local SNR threshold at each T-F unit called the local criterion (LC). The above formulation implies an LC of 0 dB. In the proposed system, a binary mask is estimated using the IBM estimation module. Given the estimated IBM and the cochleagram of the input signal, the SNRs are estimated by the SNR estimation module (Figure 1). These modules are described in the following subsections.

3.1 IBM Estimation

We aim to develop a system that generalizes well to different test conditions. Therefore, we estimate the IBM by combining a CASA algorithm that has been shown to work well in a wide range of conditions and a state-of-the-art speech enhancement algorithm. The CASA algorithm is based on the Hu-Wang'11 system described in [13]. It uses the tandem algorithm [11] to estimate the voiced IBM (the IBM in voiced frames) and a spectral subtraction based method to estimate the unvoiced IBM. The tandem algorithm is an iterative procedure that estimates both the target pitch and the corresponding binary mask for up to two voiced sound sources in the signal. It returns more than one pitch estimate for a single frame of noisy speech if the background is pitched as in multi-talker babble or bird chirp. The tandem algorithm does not link disjoint pitch contours, which is the task of sequential organization. Since we only deal with non-speech noise, multiple pitch points are typically detected only for a fraction of frames. Sequential organization in non-speech noise conditions is a relatively easy task compared to multi-talker conditions. In this work, we perform sequential organization based on: 1) plausible pitch range of speech, 2) length of a pitch contour, and 3) pitch continuity. The binary masks corresponding to the sequentially grouped pitch contours are then grouped to obtain an estimate of the voiced IBM. Given the voiced mask, the Hu-Wang'11 system estimates the unvoiced IBM by first removing periodic components from the mixture signal. It then forms a noise estimate for each unvoiced interval by averaging the energy within the noise dominant T-F units (0s in the mask) of its neighboring voiced intervals. These estimates are finally used in spectral subtraction to obtain the estimated unvoiced IBM. Figure 3(c) shows an estimated IBM obtained in this fashion. It captures most of the voiced segments (T-F regions) and a good number of unvoiced segments. Comparing with the IBM shown in Figure 3(f), we can see that it still misses a few target-dominant segments. The goal of the speech enhancement based mask estimation module described below is to recover these segments.

The motivation behind using a speech enhancement method is that such algorithms work well when the SNR is high, whereas CASA algorithms are usually designed for low SNR

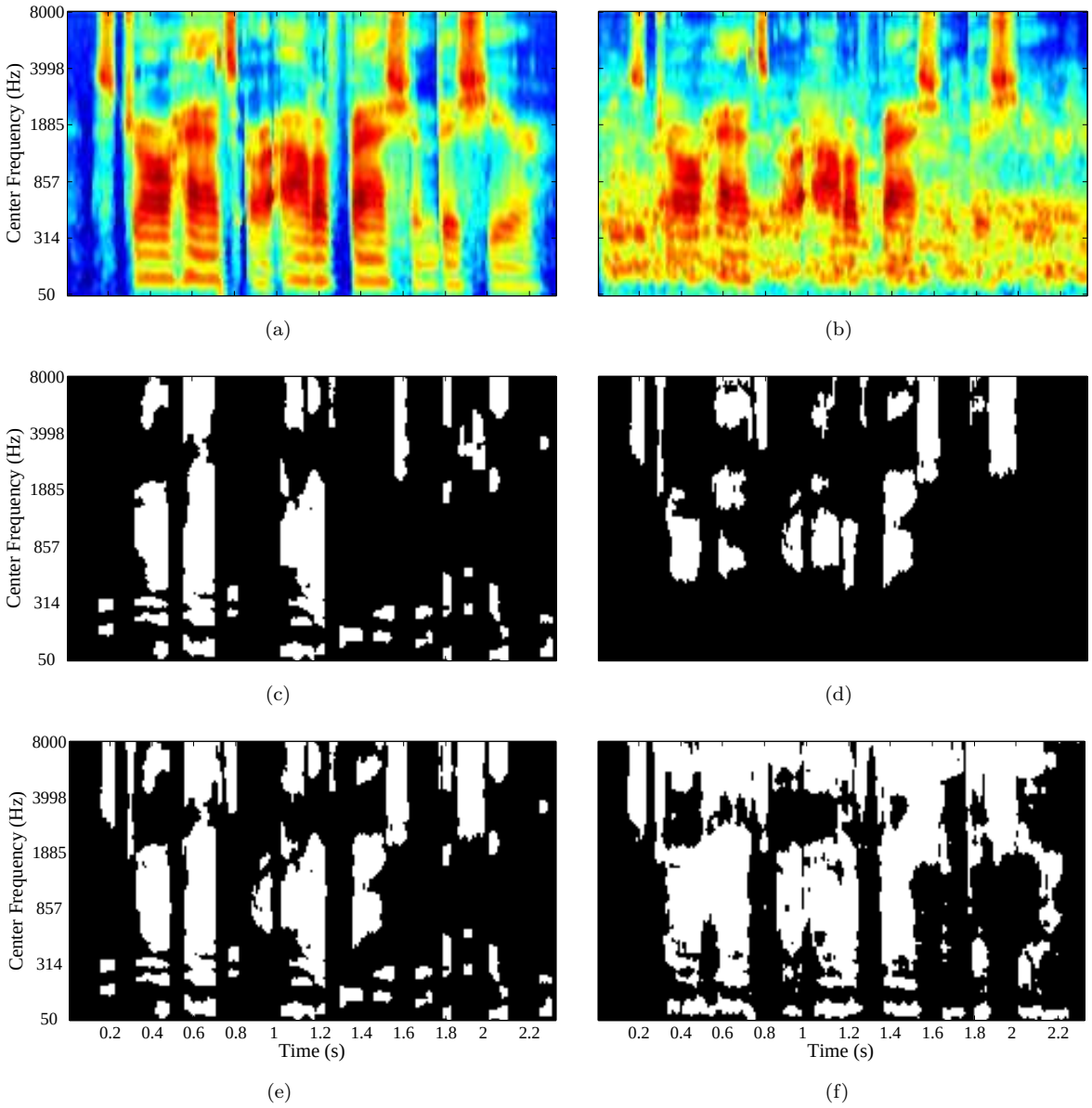

Figure 3: IBM estimation. (a) Cochleagram of the utterance 'Straw hats are out of fashion this year' from the core test set of the TIMIT corpus. (b) Cochleagram of the same utterance mixed with babble noise, where the filtered SNR is set to 5 dB. (c) The mask estimated by the Hu-Wang'11 system. (d) The mask estimated by a speech enhancement method. (e) The mask obtained by combining the two methods. (f) The IBM.

conditions. The speech enhancement mask estimation is based on a state-of-the-art noise tracking algorithm described in [8]. The algorithm operates in the linear frequency domain, using the FFT to perform T-F decomposition. To estimate the noise power, it uses an MMSE estimator of noise magnitude-squared DFT coefficients assuming that both speech and noise DFT coefficients follow a complex-Gaussian distribution. The speech DFT coefficients are estimated using the algorithm in [5], which assumes that speech magnitude-DFT coefficients follow a generalized Gamma distribution with parameters $\gamma = 1$ and $\nu = 0.6$. Given these

estimates, the noise and speech energy within a time-frequency unit are approximated as the estimated noise power and squared-magnitude of the estimated speech DFT coefficient. These estimates are then transformed to the nonlinear frequency domain of the gammatone filterbank using the frequency response of the individual gammatone filters:

$$\widehat{\mathbf{X}}(m,c) = (1/K)\sum_{k=0}^{K-1}(\widehat{\mathbf{X}}_{FFT}(m,k).|G_c(k)|^2). \tag{5}$$

Here, $\widehat{\mathbf{X}}$ is an estimate of $\mathbf{X}$, $\widehat{\mathbf{X}}_{FFT}$ the estimated speech energy in the linear frequency domain and G_c the frequency response of the filter channel c. Index k denotes a DFT coefficient and K the number of DFT bins used for T-F analysis, which is set to 512 in our experiments. A similar equation is used to estimate $\mathbf{N}$. The IBM is finally estimated by calculating the SNR at each T-F unit using $\widehat{\mathbf{X}}$ and $\widehat{\mathbf{N}}$, and comparing it to LC. Since we use the speech enhancement method to capture segments having high SNR, we set LC to a value greater than 0, unlike (4). This also helps to reduce false alarms (0s wrongly labeled as 1s) in IBM estimation. The optimal value for LC is chosen using a small development set of noisy mixtures. Figure 3(d) shows a binary mask estimated in this way. Clearly, it captures some target dominant segments missed in Figure 3(c).

To combine the two masks, we use the simple logical 'OR' operation. Figure 3(e) shows the final mask estimated by our algorithm. The final mask is more similar to the IBM than each of the individual masks.

3.2 SNR Estimation

For SNR estimation, we assume that the total filtered target energy, both at the broadband and the subband level, can be estimated using only the speech dominant T-F units and the total filtered noise energy from the noise-dominant T-F units. As shown in the evaluations, this assumption is reasonable for long-term SNR estimation.

3.2.1 Global SNR Estimation

Given an estimated IBM ($\mathbf{M}$), the total speech and noise energy are estimated as follows:

$$\widehat{E}_{speech} = \sum_{m,c}\mathbf{Y}(m,c).\mathbf{M}(m,c), \tag{6}$$

$$\widehat{E}_{noise} = \sum_{m,c}\mathbf{Y}(m,c).\neg\mathbf{M}(m,c), \tag{7}$$

where '¬' denotes the 'NOT' operation. The filtered SNR ($\widehat{SNR}_f$) is then estimated as shown below, using these estimates:

$$\widehat{SNR}_f = 10\log_{10}\left(\frac{\widehat{E}_{speech}}{\widehat{E}_{noise}}\right). \tag{8}$$

The true broadband SNR is estimated by transforming $\widehat{SNR}_f$ using an SNR transformation step. We transform the SNR based on the following observation. Recall that when speech signals are processed using the normalized gammatone filterbank, the total signal energy is not significantly altered since it applies a unit gain to most of the useful bands. Therefore, the difference between the energy of the noisy signal in the time domain and its energy after T-F decomposition using the normalized gammatone filterbank can mostly be attributed to noise. This is true especially at low SNRs, where noise energy is comparable to or greater than the target energy. With this observation, the true broadband SNR can be calculated by compensating the noise energy with this difference during SNR estimation:

$$\triangle\widehat{E} = 2\sum_t (y(t))^2 - \sum_{m,c}\mathbf{Y}(m,c), \tag{9}$$

$$\widehat{SNR}_b = 10\log_{10}\left(\frac{\widehat{E}_{speech}}{\widehat{E}_{noise} + max(0, \triangle\widehat{E})}\right). \tag{10}$$

$\widehat{SNR}_b$ is the estimated broadband SNR of the noisy signal.

3.2.2 Subband SNR Estimation

The subband SNRs are estimated similar to (8), but the energy values are summated only across time:

$$\widehat{SNR}_c = 10\log_{10}\left(\frac{\sum_m \mathbf{Y}(m,c).\mathbf{M}(m,c)}{\sum_m \mathbf{Y}(m,c).\neg\mathbf{M}(m,c)}\right). \tag{11}$$

$\widehat{SNR}_c$ denotes the estimated subband SNR for frequency channel c.

4 Evaluation Results

We start by describing the experimental setup in Section 4.1. Since the idea of using *binary* masks for SNR estimation is relatively new, we provide an initial set of results using the IBM directly in Section 4.2. We also present another set of results in the same subsection that highlights the role of the SNR transformation step. This is followed by a description of the results using the estimated IBMs and comparisons in Section 4.3.

4.1 Experimental Setup

All our experiments are conducted using the TIMIT speech corpus [7] and the NOISEX92 noise database [23]. Specifically, the experimental results are obtained on the core test set of the TIMIT database which consists of 192 clean speech utterances from 24 speakers recorded at 16 kHz. Four noises are chosen from the NOISEX92 database – white noise, car noise, babble noise and factory noise. The first two noises are stationary and the last two relatively non-stationary. Car noise is chosen as it has a considerable amount of low frequency energy as a result of which the broadband and the filtered SNRs are quite different, thereby enabling us to measure the performance of the proposed algorithm in estimating these SNRs more thoroughly. The noise signals are downsampled to 16 kHz to match the sampling rate of the speech signals.

Two test sets are created for evaluating the performance of the proposed system in estimating the broadband SNR and the filtered SNR separately. Both test sets consist of the 4 noises mixed with clean speech at 6 SNR conditions ranging from -10 dB to 15 dB, in increments of 5 dB. To create a noisy signal, a randomly selected segment of the noise is scaled to the desired level and added to the speech signal. Depending on the test set, either the broadband SNR or the filtered SNR is set to the desired SNR level. The test set created by controlling the filtered SNR is also used to evaluate subband SNR estimation.

The broadband and filtered SNR estimation results are compared with those of the following systems. The first one is the SNR estimation algorithm (WADA) proposed in [14], which was shown to significantly outperform the algorithm from NIST [1]. The second system uses the noise and speech magnitude estimates obtained using the speech enhancement algorithms [5, 8] directly to estimate the SNR (HND). The frame length and the frame shift are set to 20 msec and 10 msec, respectively, to match those used by our algorithm. The remaining approaches are based on estimated IBMs. The Hu-Wang'10 system [12] is the third, and is slightly modified so as to make use of the normalized filterbank and the SNR transform. These modifications improve the performance reported in [12]. The forth method transforms the estimates from the speech enhancement algorithms [5, 8] to the nonlinear frequency domain using (5), and estimates the IBM by setting an appropriate LC (see (4)). The estimated IBM is then used to calculate the SNRs in the same way as the proposed system, and we denote this method HND_MOD. As the fifth method we use the IBM estimated using only the CASA system described in Section 3.1; this method is denoted Hu-Wang'11. Note that the only difference between HND_MOD, Hu-Wang'11 and our method is in the way the IBM is estimated. So an improvement in performance is solely attributed to improved IBM estimation.

WADA and HND make use of all the frequencies of the signal to estimate the SNR. Therefore, before estimating the filtered SNR using these algorithms, the original mixture is processed using a filter that has a frequency response similar to the aggregated response of the

gammatone filterbank (see Figure 2). These algorithms then calculate the broadband SNR using the filtered signal, which is equivalent to estimating the filtered SNR of the signal.

A development set is created by randomly choosing 30 utterances from the training set of the TIMIT corpus to tune the following parameters: 1) LC that is used to estimate the speech enhancement mask, as described in Section 3.1. 2) LC used by HND_MOD. Values ranging from 0 dB to 10 dB in 1-dB steps are tested and the one that gives the best performance in terms of SNR estimation on the development set is finally used. To create the development set, the 30 selected utterances are mixed with all 4 noises at all 6 SNR conditions. The final chosen LC values are 8 dB for computing speech enhancement based binary mask, and 0 dB for HND_MOD.

Subband SNRs are estimated across the frequency bands of a 64-channel gammatone filterbank, which is a typical number of channels used in CASA systems. Among the algorithms described earlier, only modified versions of WADA and HND are compared with the proposed subband SNR estimation algorithm. As described in Section 2, WADA assumes that speech is Gamma distributed with a fixed parameter $\alpha = 0.4$. Although this holds for broadband signals, we have noticed that this value does not hold for band-limited signals. Therefore, the 30-utterance development set is used to find an optimal α for each subband. This is done by fitting a Gamma distribution to the clean subband signal amplitudes (in the maximum-likelihood sense). The mean α for the 30 utterances for each channel is then chosen as the final parameter for that channel. HND is adapted to estimate subband SNRs in the domain defined by the gammatone filterbank by transforming energy estimates using (5). The IBM estimation module of the proposed algorithm estimates a 128-channel mask. Instead of re-estimating a 64-channel mask for the purpose of subband SNR estimation, we sub-sample this mask to 64 channels. This is reasonable because the center frequencies (c) of the 64-channel gammatone filterbank and those of the odd numbered channels ($2c - 1$) of the 128-channel gammatone filterbank are identical, since both of them are uniformly distributed in the ERB rate scale. Sub-sampling is done by additionally accounting for the wider bandwidths of filters in the 64-channel filterbank; a T-F unit, $\mathbf{M}_{64}(m, c)$, in the 64-channel mask is labeled 1 only if at least 2 out of the 3 corresponding T-F units, $\mathbf{M}_{128}(m, 2c - 2))$, $\mathbf{M}_{128}(m, 2c - 1)$ and $\mathbf{M}_{128}(m, 2c)$, in the 128-channel mask are speech dominant. The subband SNRs are restricted to the range of -20 dB to 30 dB, i.e. any estimate not falling in this range is rounded to the boundary values.

Estimated SNR values from each of these algorithms are rounded to the nearest integer before calculating error metrics. In the case of broadband/filtered SNR estimation, the mean absolute errors and standard deviations are reported. In the case of subband SNR estimation, only the mean absolute errors are reported.

Table 1: Mean absolute error and standard deviation of the error (in parenthesis) in estimating the filtered SNR using the IBM.

Noise	**SNR**						**Mean**
type	**-10 dB**	**-5 dB**	**0 dB**	**5 dB**	**10 dB**	**15 dB**	
White	0.16(±0.37)	0.00(±0.00)	0.00(±0.00)	0.09(±0.28)	0.55(±0.50)	0.99(±0.19)	0.30(±0.22)
Car	0.00(±0.00)	0.00(±0.00)	0.00(±0.00)	0.00(±0.00)	0.00(±0.00)	0.03(±0.17)	0.01(±0.03)
Babble	0.89(±0.56)	0.05(±0.21)	0.09(±0.29)	0.79(±0.41)	1.10(±0.31)	1.72(±0.53)	0.77(±0.39)
Factory	0.77(±0.47)	0.01(±0.07)	0.07(±0.26)	0.71(±0.45)	1.02(±0.20)	1.53(±0.51)	0.68(±0.33)

Table 2: Mean absolute error and standard deviation of the error (in parenthesis) in estimating the broadband SNR using the IBM.

Noise	**SNR**						**Mean**
type	**-10 dB**	**-5 dB**	**0 dB**	**5 dB**	**10 dB**	**15 dB**	
White	0.16(±0.37)	0.00(±0.00)	0.00(±0.00)	0.07(±0.26)	0.51(±0.50)	0.98(±0.25)	0.29(±0.23)
Car	0.00(±0.00)	0.00(±0.00)	0.00(±0.00)	0.00(±0.00)	0.02(±0.12)	0.09(±0.42)	0.02(±0.09)
Babble	0.89(±0.55)	0.03(±0.16)	0.12(±0.33)	0.74(±0.44)	1.09(±0.34)	1.79(±0.51)	0.78(±0.39)
Factory	0.55(±0.50)	0.01(±0.07)	0.10(±0.31)	0.71(±0.45)	1.01(±0.22)	1.47(±0.54)	0.64(±0.35)

Table 3: Mean absolute error in estimating broadband SNR, with and without the SNR transformation step. Car noise is used for this experiment.

Noise	**Method**	**SNR**						**Mean**
type		**-10 dB**	**-5 dB**	**0 dB**	**5 dB**	**10 dB**	**15 dB**	
Car	IBM w/ SNR transformation	0.00	0.00	0.00	0.00	0.02	0.09	0.02
	IBM w/o SNR transformation	7.68	7.84	7.72	7.75	8.00	8.22	7.87

4.2 IBM Results

The mean absolute errors and the standard deviations of the errors in estimating the filtered SNR of the signal using the IBM are shown in Table 1. It can be clearly seen from the results that excellent performance is obtained using the IBM. When the noise is relatively stationary, the IBM based system is even able to perfectly estimate the SNR in a few test conditions. It is interesting to note that the errors are slightly larger in high SNR conditions. This is because at high SNRs masked T-F units are fewer, leading to an underestimation of the total noise energy. This bias is noise dependent, and the noise dependence makes it difficult to compensate for without prior knowledge about the noise type. Error statistics in estimating the broadband SNR are shown in Table 2. The error trends are quite similar to those in Table 1. These results point to the fact that the IBM, despite being binary, can indeed be used for SNR estimation.

To illustrate the role of the SNR transformation step we present the results obtained with and without it. Only the results using the car noise are shown since it has a significant amount of low-frequency energy. Note that turning off the SNR transform implies that the broadband SNR is approximately equal to the filtered SNR. The results are shown in Table 3

for 6 SNR conditions. As can be seen, without the transformation, the errors are much larger. On average, the SNR transformation step improves the performance by around 7.9 dB. The difference is less dramatic for other noise types but still significant, especially at low SNR conditions. The results corroborate our claim that the broadband and the filtered SNR can be different; the proposed SNR transform compensates for this difference for broadband SNR estimation.

4.3 Estimated IBM Results

4.3.1 Global SNR Estimation

Global SNR estimation results are tabulated in Tables 4 and 5. Each table consists of 5 sets of results – one for each noise and one for the average across the 4 noises.

The mean absolute errors in estimating the filtered SNR are shown in Table 4. The proposed algorithm obtains the best average results across all noise types. It also obtains the best results in most of the individual test conditions. Similar to the IBM results, errors gradually increase at positive SNR levels but are still reasonably small. The second best performance is obtained using another binary masking method – HND_MOD. On average, it is around 0.4 dB worse than the proposed method. The proposed algorithm outperforms WADA and HND by about 1.5 dB and 0.4 dB, respectively. WADA performs reasonably when the SNR $>$ 0 dB. But at lower SNRs, the noisy speech does not follow the Gamma distribution leading to poor estimation results. Not surprisingly, it performs the best in white noise conditions. Hu-Wang'10 outperforms the proposed system when the background noise is white and the SNR $\leq$ 0 dB. Hu-Wang'11 performs well when the SNR is low, but produces poor results at high SNRs, mainly because the algorithm is tuned to work at low SNR conditions as noted earlier. Its average performance is worse than the proposed algorithm by 1.6 dB. It outperforms the proposed method when the background noise is babble and the SNR -10 dB. It is interesting to note that the proposed algorithm works better than the IBM in a few conditions. This is possible because the IBM does make errors in SNR estimation, as can be seen from Table 1, and some errors in estimating the IBM are favorable for SNR estimation. However, on average the IBM obtains better results than the proposed algorithm in every noise condition. The standard deviations of the errors are also shown in Table 4. In terms of this error metric, the proposed algorithm also works the best in most test conditions.

The errors in estimating the broadband SNR are shown in Table 5. Again, the trends are very similar to Table 4. Compared to HND_MOD, the average mean absolute error of the proposed algorithm is better by about 0.3 dB. Compared to WADA and HND, it is better by about 1.7 dB and 1 dB, respectively. The standard deviation profiles are similar to those for filtered SNR estimation.

Table 6 shows the mean absolute errors of the different algorithms according to the utterance length. The shortest and the longest utterance in the TIMIT core test set are 1.3 sec

Table 4: The mean absolute error in estimating the filtered SNR using WADA, HND, Hu-Wang'10, HND_MOD, Hu-Wang'11 and the proposed algorithm. The standard deviation of the error is shown within parenthesis. The best result in each condition is marked in **bold**. Also shown are the average results, across SNRs and across different noise types.

Noise type	**Method**	**SNR**						**Mean**
		-10 dB	**-5 dB**	**0 dB**	**5 dB**	**10 dB**	**15 dB**	
White	WADA	3.32(±4.51)	1.17(±1.31)	0.90(±0.82)	0.88(±0.91)	1.02(±1.22)	1.55(±2.01)	1.47(±1.80)
	HND	1.66(±**0.78**)	1.27(±0.58)	0.92(±0.49)	0.76(±0.50)	0.90(±0.42)	1.08(±0.39)	1.10(±0.53)
	Hu-Wang'10	**1.15**(±0.94)	**0.45**(±0.78)	**0.39**(±0.71)	0.76(±1.81)	1.44(±1.86)	2.89(±3.20)	1.18(±1.55)
	HND_MOD	2.72(±1.08)	1.42(±**0.56**)	1.06(±**0.37**)	0.95(±**0.30**)	0.98(±**0.23**)	**1.04**(±**0.30**)	1.36(±**0.47**)
	Hu-Wang'11	1.84(±0.94)	1.06(±0.81)	0.91(±0.76)	1.45(±1.86)	2.69(±2.29)	5.03(±3.38)	2.16(±1.68)
	Proposed	1.66(±0.79)	0.78(±0.57)	0.47(±0.51)	**0.55**(±0.54)	**0.74**(±0.64)	1.12(±0.78)	**0.89**(±0.64)
Car	WADA	6.29(±7.05)	3.11(±4.88)	1.09(±1.30)	0.86(±1.03)	0.93(±1.24)	1.56(±2.01)	2.31(±2.92)
	HND	1.68(±0.94)	0.51(±0.58)	0.14(±0.36)	0.15(±0.38)	0.29(±0.48)	0.71(±0.64)	0.58(±0.56)
	Hu-Wang'10	0.77(±1.00)	0.55(±0.88)	0.60(±1.54)	0.92(±2.08)	1.56(±2.15)	3.44(±3.77)	1.31(±1.90)
	HND_MOD	0.48(±0.74)	0.16(±0.40)	0.08(±0.28)	0.17(±0.39)	0.52(±0.53)	0.82(±0.57)	0.37(±0.49)
	Hu-Wang'11	0.48(±0.88)	0.43(±0.84)	0.78(±1.51)	1.67(±2.17)	3.03(±2.41)	5.57(±3.39)	1.99(±1.87)
	Proposed	**0.35**(±**0.67**)	**0.07**(±**0.25**)	**0.01**(±**0.10**)	**0.01**(±**0.07**)	**0.06**(±**0.24**)	**0.28**(±**0.45**)	**0.13**(±**0.30**)
Babble	WADA	5.88(±2.59)	2.94(±1.86)	1.75(±1.09)	1.28(±1.06)	1.32(±1.39)	1.65(±2.09)	2.47(±1.68)
	HND	2.77(±**1.35**)	0.96(±**1.15**)	0.80(±0.91)	0.84(±0.86)	1.22(±0.99)	1.59(±1.07)	1.36(±1.05)
	Hu-Wang'10	2.32(±1.49)	1.06(±1.42)	1.12(±1.64)	1.61(±2.19)	2.29(±2.91)	3.65(±4.20)	2.01(±2.31)
	HND_MOD	2.19(±1.66)	0.96(±1.26)	0.75(±0.90)	0.73(±0.78)	0.90(±0.84)	1.04(±0.93)	1.09(±1.06)
	Hu-Wang'11	**1.45**(±1.47)	0.97(±1.20)	1.42(±1.51)	2.14(±2.01)	3.50(±2.69)	5.77(±3.71)	2.54(±2.10)
	Proposed	1.93(±1.45)	**0.84**(±1.16)	**0.43**(±**0.67**)	**0.51**(±**0.69**)	**0.64**(±**0.79**)	**0.99**(±**0.91**)	**0.89**(±**0.95**)
Factory	WADA	6.39(±3.81)	2.86(±1.71)	1.84(±1.14)	1.43(±1.18)	1.30(±1.42)	1.74(±2.17)	2.60(±1.91)
	HND	1.88(±2.31)	1.28(±1.54)	1.14(±0.97)	1.18(±0.86)	1.49(±0.87)	1.95(±1.07)	1.49(±1.27)
	Hu-Wang'10	1.75(±2.09)	1.14(±1.72)	1.03(±1.53)	1.05(±1.45)	1.65(±2.57)	3.02(±3.40)	1.61(±2.13)
	HND_MOD	2.36(±3.23)	1.43(±1.64)	1.05(±0.94)	1.04(±0.84)	1.27(±**0.78**)	1.40(±**0.99**)	1.43(±1.40)
	Hu-Wang'11	1.62(±**1.79**)	1.46(±1.53)	1.23(±1.20)	1.56(±1.36)	3.02(±2.63)	5.58(±3.42)	2.41(±1.99)
	Proposed	**1.29**(±1.80)	**0.99**(±**1.23**)	**0.56**(±**0.71**)	**0.52**(±**0.67**)	**0.73**(±**0.78**)	**1.17**(±1.02)	**0.88**(±**1.03**)
All	WADA	5.47(±5.73)	2.52(±3.17)	1.39(±1.23)	1.11(±1.10)	1.14(±1.35)	1.63(±2.08)	2.21(±2.45)
	HND	1.99(±2.20)	1.00(±1.28)	0.75(±0.85)	0.73(±0.80)	0.98(±0.85)	1.33(±0.97)	1.13(±1.16)
	Hu-Wang'10	1.50(±1.99)	0.80(±1.30)	0.79(±1.41)	1.08(±1.94)	1.74(±2.43)	3.25(±3.68)	1.53(±2.12)
	HND_MOD	1.94(±2.57)	0.99(±1.25)	0.74(±0.79)	0.72(±0.71)	0.92(±**0.69**)	1.07(±**0.78**)	1.06(±1.13)
	Hu-Wang'11	1.35(±**1.73**)	0.98(±1.20)	1.08(±1.30)	1.70(±1.89)	3.06(±2.53)	5.49(±3.49)	2.28(±2.02)
	Proposed	**1.31**(±1.77)	**0.67**(±**0.99**)	**0.37**(±**0.58**)	**0.39**(±**0.59**)	**0.54**(±0.70)	**0.89**(±0.89)	**0.70**(±**0.92**)

Table 5: The mean absolute error in estimating the broadband SNR using WADA, HND, Hu-Wang'10, HND_MOD, Hu-Wang'11 and the proposed algorithm. The standard deviation of the error is shown within parenthesis. The best result in each condition is marked in **bold**. Also shown are the average results, across SNRs and across different noise types.

Noise type	**Method**	**SNR**						**Mean**
		-10 dB	**-5 dB**	**0 dB**	**5 dB**	**10 dB**	**15 dB**	
White	WADA	3.52(±4.69)	1.10(±1.25)	0.87(±0.77)	0.86(±0.84)	1.02(±1.13)	1.57(±1.88)	1.49(±1.76)
	HND	1.85(±**0.75**)	1.40(±0.60)	1.08(±0.41)	0.95(±0.40)	0.99(±0.34)	1.19(±0.41)	1.25(±0.49)
	Hu-Wang'10	**1.21**(±0.92)	**0.44**(±0.69)	**0.39**(±0.70)	0.67(±1.08)	1.51(±2.89)	3.03(±3.77)	1.21(±1.68)
	HND_MOD	2.73(±0.99)	1.42(±0.58)	1.05(±**0.32**)	0.97(±**0.25**)	0.97(±**0.24**)	1.06(±**0.30**)	1.37(±**0.45**)
	Hu-Wang'11	1.90(±0.95)	1.01(±0.73)	0.90(±0.76)	1.41(±1.22)	2.67(±2.92)	5.04(±3.33)	2.15(±1.65)
	Proposed	1.68(±0.78)	0.80(±**0.54**)	0.53(±0.55)	**0.55**(±0.52)	**0.74**(±0.62)	**1.12**(±0.75)	**0.90**(±0.63)
Car	WADA	6.93(±7.61)	4.48(±6.23)	1.53(±1.78)	1.21(±1.29)	1.22(±1.34)	1.64(±1.93)	2.84(±3.36)
	HND	5.71(±1.81)	3.41(±1.29)	2.10(±0.88)	1.74(±0.92)	1.40(±0.69)	1.21(±0.78)	2.60(±1.06)
	Hu-Wang'10	0.35(±1.52)	0.31(±1.51)	0.46(±1.71)	0.93(±2.58)	1.62(±2.30)	3.57(±4.05)	1.21(±2.28)
	HND_MOD	0.04(±0.20)	0.01(±0.07)	**0.00**(±**0.00**)	0.01(±0.10)	0.03(±0.18)	0.23(±0.52)	0.05(±0.18)
	Hu-Wang'11	0.48(±1.68)	0.56(±1.21)	0.85(±1.51)	1.52(±2.08)	2.93(±3.14)	5.41(±3.43)	1.96(±2.18)
	Proposed	**0.01**(±**0.07**)	**0.00**(±**0.00**)	**0.00**(±**0.00**)	**0.00**(±**0.00**)	**0.01**(±**0.07**)	**0.07**(±**0.33**)	**0.01**(±**0.08**)
Babble	WADA	5.72(±2.58)	2.94(±1.66)	1.70(±1.08)	1.30(±1.06)	1.28(±1.23)	1.70(±2.05)	2.44(±1.61)
	HND	3.22(±1.93)	0.94(±1.22)	0.69(±0.93)	0.90(±1.04)	1.21(±1.00)	1.57(±1.11)	1.42(±1.20)
	Hu-Wang'10	2.18(±1.66)	0.98(±1.39)	1.06(±1.44)	1.62(±2.49)	2.22(±2.80)	4.08(±4.67)	2.02(±2.41)
	HND_MOD	1.90(±1.69)	0.91(±1.25)	0.57(±0.74)	0.67(±0.70)	0.90(±0.82)	1.01(±**0.89**)	0.99(±1.01)
	Hu-Wang'11	**1.48**(±1.65)	1.06(±1.29)	1.32(±1.30)	2.11(±2.12)	3.43(±2.51)	6.03(±3.66)	2.57(±2.09)
	Proposed	1.80(±**1.60**)	**0.74**(±**1.05**)	**0.41**(±**0.66**)	**0.44**(±**0.65**)	**0.61**(±**0.81**)	**0.96**(±0.93)	**0.83**(±**0.95**)
Factory	WADA	5.70(±3.92)	2.82(±1.66)	1.72(±1.14)	1.24(±1.03)	1.30(±1.22)	1.81(±2.02)	2.43(±1.83)
	HND	1.62(±**1.56**)	0.94(±1.30)	0.94(±0.90)	1.03(±0.77)	1.30(±0.82)	1.62(±**0.91**)	1.24(±1.04)
	Hu-Wang'10	1.53(±1.96)	0.89(±1.23)	0.82(±1.21)	0.93(±1.39)	1.40(±1.89)	2.95(±3.66)	1.42(±1.89)
	HND_MOD	1.73(±2.47)	1.19(±1.37)	0.89(±0.81)	0.90(±0.66)	1.09(±0.76)	1.10(±0.95)	1.15(±1.17)
	Hu-Wang'11	1.40(±1.70)	1.21(±1.05)	1.12(±1.12)	1.55(±1.51)	2.92(±2.35)	5.47(±3.63)	2.28(±1.89)
	Proposed	**1.17**(±1.64)	**0.78**(±**0.94**)	**0.47**(±**0.68**)	**0.38**(±**0.55**)	**0.63**(±**0.75**)	**1.05**(±**0.91**)	**0.74**(±**0.91**)
All	WADA	5.47(±5.81)	2.83(±3.63)	1.46(±1.32)	1.15(±1.09)	1.21(±1.24)	1.68(±1.97)	2.30(±2.51)
	HND	3.10(±3.18)	1.67(±2.13)	1.20(±1.50)	1.15(±1.39)	1.23(±1.33)	1.40(±1.43)	1.63(±1.83)
	Hu-Wang'10	1.32(±2.00)	0.65(±1.27)	0.68(±1.33)	1.04(±2.03)	1.69(±2.53)	3.41(±4.08)	1.46(±2.21)
	HND_MOD	1.60(±2.22)	0.88(±1.14)	0.63(±0.69)	0.64(±0.62)	0.75(±0.71)	0.85(±**0.81**)	0.89(±1.03)
	Hu-Wang'11	1.32(±1.84)	0.96(±1.12)	1.05(±1.22)	1.65(±1.80)	2.98(±2.76)	5.49(±3.53)	2.24(±1.03)
	Proposed	**1.16**(±**1.67**)	**0.58**(±**0.84**)	**0.35**(±**0.58**)	**0.34**(±**0.54**)	**0.50**(±**0.70**)	**0.80**(±0.89)	**0.62**(±**0.87**)

Table 6: The mean absolute error in estimating the broadband SNR using using WADA, HND, Hu-Wang'10, HND_MOD, Hu-Wang'11 and the proposed algorithm, according to the utterance length. Errors are averaged across all 4 noises. The best result in each condition is marked in **bold**.

Utterance length (sec) (# of utterances)	**Method**	**SNR**						**Mean**
		-10 dB	**-5 dB**	**0 dB**	**5 dB**	**10 dB**	**15 dB**	
$0-2$ (19)	WADA	5.50	2.92	1.37	1.20	1.20	1.50	2.28
	HND	2.68	1.68	1.46	1.30	1.21	1.39	1.62
	Hu-Wang'10	1.42	0.86	0.76	1.18	1.70	3.51	1.57
	HND_MOD	1.67	1.03	0.79	0.74	0.83	0.82	0.98
	Hu-Wang'11	1.55	1.22	1.08	1.61	2.70	5.51	2.28
	IBM	0.42	0.03	0.04	0.42	0.71	1.13	0.46
	Proposed	**1.29**	**0.78**	**0.51**	**0.39**	**0.57**	**0.80**	**0.72**
$2-3$ (83)	WADA	5.76	2.79	1.50	1.16	1.21	1.72	2.36
	HND	3.11	1.61	1.17	1.10	1.20	1.34	1.59
	Hu-Wang'10	1.40	0.70	0.70	1.05	1.67	3.45	1.49
	HND_MOD	1.62	0.89	0.59	0.60	0.71	0.83	0.87
	Hu-Wang'11	1.34	0.94	1.00	1.52	2.86	5.16	2.14
	IBM	0.38	0.01	0.07	0.34	0.61	1.04	0.41
	Proposed	**1.19**	**0.60**	**0.33**	**0.32**	**0.48**	**0.76**	**0.61**
$3-4$ (66)	WADA	5.23	3.04	1.45	1.17	1.27	1.87	2.34
	HND	3.16	1.73	1.16	1.14	1.22	1.44	1.64
	Hu-Wang'10	1.23	0.57	0.66	0.97	1.67	3.30	1.40
	HND_MOD	1.53	0.87	0.62	0.63	0.75	0.89	0.88
	Hu-Wang'11	1.25	0.91	1.12	1.83	3.23	5.83	2.36
	IBM	0.41	0.01	0.06	0.41	0.67	1.11	0.45
	Proposed	**1.10**	**0.52**	**0.34**	**0.36**	**0.52**	**0.85**	**0.61**
$4-5$ (19)	WADA	5.25	2.29	1.42	1.14	1.07	1.20	2.06
	HND	3.13	1.74	1.18	1.28	1.33	1.53	1.70
	Hu-Wang'10	**1.14**	**0.49**	0.47	0.76	1.38	2.93	1.20
	HND_MOD	1.70	0.78	0.61	0.72	0.84	0.89	0.92
	Hu-Wang'11	1.25	0.84	0.89	1.36	2.63	5.18	2.03
	IBM	0.42	0.00	0.03	0.42	0.74	1.11	0.45
	Proposed	**1.14**	0.50	**0.28**	**0.25**	**0.33**	**0.67**	**0.53**
> 5 (5)	WADA	4.50	2.55	1.25	0.75	0.90	1.15	1.85
	HND	3.65	1.80	1.30	1.15	1.40	1.45	1.79
	Hu-Wang'10	1.35	0.90	1.10	2.15	3.30	5.60	2.40
	HND_MOD	1.60	0.70	0.70	0.70	**0.80**	**0.70**	**0.87**
	Hu-Wang'11	**1.20**	1.35	1.35	2.60	4.35	7.40	3.04
	IBM	0.50	0.00	0.00	0.45	0.60	1.05	0.43
	Proposed	**1.20**	**0.60**	**0.60**	**0.60**	0.95	1.25	**0.87**

and 6.2 sec, respectively. The utterances are grouped into 1-sec bins. The errors across all 4 noise types are aggregated to obtain the average performance in the table. The proposed algorithm performs the best in most conditions. It can be seen that our algorithm works best when the utterance length is between 2 – 5 sec and the SNR between -5 dB and 10 dB. Note that not enough samples are available to draw meaningful conclusions when the utterance

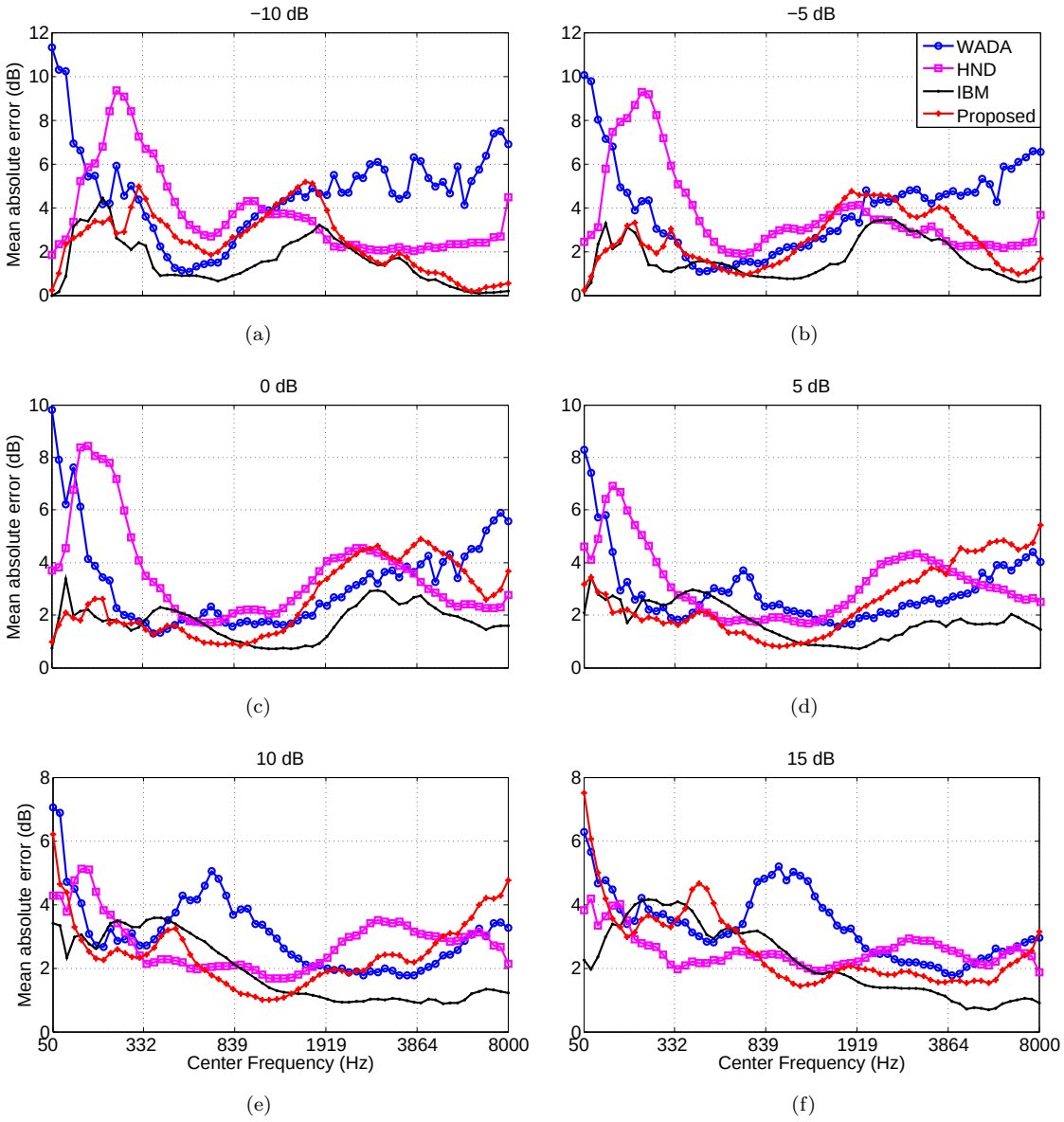

Figure 4: Subband SNR estimation results using WADA, HND, the IBM, and the estimated IBM by the proposed algorithm, for speech mixed with white noise. Mean absolute errors across the 64 sub-bands are shown, for the following filtered SNR conditions: (a) -10 dB. (b) -5 dB. (c) 0 dB. (d) 5 dB. (e) 10 dB. (f) 15 dB.

length $>$ 5 sec.

These results clearly show that the proposed algorithm is able to obtain accurate estimates of global SNR – both broadband and filtered.

4.3.2 Subband SNR Estimation

Subband SNR estimation results are shown in Figs. 4-8. The figures show the performance for the four noises individually, and the average across them. Unlike the global SNR estimation

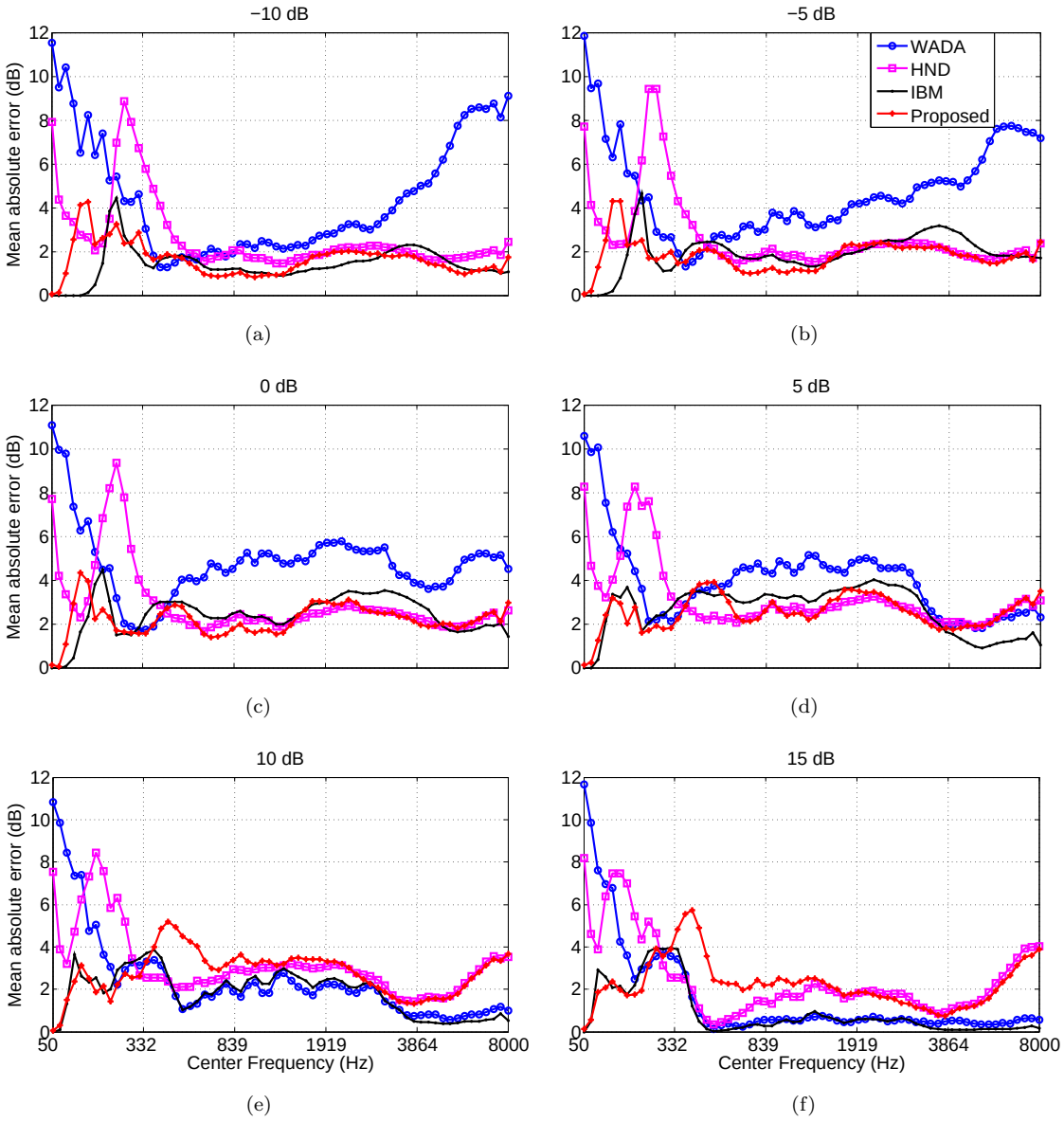

Figure 5: Subband SNR estimation results using WADA, HND, the IBM, and the estimated IBM by the proposed algorithm, for speech mixed with car noise. Mean absolute errors across the 64 sub-bands are shown, for the following filtered SNR conditions: (a) -10 dB. (b) -5 dB. (c) 0 dB. (d) 5 dB. (e) 10 dB. (f) 15 dB.

results, the errors are larger even when the IBM is used, where the best performance is typically obtained. Similar to broadband SNR estimation, HND and the proposed algorithm outperform the IBM based results for a few channels and test conditions. For the proposed algorithm, better performance is usually obtained when the noise type is stationary. Two conditions especially unfavorable to the proposed algorithm are: low frequency channels when the background noise is babble, and high frequency channels when the background noise is factory, both at SNRs $\leq$ 0 dB. Even then, the proposed algorithm is not the worst performing

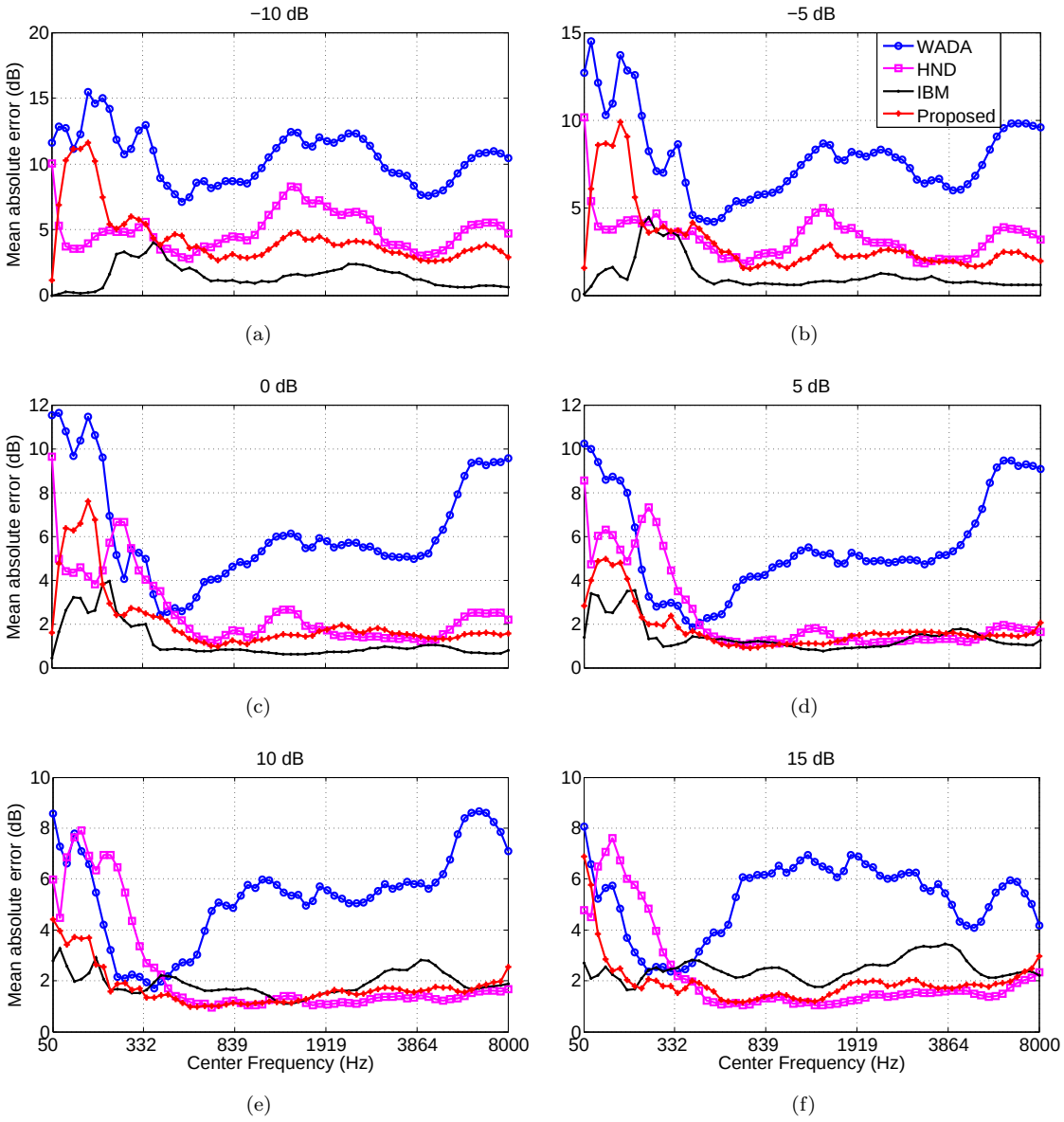

Figure 6: Subband SNR estimation results using WADA, HND, the IBM, and the estimated IBM by the proposed algorithm, for speech mixed with babble noise. Mean absolute errors across the 64 sub-bands are shown, for the following filtered SNR conditions: (a) -10 dB. (b) -5 dB. (c) 0 dB. (d) 5 dB. (e) 10 dB. (f) 15 dB.

method at these conditions. In fact, at the latter condition, it is still better than both HND and WADA. The mean absolute error of the proposed algorithm is almost always $\leq$ 5 dB at the remaining conditions.

Excluding the IBM results, the best performance in the low frequency channels (center frequency $\leq$ 350 Hz or the first 10–15 channels) is typically obtained by the proposed algorithm. The only noted exception is when the noise is babble and the SNR $\leq$ 0 dB. In these conditions, HND works better. If we consider the average performance across all noise con-

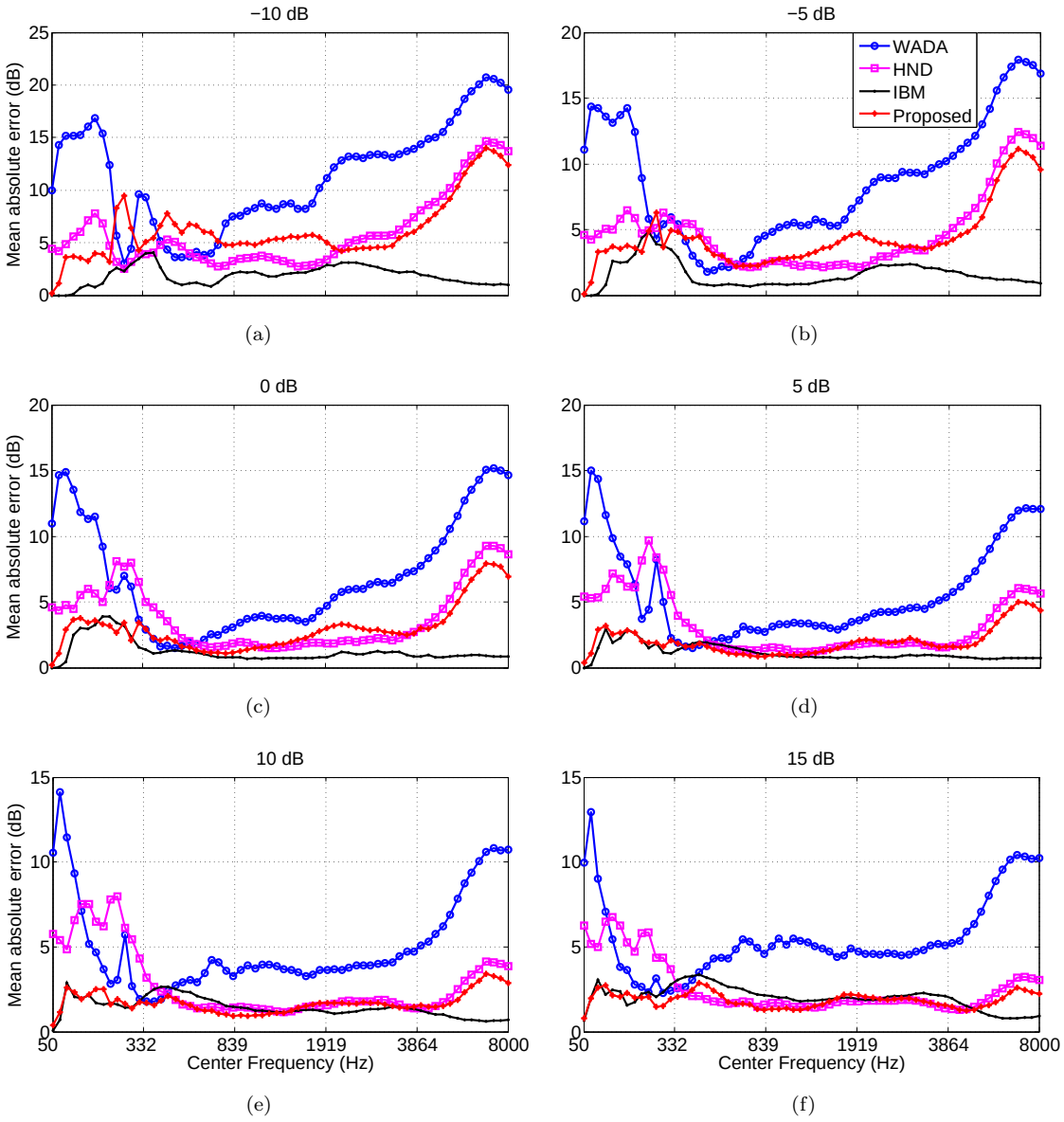

Figure 7: Subband SNR estimation results using WADA, HND, the IBM, and the estimated IBM by the proposed algorithm, for speech mixed with factory noise. Mean absolute errors across the 64 sub-bands are shown, for the following filtered SNR conditions: (a) -10 dB. (b) -5 dB. (c) 0 dB. (d) 5 dB. (e) 10 dB. (f) 15 dB.

ditions (Figure 8), the mean absolute error of the proposed algorithm is well within 5 dB for these frequency channels, significantly better than both HND and WADA. The performance of WADA at low frequency channels is poor because the noisy signals, or even the speech signals for that matter, in these channels are not Gamma distributed anymore.

For the mid-frequency channels (center frequency between 300 Hz and 3800 Hz, or frequency channels 13–51), no one method works uniformly better than the rest. Both HND and the proposed algorithm work well in most conditions. WADA obtains results similar to HND and

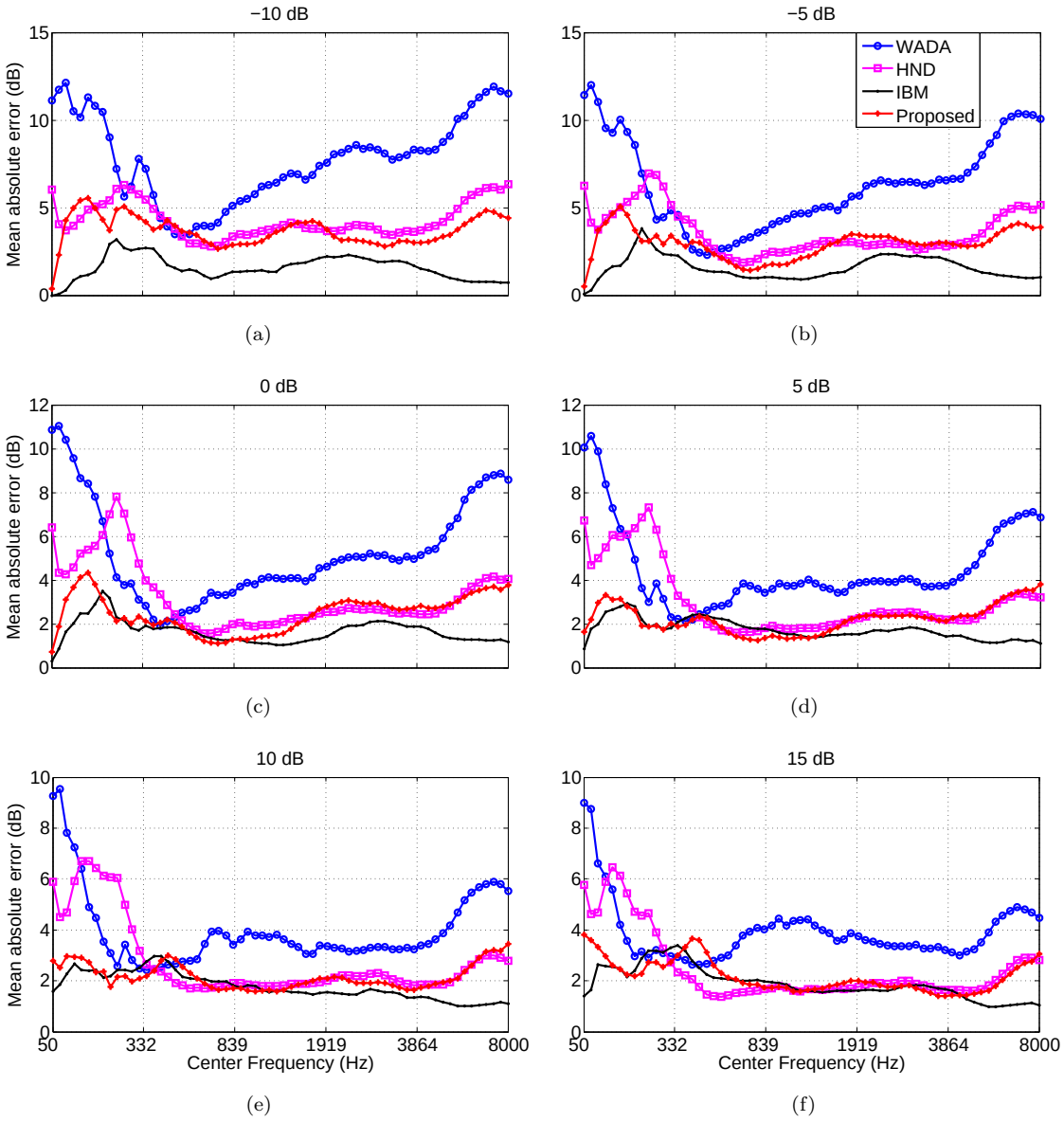

Figure 8: Subband SNR estimation results using WADA, HND, the IBM, and the estimated IBM by the proposed algorithm, averaged across all noise types. Mean absolute errors across the 64 sub-bands are shown, for the following filtered SNR conditions: (a) -10 dB. (b) -5 dB. (c) 0 dB. (d) 5 dB. (e) 10 dB. (f) 15 dB.

the proposed algorithm when the background noise is white. For car noise it works quite well when the SNR $\geq$ 10 dB, almost as well as the IBM and better than both HND and the proposed algorithm. This is largely because the true subband SNRs in these conditions are well above 0 dB. At other conditions, performance of WADA is significantly worse than the other methods, as reflected in the average performance shown in Figure 8. When the background noise is non-stationary, the proposed algorithm is slightly better than HND at most SNRs. Under stationary conditions, the performance of the proposed algorithm is mostly comparable or better than HND. In a few cases, especially when the SNR is high, HND works

slightly better. Similar mixed trends can be observed for the high frequency channels (center frequency $\geq$ 3800 Hz, or the last 10–15 channels), with the proposed algorithm working slightly better than HND especially when the noise type is non-stationary.

We can observe a few overall trends in estimation errors from the figures. For example, from Figure 8 we can see that as the filtered SNR of the signal increases, the performance of the proposed algorithm also improves. When the SNR is -10 dB, the mean absolute errors are about 4 dB. And when the SNR is 10 dB, the errors are about 2 dB. Also note that improvements in mask estimation can clearly improve the average performance of the proposed method, since the IBM results are significantly better especially at low SNR conditions. These results indicate that the proposed algorithm can additionally be used to estimate subband SNRs with considerable accuracy.

5 Discussion

The results presented in this paper show that binary masks can be used for long-term SNR estimation – both at subband and broadband levels. The results further indicate that we only need a reasonable estimate of the IBM to obtain good SNR estimates. If an algorithm is able to correctly label the high energy regions as belonging to the target or the noise, the long-term SNR can be estimated with very good accuracy as the energy in these regions dominates the total energy. In most of the test conditions, the best performance is obtained when the masks estimated by CASA and speech enhancement algorithms are combined.

The proposed algorithm cannot be used to estimate short-time SNR of a signal, which would lead to a chicken-and-egg problem as the short-time SNR can directly be used to estimate the IBM. A disadvantage of the proposed algorithm is its computational complexity. The CASA component involves computation of autocorrelation and envelope extraction at each T-F unit during the feature extraction stage, both of which are computationally expensive. The feature extraction stage dominates the time complexity of the proposed algorithm. Autocorrelations can be efficiently calculated in $O(N \log N)$ time and since frequency channels are independent of each other, computations can be parallelized [11, 13]. Even so, the algorithm takes longer than WADA or HND. Nevertheless, the performance in SNR estimation obtained by the proposed system is significantly better than these approaches.

Binary masking described in this work is quite different from the VAD based algorithms that have been proposed in the literature for SNR estimation [16, 20]. A VAD tries to identify *noise-only* frames to obtain an estimate of the noise energy by assuming stationarity. On the other hand, our approach identifies *noise-dominant* T-F units, which are used to approximate the total noise energy in the algorithm. The algorithm can easily be extended to estimate the SNR in speech-present frames, by simply dropping noise-only frames during estimation.

Note that, the mask estimation and the SNR estimation in the proposed system are two separate modules. The IBM estimation module used by the current system can be replaced

with any other mask estimation algorithm. Therefore, the proposed algorithm can potentially be used in more challenging conditions like reverberant noisy environments and multi-talker conditions by replacing the existing mask estimation algorithm with those that work well in such conditions.

To summarize, we have proposed a novel CASA based SNR estimation algorithm. The algorithm estimates the filtered SNR, the broadband SNR and the subband SNRs with high accuracy. Results show that the performance of the proposed system is better than existing long-term SNR estimation algorithms. The algorithm additionally estimates the IBM, which can be used for speech separation purposes. An insight from our work is that binary masks can be effectively used for SNR estimation.

Acknowledgment

The authors would like to thank Guoning Hu, Ke Hu and Kun Han for helpful discussions and providing software implementations; Richard Hendriks, Richard Heusdens, Jesper Jensen and Jan Erkelens for sharing MATLAB codes for the work described in [5, 8]; and Chanwoo Kim for providing an implementation of WADA.

References

[1] NIST Speech Quality Assuracne (SPQA) Package v2.3. 1994 [Online]. Available: http://www.itl.nist.gov/iad/mig/tools/.

[2] M. Berouti, R. Schwartz, and R. Makhoul, "Enhancement of speech corrupted by acoustic noise," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 1979, pp. 208–211.

[3] H. Bourlard, S. Dupont, H. Hermansky, and N. Morgan, "Towards sub-band-based speech recognition," in *Proceedings of European Signal Processing Conference*, 1996, pp. 1579–1582.

[4] T. H. Dat, K. Takeda, and F. Itakura, "On-line Gaussian mixture modeling in the log-power domain for signal-to-noise ratio estimation and speech enhancement," *Speech Communication*, vol. 48, pp. 1515–1527, 2006.

[5] J. Erkelens, R. Hendriks, R. Heusdens, and J. Jensen, "Minimum mean-square error estimation of discrete fourier coefficients with generalized gamma priors," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, no. 6, pp. 1741–1752, 2007.

[6] S. Furui, *Digital speech processing, synthesis, and recognition*, 2nd ed. New York, NY: Marcel Dekker, 2000.

[7] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren. DARPA TIMIT acoustic phonetic continuous speech corpus. 1993 [Online]. Available: http://www.ldc.upenn.edu/Catalog/LDC93S1.html.

[8] R. Hendriks, R. Heusdens, and J. Jensen, "MMSE based noise PSD tracking with low complexity," in *Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, 2010, pp. 4266–4269.

[9] H. G. Hirsch, "Estimation of noise spectrum and its applications to SNR-estiamtion and speech enhancement," International Computer Science Institute, Berkeley, California, USA, TR-93-012, Tech. Rep., 1993.

[10] H. G. Hirsch and C. Ehrlicher, "Noise estimation techniques for robust speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 1995, pp. 153–156.

[11] G. Hu and D. L. Wang, "A tandem algorithm for pitch estimation and voiced speech segregation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, pp. 2067–2079, 2010.

[12] ——, "Segregation of unvoiced speech from nonspeech interference," *Journal of Acoustical Society of America*, vol. 124, pp. 1306–1319, 2008.

[13] K. Hu and D. L. Wang, "Unvoiced speech segregation from nonspeech interference via CASA and spectral subtraction," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 19, no. 6, pp. 1600–1609, 2011.

[14] C. Kim and R. Stern, "Robust signal-to-noise ratio estimation based on waveform amplitude distribution analysis," in *Proceedings of Interspeech*, 2008, pp. 2598–2601.

[15] M. Kleinschmidt and V. Hohmann, "Sub-band SNR estimation using auditory feature processing," *Speech Communication*, vol. 39, pp. 47–64, 2003.

[16] A. Korthauer, "Robust estimation of the SNR of noisy speech signals for the quality evaluation of speech databases," in *Proceedings of ROBUST'99 Workshop*, 1999, pp. 123–126.

[17] Y. Lu and P. Loizou, "Estimators of the magnitude-squared spectrum and methods for incorporating SNR uncertainty," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 19, no. 5, pp. 1123–1137, 2011.

[18] R. Martin, "An efficient algorithm to estimate the instantaneous SNR of speech signals," in *Proceedings of Eurospeech*, 1993, pp. 1093–1096.

[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. 504–512, 1999.

[20] C. Ris and S. Dupont, "Assessing local noise level estimation methods: Application to noise robust ASR," *Speech Communication*, vol. 34, pp. 141–158, 2001.

[21] J. Tchorz and B. Kollmeier, "SNR estimation based on amplitude modulation analysis with applications to noise suppression," *IEEE Transactions on Audio, Speech, and Signal Processing*, vol. 11, pp. 184–192, 2003.

[22] D. van Compernolle, "Noise adaptation in a hidden Markov model speech recognition system," *Computer Speech and Language*, vol. 3, pp. 151–168, 1989.

[23] A. Varga and H. J. M. 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 Communication*, vol. 12, pp. 247–251, 1993.

[24] D. L. Wang, "On ideal binary masks as the computational goal of auditory scene analysis." in *Speech Separation by Humans and Machines*, P. Divenyi, Ed. Boston, MA: Kluwer Academic, 2005, pp. 181–197.

[25] D. L. Wang and G. J. Brown, Eds., *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*. Hoboken, NJ: Wiley/IEEE Press, 2006.

[26] X. Zhao, Y. Shao, and D. L. Wang, "Robust speaker identification using a CASA front-end," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2011, pp. 5468–5471.