On the use of wavelets for audio compression.

Rodrigo Capobianco Guido,
José Carlos Pereira, Everthon Silva Fonseca
Depto de Engenharia Elétrica, EESC, USP,
Av. Trabalhador São Carlense 400 - 13566-590 - São Carlos - SP - Brazil.
E-mail: guido@ifsc.usp.br, pereira@sel.eesc.usp.br, everthon@sel.eesc.usp.br

Lucimar Sasso Vieira, Fabrício Lopes Sanchez,
Maurício Monteiro, Emerson Jesus Rodrigues de Moura,
Paulo César Fantinato
Núcleo de Estudos de Computação, Centro Universitário do Norte Paulista, UNORP,
R. Ipiranga 3460 - Jd. Alto Rio Preto - 15020-040 - São José do Rio Preto - SP - Brazil.
E-mail: nec@unorp.br

Abstract

We have tested several wavelet transforms, including Daubechies, Haar, Coiflets, Vaidyanathan, and other perfect-reconstruction filter banks (PRFB) with several support sizes and vanishing moments applied to digital audio compression, spanning music and speech signals. A criterion that quantifies the lossy of energy the compression imposes has been established so that we find out what is the best wavelet for compressing the signals by 2:1 using the 1-level discrete wavelet transform. The results extend to higher levels of compression as well.

1 Introduction

Much literature has appeared recently describing different methods and techniques for audio compression [1] using different lossy and lossless [2] algorithms, like Huffman code, Lempel-Ziv-Welch and so on.

The purpose of this work is the lossy compression of audio signals using Perfect-Reconstruction Filter Banks (PRFB) of wavelets filters [3]. We have tested different discrete wavelet transforms (DWT) and different styles of audio data. A simple algorithm is used to quantify the lossy of energy the compression imposes so that we find out what is the best wavelet for compressing the signals. The motivation for the use of wavelets consists of promising results we have obtained with them for similar purposes [4], [5].

All the audio files we have tested are previously pulse-code-modulation (PCM) coded [6] so that the quantization precision is 16 bits and the sampling rate is 22050 samples/second. Therefore and according to the Nyquist rule [7] its maximum frequency is 22050 Hz. A set of three different style audio clips, 2 minute-long each, were applied to the proposed algorithm. The styles used are rock and love songs and speech.

The remainder of this paper is comprised of 3 sections, excluding the acknowledgments and references. Section 2 describes the proposed procedure. Sections 3 describes the tests and results. Lastly, section 4 lists the conclusions.

2 The Proposed Procedure

Table 1 describes the complete procedure to test the wavelets with the audio files. These files are according to the above-mentioned conditions. The energy criterion adopted is the usual energy operator, $E = \sum_{k=0}^{255} {x_k}^2$, x being each sample of the frame. The limits in the sum are due to the length of the audio frames used.

3 Tests and Results

We have found the results summarized in table 2 according to the averages of the compression rates loaded in the array *fcr*. A total of 4096 frames, 256 sample-long, were tested for each style of audio clip.

The proposed algorithm was implemented using C++ programming language under RedHat Linux 9.0.

———————————————————————————

- BEGINNING
- *Step E-1:* Input the raw data of the original discrete-time audio signal under analysis and divide it into n sequential frames of 256 samples each. Only the last frame may contain less than 256 samples just when the length of the signal is not multiple of 256. In this case the last frame will be discarded;
- *Step E-2:* For wavelets $j = 1$ to 16, according to table 2:
  - *Step E-3.2.1:* Make $ep \leftarrow 0$. This variable loads the average compression rates;
  - *Step E-3.2.2:* For frames $i = 1$ to n:
    * *Step E-3.2.2.1:* Measure the original energy of the frame, $E_{ORIGINAL}$;
    * *Step E-3.2.2.2:* Convert the data into its 1-level W_j discrete wavelet transform, W_j being the wavelet basis chosen, according to the table 2 in section 3;
    * *Step E-3.2.2.3:* Measure the energy of the trend part (good resolution) of the transformed data and label it as $E_{TRANSFORMED}$. This part of the transformed data corresponds to a half-band low-pass filtering followed by a sub-sample by 2 [7], according to figure 1;
    * Make $ep \leftarrow ep + \left(\frac{E_{TRANSFORMED}}{E_{ORIGINAL}}\right)$;
  - *Step E-3.2.3:* Make

    $$tep[j] \leftarrow \left(100 * (\frac{ep}{n})\right).$$

    Therefore the index j of the array tep loads the total percentage of energy that is preserved after the 1-level wavelet transform is applied to the data, for the entire audio file under analysis with the DWT_j.
- END.

———————————————————————————

Table 1: The procedure used to discover the best wavelet for audio compression.

style	**wavelet**	**energy**
rock	Haar	94.0520%
rock	Daubechies support 4	96.1200%
rock	Daubechies support 6	96.5588%
rock	Daubechies support 8	96.7236%
rock	Daubechies support 10	96.8382%
rock	Daubechies support 20	96.9528%
rock	Daubechies support 40	96.9759%
rock	Vaidyanathan support 24	96.9740%
rock	Coiflet support 6	96.3818%
rock	Coiflet support 12	96.8821%
rock	Coiflet support 18	96.9510%
rock	Burt Adelson support 6	97.1721%
rock	Beylkin support 18	97.0148%
love	Haar	99.3104%
love	Daubechies support 4	99.7617%
love	Daubechies support 6	99.7230%
love	Daubechies support 8	99.7169%
love	Daubechies support 10	99.7625%
love	Daubechies support 20	99.7990%
love	Daubechies support 40	99.8077%
love	Vaidyanathan support 24	99.8101%
love	Coiflet support 6	99.6595%
love	Coiflet support 12	99.8723%
love	Coiflet support 18	99.8300%
love	Burt Adelson support 6	99.4706%
love	Beylkin support 18	99.8593%
speech	Haar	99.8721%
speech	Daubechies support 4	99.8928%
speech	Daubechies support 6	99.8889%
speech	Daubechies support 8	99.8925%
speech	Daubechies support 10	99.9029%
speech	Daubechies support 20	99.9149%
speech	Daubechies support 40	99.9292%
speech	Vaidyanathan support 24	99.9300%
speech	Coiflet support 6	99.8873%
speech	Coiflet support 12	99.9095%
speech	Coiflet support 18	99.9122%
speech	Burt Adelson 6	99.8755%
speech	Beylkin support 18	99.8337%

Table 2: Percentage of energy preserved in the trend-part of the 1-level wavelet transform, according to the wavelet used and audio file style.

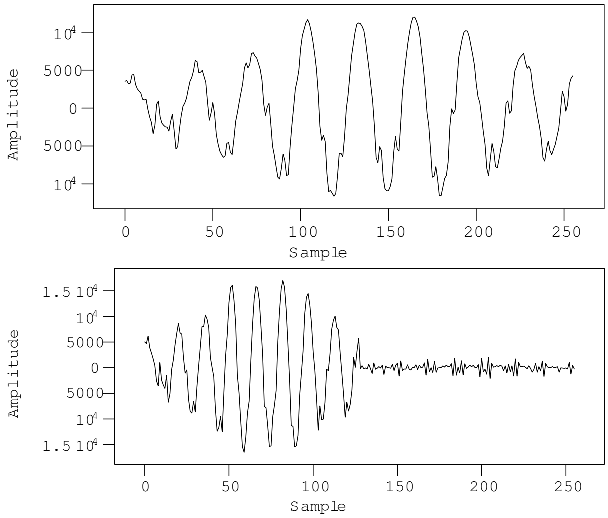

Figure 1: (top): a 256-length audio clip. (bottom): the corresponding 1-level discrete wavelet transform. The *trend* corresponds to the first half of the transformed signal (from left to right). The second half is called *fluctuation*. For audio signals, there is usually much energy in the trend and almost no energy in the fluctuation.

4 Conclusions

We have proposed a simple algorithm for choosing the best wavelet for audio compression, according to the characteristics of the proposed data. Table 2 clearly shows that the Coiflet family of wavelets is the best one for audio compression according to the criterion we have established. It preserves much part of the energy of the original audio file after a 1-level decomposition.

Although there are different results among the audio styles, one characteristic is usually kept: the closer the frequency response of the filters is to the ideal response, the higher the amount of energy in the trend part of the transformed signal.

A lot of subjects who have listened to the audio files could not distinguish between the original and the corresponding decompressed data. The latter is the one obtained by applying the inverse discrete wavelet transform discarding the fluctuation-part of the transformed signal, the one that contains the high-pass filtered data.

The major set of tests have used the 1-level discrete wavelet transform. It compresses the data by 2:1. Actually, we have extended the tests to higher levels of transforms. There are promising results for levels 3 or 4 of the transform, which lead to a significant reduction of the data by 8:1 or 16:1. The 3-level DWT takes to a decompressed data that can not be perceptually distinguished from the original too.

Please send questions and comments directly to the corresponding author (R.C.G.).

5 Acknowledgments

We would like to thank *FAPESP - The State of São Paulo Research Foundation* (http://www.fapesp.br) by the grants and support given to this work. We are very grateful to the reviewers for all their relevant suggestions.

References

[1] D. Hankerson, G.A. Harris, P.D. Johnson, "Introduction to Information Theory and Data Compression", 2nd ed. Boca Raton: CRC Press, 2003.

[2] M. Bosi, R. Goldberg, "Introduction to Digital Audio Coding and Standards". 2nd ed. Massachusetts: Kluwer Academic Publishers, 2003.

[3] P. S. Addison, "The Illustrated Wavelet Transform Handbook: Introductory theory and applications in science, engineering, medicine and finance". Edinburg: Institute of Physics Publishing, 2002.

[4] R.C. Guido, J.C. Pereira, E. Fonseca, L.S. Vieira, F.L. Sanchez. "Trying different wavelets on the search for voice disorders sorting", in *37th IEEE International Southeastern Symposium on System Theory*, Tuskegee-AL-US, v. 1 (2005) p. 495-499.

[5] R.C. Guido, J.F.W. Slaets, L.O.B. Almeida, R. Koberle, J.C. Pereira, "A new technique to construct a wavelet transform matching a specified signal with applications to digital, real time, spike and overlap pattern recognition", *Digital Signal Processing*, Elsevier, v. 15, (2005), n. – , pp. – (article in press, available on www.sciencedirect.com website, to appear in 2005).

[6] J. Watkinson, "The MPEG Handbook", 2nd ed. Burlington-MA-US: Elsevier-Focal Press, 2004

[7] G. Strang, T. Nguyen,"Wavelets and Filter Banks". Wellesley: Wellesley Cambridge Press, 1997.