ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.2 , No.1, Pages : 424 – 429 (2013)
Special Issue of ICACSE 2013 - Held on 7-8 January, 2013 in Lords Institute of Engineering and Technology, Hyderabad

Image Compression using Combined FIR-IIR Filters

Pavan Kumar K
Sree Visvesvaraya Institute of Technology & Science
Mahabubnagar AP
pavankonapoor@gmail.com

K.Pravallika
Asst.Professor, ECE Dept
Lords Institute of Engineering and Technology
Hyderabad-08
Pravallikareddyk.ece@gmail.com

ABSTRACT

The paper discussed suggests a new algorithm for image compression that combines the features useful of Finite impulse type and Infinite impulse type filters. The simulation results show that the new algorithm improves good compression ratio as it is needed for many advanced image processing applications.

Key words: FIR filter, IIR filter, Image compression, Hybrid filter.

1. INTRODUCTION

Data compression is the process of converting data files into smaller files for efficiency of storage and transmission. As one of the enabling technologies of the multimedia revolution, data compression is a key to rapid progress being made in information technology. It would not be practical to put images, audio, and video alone on websites without compression. Wavelet transform based image compression is lossy compression method. There are also other methods of lossy image compression like vector quantization (VQ), predictive coding, and Fractal compression.

Wavelet transform analysis has emerged as a major new time- frequency decomposition tool for data analysis. The wavelet transform has been found to be particularly useful for analyzing signals which are transitory, discontinuous, noisy, and so on. Its ability to examine the signal in both time and frequency resolution is distinctive and enables myriads of applications possible that traditional signal analysis tools such as Fourier transform cannot handle. It has now been applied to diverse realm of data analysis/process: climate analysis, financial indices analysis, signals de noising, characterization, feature extraction, data compression, and so on.

2. IMAGE COMPRESSION TECHNIQUES

A digital color image can be viewed as a three valued (channels) positive function $I=I(x,y)$ defined onto a plane. Its algebraic representation is obtained through an N by M by 3 matrix A. Thus, each entry of A is a three component integer vector (pixel color) expressing an intensity value at discrete location (x,y) with a precision p (for instance, one bit for each channel). Each component or layer of the image can be viewed as a single channel image, which, under particular conditions, can be analyzed independently from the others. This is not the case for RGB space, if two channels are fixed, human visual perception is very sensitive to small changes of the value of the remaining channel. Thus, even though RGB is the most common storage format for images, other formats may be better for compression.

The key step in lossy data compression in which data cannot be recovered exactly is the quantization phase, which exploits a data reduction based on their low information content. This is not optimal for RGB images. Nevertheless, for three layers, this can lead to the elimination of some low coefficients in a channel in a certain spatial location, even though the corresponding coefficients in the other layers are not eliminated because they carry high information content. When reconstructing the image at that location, a high visual distortion is introduced. The assumption of analyzing the three layers separately is valid only if they are not correlated with respect the visual appearance.

Lossless Compression

If data have been lossless compressed, the original data can be recovered exactly from the compressed data. It is generally used for applications that cannot allow any difference between the original and reconstructed data.
Run Length Encoding. Run length encoding, sometimes called recurrence coding, is one of the simplest data compression algorithms. It is effective for data sets that are comprised of long sequences of a single repeated character. For instance, text files with large runs of spaces or tabs may compress well with this algorithm. Old versions of the arc compression program used this method.

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.2 , No.1, Pages : 424 – 429 (2013)
Special Issue of ICACSE 2013 - Held on 7-8 January, 2013 in Lords Institute of Engineering and Technology, Hyderabad

RLE finds runs of repeated characters in the input stream and replaces them with a three-byte code. The code consists of a flag character, a count byte, and the repeated characters. For instance, the string `AAAAAABBBBCCCCC" could be more efficiently represented as ``*A6*B4*C5". That saves us six bytes. Of course, since it does not make sense to represent runs less than three characters in length with a code, none is used.

Thus ``AAAAAABBCCCDDDD'' might be represented as ``*A6BBCCC*D4". The flag byte is called a sentinel byte.

Lossy Compression Methods

Lossy compression techniques involve some loss of information, and data cannot be recovered or reconstructed exactly. In some applications, exact reconstruction is not necessary. For example, it is acceptable that a reconstructed video signal is different from the original as long as the differences do not result in annoying artifacts. However, we can generally obtain higher compression ratios than is possible with lossless compression.

Vector Quantization:

Vector Quantization (VQ) is a lossy compression method. It uses a codebook containing pixel patterns with corresponding indexes on each of them. The main idea of VQ is to represent arrays of pixels by an index in the codebook. In this way, compression is achieved because the size of the index is usually a small fraction of that of the block of pixels.

The main advantages of VQ are the simplicity of its idea and the possible efficient implementation of the decoder. Moreover, VQ is theoretically an efficient method for image compression, and superior performance will be gained for large vectors. However, in order to use large vectors, VQ becomes complex and requires many computational resources (e.g. memory, computations per pixel) in order to efficiently construct and search a codebook. More research on reducing this complexity has to be done in order to make VQ a practical image compression method with superior quality.

Predictive Coding:

Predictive coding has been used extensively in image compression. Predictive image coding algorithms are used primarily to exploit the correlation between adjacent pixels. They predict the value of a given pixel based on the values of the surrounding pixels. Due to the correlation property among adjacent pixels in image, the use of a predictor can reduce the amount of information bits to represent image.

This type of lossy image compression technique is not as competitive as transform coding techniques used in modern lossy image compression, because predictive techniques have inferior compression ratios and worse reconstructed image quality than those of transform coding.

Fractal Compression:

The application of fractals in image compression started with M.F. Barnsley and A.Jacquin. Fractal image compression is a process to find a small set of mathematical equations that can describe the image. By sending the parameters of these equations to the decoder, we can reconstruct the original image. In general, the theory of fractal compression is based on the contraction mapping theorem in the mathematics of metric spaces. The Partitioned Iterated Function System (PIFS), which is essentially a set of contraction mappings, is formed by analyzing the image. Those mappings can exploit the redundancy that is commonly present in most images. This redundancy is related to the similarity of an image with itself, that is, part *A* of a certain image is similar to another part *B* of the image, by doing an arbitrary number of contractive transformations that can bring *A* and *B* together.

These contractive transformations are actually common geometrical operations such as rotation, scaling, skewing and shifting. By applying the resulting PIFS on an initially blank image iteratively, we can completely regenerate the original image at the decoder. Since the PIFS often consists of a small number of parameters, a huge compression ratio (e.g. 500 to 1000 times) can be achieved by representing the original image using these parameters. However fractal image compression has its disadvantages. Because fractal image compression usually involves a large amount of matching and geometric operations, it is time consuming.

The coding process is so asymmetrical that encoding of an image takes much longer time than decoding.

Reason to Use Wavelet Based Compression

As discussed earlier, for image compression, loss of some information is acceptable. Among all of the above lossy compression methods, vector quantization requires many computational resources for large vectors; fractal compression is time consuming for coding; predictive coding has inferior compression ratio and worse reconstructed image quality than those of transform based coding. So, transform based compression methods are generally best for image compression.

For transform based compression, JPEG compression schemes based on DCT (Discrete Cosine Transform) have some advantages such as simplicity, satisfactory performance, and availability of special purpose hardware for implementation. However, because the input image is blocked, correlation across the block boundaries cannot be eliminated.

3. BLOCK DIAGRAM

A block diagram of a wavelet based image compression system is shown in Figure 3.3. At the heart of the analysis (or compression) stage of the system is the forward discrete

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.2 , No.1, Pages : 424 – 429 (2013)
Special Issue of ICACSE 2013 - Held on 7-8 January, 2013 in Lords Institute of Engineering and Technology, Hyderabad

wavelet transform (DWT) [26]. Here, the input image is mapped from a spatial domain, to a scale-shift domain. This transform separates the image information into octave frequency sub bands. The expectation is that certain frequency bands will have zero or negligible energy content; thus, information in these bands can be thrown away or reduced so that the image is compressed without much loss of information.

The DWT coefficients are then quantized to achieve compression. Information lost during the quantization process cannot be recovered and this impacts the quality of the reconstructed image. Due to the nature of the transform, DWT coefficients exhibit spatial correlation, that are exploited by quantization algorithms like the embedded zero-tree wavelet (EZW) [2] and set partitioning in hierarchical trees (SPIHT) [28] for efficient quantization. The quantized coefficients may then be entropy coded; this is a reversible process that eliminates any redundancy at the output of the quantizer.

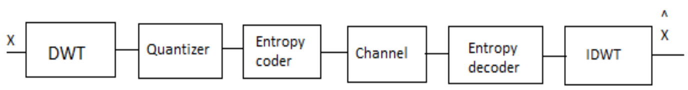

Figure 3.3: A Wavelet Based image compression system

In the synthesis (or decompression) stage, the inverse discrete wavelet transform recovers the original image from the DWT coefficients. In the absence of any quantization the reconstructed image will be identical to the input image. However, if any information was discarded during the quantization process, the reconstructed image will only be an approxi- mation of the original image. Hence this is called lossy compression. The more an image is compressed, the more information is discarded by the quantizer; the result is a reconstructed image that exhibits increasingly more artifacts. Certain integer wavelet transforms exist that result in DWT coefficients that can be quantized without any loss of information. These result in lossless compression, where the reconstructed image is an exact replica of the input image. However, compression ratios achieved by these transforms are small compared to lossy transforms (e.g. 4:1 compared to 40:1).

The remainder of this chapter explores the discrete wavelet transform [27], and the hardware implementation of the transform stage for computation of the 2D DWT for an image.

4. The Fast Wavelet Transform Algorithm

The Discrete Wavelet Transform (DWT) coefficients can be computed by using Mallat.s Fast Wavelet Transform algorithm. This algorithm is sometimes referred to as the two-channel sub-band coder and involves filtering the input signal based on the wavelet function used.

Implementation Using Filters:

To explain the implementation of the Fast Wavelet Transform algorithm consider the following equations:

$$\phi(t) = \sum_k c(k)\phi(2t-k)$$

$$\psi(t) = \sum_k (-1)^k c(1-k)\phi(2t-k)$$

$$\sum_k c_k c_{k-2m} = 2\delta_{0,m}$$

The first equation is known as the *twin-scale relation* (or the dilation equation) and defines the scaling function $\varnothing(t)$. The next equation expresses the wavelet $\psi(t)$ in terms of the scaling function $\varnothing(t)$. The third equation is the condition required for the wavelet to be orthogonal to the scaling function and its translates. The coefficients c(k) or {c0, .., c2N-1} in the above equations represent the impulse response coefficients for a low pass filter of length 2N, with a sum of 1 and a norm of 1/$\sqrt{2}$.

The high pass filter is obtained from the low pass filter using the relationship

$$g_k = (-1)^k c(1-k)$$

where k varies over the range (1 . (2N . 1)) to 1.

Equation 3.19 shows that the scaling function is essentially a low pass filter and is used to define the approximations. The wavelet function defined by equation 3.19 is a high pass filter and defines the details. Starting with a discrete input signal vector s, the first stage of the FWT algorithm decomposes the signal into two sets of coefficients. These are the approximation coefficients cA1 (low frequency information) and the detail coefficients c_D1 (high frequency information), as shown in the figure below.

The coefficient vectors are obtained by convolving s with the low-pass filter Lo_D for approximation and with the high-pass filter Hi_D for details. This filtering operation is then followed by dyadic decimation or down sampling by a factor of 2.

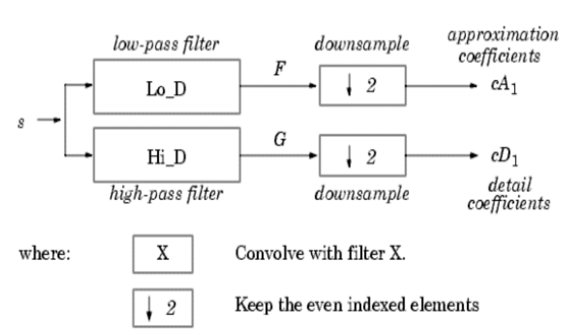

Figure 3.12: Filter operation during DWT

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.2 , No.1, Pages : 424 – 429 (2013)
Special Issue of ICACSE 2013 - Held on 7-8 January, 2013 in Lords Institute of Engineering and Technology, Hyderabad

Signal Reconstruction

The original signal can be reconstructed or synthesized using the inverse discrete wavelet transform (IDWT). The synthesis starts with the approximation and detail coefficients cAj and cDj, and then reconstructs cAj-1 by up sampling and filtering with the reconstruction filters.

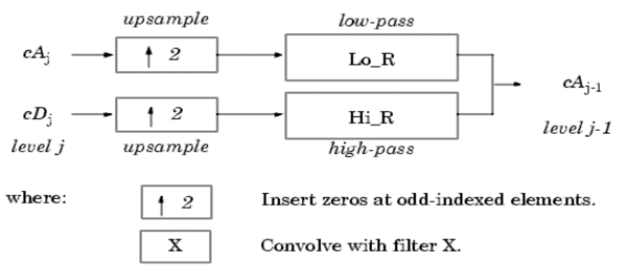

Figure 3.14: Wavelets reconstruction

The reconstruction filters are designed in such a way to cancel out the effects of aliasing introduced in the wavelet decomposition phase. The reconstruction filters (Lo_R and Hi_R) together with the low and high pass decomposition filters, forms a system known as quadrature mirror filters (QMF). For a multilevel analysis, the reconstruction process can itself be iterated producing successive approximations at finer resolutions and finally synthesising the original signal.

Most of the gain associated with sub band/wavelet coders comes from the compaction achieved by the frequency-sensitive analysis filter bank. As discussed in the earlier section, on top of having food frequency selectivity, important secondary gains can be achieved through phase linearity and transition band modifications to the magnitude response that lower high-frequency channel aliasing energy. In this section, we consider the formulation of new analysis filters that are short (comparable to the 5/3 bi-orthogonal filters) but with improved frequency-domain characteristics, consistent with earlier observation. Simple reoptimization of the filter coefficient does not result in demonstrably consistent improvement, owing primarily to the constrained nature of the exact reconstruction conditions for FIR filter banks

$$G_0(z) = -H_1(-z)$$
$$G_1(z) = H_0(-z)$$
$$G_0(z)H_0(z) + G_1(z)H_1(z) = 2z^{-1} \qquad (k\ odd)$$

Where $H_0(z)\ and\ H_1(z)$ are the low- and high-pass analysis filters, respectively, $G_0(z)$ and $G_1(z)$ are the low- and high-pass synthesis filters, respectively. However, employing the more general perfect reconstruction conditions

$$G_0(z)H_0(-z) + G_1(z)H_1(-z) = 0$$

$$G_0(z)H_0(z) + G_1(z)H_1(z) = 2$$

………… (4.2)

We can consider a boarder class of solutions. Given an arbitrary pair of analysis filters $H_0(z)$ and $H_1(z)$ the synthesis filters $G_0(z)$ and $G_1(z)$ as determined by the reconstruction conditions above, are

$$G_0(z) = -\frac{H_1(-z)}{N(z)}$$

$$G_1(z) = \frac{H_0(-z)}{N(z)}$$

Where,

$$N(z) = \frac{1}{2}[-H_0(z)H_1(-z) + H_1(z)H_0(-z)]$$

These reconstruction conditions have the attractive feature that they allow the analysis filters $H_0(z)$ and $H_1(z)$ to be chosen independently. The only restriction is that N (z) should be stable, which means no zeros on the unit circle. A consequence of the generality of (4.3) is that the synthesis filters could be recursive, but still quite interesting if these filters can be implemented with efficiency greater than or equal to their FIR counterpart.

Using (4.3), we can choose analysis filters in a virtually unconstrained fashion based on desired frequency-domain characteristics. The Sym2 analysis *low-pass* filter (no pass band deviation) and the 5/3 analysis *high-pass* filter (strong rejection of aliasing) were chosen for the analysis pair of the new filters,

$$\begin{cases} H_0(z) = -C_0(1+z^{-1})^2(1-xz^{-1}) \\ H_1(z) = C_1(1-z^{-1})^2 \end{cases}$$

Where H_0 (z) is the low-pass analysis filter of the Sym2, with x=1/(2-√3) , and Co and C1 are the normalization constants. Computing N (z) using (4.1), we obtain

$$N(z) = C_0C_1z^{-1}[(4\text{-}x)+(4\text{-}6x)z^{-2}xz^{-4}]$$

The corresponding synthesis filters are then derived directly from (4.3). To increase computational efficiency, we modify the value of x from 1/ (2-√3) ≈3.732 to 4.0. This conversion from floating point precision to integer results in a simple and stable denominator

$$N(z) = C_Nz^{-3}(1+0.2z^{-2})$$

Where C_N is the normalization constant. In direct form, the analysis filters of are given by

$$\begin{cases} H_0(z) = -C_0(1-2z^{-1}-7z^{-2}-4z^{-3}) \\ H_1(z) = C_1(1-2z^{-1})+z^{-2}) \end{cases}$$

Perhaps not surprisingly, the coefficient conversion to integer precision does not change the analysis filter frequency response dramatically, nor is the compression performance in our test results affected by this change. We call this new FIR–IIR hybrid filter set “Hybrid1.” Coefficients of the Hybrid1 filters are listed in Table 4.1.

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.2 , No.1, Pages : 424 – 429 (2013)

Special Issue of ICACSE 2013 - Held on 7-8 January, 2013 in Lords Institute of Engineering and Technology, Hyderabad

Only the 15 largest coefficients are shown for the IIR filters. Its compression performance is evaluated for many test images; results on seven specific test images are presented in Table 4.1, along with benchmark comparisons against the Sym2 and5/3 filters.

To construct linear-phase FIR–IIR hybrid filters, we start with the linear-phase 5/3 low-pass filter and optimize the reciprocal roots affecting the pass band so that the pass band magnitude response approximates that of the Sym2. As previously, the optimized coefficients are converted to integers in order to realize computational savings. The resulting filters, which we call Hybrid2, are given by

$$\begin{cases} H_0(z) = -C_0(1+z^{-1})^2(1-6z^{-1}+z^{-2}) \\ \quad = -C_0(1-4z^{-1}-10z^{-2}-4z^{-3}+z^{-4}) \\ H_1(z) = C_1(1-z^{-1})^2 \\ \quad = C_1(1-2z^{-1}+z^{-2}) \end{cases} \quad \text{.........}(4.9)$$

The low-pass filter frequency response is shown in Fig. 7.6. The close approximation to the Sym2 and Hybrid1 is evident, as is the notable improvement over the 5/3 low-pass filter. Using (4.4), we obtain

$$N(z) = C_N z^{-1}(1 + 14z^{-2} + z^{-4})$$

Then, the synthesis filters are derived from Filter coefficients of the Hybrid2 are shown in Table 4.1

The arithmetic complexity for the Hybrid1 filters, as conveyed by multiplies and adds, is quite modest. The FIR part can be implemented using two additions per data point (1-D transform) for decomposition and three additions for reconstruction. For the recursion in reconstruction, one extra addition and multiplication is needed. Thus, the combined complexity of both decomposition and reconstruction is six additions and one multiplication.

4. SIMULATION RESULTS

Filter	PSNR	Compression Ratio
Hybrid	42.4490	6.2654
Sym2	36.6588	5.7344
Biorth 5/3	36.7710	6.6075

Comparison of wavelet transforms methods

To compare the objective equality for different wavelet types, we use the standard testing image lena.

Table 7.1: Comparison of Compression Results by Using Different Wavelets for standard image Lena

Table 7.2 Shows Peak Signal to Noise Ratio(PSNR) and Compression Ratio (CR) of different type of bi-orthogonal wavelets i.e sym2,biorth 5/3, hybrid1 and hybrid2 ; for different images i.e. lena, Barbara, Light house and Mandrill. From above tables, we can observe that the bi-orthogonal hybrid filters giving better performance compared to all other bi-orthogonal filters.

5. CONCLUSION

Image compression is currently an active topic for research in the areas of Very Large Scale Integrated (VLSI) circuit technologies and Digital Signal Processing (DSP). The Discrete Wavelet Transform performs very well in the compression of images. For real time image processing however, its performance is not as good. Therefore for real time image compression it is recommended to use a wavelet with a small number of vanishing moments at level 5 decomposition or less. Using wavelets, the compression ratio can be easily varied, while most other compression techniques have fixed compression ratios.

Further data compaction is possible by exploiting the redundancy in the encoded transform coefficients. A bit encoding scheme could be used to represent the data more efficiently. A common loss-less coding technique is Entropy coding. Two common entropy coding schemes are Prefix coding and tree-structured Huffman coding. Both these forms of entropy coding require a prior knowledge of the nature of the source data, such as probability distribution of the source output data . In practice however, probabilistic models are usually not known a priori. Thus a model of the data must be constructed from the data set itself.

.

REFERENCES

1. **J**. Woods and S. O'Neil, "Subband coding of images," *IEEE Trans. Acoust. Speech, Signal Process.*, vol. 34, no. 5, pp. 1278–1288, Oct. 1986.
2. J. M. Shapiro, "Embedded image coding using zerotrees of wavelets coefficients," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3445–3462, Dec. 1993.
3. A. Said and W. A. Pearlman, "A new fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 3, pp. 243–250, May 1996.
4. D. S. Taubman and M. W. Marcellin, *JPEG2000: Image Compression Fundamentals*. Norwell, MA: Kluwer, 2002, standards, and practice.
5. M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Trans. Image Process.*, vol. 1, no. 2, pp. 205–220, Apr. 1992.
6. A. Z. Averbuch and V. A. Zheludev, "A new family of spline-based biorthogonal wavelet transforms and their

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.2 , No.1, Pages : 424 – 429 (2013)
Special Issue of ICACSE 2013 - Held on 7-8 January, 2013 in Lords Institute of Engineering and Technology, Hyderabad

application to image com- pression," *IEEE Trans. Image Process.*, vol. 13, no. 7, pp. 993–1007, Jul. 2004.

7. X. Zhang, W. Wang, T. Yoshikawa, and Y. Takei, "Design of IIR or- thogonal wavelet filter banks using lifting scheme," *IEEE Trans. Signal Process.*, vol. 54, no. 7, pp. 2616–2624, Jul. 2006.
8. C. W. Kim and R. Ansari, "FIR/IIR exact reconstruction filter banks with applications to subband coding of images," in *Proc. 34th Midwest Symp. Circuits Syst.*, May 1991, pp. 227–230.
9. F. Argenti and E. Del Re, "Design of biorthogonal M-channel cosine- modulated FIR/IIR filter banks," *IEEE Trans. Signal Process.*, vol. 48, no. 3, pp. 876–881, Mar. 2000.
10. G. Strang and T. Q. Nguyen, *Wavelets and Filter Banks.* Wellesley, MA: Wellesley-Cambridge, 1996.
11. J.Villasenor, B. Belzer, and J. Liao, "Wavelet filter evaluation for image compression," *IEEE Trans. Image Process.*, vol. 4, no. 8, pp. 1053–1060, Aug. 1995.
12. J. Lin and M. T. J. Smith, "New perspectives and improvements on the symmetric extension filter bank for subband/wavelet image compres- sion," *IEEE Trans. Image Process.*, vol. 17, no. 2, pp. 177–189, Feb. 2008.
13. M. J. T. Smith and T. Barnwell, "Exact reconstruction techniques for tree-structured subband coders," *IEEE Trans. Acoust. Speech, Signal Process.*, vol. ASSP-34, no. 3, pp. 434–441, Jun. 1986.
14. M. J. T. Smith and S. L. Eddins, "Analysis/synthesis techniques for subband image coding," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 38, no. 8, pp. 1446–1456, Aug. 1990.
15. S. Aase and T. Ramstad, "On the optimality of nonunitary filter banks in subband coders," *IEEE Trans. Image Process.*, vol. 4, no. 12, pp. 1585–1591, Dec. 1995.
16. T. P. Barnwell, "Subband coder design incorporating recursive quadra- ture filters and optimum ADPCM coders," *IEEE Trans. Acoust. Speech, Signal Process.*, vol. ASSP-30, no. 5, pp. 751–765, Oct. 1982.
17. S. Rout and A. E. Bell, "Narrowing the performance gap between orthogonal and biorthogonal wavelets," in *Conf. Rec. 38th Asilomar Conf. Signals, Syst. Comput. 2004*, vol. 2, pp. 1757–1761.
18. B. E. Usevitch, "A tutorial on modern lossy wavelet image compres- sion: Foundations of JPEG 2000," *IEEE Signal Process. Mag.*, vol. 18, no. 5, pp. 22–35, Sep. 2001.
19. T. D. Tran and T. Q. Nguyen, "A progressive transmission image coder using linear phase uniform filter banks as block transforms," *IEEE Trans. Image Process.*, vol. 8, no. 11, pp. 1493–1507, Nov. 1999.
20. John D.Villasenor, Benjamin Belzer, and Judy Liao. Wavelet filter evaluation for image compression. IEEE Trans. on Image Processing, 4(8):1053–1060, August 1995.
21. Timo Eirola. Sobolev characterization of solutions of dilation equations. SIAM J. Math. Anal, 23:1015–1030, 1992
22. Michael Lightstone, Eric Majani, and Sanjit K. Mitra. Low bit-rate design considera- tions for wavelet-based image coding. Multidimensional systems and signal processing, 8(1-2):111–128, January 1997
23. J. E. Odegard and C. S. Burrus. Discrete finite variation: a new measure of smoothness for the design of wavelet basis. Proceedings of the International Conference on Acoustics, Speech and Signal Processing, 3:1467–70, 1996.
24. Gilbert Strang and Truong Nguyen. Wavelets and Filter Banks. Wellesley-Cambridge Press, Wellesley MA, first edition, 1996.
25. Michael Unser and Thierry Blu. Mathematical properties of the jpeg2000 wavelet filters.IEEE Trans. on Image Processing, to appear in 2003
26. M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," IEEE Trans. Image Processing, vol. 1, no. 2, pp. 205–220, April 1992.
27. S. Rout, "Orthogonal vs. biorthogonal wavelets for image compression," Master's thesis, Virginia Polytechnic Institute and State University, 2003.
28. A. Said and W. A. Pearlman. A new, fast, and e_cient image codec based on set partitioning in hierarchical trees. IEEE Trans. on Circ. and Syst. for Video Tech., 6(3):243{250, June 1996.
29. J. M. Shapiro. Embedded image coding using zerotrees of wavelet coe_cients. IEEE Trans. on Image Proc., 41(12):3445{3462, December 1993.
30. Olivier Rioul. Regular wavelets: a discrete-time approach. IEEE Trans. on Signal Processing, 41(12):3572 { 3579, Dec. 1993.