Spectral contrast enhancement: Algorithms and comparisons

Jun Yang a, Fa-Long Luo b,*, Arye Nehorai c

a Fortemedia Inc., 20111 Stevens Creek Boulevard, Suite 150, Cupertino, CA 95014, USA
b Quicksilver Technology, 6640 Via Del Oro, San Jose, CA 95119, USA
c ECE Department, University of Illinois at Chicago, 851 S. Morgan Street, 1120 SEO, Chicago, IL 60607, USA

Abstract

This paper investigates spectral contrast enhancement techniques and their implementation complexity. Three algorithms are dealt with in this paper. The first is the method described by Baer, Moore and Gatehouse. Two alternative methods are also proposed and investigated in this paper from a practical application and implementation point of view. Theoretical analyses and results from laboratory, simulation and subject listening show that spectral contrast enhancement and performance improvement can be achieved by use of these three methods with the appropriate selection of their relevant parameters.

© 2002 Elsevier Science B.V. All rights reserved.

Keywords: Signal processing; Noise reduction; Speech enhancement; Human audition; Auditory system; Real-time implementation

1. Introduction

Spectral contrast is defined as the decibel difference between peaks and valleys in the spectrum. There are two general motivations behind spectral contrast enhancement for hearing-impaired (HI) people. First, in a sensorineural-impaired cochlea, auditory filters are generally broader than the normal and are in many cases abnormally asymmetrical. Processing through these abnormal filters may produce a smearing of spectral detail in the internal representation of acoustic stimuli. Differences in amplitudes between peaks and valleys in the input spectrum may be reduced, making it more difficult to locate spectral prominence (i.e., formants) which provide crucial cues to speech intelligibility. To enhance spectral contrast may be of some help in compensating for the effects of this reduced frequency selectivity. Second, spectral analysis of speech in noise typically shows that these formants are well represented only when the input signal-to-noise ratio (SNR) is large enough but the spectral valleys between the formants are filled with noise. HI people have a reduced ability to pick out the spectral prominence, and are more affected by the noise filling in the valleys, partly because of their reduced frequency selectivity. Therefore, spectral contrast enhancement may be beneficial for the noise reduction. As a matter of fact, from a noise reduction point of view, spectral contrast enhancement can result in speech

*Corresponding author.
E-mail address: falongl@yahoo.com (F.-L. Luo).
enhancement in noise which is also useful for normal people.

Spectral contrast enhancement has received intensive attention and a number of techniques have been proposed during the past two decades. The simplest idea, as proposed by Boers (1980), is to square spectrum levels and then to normalize amplitude. High-amplitude regions of the spectrum will grow more in amplitude when squared than will low-amplitude regions. Another method, proposed by Summerfield et al. (1985), is to decrease formant bandwidth for the synthesis. Narrowing these bandwidths leads to both sharper spectral peaks and greater peak-to-valley ratios (and hence, increased contrast). In the method of Bustamante and Braida (1986), contrast enhancement is based on principal components decomposition of short-term spectrum and made by inflating the amplitude of higher-order principal components that are most strongly associated with narrow-band features of spectral shape. Bunnell (1990) modified the spectrum using the following relation:

\[ T_i = C \left( S_i - S \right) + S, \tag{1} \]

where \( T_i \) is the target magnitude at frequency bin \( i \), \( S_i \) is the original magnitude at frequency bin \( i \), \( S \) is the average spectrum level, and \( C \) is a contrast weight. All spectrum levels are in decibels.

When \( C = 1 \), the target envelope is the same as the original envelope. When \( C < 1 \), contrast reduction is produced. \( C > 1 \) produces contrast enhancement. Because the first four formants play the most important role, Finan and Liu (1994) proposed a linear prediction (LP) based formant enhancement technique. In this method, an all-pole digital filter, determined by LP, was used to model the resonances of the vocal tract for each frame. The formants and their bandwidths were evaluated from the poles of the filter. FIR filters centered on the formants were used to enhance the first four formants. The outputs of the filters were summed together to give the enhanced frame of speech. The final enhanced speech signal was generated by rejoining the overlapping frames. Ribic et al. (1996) proposed another formant-extraction based enhancement technique that extracts the first three formants and then modifies the spectrum values in frequency bins around these formants by designing appropriate contrast weights, which can be accomplished in either the frequency domain or the time domain. Simpson et al. (1990) described a method for increasing the difference in level between peaks and valleys in the spectrum which involves convolving the spectrum with a difference-of-Gaussians (DoG) filter. This operation is similar to taking a smoothed second derivative of the spectrum. Moore's group at Cambridge University carried out a comprehensive investigation concerning the performance and implementation of this type of approach (Stone and Moore, 1992; Baer et al., 1993).

However, despite these efforts and available methods, a good compromise involving quality and complexity in spectral contrast enhancement has not yet been reached, and more effort is highly desirable for the real-time implementation and real-world uses of spectral contrast enhancement techniques. For that reason, we further investigated the method described in (Stone and Moore, 1992; Baer et al., 1993) (for short, the method is hereafter referred as Cambridge's method) under various conditions such as different noise environments, different signal-to-noise ratios, changing the equivalent rectangular bandwidth (ERB) of auditory filters and the enhancement degree of DoG filters, with a specific frame configuration. Furthermore, from a practical application and implementation point of view, two alternative methods are also proposed. These two new proposed methods require much less computational complexity in comparison with Cambridge's method and make it possible to achieve real-time implementation.

The rest of this paper is organized as follows. Section 2 presents a brief description of Cambridge's method and our simulation results under various conditions. Section 3 proposes a simple contrast enhancement algorithm and presents its combination with other processing in hearings-aid products along with illustrations of our results. Section 4 presents another proposed spectral contrast enhancement algorithm and the related results. In Section 5, we make further comparisons among three algorithms. Finally, we will offer some conclusions.
2. Cambridge's method on speech contrast enhancement

The schematic of the method proposed in (Baer et al., 1993) is shown in Fig. 1 which mainly consists of four steps:

1. Transform of the input signal to frequency domain by performing FFT.
2. Calculation of excitation pattern. This involves calculating the output of an array of simulated auditory filters in response to the magnitude spectrum. Each side of each auditory filter is modeled as an intensity-weighting function, assumed to have the form

\[
w(f) = \left( 1 + p \frac{|f - f_c|}{f_c} \right) \exp \left( -p \frac{|f - f_c|}{f_c} \right),
\]

where \( f_c \) is the center frequency of the filter and \( p \) is a parameter determining the slope of the filter skirts. The value of \( p \) is assumed to be the same for the two sides of the filter. The ERB of these filters is \( 4f_c/p \). According to the definition of ERB (Moore and Gatehouse, 1983), we have

\[
p \frac{f - f_c}{f_c} = \frac{4(f - f_c)}{f_c(0.00000623f_c + 0.09339) + 28.52}.
\]

The purpose of this step is to remove minor irregularities in the spectrum while preserving peaks corresponding to major spectral prominence in the speech.

3. Calculation of the enhanced magnitude spectrum. First, an enhancement function is derived from the above excitation pattern by a convolution-like process with a DoG function on an ERB scale. This DoG function is the sum of a positive Gaussian and a negative Gaussian that has twice the bandwidth of the positive Gaussian, that is,

\[
\text{DoG}(f) = \frac{1}{\sqrt{2\pi}} \left( \exp \left( -\frac{(f - f_c)^2}{2b^2} \right) - \frac{1}{2} \exp \left( -\frac{(f - f_c)^2}{8b^2} \right) \right),
\]

where \( b \) is a parameter determining the bandwidth of the DoG function and is selected, as Baer et al. (1993) suggested, by using

\[
b = k \frac{0.00000623f_c^2 + 0.09339f_c + 28.52}{\sqrt{8\ln2/3}}
\]

where \( k \) is an adjustable constant and whose selection will be discussed later. The details of this convolution-like process are described as follows: for a given center frequency of the DoG function, the value of the excitation pattern at each frequency (in linear power units) is multiplied by the value of the DoG function at the same frequency, and the products obtained in this way are summed. The magnitude value of the excitation pattern at that center frequency is then replaced by that sum. The enhancement function \( \text{en}(f) \) derived in the above convolution is then used to modify the excitation pattern. At each frequency where the enhancement function is positive, the excitation pattern is increased in magnitude; at frequency where the enhancement function is negative, the excitation pattern is decreased in magnitude. This can be achieved by the following operation:

![Fig. 1. Schematic diagram of Cambridge's method.](image-url)
spen(f) = M \frac{|en(f)|}{|en(f)|} \log(|en(f)|) + \log(ex(f)),

(6)

where spen(f) is the enhanced magnitude spectrum, ex(f) is the input excitation pattern and M is a parameter which determines the degree of the enhancement. The first term on the right side of Eq. (6) is called a gain function.

4. The magnitude value spen(f) is expressed in linear amplitude units and then combined with the original phase values and finally the IFFT is used to obtain the processed speech.

With a specific frame configuration, we investigated the performance of this scheme under various conditions:

(1) Different signal and noise sources: speech in traffic noise, speech with water noise, speech in restaurant and cafeteria, speech in kitchen, speech with music, and so forth.

(2) Different SNRs. SNR levels that we have dealt with are $\frac{C_0}{C_0}$ 15, $\frac{C_0}{C_0}$ 10, $\frac{C_0}{C_0}$ 5, 0, 5 and 10 dB.

(3) Different bandwidth parameter $k$: from 0.1 to 10.

(4) Different enhancement degree parameter $M$: from 0.1 to 0.5.

With these conditions, we made extensive simulation investigations. On the basis of informal subject listening, we then arrived at some conclusions.

(1) This scheme becomes effective only in large SNR conditions (usually, larger than $\frac{C_0}{C_0}$ 10 dB). Because this method does not identify the noise and the desired speech, processing the input with lower SNR in effect enhances noise rather than the desired speech. Figs. 2(a)–(d) illustrate an example with (female) speech in traffic noise. The sampling rate is 16000 Hz and the length of FFT and IFFT is 128. The duration of each input is about 20 s. In this example, we selected $k = 1$ and $M = 0.1$. Figs. 2(a)–(d) correspond to SNRs of 0, $\frac{C_0}{C_0}$, $\frac{C_0}{C_0}$ 10, $\frac{C_0}{C_0}$ 15 dB, respectively. It can be seen from these figures that the second processing unit in Fig. 1 first removes minor irregularities in the spectrum (similar to some kind of smooth processing) and then provides the excitation pattern.

Main peaks in the excitation pattern are located at frequency 800, 1200 and 2050 Hz, respectively. The main valleys of the spectrum after the second step processing mainly are located at 1800, 2800 and 4500 Hz. In comparison with the excitation pattern curve of each figure, the spectrum of the system output is obviously sharpened to some extent, that is, the spectral contrast of the output has been enhanced. For example, the differences between the peak at 2050 Hz and the valley at 2800 Hz in the related spectra of Fig. 2(a) are 17 and 21 dB before the enhancement processing and after the enhancement processing, respectively.

In addition, the degree of the contrast enhancement depends on the SNR of the input signal, because excitation patterns and the enhancement gain functions will be different for different SNRs of input signal. As a matter of fact, the effect of SNR on the enhancement of the spectrum around 800 Hz is from the noise in this example. This finding also means that this scheme does not apply to the speech-like noise environments, especially in the case of low SNRs. It should be noted that the excitation pattern processing unit (Step 2) boosts the high-frequency part of the input signal. As a result, the system in Fig. 1 also could serve as a high-pass filter to some extent.

(2) The width parameter $k$ has a large effect on the enhancement result. Figs. 3(a)–(e) illustrate a set of results for $k$ at 0.1, 0.5, 1.0, 2.0 and 10.0, respectively. In this example, speech with water noise was considered and the SNR of the input is 0 dB. The enhancement factor $M$ is selected to be 0.1. Main peaks in the excitation pattern are located at frequency 1600, 2700 and 4300 Hz, respectively. The main valleys of the spectrum after the second step processing mainly are located at 2020 and 3200 Hz. The conflict between the signal distortion and spectral contrast enhancement exists in selecting the width parameter $k$. It is worth mentioning that the amount of spectral contrast enhancement decreases in the cases of both large and small values of $k$ with the enhancement occurring mainly when $k$ is around 1.0. This is mainly because the DoG function approaches a constant when $k$ takes either large value or small value. To illustrate this effect, let us consider the spectral magnitude difference
between the frequency bins 1600 and 2020 Hz. Their magnitude difference in the excitation pattern is 13 dB. With the enhancement processing, the differences become 13.7, 15.2, 19.2, 19.8 and 13.8 dB for \( k \) at 0.1, 0.5, 1.0, 2.0 and 10.0, respectively. Obviously, there is almost no enhancement for \( k = 0.1 \) or 10.0 and there is a significant enhancement for \( k = 1.0 \) and 2.0. However, it would be difficult to give a relationship in numerical quantities between the enhanced amount and the width parameter \( k \). For simplicity, we generally select \( k = 1.0 \), which means that the width of the positive lobe (between the zero-crossing points) of the DoG function equals the ERB of the auditory filter with the same center frequency.

(3) The enhancement degree parameter \( M \) is another important factor that affects the output signal. It is easy to see from Eq. (6) that the enhancement amount of the spectral contrast is a monotonically increasing function of the parameter \( M \). Although a large value of \( M \) will result in large enhancement in spectral contrast, this would give rise to a distortion of the signal. On the contrary, a small value of \( M \) will not distort the signal but will carry the cost of low enhancement and poor quality improvement. Figs. 4(a)–(c) show a set of simulation results with \( M \) being 0.1, 0.3 and 0.5, respectively. In this simulation, male speech in restaurant noise (a speech-like noise) is considered and the SNR is \(-5\) dB. All our simulations show that the appropriate value of \( M \) is about 0.1, and it would be better to allow this value to be adjustable rather than fixed in hardware implementation.
In summary, with appropriate selection of the related parameters, the effective enhancement of spectral contrast can be achieved by Cambridge's method. The key problem that this method suffers from is the extensive computational complexity, which we will deal with in Sections 3 and 5.

Fig. 3. The results of Cambridge's method with different bandwidth parameter $k$. (a) $k = 0.1$; (b) $k = 0.5$; (c) $k = 1.0$; (d) $k = 2.0$; (e) $k = 10.0$. 
3. A simple spectral contrast enhancement technique

Although Cambridge's method can improve performance, the computational complexity involved in this method is very extensive. Steps 2 and 3 both involve convolution-like computation in the frequency domain, which makes it difficult to implement in real time. Based on this problem, we will propose a simple spectral contrast enhancement (SSCE) technique which includes the following steps:

1. Transform of the input signal to frequency domain by performing FFT.
2. Calculation of the enhancement magnitude spectrum by
   \[
   \text{spout}(f) = M \log(\text{spin}(f)) + \log(\text{spin}(f)),
   \]
   where \( \text{spin}(f) \) is the magnitude spectrum obtained by Step 1 and \( M \) is the enhancement factor with positive value.
3. Generation of processed speech by taking the magnitude value \( \text{spout}(f) \), expressed in linear amplitude units, combined with the original phase values, and finally using IFFT.

In comparison with Cambridge's method, this proposed method avoids the calculation of excitation pattern \( \text{ex}(f) \) and the enhancement function \( \text{en}(f) \) which are the major burden of Cambridge's method and hence does not need any convolution-like computation at all. As a result, this proposed method is very simple from a computation complexity and real-time implementation point of view.

Now, we will prove that this proposed method can effectively enhance the spectral contrast.

![Fig. 4. The results of Cambridge's method with different enhancement degree parameter \( M \). (a) \( M = 0.1 \); (b) \( M = 0.3 \); (c) \( M = 0.5 \).](image)
Assuming \(\text{spin}(f_1)\) and \(\text{spin}(f_2)\) are the magnitude spectrum at the peak and valley, respectively, then we have

\[
|\text{spout}(f_1) - \text{spout}(f_2)| = (1 + M) |\log(\text{spin}(f_1)) - \log(\text{spin}(f_2))|.
\] (8)

Because \(M\) is a positive constant, Eq. (8) shows the enhancement of the spectral contrast with an amount of \(1 + M\).

It is worth mentioning that if we replace the contrast weight \(C\) of Eq. (1) by \(1 + M\), then Eq. (1) becomes

\[
T_i = (1 + M)S_i - MS,
\] (9)

which is similar to Eq. (7). The major difference between Eqs. (9) and (7) is the involvement of the average spectrum level \(S\) in Eq. (9), which requires additional computation. Moreover, as pointed out by Bunnell (1990), for the purpose of getting the desired performance improvement and overcoming disadvantages of his algorithm, non-uniform contrast weights should be used in his algorithm, that is, contrasts were enhanced mainly at middle frequencies, leaving high and low frequencies relatively unaffected. All these make the real-time implementation of Bunnell’s algorithm more complicated and more difficult than that of our proposed algorithm. In addition, Bunnell (1990) dealt with speech in quiet, rather than speech in noise; that is, the results of Bunnell’s algorithm for processing speech in noise were not reported.

It should also be noted that if we choose \(M = 1\) in Eq. (7) then the processing of this proposed algorithm becomes to square spectrum as Boers used in (Boers, 1980). However, our experimental results have shown that \(M = 1\) always results in an unacceptable signal distortion, although this choice of enhancement degree parameter can offer the simplest implementation structure. As we can see from the following results, \(M\) should be less than 0.5 for real applications of this proposed algorithm.

Because of its simplicity, we can implement this proposed algorithm in hardware and include it in the DSP based digital hearing-aid products. This hearing-aid system with this technique may include the following parts: A/D converter, window overlap, FFT, compression gain calculation, spectral contrast enhancement gain calculation, IFFT, overlap-add, and D/A converter. The A/D unit converts the microphone signal to the digital domain and then sends it to one programmable (assembly code) DSP chip which performs all the above processing. Hanning window overlap processing before FFT is necessary to overcome the time aliasing problem and its artifacts at the final output. FFT processing provides both the magnitude and phase in linear domain for each frequency bin. Compression gain part determines how large gain (amplification) would be used for each perceptual frequency band on the basis of the hearing loss characteristics (typically, the audiogram) by some available fitting algorithm. The contrast enhancement gain for each frequency bin is calculated according to the proposed algorithm. As a matter of fact, the contrast enhancement gain calculation is accomplished in linear domain mainly because all values in digital implementation are limited to the range from \(-1\) to \(1\). After compression processing and contrast enhancement processing, the magnitude of each frequency bin is combined with the corresponding phase obtained by FFT and then IFFT processing is carried out. The overlap-add after IFFT is also necessary for undoing the processing of the window overlap before FFT. Finally the output is converted to acoustical signal via the receiver. In addition, because the output of the receiver may cause feedback via the microphone, an adaptive feedback cancellation processing should be included in this hearing-aid system.

Investigation of the performance of this proposed algorithm has been made in various ways such as laboratory tests and subject listening (for both HI people and normal people), and under various conditions such as

1. different signal and noise sources: speech in traffic noise, speech with water noise, speech in restaurant and cafeteria, speech in kitchen, speech with music, and so forth.
2. different SNRs: \(-15, -10, -5, 0, 5\) and \(10\) dB.
3. different enhancement degree parameters \(M\): from 0.1 to 0.5.
Figs. 5(a)–(d) are the results for speech in traffic noise and SNRs of 0, −5, −10 and −15 dB, respectively. In this example, we selected $M = 0.3$. Main peaks in the spectrum are located at frequency 810, 1180 and 2030 Hz, respectively; main valleys are located at 920, 1850 and 2880 Hz. Obviously, the differences between peaks and valleys have been enhanced by this proposed processing method. For example, the differences between the peak at 2030 Hz and the valley at 2880 Hz in the spectra of Fig. 5(a) are 25.8 and 33.9 dB before and after enhancement processing, respectively. As a matter of fact, according to Eq. (8), the desired enhanced spectral difference is 33.6 dB, which is approximated by the real result 33.9 dB in this example.

Table 1 gives a set of comparisons between the desired and measured enhancement amounts for different frequency bins in this example.

Note that in this table, 500 Hz is taken as a reference frequency bin, the enhancement degree parameter $M$ is selected to be 0.25, and the compression gain is 0 dB for all frequency bins.

Unlike Cambridge’s method, the degree of the contrast enhancement in this proposed algorithm does not depend on the SNR of the input signal. This can been seen from Eq. (8) and the above results. However, the enhancement degree highly depends on values of the parameter $M$. To illustrate this, Figs. 6(a)–(c) show a set of simulation

![Figures](image-url)
results for $M$ at 0.1, 0.3 and 0.5, respectively. In this example, speech in restaurant noise is considered and the SNR is $-5$ dB.

These results and theoretical analyses demonstrate that this proposed method can enhance the spectral contrast effectively. However, this proposed method suffers from the same problem that Cambridge’s method does, namely the conflict between the enhancement degree and distortion of the desired signal. In addition, this proposed method enhances all spectral details, even when they result from noise. Section 4 presents another method that avoids this problem by combining the above two methods: it is a modified version of Cambridge’s method, using the enhancement method proposed in Section 3.

Table 1
Enhancement comparisons between the desired and measured outputs

<table>
<thead>
<tr>
<th>dB SPL</th>
<th>500 Hz</th>
<th>1000 Hz</th>
<th>1500 Hz</th>
<th>2000 Hz</th>
<th>2500 Hz</th>
<th>3000 Hz</th>
<th>3500 Hz</th>
<th>4000 Hz</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input signal</td>
<td>50.7</td>
<td>70.8</td>
<td>79.1</td>
<td>59.2</td>
<td>63.2</td>
<td>65.2</td>
<td>58.7</td>
<td>54.3</td>
</tr>
<tr>
<td>Desired output</td>
<td>50.7</td>
<td>75.8</td>
<td>86.2</td>
<td>61.3</td>
<td>66.3</td>
<td>68.8</td>
<td>60.7</td>
<td>55.2</td>
</tr>
<tr>
<td>Measured output</td>
<td>50.7</td>
<td>75.4</td>
<td>86.3</td>
<td>61.1</td>
<td>66.8</td>
<td>69.1</td>
<td>60.3</td>
<td>55.5</td>
</tr>
</tbody>
</table>

Fig. 6. The results of the SSCE algorithm with different enhancement degree parameter $M$. (a) $M = 0.1$; (b) $M = 0.3$; (c) $M = 0.5$. 
4. Excitation pattern based method

As pointed out in Section 2, the calculation of excitation pattern $ex(f)$ is mainly used to avoid enhancing spectral details and to preserve peaks corresponding to major spectral prominence in the speech. With this, if we simply replace the magnitude spectrum $spin(f)$ of Eq. (7) by the excitation pattern, then we can get another spectral contrast enhancement method as follows:

1. Transform of the input signal to frequency domain by performing FFT.
2. Calculation of excitation pattern $ex(f)$ by use of Eqs. (2) and (3).
3. Enhancement of the excitation pattern by use of

$$spout(f) = M \log(ex(f)) + \log(ex(f)),$$

which is based on Eq. (7).

4. Combination of the magnitude value $spout(f)$ expressed in linear amplitude units with the original phase values and obtaining the processed speech using IFFT.

In comparison with Cambridge’s method, this new method needs only the calculation of excitation pattern; hence its computational complexity is about half that of Cambridge’s method. On the other hand, the cost to avoid enhancing spectral details is the calculation of excitation pattern, which makes this excitation-pattern based (EPB) method much more computationally expensive in

Fig. 7. The results of the EPB algorithm with different SNRs. (a) SNR = 0 dB; (b) SNR = -5 dB; (c) SNR = -10 dB; (d) SNR = -15 dB.
comparison with the proposed method in Section 3.

Under the same situations and the same frame configuration as in Section 3, we investigated the performance of this EPB method. Figs. 7(a)–(d) are the results with a speech in traffic noise and with SNRs of 0, −5, −10 and −15 dB, respectively. In this example, we selected $M = 0.3$. Note that the input signal in this example is the same as that of Fig. 2(a)–(d). Because the enhancement gain function is no longer involved in this proposed method, the effect of the SNR of the input signal on the enhancement amount of the output spectrum is much less than that with Cambridge’s method. However, the parameter $M$ is still the principal factor determining the enhancement amount of the output spectrum. To illustrate it, Figs. 8(a)–(c) show a set of simulation results with $M$ being 0.1, 0.3 and 0.5, respectively. In this example, speech in restaurant noise is considered and the SNR is −5 dB.

5. Comparisons and discussions of the three algorithms

In this section, we will make further comparisons and discussions about these three spectral contrast enhancement algorithms with emphasis on their computational complexity and implementation. As mentioned in the above sections, Cambridge’s algorithm is the most complicated from an implementation point of view. As a further comparison, Tables 2–4 give the number of multiplications, additions and coefficients (corresponding to data memory) required in each step.
for implementing Cambridge’s algorithm, the SSCE algorithm proposed in Section 3, and the EPB algorithm proposed in Section 4, respectively, where $2N$ is the length of FFT and IFFT. It should be noted that Stages 3 and 4 of Table 2 form Step 3 of Cambridge’s algorithm. Also, in these tables $N$ must be a power of 2 and greater than 4. Moreover, because the DFT of real-data has the conjugate symmetry, we can use length-$N$ complex-valued FFT to calculate length-$2N$ real-valued FFT and IFFT with some additional pre-processing and post-processing to further reduce the computational complexity (Guo et al., 1998). This reduction of complexity has been taken into account in the calculation related to numbers in these tables.

In addition, there are two ways to get the weight coefficients required in Cambridge’s algorithm and the EPB algorithm in real-time implementation. One is to calculate them according to Eq. (2) online, which does not need additional data memory to store these coefficients. The other is to first calculate them off-line and then to store them in additional data memory. Because there is a symmetric property in these coefficients, only half of coefficients need to be stored. In the results of Tables 2 and 4 this symmetry has been used. The DoG function required in Cambridge’s algorithm has the same problem and the same property as above for weight coefficients, which has been taken into account in Table 2 as well. In these tables, we assume that one coefficient needs two bytes (16 bits).

For the configuration with the frame length is 128, that is, $N = 64$, Table 5 gives the main operation (multiplication and addition) number and data memory size for implementing the three algorithms. It can be seen from Table 5 that the number of main operations required in the SSCE

<table>
<thead>
<tr>
<th>Table 2</th>
<th>Complexity for implementing Cambridge’s algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stage 1</td>
<td>$\frac{1}{2} \log_2(N) - \frac{1}{2} N + 1$</td>
</tr>
<tr>
<td>Stage 2</td>
<td>$N^2$</td>
</tr>
<tr>
<td>Stage 3</td>
<td>$N^2 + 2N$</td>
</tr>
<tr>
<td>Stage 4</td>
<td>$N$</td>
</tr>
<tr>
<td>Stage 5</td>
<td>$\frac{1}{2} \log_2(N) - \frac{1}{2} N + 2$</td>
</tr>
<tr>
<td>Total</td>
<td>$2N^2 + N \log_2(N) + \frac{1}{2} N + 3$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 3</th>
<th>Complexity for implementing SSCE algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stage 1</td>
<td>$\frac{1}{2} \log_2(N) - \frac{1}{2} N + 1$</td>
</tr>
<tr>
<td>Stage 2</td>
<td>$N$</td>
</tr>
<tr>
<td>Stage 3</td>
<td>$\frac{1}{2} \log_2(N) - \frac{1}{2} N + 2$</td>
</tr>
<tr>
<td>Total</td>
<td>$N \log_2(N) - \frac{1}{2} N + 3$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 4</th>
<th>Complexity for implementing EPB algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stage 1</td>
<td>$\frac{1}{2} \log_2(N) - \frac{1}{2} N + 1$</td>
</tr>
<tr>
<td>Stage 2</td>
<td>$N^2$</td>
</tr>
<tr>
<td>Stage 3</td>
<td>$N$</td>
</tr>
<tr>
<td>Stage 4</td>
<td>$\frac{1}{2} \log_2(N) - \frac{1}{2} N + 2$</td>
</tr>
<tr>
<td>Total</td>
<td>$N^2 + N \log_2(N) - \frac{1}{2} N + 3$</td>
</tr>
</tbody>
</table>
algorithm is only one twelfth of that required in Cambridge's algorithm and one sixth of that required in the EPB algorithm. In addition, no additional data memory is needed at all in the SSCE algorithm. These properties of the SSCE algorithm provide great simplicity for implementing this algorithm in real time.

6. Conclusions

This paper investigated spectral contrast enhancement techniques and their hardware implementation complexity. Because of the extensive computational complexity of Cambridge's method, we proposed two alternative methods and investigated their performance. We have implemented one of these two proposed algorithms in hardware. The theoretical analysis, laboratory testing and subject listening results have shown that by using these three methods, the desired spectral contrast enhancement and performance improvement can be achieved with the appropriate selection of the related parameters. The common problem of these three methods is the conflict between the enhancement of spectral contrast and the distortion of the desired signal. Consequently, we make a trade off and select the appropriate values of the related parameters according to application situations.

Acknowledgements

We are grateful to the anonymous reviewers for their very useful suggestions and valuable comments.

References


