Speech Enhancement and Multi-band Frequency Compression for Suppression of Noise and Intraspeech Spectral Masking in Hearing Aids

Nitya Tiwari
Dept. of Electrical Engineering
IIT Bombay, Mumbai, India
<nitya@ee.iitb.ac.in>

Santosh K. Waddi
Samsung Research India
Bangalore, India
<santosh4b6@gmail.com >

Prem C. Pandey
Dept. of Electrical Engineering
IIT Bombay, Mumbai, India
<pcpandey@ee.iitb.ac.in>

Abstract— **Sensorineural hearing impairment is associated with increased intraspeech spectral masking and results in degraded speech perception in noisy environment due to increased masking. Speech enhancement using spectral subtraction can be used for suppressing the external noise. Multi-band frequency compression of the complex spectral samples has been reported to reduce the effects of increased intraspeech masking. A combination of these techniques is implemented for real-time processing for improving speech perception by persons with moderate sensorineural loss. For reducing computational complexity and memory requirement, spectral subtraction is carried out using a cascaded-median based estimation of the noise spectrum without voice activity detection. Multi-band frequency compression, based on auditory critical bandwidths, is carried out using fixed-frame processing along with least-squares error based signal estimation to reduce the processing delay. To reduce computational complexity, the two processing stages share the FFT based analysis-synthesis. The processing is implemented and tested for satisfactory operation, with sampling frequency of 10 kHz, 25.6 ms window with 75% overlap, using a 16-bit fixed-point DSP processor. The real-time operation is achieved with signal delay of approximately 36 ms and using about one-third of the computing capacity of the processor.**

Keywords— ***hearing aid; multi-band frequency compression; speech enhancement; sensorineural hearing loss***

I. Introduction

Sensorineural hearing impairment is characterized by frequency-dependent elevation of hearing thresholds, reduced dynamic range of hearing accompanied by an abnormal loudness growth, and increased temporal and spectral masking. It is generally associated with degraded speech perception [1] – [3]. In addition to providing frequency-selective gain and automatic gain control, many hearing aids have multi-channel dynamic range compression with settable attack time, release time, number of channels, and compression ratios [3], [4]. Spectral contrast enhancement [5], [6] and multi-band frequency compression [7] – [9] have been used for reducing the effect of increased intraspeech spectral masking caused by widening of auditory filters. Listeners with sensorineural loss experience severe difficulty in noisy environments. Spectral subtraction, a single-input speech enhancement technique [10], may be used for suppressing noise to improve speech perception.

For use in a hearing aid, the processing techniques should have low algorithmic delay and computational complexity for implementation on a low-power processor. A combination of noise suppression by spectral subtraction and multi-band frequency compression for reducing the adverse effects of intraspeech spectral masking is implemented for real-time processing using a 16-bit fixed-point DSP chip. For computational efficiency, multi-band frequency compression and spectral subtraction share input-output and FFT based analysis-synthesis stages. The following sections describe the signal processing techniques with emphasis on adaptations made for reducing the storage and computation requirements, real-time implementation, test results, and conclusions.

II. Noise Suppression

Several variations of speech enhancement using spectral subtraction have been reported for use in audio codecs and speech recognition [10] – [18]. It involves estimating the noise spectrum, subtracting it from the noisy speech spectrum, and re-synthesizing the speech signal. As the interfering noise is non-stationary, its spectrum needs to be dynamically estimated. An under-estimation results in residual noise and over-estimation leads to distortion. Noise can be estimated during the silence intervals as identified by a voice activity detector [10], but the detection may not be satisfactory under low-SNR conditions. Further, the method does not track the noise spectrum during long speech segments. Several statistical techniques [10] – [14] have also been reported for estimating the noise spectrum. They do not involve voice activity detection, but their memory requirement and computational complexity make it difficult to implement them for real-time processing using a low-power processor. A cascaded-median based estimation of the noise spectrum has been reported earlier for real-time implementation on a fixed-point DSP chip [19]. This technique is further modified for improving the dynamic response of noise estimation.

Speech enhancement using spectral subtraction involves windowing, FFT calculation, noise spectrum estimation, spectral subtraction, complex spectrum calculation, and re-synthesis using IFFT with overlap-add [10]. The magnitude spectra calculated from FFT of the windowed frames of the input speech signal $x(n)$ are used to estimate the noise magnitude spectrum $D_n(k)$. The enhanced magnitude spectrum $|Y_n(k)|$ is computed, using generalized spectral subtraction, as

$$|Y_n(k)| = \begin{cases} \beta^{1/\gamma} D_n(k), & \text{if } |X_n(k)| < (\alpha+\beta)^{1/\gamma} D_n(k) \\ [|X_n(k)|^{\gamma} - \alpha(D_n(k))^{\gamma}]^{1/\gamma} & \text{otherwise} \end{cases} \tag{1}$$

The exponent factor γ = 2 results in power subtraction and γ = 1 results in magnitude subtraction. Use of subtraction factor α greater than one helps in reducing the broadband peaks in the residual noise, but it may result in deep valleys causing warbling or musical noise which adversely affects the speech quality. A floor noise controlled by the spectral floor factor β is introduced to mask the musical noise. Several variations of the method using frequency-dependent factors and factors as functions of *a posteriori* estimate of SNR have been used for improving speech enhancement [10]. The enhanced magnitude spectrum is combined with the original noisy phase, to get the complex spectrum. To avoid phase calculation, the complex spectrum is calculated using

$$Y_n(k) = |Y_n(k)|\, X_n(k) / |X_n(k)| \tag{2}$$

The resulting short-time spectrum is used to re-synthesize the speech signal. Discontinuities between segments corresponding to the modified complex spectra of the consecutive frames are masked by overlap-add based re-synthesis.

Minimum statistics based noise estimation [11] needs estimation of an SNR-dependent subtraction factor and it may result in removal of some parts of speech signal during the weaker segments. Based on the observation that the speech signal energy in a particular frequency bin is low in most of the frames and high only in 10 – 20% frames corresponding to voiced speech segments, a quantile-based method [14] has been reported for dynamically estimating the noise spectrum and a median-based estimation has been found to work satisfactorily. This method is not well suited for real-time operation because of the computation and memory requirement of the sorting operations involved in finding the median. In [19], a cascaded-median has been used as an approximation to median, with a significant reduction in computation and memory requirement. A p-frame q-stage cascaded-median has q stages with each stage having a first-in-first-out buffer holding p magnitude spectra. In each stage, an ensemble median is calculated after receiving p inputs and is given as input to the next stage. The first stage receives the frame spectrum as the input and the output of the last stage is taken as an approximation of the ensemble median of the spectra over p^q past frames. With reference to the true median based estimation every M (= p^q) frames, the cascaded-median based estimation gives a storage saving ratio of $2M/(pq)$ with the highest saving for $q \approx \ln(M)$. The saving ratio for sorting operation per frame is $(M\text{-}1)/p(p\text{-}1)$, i.e. a lower p requires lesser computation. Further, use of p = 3 simplifies the programming for sorting operations [19].

Speech enhancement using median based noise estimation for speech signals and different noises (white noise, pink noise, babble, etc.) showed that a duration of about 0.8 – 1.5 s could be used for most of them. The processing was implemented with sampling frequency of 10 kHz and window length of 256 samples with 75 % overlap. Use of a 3-frame 5-stage cascaded-median approach (M = 243, p = 3, q = 5) gives an approximation of median of frames over past 1.55 s. The outputs processed using spectral subtraction with the noise spectrum estimated by the true-median and the cascaded-median approaches were indistinguishable. For each frequency bin, the storage requirement was reduced from 486 samples for true median to 15 samples for cascaded median, and the number of sorting operations per frame was reduced from 121 to 3. A weighted average of the medians from different stages can be used to improve the dynamics of noise estimation without increasing storage and sorting operations. Thus our noise estimation method is based on a cascaded-median weighted average as an approximation of the median.

III. MULTI-BAND FREQUENCY COMPRESSION

Multi-band frequency compression [7] – [9] involves dividing speech spectrum into analysis bands and compressing the spectral samples in each band towards the band center. The objective is to present speech energy in relatively narrow bands in order to avoid masking by adjacent spectral components and to reduce the effects of increased intraspeech spectral masking. The processing involves segmentation and spectral analysis, spectral modification, and resynthesis. Arai et al. [7] applied the technique using auditory critical bandwidths on the magnitude spectrum and the complex spectrum was obtained by associating the compressed magnitude spectrum with the original phase spectrum. For decreasing the computation and reducing the processing related artifact, Kulkarni et al. [9] applied the compression on the complex spectrum without calculating the magnitude and phase spectra. Investigations with different types of bandwidths, frequency mappings for spectral modification, and segmentation for analysis-synthesis showed that maximum improvement in speech perception was obtained for auditory critical bandwidth based compression using spectral segment mapping and pitch-synchronous analysis-synthesis. Use of fixed frame analysis-synthesis resulted in perceptible distortions and adversely affected the advantage of multi-band frequency compression. Evaluation of pitch-synchronous implementation using Modified Rhyme Test on eight subjects with moderate

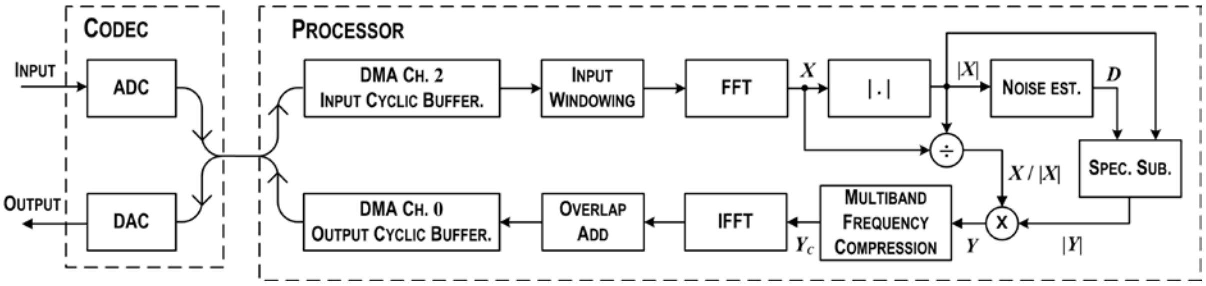


Fig. 1. Implementation of spectral subtraction and multi-band frequency compression on the DSP board.

sensorineural loss showed best performance for the compression factor of 0.6, with a mean increase of 16.5% in recognition scores and a mean decrease of 0.89 s in response time [9]. The results indicated that the processing may improve the usefulness of hearing aids for persons with moderate sensorineural loss.

Implementation of multi-band frequency compression using pitch-synchronous analysis-synthesis was not found feasible on a low-power processor, due to delay and computational complexity associated with pitch estimation. In [20], it has been reported that fixed frame analysis-synthesis with the Griffin-Lim method [21] of signal estimation from modified short-time spectrum can be used for multi-band frequency compression and the resulting output is perceptually indistinguishable from that obtained using pitch-synchronous analysis-synthesis. The method is based on least squared error estimation (LSEE), i.e., minimizing the mean squared error between the modified short-time spectrum and the short-time spectrum of the estimated signal. The input signal is windowed and short-time complex spectrum is computed. After spectral modification, the output signal is re-synthesized by overlap-add after multiplying the output segments with the analysis window. The window should meet the requirement that sum of the squares of all the overlapped window samples is unity. For window length *L* and window shift *S* = *L*/4 corresponding to 75% overlap, this requirement is met by modified Hamming window, given as

$$w(n) = [1/\sqrt{(4d^2 + 2e^2)}][d + e\cos(2\pi(n+0.5)/L)] \qquad (3)$$

with $d = 0.54$ and $e = -0.46$.

For processing, the input signal is segmented using the modified Hamming window of length *L* and 75% overlap. The samples are zero padded, and *N*-point FFT is calculated. Spectral modification is applied on the complex spectral samples using auditory critical bands and spectral segment mapping [9]. For each output frequency sample, a one-sample interval centered on it is mapped to a corresponding segment of the frequency axis of the input spectrum. The edges *a* and *b* of the input frequency segment for the output spectral sample with frequency index *k'* in the *i*th analysis band with center frequency k_{ic} are given as

$$a = k_{ic} - [(k_{ic} - (k' - 0.5))/c] \qquad (4)$$

$$b = a + 1/c \qquad (5)$$

where *c* is the compression factor. With *m* and *n* as the indices of the first and the last spectral samples, respectively, in [*a*, *b*], the frequency sample in the output spectrum is calculated from the samples of the input spectrum as

$$Y_C(k') = (m\text{-}a)\,Y(m) + \sum\nolimits_{j=m+1}^{n-1} Y(j) + (b-n)\,Y(n) \qquad (6)$$

After calculating IFFT and *L*-point windowing, overlap-add is used to re-synthesize the output.

IV. Implementation for Real-time Processing

In order to use noise suppression and multi-band frequency compression for improving speech perception by persons with sensorineural impairment, the techniques need to be implemented for real-time operation on a low-power DSP chip. Both techniques have been earlier separately implemented on a fixed-point processor [19], [20]. For a combined implementation, spectral modification for multi-band frequency compression is applied on the short-time complex spectrum obtained after noise suppression. The input-output and FFT based analysis-synthesis operations are shared for improving the processing efficiency.

The processing was implemented using the 16-bit fixed-point DSP chip TI/TMS320C5515 [22]. It has 16 MB memory space, 320 KB on-chip RAM including 64 KB dual access RAM, three 32-bit programmable timers, four DMA controllers each with four channels, a hardware accelerator supporting 8 to 1024-point FFT, and maximum clock rate of 120 MHz. The DSP board “eZdsp” [23] used for implementation has 4 MB on-board NOR flash for user program and stereo codec TLV320AIC3204 [24] supporting 16/20/24/32-bit quantization and sampling frequency of 8 – 192 kHz. The program was developed in C using TI's “CCStudio, ver. 4.0”. The implementation uses one channel of the stereo codec, with 16-bit quantization and 10 kHz sampling. For reducing conversion overheads, the input samples, spectral values, and the processed samples are stored as complex numbers. The imaginary part of input samples is set to zero.

Fig.1 shows the block diagram of the implementation. Codec and DMA are used to acquire and output the signal. Cyclic DMA buffers are used for segment-based signal processing. The input signal is windowed and short-time spectrum *X* is calculated using FFT. The complex spectrum *Y* is obtained after spectral subtraction and given as input for multi-band frequency compression. The resulting spectrum is used for IFFT calculation and the signal segments are windowed. The output signal is calculated using overlap-add.

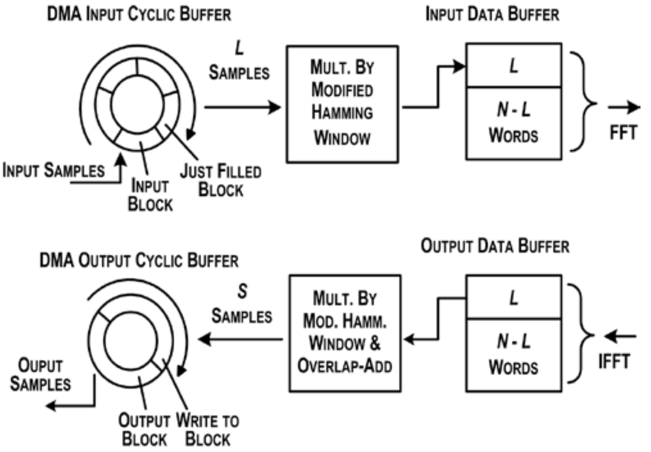
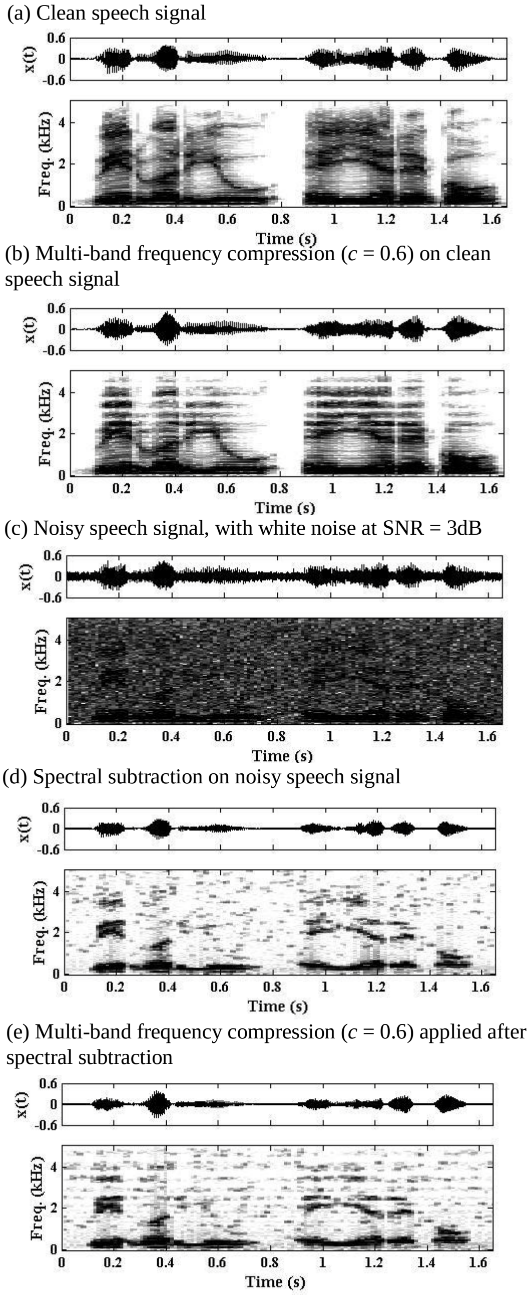


Fig. 2. Data transfer and buffering (S = L/4) [20].

Data transfer and buffering using cyclic DMA buffers for efficient realization of 75% overlap and zero padding, as used in [20], is shown in Fig.2. Signal is acquired using a 5-block DMA input cyclic buffer, with *S*-word blocks. An *N*-word buffer, initialized with zero values, serves as input data buffer. DMA channel-2 reads the input from ADC at regular sampling intervals and writes to the input cyclic buffer. The output is handled using a 2-block DMA output cyclic buffer, with *S*-word blocks. DMA channel-0 is used to cyclically output to DAC. Pointers keep track of the current input, just-filled input, current output, and write-to output blocks, and these are initialized to 0, 4, 0, and 1, respectively. A DMA interrupt is generated when a block gets filled. All pointers are incremented cyclically. The DMA-mediated reading from ADC and writing to DAC are continued. The samples of the just-filled and the previous three blocks are copied to the input data buffer and multiplied by modified Hamming window of length *L* as given in (3). These samples padded with *N*−*L* zero-valued samples serve as input to *N*-point FFT and the result is stored in an *N*-word buffer. The first *N*/2 spectral samples are used for operations related to noise suppression and multi-band frequency compression.

The complex spectrum is used to calculate the magnitude spectrum. The noise spectrum for spectral subtraction is estimated using 3-point 5-stage cascaded-median weighted average. The weights are to be selected emperically. The enhanced magnitude spectrum is obtained by subtracting the estimated noise spectrum from original magnitude spectrum. The corresponding complex spectrum is calculated using (2) and the resulting complex samples are used as input for multi-band frequency compression. The output spectral samples are calculated using spectral segment mapping as given in (6). A look-up table of pre-calculated values of *m*, *n*, *m*–*a*, and *b*–*n* for each output spectral index is used.

The *N*/2 complex samples obtained after multi-band frequency compression along with zero-valued samples are used for calculating *N*-point IFFT. First *L* samples of the real part of resulting sequence are taken as the time domain segment. The segment is multiplied by twice the modified Hamming window. The result is stored in the output data buffer. Overlap-add operation uses a buffer of 3*S* samples. The first *S* samples of the output data buffer are added to the first *S* samples of the overlap buffer containing the partial results from the previous operation. The resulting samples are written as the processed output to the write-to output block. The next 2*S* samples of the output data buffer and the overlap buffer are added together and copied as the first 2*S* samples of the overlap buffer. The last *S* samples of the output data buffer are copied as the last *S* samples of the overlap buffer. As a new input block is received every *S* sampling intervals, all processing operations on the input data buffer should get completed within this duration.



(a) Clean speech signal

(b) Multi-band frequency compression (*c* = 0.6) on clean speech signal

(c) Noisy speech signal, with white noise at SNR = 3dB

(d) Spectral subtraction on noisy speech signal

(e) Multi-band frequency compression (*c* = 0.6) applied after spectral subtraction

Fig. 3. Processing example: “Where were you a year ago”, from a male speaker.

V. Test results

The processing used sampling frequency of 10 kHz, window length L = 256 (i.e. 25.6 ms segments), and FFT size N = 512. The processing methods for noise suppression and multi-band frequency compression as modified for real-time implementation on a DSP chip were first implemented in Matlab using floating-point. The results of noise suppression using noise spectrum estimated using 243-point median and 3-point 5-stage cascaded median for different types of noises and SNRs were almost similar, resulting in about 4 – 13 dB improvement in SNR. For 0 dB input SNR, processing improved PESQ score [25] by 0.37 – 0.86. Use of cascaded-median weighted average was found to be advantageous in cases of non-stationary noise, with the optimal weights varying with the type of noise and SNR. The most acceptable set of values of the relative weights of the 5 median stages for babble were 0, 0, 0.2, 0.6, and 0.2. For noise-free speech input, the results of multi-band frequency compression using fixed-frame processing was perceptually indistinguishable from that of pitch-synchronous processing [9] and the PESQ score for fixed-frame processing with reference to pitch-synchronous processing was 3.7. The fixed-frame processing was found to be useable for noisy speech and music as well, where pitch-synchronous processing could not be applied. An example of processing is shown in Fig.3.

To test the real-time implementation on the DSP board, the speech signal with added noise was output from the PC sound card and applied as input to the codec of the DSP board. The processed output was acquired through the PC sound card. Informal listening showed that the processed output from the DSP board was perceptually similar to the corresponding output from the offline implementation for speech as well as other audio signals. For multi-band compression, PESQ score of the real-time implementation with reference to offline implementation was 2.5 – 3.4 for compression factors of 0.6 – 1.0, indicating a certain but acceptable degradation due to fixed-point processing.

The program operation was tested by progressively decreasing the clock frequency from 120 MHz and it worked satisfactorily down to 39 MHz. The processing has an algorithmic delay of L samples and computational delay of less than L/4 samples. The real-time implementation exhibited a signal delay of about 36 ms, which may be considered as acceptable for its use in the hearing aids along with lipreading.

VI. Conclusion

A real-time implementation of signal processing for noise suppression by spectral subtraction and reduction of intra-speech spectral masking by multi-band frequency compression has been presented. Spectral subtraction technique is implemented using a cascaded-median weighted-average approximation of median for reducing computational complexity and memory requirement. Multi-band frequency compression has been implemented using auditory critical bandwidths, spectral segment mapping, and fixed frame analysis-synthesis based on least square error estimation. The two processing stages share the analysis-synthesis stages for reducing the computation. The technique has been implemented with sampling frequency of 10 kHz on the 16-bit fixed-point processor TI/TMS320C5515, using about one-third of its processing capacity, and with a total signal delay of approximately 36 ms. The implementation needs to be combined with automatic gain control and tested for its usefulness in improving speech perception.

Acknowledgment

The research is supported by the project “National Program on Perception Engineering”, sponsored by the Department of Information Technology, MCIT, Government of India.

References

[1] H. Levitt, J. M. Pickett, and R. A. Houde, Eds., *Senosry Aids for the Hearing Impaired.* New York: IEEE Press, 1980.

[2] B. C. J. Moore, *An Introduction to the Psychology of Hearing*, London, UK: Academic, 1997, pp 66–107.

[3] J. M. Pickett, The Acoustics of Speech Communication: Fundamentals, Speech Perception Theory, and Technology. Boston, Mass.: Allyn Bacon, 1999, pp. 289–323.

[4] H. Dillon, *Hearing Aids*. New York: Thieme Medical, 2001.

[5] T. Baer, B. C. J. Moore, and S. Gatehouse, “Spectral contrast enhancement of speech in noise for listeners with sensorineural hearing impairment: effects on intelligibility, quality, and response times”, *Int. J. Rehab. Res.*, vol. 30, no. 1, pp. 49–72, 1993.

[6] J. Yang, F. Luo, and A. Nehorai, “Spectral contrast enhancement: Algorithms and comparisons,” *Speech Commun*., vol. 39, no. 1–2, pp. 33–46, 2003.

[7] T. Arai, K. Yasu, and N. Hodoshima, “Effective speech processing for various impaired listeners,” in *Proc. 18th Int. Cong. Acoust. (ICA 2004),* Kyoto, Japan, 2004, pp. 1389–1392.

[8] K. Yasu, M. Hishitani, T. Arai, and Y. Murahara, “Critical-band based frequency compression for digital hearing aids,” *Acoustical Science and Technology*, vol. 25, no. 1, pp. 61-63, 2004.

[9] P. N. Kulkarni, P. C. Pandey, and D. S. Jangamashetti, “Multi-band frequency compression for improving speech perception by listeners with moderate sensorineural hearing loss,” *Speech Commun.*, vol. 54, no. 3 pp. 341–350, 2012.

[10] P. C. Loizou, *Speech Enhancement: Theory and Practice*. New York: CRC, 2007.

[11] R. Martin, “Spectral subtraction based on minimum statistics,” in *Proc. 7th Eur. Signal Processing Conf. (EUSIPCO'94)***,** Edinburgh, U.K., 1994, pp. 1182-1185.

[12] I. Cohen, “Noise spectrum estimation in adverse environments: improved minima controlled recursive averaging,” *IEEE Trans. Speech Audio Process*., vol. 11, no. 5, pp. 466-475, 2003.

[13] H. Hirsch and C. Ehrlicher, “Noise estimation techniques for robust speech recognition,” in *Proc. IEEE ICASSP* 1995, Detroit, MI, pp. 153-156.

[14] V. Stahl, A. Fisher, and R. Bipus, “Quantile based noise estimation for spectral subtraction and Wiener filtering,” in *Proc. IEEE ICASSP* 2000, Istanbul, Turkey, pp. 1875-1878.

[15] M. Berouti, R. Schwartz, and J. Makhoul, “Enhancement of speech corrupted by acoustic noise,” in *Proc. IEEE ICASSP* 1979, Washington, DC, pp. 208-211.

[16] S. F. Boll, “Suppression of acoustic noise in speech using spectral subtraction,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 27, no. 2, pp. 113-120, 1979.

[17] Y. Lu and P. C. Loizou, “A geometric approach to spectral subtraction,” *Speech Commun.*, vol. 50, no. 6, pp. 453-466, 2008.

[18] K. Paliwal, K. Wójcicki, and B. Schwerin, “Single-channel speech enhancement using spectral subtraction in the short-time modulation domain,” *Speech Commun.*, vol. 52, no. 5, pp. 450–475, 2010.

[19] S. K. Waddi, P. C. Pandey, and N. Tiwari, “Speech enhancement using spectral subtraction and cascaded-median based noise estimation for hearing impaired listeners,” in *Proc. Nat. Conf. Commun. (NCC 2013)*, Delhi, India, 2013, paper no. 1569696063.

[20] N. Tiwari, P. C. Pandey, and P. N. Kulkarni, “Real-time implementation of multi-band frequency compression for listeners with moderate sensorineural impairment,” in *Proc. 13th Annual Conf. of the Int. Speech Commun. Assoc. (Interspeech 2012)*, Portland, Oregon, 2012, paper no. 689.

[21] D. W. Griffin and J. S. Lim, “Signal estimation from modified short-time Fourier transform,” *IEEE Trans. Acoustics, Speech, Signal Proc.*, vol. 32, no. 2, pp. 236-243, 1984.

[22] Texas Instruments, Inc. (2011) TMS320C5515 Fixed-Point Digital Signal Processor. [online]. Available: focus.ti.com/lit/ds/ symlink/ tms320c5515.pdf

[23] Spectrum Digital, Inc. (2010) TMS320C5515 eZdsp USB Stick Technical Reference. [online]. Available: support.spectrum digital.com/boards/usbstk5515/reva/files/usbstk5515_TechRef_RevA.pdf

[24] Texas Instruments, Inc. (2008) TLV320AIC3204 Ultra Low Power Stereo Audio Codec. [online]. Available: focus.ti.com/lit/ds/ symlink/ tlv320aic3204.pdf

[25] 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,” *ITU-T Rec*., P.862, 2001.