Navpreet Singh, Mandeep Kaur,Rajveer Kaur / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 3, Issue 4, Jul-Aug 2013, pp.2222-2228

An Enhanced Low Bit Rate Audio Codec Using Discrete Wavelet Transform

Navpreet Singh[1], Mandeep Kaur[2], Rajveer Kaur[3]
[1,2](M. tech Students, Department of ECE, Guru kashi University, Talwandi Sabo(BTI.), Punjab, INDIA
[3](Asst. Prof. Department of ECE, Guru Kashi University, Talwandi Sabo (BTI.), Punjab,INDIA

Abstract

Audio coding is the technology to represent audio in digital form with as few bits as possible while maintaining the intelligibility and quality required for particular application. Interest in audio coding is motivated by the evolution to digital communications and the requirement to minimize bit rate, and hence conserve bandwidth. There is always a tradeoff between lowering the bit rate and maintaining the delivered audio quality and intelligibility. The wavelet transform has proven to be a valuable tool in many application areas for analysis of nonstationary signals such as image and audio signals. In this paper a low bit rate audio codec algorithm using wavelet transform has been proposed. The major issues concerning the development of algorithm are choosing optimal wavelets for audio signals, decomposition level in the digital wavelet transform and thresholding criteria for coefficient truncation which is the basis to provide compression ratio for audio with suitable peak signal to noise ratio (PSNR), wavelet packet compression technique has also been used to compare the performance of audio codec using wavelet transform. After reconstructing the audio signal a postfiltering technique is used to improve the quality of reconstructed audio signal. In postfiltering technique the error of the coded audio signal is estimated and subtracted from the coded audio signal.

***Keywords* -** Audio coding, MATLAB 7.5, DWT, DWPT, Post-Filter.

I. INTRODUCTION

Audio signal compression has found application in many areas, such as multimedia signal coding, high-fidelity audio for radio broadcasting, audio transmission for HDTV, audio data transmission/sharing through Internet, etc. High-fidelity audio signal coding demands a relatively high bit rate of 705.6 kbps per channel using the compact disc format with 44.1 kHz sampling and 16-bit resolution. For large amount of exchange and transmission of audio information through internet and wireless systems, efficient (i.e., low bit rate) audio coding algorithms need to be devised. Two major classes of techniques can be used in audio to source coding reduce coded bit rate. The first class employs some signal processing so that essential information and perceptually irrelevant signal components can be separated and later removed. This class includes techniques such as subband coding; transform coding, critical band analysis, and masking effects. The second class takes advantage of the statistical redundancy in audio signal and applies some form of digital encoding. Examples of this class include entropy coding in lossless compression and scalar/vector quantization in lossy compression [1].

Digital audio compression allows the efficient storage and transmission of audio data. The various audio compression techniques offer different levels of complexity, compressed audio quality, and amount of data compression. Digital audio processing means processing of the digital audio by digital computers. The digital audio processing techniques serve two main purposes, first improving quality of the audio signal and the second; processing digital audio for storage, transmission and representation according to machine perception.

1.1 Basics of Audio Signal

The sounds heard in all of nature are analog in nature, i.e. they are continuous in time. Our auditory systems process these waveforms in their natural analog forms. An audio signal is usually not stored in an analog format however, in modern audio systems, there is an increasing use of digital technology. It is advantageous to store, retrieve, and process an audio signal in the digital domain. Audio coding is a digital operation. It must therefore exist in the digital domain first before it can be processed. The standard digital coding format which has gained acceptance is the pulse code modulation (PCM) format. Data rate requirement for stereo PCM is very high. Using the CD-Audio format for example, requires 16 bits/sample × 44,100 samples/sec × 2 channels = 1.41 Mbits/sec. With multi-channel sources, it can be seen how the bandwidth increases if PCM audio is chosen as the format. It becomes clear that storage requirements and transmission problems can be improved if data reduction is used to minimize the size of the digital file [2].

1.2 Digital Audio Data

The digital representation of audio data offers many advantages: high noise immunity, stability, and reproducibility. Audio in digital form also allows the efficient implementation of many audio processing functions (e.g., mixing, filtering, and equalization) through the digital computer. The conversion from the analog to the digital domain begins by sampling the audio input in regular, discrete intervals of time and quantizing the sampled values into a discrete number of evenly spaced levels. The digital audio data consists of a sequence of binary values representing the number of quantizing levels for each audio sample. The method of representing each sample with an independent code word is called pulse code modulation (PCM). Fig 1.1 shows the digital audio process.

According to the Nyquist theory, a time-sampled signal can faithfully represent signals up to half the sampling rate. Typical sampling rates range from 8 kilohertz (kHz) to 48 kHz. The 8-kHz rate covers a frequency range up to 4 kHz and so covers most of the frequencies produced by the human voice. The 48-kHz rate covers a frequency range up to 24 kHz and more than adequately covers the entire audible frequency range, which for humans typically extends to only 20 kHz. In practice, the frequency range is somewhat less than half the sampling rate because of the practical system limitations

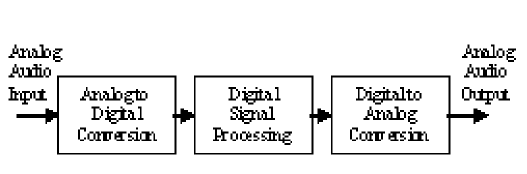

Fig 1.1 Digital Audio Process

1.3 Conventional Data Rate Reduction for Audio

The simple methods used for reducing the data requirement for PCM data storage are as follows:

• Reduce the word length. For example, instead of using 16-bits to represent each audio sample, use 12-, 10-, or 8-bit per samples.

• Reduce the sample rate. Instead of a 44.1 kHz sample rate, use a 22.05 or 11.025 kHz sampling rate. This will reduce the total number of values stored.

For example, if 8 bits per sample and 22.05 kHz sampling rate is chosen, the overall data rate can be reduced by a factor of four. In practice, however, neither of these strategies is useful, since audio quality is impacted greatly when using either or both of these methods, enough to have any benefits. If the word length is truncated, the quality of each digital sample is limited, which in turn limits the overall perceptual quality of the audio. If sampling frequency is lowered, higher frequencies are unable to be coded due to the Nyquist limit that states one can only represent frequencies equal to one-half of the sampling rate chosen. It is desired to reduce the data rate without impacting audio quality, so other options must be looked to for reducing data rate.

1.4 Need of Compression for Audio

Emerging digital audio applications in networks, wireless, and multimedia computers face serious shortfalls such as bandwidth limitations, and limited storage capacity. These technologies have created a demand for high quality audio that can be transferred and stored at low bit rates. This creates a need of compression, whose role is to minimize the number of bits needed to retain acceptable quality of the original source signal [3]. One disadvantage of digital audio music is that it takes a great amount of information to represent a small amount of sound. It takes nearly 1.41 MB of information to represent one second of music, and an entire song can take up to 50 MB. It is not feasible or efficient to transfer this uncompressed audio over the internet, and therefore audio compression helps to reduce the amount of information needed for the digital content and the amount of time needed to download the content.

II. THEORY

2.1 Discrete Wavelet Transform

The DWT is based on subband coding and is found to yield a fast computation of Wavelet Transform. It is easy to implement and reduces the computation time and resources required. In the discrete wavelet transform, a signal can be analyzed by passing it through an analysis filter bank followed by a decimation operation. This analysis filter bank, which consists of a low pass and a high pass filter at each decomposition stage, is commonly used in audio signal compression. When a signal passes through these filters, it is split into two bands. The low pass filter, which corresponds to an averaging operation, extracts the coarse information of the signal. The high pass filter, which corresponds to a differencing operation, extracts the detail information of the signal. The output of the filtering operations is then decimated by two [4]. The wavelet decomposition steps are shown in fig 2.1. At each decomposition level, the half band filters produce signals spanning only half the frequency band. This doubles the frequency resolution as the uncertainty in frequency is reduced by half.

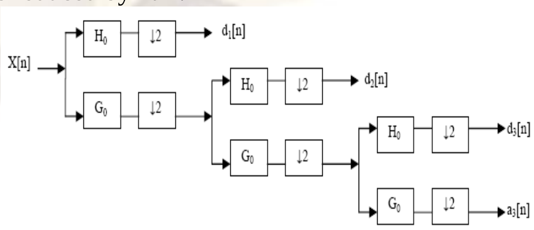

Fig 2.1 Wavelet Decomposition (3 levels)

In accordance with Nyquist's rule if the original signal has a highest frequency of ω, which requires a sampling frequency of 2ω radians, then it now has a highest frequency of ω/2 radians. It can now be sampled at a frequency of ω radians thus discarding half the samples with no loss of information. This decimation by 2 halves the time resolution as the entire signal is now represented by only half the number of samples. Thus, while the half band low pass filtering removes half of the frequencies and thus halves the resolution, the decimation by 2 doubles the scale.

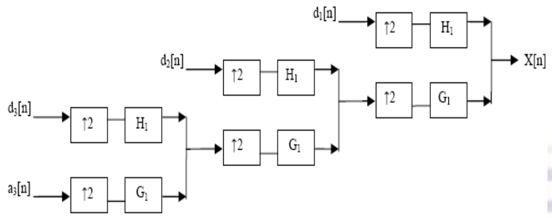

Fig 2.2 Wavelet Reconstruction Tree (3 levels)

Fig 2.2 shows the reconstruction of the original signal from the wavelet coefficients. Basically, reconstruction is the reverse process of decomposition. The approximation and detail coefficients at every level are upsampled by two, passed through the low pass and high pass synthesis filters and then added. This process is continued through the same number of levels as in the decomposition process to obtain the original signal. The Mallat algorithm works equally well if the analysis filters, G_0 and H_0, are exchanged with the synthesis filters, G_1 and H_1.

2.2 Discrete Wavelet Packet Transform

The wavelet packet method is a generalization of wavelet decomposition that offers a richer range of possibilities for signal analysis. In wavelet analysis, a signal is split into an approximation and a detail. The approximation is then itself split into a second-level approximation and detail, and the process is repeated. For n-level decomposition, there are n+1 possible ways to decompose or encode the signal.

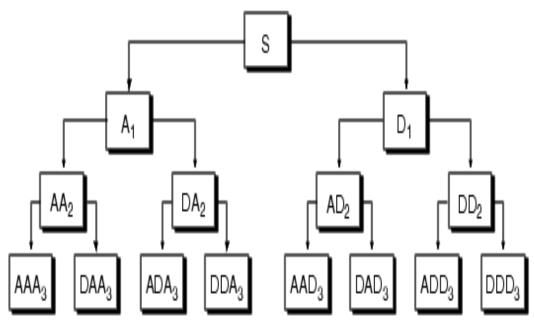

Fig 2.3 Wavelet Packet Decomposition

In wavelet packet analysis, the details as well as the approximations can be split. This yields more than $2^{2^{n-1}}$ different ways to encode the signal. Figure 2.3 shows the level 3 decomposition using wavelet packet transform. This is the wavelet packet decomposition tree. The wavelet decomposition tree is a part of this complete binary tree. For instance, wavelet packet analysis allows the signal S to be represented as A1 + AAD3 + DAD3 + DD2. This is an example of a representation that is not possible with ordinary wavelet analysis. Choosing one out of all these possible encodings presents an interesting problem. In wavelet packet analysis, an entropy-based criterion is used to select the most suitable decomposition of a given signal. This means we look at each node of the decomposition tree and quantify the information to be gained by performing each split.

2.3 Design Considerations for Audio Coders

When selecting a digital audio encoder for an application, various considerations needed to be taken into account. These considerations include the audio compression rate, audio quality, format of the compressed bit stream, complexity of the encoding and decoding algorithm, and speed of compression and decompression [5] [6].

2.3.1 Compression Ratio

Depending on the algorithms that are used, different coders will possess a different maximum compression ratio. If the maximum possible compression ratio is desired, the best solution will be an audio coder that implements both lossy and lossless techniques. The compression ratio of a coder is usually defined as

$R_{cr} = S/C$ (dimensionless)

where S is the size of the source file and C is the size of the compression file.

2.3.2 Audio Quality

Most audio coders use lossy compression technique as they are willing to sacrifice small degradations in audio quality for a higher compression ratio. Lossless compression is used either for data compression, where it is important that all the data is preserved, or in conjunction with a lossy technique to provide additional compression. There is no need to determine audio quality for lossless compression as it perfectly preserves the signal. Quality of a lossy compression signal can be determined either objectively or subjectively. Objective metrics use the peak signal-to-noise ratio (PSNR) which can also be defined as

$$PSNR = 10\log_{10}(NX^2/(x-y)^2)$$

where N is the length of reconstructed audio signal, X is the maximum absolute square value of original audio x and y is the corresponding reconstructed audio [6].

2.3.3 Complexity and Delay of the Encoder/Decoder

Complexity and speed are usually directly related for audio encoders: those that achieve higher compression ratios require more complex algorithms and produce higher delays during compression and decompression. Examples of this are the various layers of the MPEG-1 family. While each successive layer provides additional compression, they are more complex and therefore more delay.

III. STRUCTURE OF THE CODEC

The block diagram of low bit rate audio codec algorithm using wavelet transform is shown in fig 3.1. The original audio files used for coding are .wav files. Wavelets concentrate audio information (energy and perception) into a few neighboring coefficients. Therefore after finding out wavelet transform of a signal, many coefficients will either be zero or have negligible magnitudes. Signal compression is achieved by treating small valued coefficients as insignificant data and thus discarding them. The process of coding an audio signal using wavelets involves a number of stages, each of them are discussed as follows:

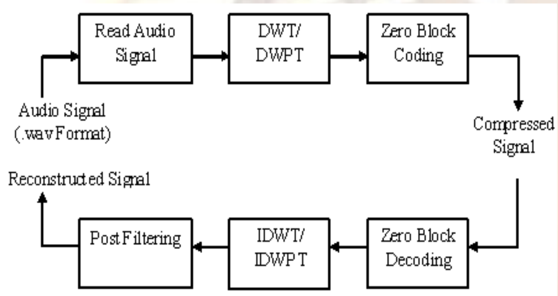

Fig 3.1 Block Diagram of Codec

3.1 Choice of Wavelet

The choice of the mother-wavelet function used in designing high quality audio coders is of prime importance. Choosing a wavelet that has compact support in both time and frequency in addition to a significant number of vanishing moments is essential for an optimum wavelet audio coding. Wavelets with more vanishing moments provide better reconstruction quality, as they introduce less distortion into the processed audio signal and concentrate more signal energy in a few neighboring coefficients. However the computational complexity of the DWT increases with the number of vanishing moments. Several different criteria are used in selecting an optimal wavelet function. The objective is to minimize reconstructed error variance and maximize peak signal to noise ratio (PSNR). In general optimum wavelets can be selected based on the energy conservation properties in the approximation part of the wavelet coefficients. The Haar, Daubechies, Coiflets, Symlets and Biorthogonal wavelets have been used for codec algorithm.

3.2 Decomposition Level

Signal coding is based on the concept that selecting a small number of approximation coefficients (at a suitably chosen level) and some of the detail coefficients can accurately represent regular signal components. Choosing a decomposition level for the DWT usually depends on the type of signal being analyzed or some other suitable criterion such as entropy. For the processing of audio signals decomposition up to level 5 is adequate, with no further advantage gained in processing beyond that. Therefore the audio signal has been tested using level 3 and 5 only.

3.3 Truncation of coefficients

Since most of the energy of audio signal is present in high-valued coefficients. Therefore the small valued coefficients are truncated or zeroed and later on used to reconstruct the signal [6].
Two different approaches are available for calculating thresholds. The first, known as Global Thresholding involves taking the wavelet expansion of the signal and keeping the largest absolute value coefficients and a global threshold is set manually. The second approach known as By Level Thresholding consists of applying visually determined level dependent thresholds to each decomposition level in the wavelet transform.
Global thresholding has been used where thresholding parameter is set manually. The normalized amplitude of sampled audio signal is in the range from [-1, +1]. The audio signal has been tested at various threshold values ranging from 0.02 to 0.2, and providing bit rates from 20 to 238 kbps after encoding the signal.

3.4 Encoding Zero Valued Coefficients

After zeroing wavelet coefficients with negligible values based on manually set threshold value, the transform vector needs to be encoded. Consecutive zero valued coefficients are encoded with two bytes. One byte is used to specify a starting string of zeros and the second byte keeps track of the number of successive zeros. Due to the scarcity of the wavelet representation of the audio signal, this encoding method leads to a higher compression ratio than storing the non-zero coefficients along with their respective positions in the wavelet transform vector. This encoding scheme is the primary means of achieving signal with low bit rate [7] [8].

3.5 Postfiltering

After reconstructing the signal, there is always some coding error, which degrades the quality of reconstructed audio signal. The postfiltering technique is used to enhance the perceptual quality of audio signal coded at low bit rates. In postfiltering the reconstruction error of the coded audio signal is estimated and subtracted from the coded audio signal, so that the noise level in the coded audio signal is suppressed and hence better perceptual quality is achieved [9]. The postfilter is modeled as a time variant non linear filter performed on the noisy signal [7]. Noise reduction is achieved by attenuating short time spectrum components that are with low signal to noise ratio (SNR), as shown in fig 3.2

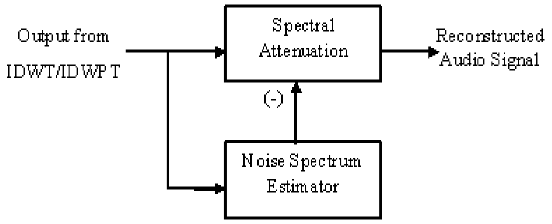

Fig 3.2 Basic Structure of Postfiltering

IV. RESULTS

The performance of the algorithm is evaluated by considering different parameters such as decomposition levels, optimal wavelets and threshold value for wavelet coefficients to obtain low bit rate signal. PSNR is also calculated by varying above parameters which affect the quality of reconstructed signal.

The proposed method has been implemented in Matlab 7.5 . Several audio signals of different time duration and quality have been used for testing purposes. The audio signals have been tested for different wavelets (Haar, Daubechies, Symlets, Coiflets,Biorthogonal and complex gaussian) function at level 3, 5, 7 and 9, of which Biorthogonal wavelets gives significant improvement in PSNR at level 5. Therefore results have been shown only for complex gaussian wavelets for all tested signal. The input audio signals have been compressed at different threshold value for different bit rates and compression ratio.

4.1 Results for 'audio1.wav' Audio signal

The test signal 'audio1.wav'of size 172 KB is formed by converting the MP3 file of audio into .wav file by 'Meda mp3 splitter' software. The 'audio1.wav' has 85 sampled Frames of 16 bits/sample with sampling frequency 44.1 kHz. The input signal has bit rate of 705.6 kbps, and 26 seconds long duration. Since the amplitude values of sampled data are in the range from [-1, +1], so the input audio has been tested at threshold value of 0.226, 0.195, 0.168, 0.133, 0.108 and 0.09 which provide the bit rate of 20, 30, 40, 60, 80 and 100 kbps respectively after encoding the signal. The algorithm was tested for different wavelet functions with different level, of which only cgau3 gives significant improvement in PSNR, as shown in Fig 4.1

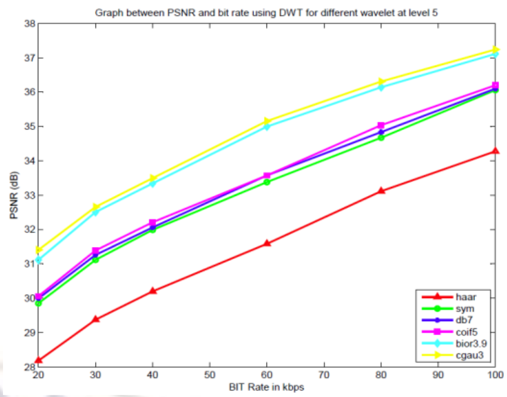

Fig 4.1: Graph between PSNR and bit rate using DWT for different wavelet at level 5 for 'audio1.wav'.

The PSNR values at different bit rates are shown in Table 4.1 with cgau3 wavelet using DWT at level 3 and 5 for 'audio1.wav' with and without postfiltering and Table 4.2 shows PSNR values with cgau3 wavelet using DWPT for 'audio1.wav' with and without postfiltering. There is small improvement in PSNR with wavelet transform over wavelet packet transform.

Audio : audio1.wav					
Wavelet : cgau3 DWT					
Compression Ratio	Bit Rate (kbps)	PSNR in db without postfiltering		PSNR in db with postfiltering	
		Level 3	Level 5	Level 3	Level5
35.3000	20	31.1821	31.4180	31.0083	31.1048
23.3940	30	32.5038	32.7485	32.5591	32.7102
17.6550	40	33.2296	33.4398	33.4609	33.7193
11.7666	60	34.9218	35.1085	35.2407	35.8220
8.8250	80	35.9491	36.3671	36.8113	37.2178
7.0600	100	37.3376	36.3724	38.1433	38.6616

Table 4.1: PSNR values for 'audio1.wav' using DWT for cgau3 wavelet

Audio : audio1.wav					
Wavelet : cgau3 DWPT					
Compression Ratio	Bit Rate (kbps)	PSNR in db without postfiltering		PSNR in db with postfiltering	
		Level 3	Level 5	Level 3	Level5
35.3000	20	31.1821	31.2557	31.0083	31.0066
23.3940	30	32.5038	31.7298	32.5591	32.5969
17.6550	40	33.2296	32.6561	33.4609	33.5425
11.7666	60	34.9218	34.1311	35.2407	35.4528
8.8250	80	35.9491	35.4896	36.8113	37.1448
7.0600	100	37.3376	36.6551	38.1433	36.4543

Table 4.2: PSNR values for 'audio1.wav' using DWPT for cgau3 wavelet

Fig 4.2 shows a graphical representation for the comparison of PSNR values at decomposition level 5 for cgau3 wavelet using DWT for 'audio1.wav' with and without post filtering. There is small improvement in PSNR with postfiltering.

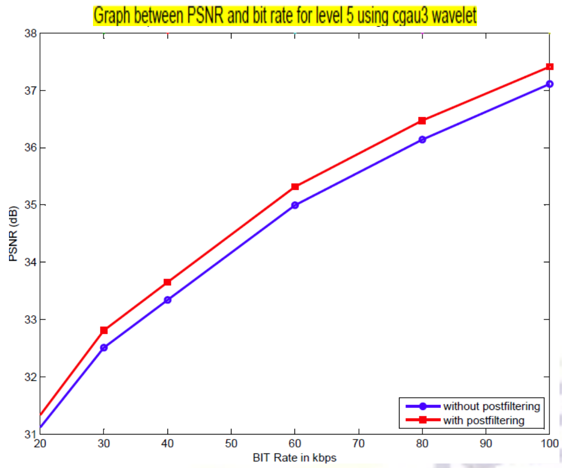

Fig 4.2: Graph between PSNR and bit rate for level 5 using cgau3 wavelet for 'audio1.wav' with and without postfiltering

Fig 4.3 shows a graphical representation for the comparison of PSNR values at level 5 for cgau3 wavelet using DWT and DWPT for 'audio1.wav'.

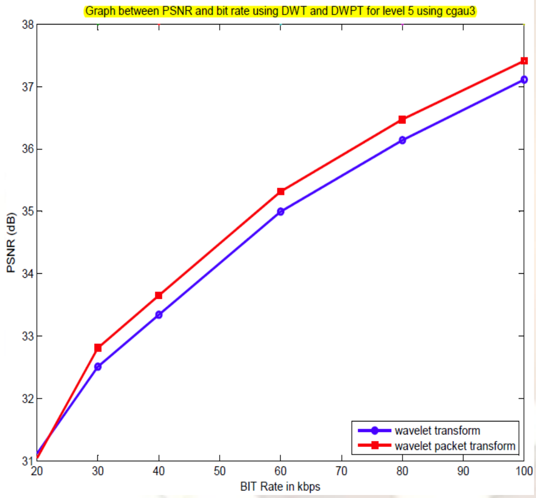

Fig 4.3: Graph between PSNR and bit rate using DWT and DWPT for level 5 using cgau3 wavelet for 'audio1.wav'.

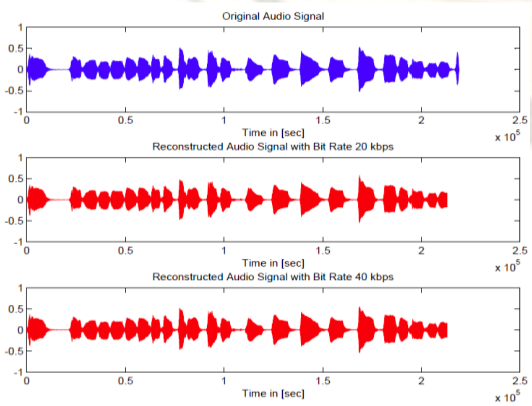

Fig 4.4: Original and reconstructed 'audio1.wav'

Fig 4.4 shows Original and reconstructed signal 'audio1.wav' using cgau3 wavelet for level 5 at 20 and 40 kbps. At 20 kbps quality of reconstructed signal is not good due to higher compression ratio, whereas at 40 kbps the quality of reconstructed signal is good and it is comparable with original audio signal.

V. CONCLUSION

In this paper, a low bit rate audio codec algorithm using wavelet transform and wavelet packet transform has been developed, which is simple yet effective compression technique. The algorithm successfully improves the quality of the reconstructed audio signal by using postfiltering at suitable bit rates. To test the algorithm several audio signal of different time duration has been used. For the same compression ratio better PSNR values have been obtained with cgau3 wavelet. Thus cgau3 wavelet had been chosen for the proposed codec. The comparative analysis of the results show that for good quality of reconstructed signal, the bit rates of the proposed codec should be in the range of 40 - 60 kbps with PSNR values 31.1047 - 35.2755 dB respectively. The results shows that the wavelet packet transform instead of wavelet transform provides the improvement in quality of reconstructed audio signal for all wavelets function except cgau3. It has been observed that postfiltering improves the quality of the reconstructed audio signal with wavelet packet transform as well as wavelet transform.

REFERENCES

[1]. Yuan-Hao Huang and Tzi-Dar Chiueh, "A New Audio Coding Scheme Using a Forward Masking Model and Perceptually Weighted Vector Quantization", IEEE transactions on speech and audio processing, July 2002.

[2] Davis Yen Pan, "Digital Audio Compression" Digital Technical Journal Vol. 5 No. 2, Spring 1993.

[3] Eric D. Majewicz, "An Analysis of Digital Audio Compression and Digital Rights Management Technique" Department of Computer Engineering Rochester Institute of Technology, November 2004.

[4] Amara Graps, "An Introduction to Wavelets", IEEE transaction on wavelets.

[5] Eric D. Majewicz, "An Analysis of Digital Audio Compression and Digital Rights Management Technique" Department of Computer Engineering Rochester Institute of Technology, November 2004.

[6] El-Bahlul Fgee, W. J. Phillips and W. Robertson, "Comparing Audio Compression Using Wavelets with other Audio Compression Schemes", Proceedings of the

1999 IEEE Canadian Conference on Electrical and Computer Engineering.

[7] Y. Karelic and D. Malah, “Compression of High-Quality Audio Signals Using Adaptive Filterbanks and a Zero-Tree Coder”, Department of Electrical Engineering Technion - Israel Institute of Technology Haifa 32000, Israel.

[8] W. Kinsner and A. Langi, “Speech and Image Signal Compression with Wavelets”, IEEE Wescanex Conference Proceedings, New York 1998.

[9] Yu Rongshan. “Improving Quality of Low Bit Rate Audio Coding by Using Short-Time Spectral Attenuation”, Centre for Signal Processing, School of EEE Nanyang Technological University Singapore, 200