Spectrum Flipping for Wavelet Packet Decomposition

Jianyu Lin
The Brain & Mind Research Institute
The University of Sydney
NSW 2006, Australia

Abstract – In this paper, it is shown that the higher compression performance of biorthogonal wavelet filters over orthogonal wavelet filters is the combined effect from (1) the asymmetry of the frequency responses of the biorthogonal wavelet filters and (2) the non-flat spectra of the signals to be decomposed. Based upon this observation, an improved wavelet packet decomposition method for natural images through swapping the analysis and synthesis filters is proposed. The method leads to consistent improved compression performance at a very wide range of data rates without increasing computational complexity.

Key words – Compression performance of wavelets, Frequency responses of biorthogonal wavelet filters, Wavelet packet decomposition

1 Introduction

Wavelet transform is widely used in image compression [1] [2]. Only biorthogonal wavelet filters have been used in this area, primarily due to the following two reasons. First, linear-phase biorthogonal FIR wavelet filters can be easily constructed while it is impossible to construct linear-phase orthogonal FIR wavelet filters. Second, biorthogonal wavelet filters have high compression performance for natural images [2][3][4]. However, factors that affect the performance of wavelet transform for image compression are not completely understood. In this paper, we show that the frequency response of wavelet filters is an important factor that affects the compression performance and explain why biorthogonal wavelet filters have better compression performance than orthogonal wavelet filters for natural images. Our explanation leads to an improved method for wavelet packet decomposition which results in higher compression performance without any increase of computational complexity.

2 Comparison of Biorthogonal Filters and Orthogonal Filters

The support and the number of vanishing moments are two critical properties that are known to affect the compression performance of wavelets [4][8][9], but not much attention has been paid to the frequency response of wavelet filters. Here, we compare the frequency response of two closely related wavelets, the least asymmetric Daubechies orthogonal wavelet with four vanishing moments (called the "sym4" wavelet) and the very popular linear-phase 9/7-filter-tap Daubechies biorthogonal wavelet [3] (called the "9/7" wavelet), which is used in JPEG2000 [2]. These two wavelets are identical for the two wavelet properties mentioned above but have different frequency responses. Both the analysis and synthesis wavelets of the biorthogonal 9/7 have the support of 7 [4][8][9], which is exactly the same as the sym4, and they also have 4 vanishing moments [9], just like the sym4. In fact, from the construction of the 9/7 biorthogonal wavelets and the sym4 wavelets, it can be seen that they are intrinsically related [4].

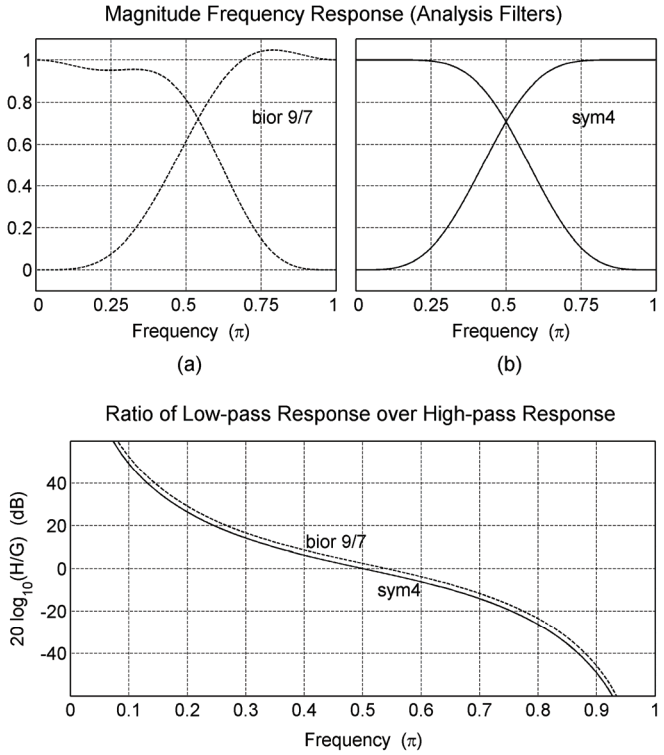

Fig 1. Magnitude frequency response of **(a)** the 9/7 wavelet filters and **(b)** the sym4 wavelet filters. **(c)** Plot of $20\log_{10}(H/G)$ for the two pairs of filters.

Figure 1 shows the magnitude frequency response

of the biorthogonal 9/7 analysis filters and the orthogonal sym4 filters. Clearly, the response curves of the biorthogonal 9/7 filters are not symmetric. The *cross-over point* for the lowpass and highpass response curves, which is the point where the curves have the same magnitude response, appears at a frequency higher than 0.5π (half the whole frequency band). On the other hand, the cross-over point for the sym4 is exactly at 0.5π.

To clearly show the effects of the frequency response, we define:

$$R(\omega) = \left| H(\omega) / G(\omega) \right| \tag{1}$$

and

$$S = \frac{\text{total energy of the low frequency band after filtering}}{\text{total energy of the high frequency band after filtering}} \tag{2}$$

where $H(\omega)$ and $G(\omega)$ are the lowpass and the highpass frequency responses respectively. Apparently, at $\omega = 0.5\pi$, $R_{9/7}(0.5\pi) > R_{\text{sym4}}(0.5\pi)$, due to the difference in the cross-over points (see Figure 1a and 1b). In fact, as shown in Figure 1c, not only at $\omega = 0.5\pi$ but for all ω, $R_{9/7}(\omega) > R_{\text{sym4}}(\omega)$.

Parameter S defined by (2) measures the compression performance of a transform from the point of view of the coding gain. $S = 1$ after a wavelet transformation means that energy is equally distributed into low and high frequency bands or there is no energy compaction from the transform, leading to no compression of the transformed signal. $S \neq 1$ means that the transform packs more energy into the low or high frequency band, or equivalently the transformed signal gets compressed. Furthermore, when $S > 1$, the larger the S, the more compressed the signal, because of higher energy compaction. Similarly, when $S < 1$, the smaller the S, the more compressed the signal.

The energy of a natural image is not uniformly distributed in the frequency domain but concentrated in the low frequency region [10]. Given $R_{9/7}(\omega) > R_{\text{sym4}}(\omega)$, if a low-frequency rich natural image is separately filtered by the 9/7 and the sym4, then the filter outputs would be such that $S_{9/7} > S_{\text{sym4}} > 1$, where S is defined by (2). This indicates that energy gets even more packed into the low-frequency band for the 9/7 than for the sym4, suggesting higher compression for the 9/7. On the other hand, suppose the signal to be decomposed has more high-frequency energy than low-frequency energy, then $S_{\text{sym4}} < S_{9/7} < 1$ after wavelet transformation, which means that the 9/7 packs less energy into the high-frequency band than the sym4, and thus leading to lower compression performance for the 9/7 than for the sym4 in this case.

Without exception, high performance biorthogonal wavelet filters, including the popular 5/3 LeGall [9], 10/18 [5], and 28/28 [6] etc., all have asymmetric frequency response curves just like the 9/7 wavelet filters.

One might not agree with our above explanation or argument, since an image not only has the low-frequency rich property, but also has complicated correlations among the pixels. Here we design an experiment in which signals are originated from random noise. Therefore, these signals can be viewed as only having the non-flat spectrum redundancy. In this way, we isolate the factor – non-flat spectrum redundancy (either low-frequency rich or high-frequency rich), and see whether our argument is true. If it is tested true, the argument should also apply to any signal that has this non-flat spectral property regardless whether the signal has other properties (which might, if any, counteract or support the effect from this non-flat spectral property), for example, whatever correlation models for the time domain pixels of an image.

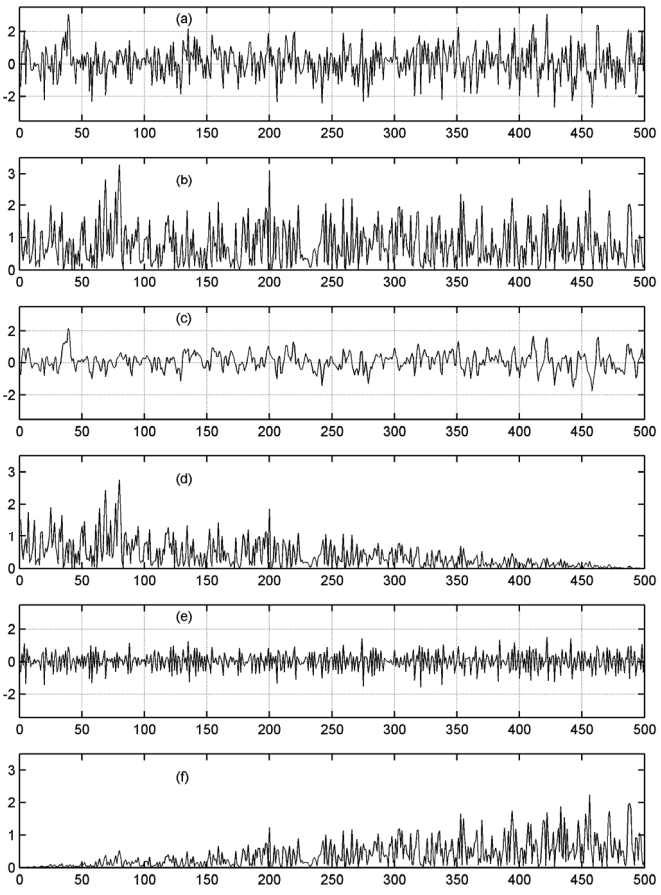

Fig 2. Three test signals and their respective magnitude spectrums: **(a)** Discrete Gaussian white noise and **(b)** its magnitude spectrum. **(c)** Discrete low-frequency-rich pink noise and **(d)** its magnitude spectrum. **(e)** Discrete high-frequency-rich pink noise and **(f)** its magnitude spectrum.

The first signal (Figure 2a) is the Gaussian white noise. Since the DFT of the Gaussian white noise is also

a Gaussian white noise, its magnitude spectrum (Figure 2b) essentially has a flat average. The second signal (Figure 2c) is a low-frequency-rich pink noise generated by linearly decreasing the spectrum magnitude of the Gaussian white noise as evident from its magnitude spectrum in Figure 2d. The third signal (Figure 2e) is a high-frequency-rich pink noise, which is obtained by linearly decreasing, from high to low frequency, the magnitude spectrum of the Gaussian white noise. The magnitude spectrum the third signal is shown in Figure 2f.

The coding gain is computed using the generalized coding gain formula [7]:

$$C_{\text{coding gain}} = 10\log_{10}\frac{\sigma_x^2}{\left(\prod_{i=0}^{M-1}\sigma_{x_i}^2\|f_i\|^2\right)^{1/M}} \quad (3)$$

where σ_x^2 is the variance of the input signal, $\sigma_{x_i}^2$ is the variance of the ith subband, $\|f_i\|^2$ is the norm of the ith synthesis filter, M is the total number of decomposed subbands. In our case, the one-dimensional signals are decomposed into high-frequency and low-frequency bands, i.e. $M = 2$.

Table 1 shows our test results. As expected, for the low-frequency-rich pink noise (Figure 2c), the 9/7 filters results in higher coding gain than the sym4 and therefore better compression performance. And for high-frequency-rich pink noise (Figure 2e), the sym4 has higher coding gain than the 9/7 and therefore better compression performance.

Signal	Filters (Wavelets)			
	9/7	Sym4	5/3	db2
Gaussian white noise	-0.10	0	-0.33	0
Low-frequency-rich pink noise	1.84	1.54	1.95	1.32
High-frequency-rich pink noise	1.03	1.54	0.15	1.32

Table 1. Coding gains (in dB) for the three artificial test signals

Table 1 also shows coding gains for another pair of intrinsically related wavelets: the 5/3 LeGall biorthogonal wavelet (called the "5/3" wavelet, which is also used in JPEG2000), and the Daubechies orthogonal wavelet with two vanishing moments [4] (called the "db2" wavelet). Both the 5/3 biorthogonal wavelets and the db2 wavelets have the same support of 3 [9], and the same number of vanishing moments of 2 [9]. Only the filter frequency responses of the two wavelets are different – in the same way as the pair biorthogonal 9/7 and sym4 (figure 1).

We now analyze Table 1 further. The coding gain for sym4 and db2 is zero for the Gaussian white noise (Figure 2a) because Gaussian white noise remains as Gaussian white noise after the orthogonal transform. Intuitively, since there is no correlation among the samples of a random i.i.d. Gaussian source, the discrete Gaussian white noise cannot be compressed. The two orthogonal filters achieve higher coding gain for the Gaussian white noise than the biorthogonal filters. This is also understandable as Gaussian white noise is transformed to pink noise after the biorthogonal transformation. Since there is correlation among the samples of a pink noise, the biorthogonal 9/7 and 5/3 filters actually "expand" (rather than compress) the uncompressible Gaussian white noise into the compressible pink noise, which leads to negative gains. For the low-frequency-rich pink noise, the biorthogonal 9/7 and 5/3 filters have higher coding gains than the orthogonal sym4 and db2 respectively as expected. Also as expected, for the high-frequency-rich pink noise, the biorthogonal 9/7 and 5/3 filters have lower coding gain than sym4 and db2 respectively. Finally, the sym4 and the db2 respectively produce an identical coding gain for the high-frequency-rich and the low-frequency-rich pink noises because the high-frequency-rich pink noise is simply a spectrally "flipped" version of the low-frequency-rich pink noise.

3 Spectrum Flipping for Biorthogonal Wavelet Packet Decomposition of Images

In this section, biorthogonal wavelet packet decomposition is modified according to the argument made the previous section to achieve higher coding gain. In wavelet packet decomposition, a decomposed frequency band (no matter high-frequency or low-frequency band) is further decomposed whenever it is necessary. Such an optimal wavelet packet decomposition structure can always be found for a given signal or image. Of course, uniformly decomposed subbands using wavelet filters is just a simple version of wavelet packet decomposition.

As is known, the decimation procedure during the wavelet packet decomposition results in the "flipping" of the high-frequency band spectrum, i.e., high-frequency contents become low-frequency contents. For a natural image, such a decimated high-frequency band will contain more high-frequency energy because of this spectrum flipping. This high-frequency band (with more high-frequency energy) is further decomposed by, say, the 9/7 filters during the wavelet packet decomposition. As discussed in section II, the compression performance of the 9/7 filters for signals with more high-frequency energy is even poorer than the sym4. Apparently, the flipped spectrum needs to be flipped back in order to achieve high performance for the 9/7 filters. This is easily done in the time domain by multiplying the decomposed coefficients with $(-1)^n$. To see this clearly,

Bit Rate	9/7 filters			5/3 filters			10/18 filters		
	Flip	No flip	Gain	Flip	No flip	Gain	Flip	No flip	Gain
Barbara (512×512)									
0.125	**25.838**	25.835	0.003	**24.791**	24.612	0.179	**26.064**	25.923	0.141
0.25	**28.680**	28.638	0.042	**27.299**	26.942	0.357	**29.099**	28.923	0.176
0.5	**32.484**	32.380	0.104	**30.699**	30.238	0.461	**32.957**	32.706	0.251
1	**37.323**	37.106	0.216	**35.378**	34.661	0.717	**37.798**	37.455	0.343
2	**43.260**	43.017	0.243	**41.433**	40.792	0.641	**43.583**	43.146	0.437
Goldhill (512×512)									
0.125	**28.561**	28.554	0.007	**28.321**	28.223	0.098	**28.667**	28.586	0.081
0.25	**30.718**	30.665	0.053	**30.222**	30.142	0.080	**30.797**	30.688	0.109
0.5	**33.217**	33.134	0.083	**32.626**	32.473	0.153	**33.337**	33.167	0.170
1	**36.547**	36.450	0.097	**35.743**	35.528	0.215	**36.691**	36.426	0.266
2	**41.943**	41.807	0.136	**40.697**	40.265	0.432	**42.019**	41.658	0.362
Lena (512×512)									
0.125	**31.133**	31.081	0.052	**30.509**	30.325	0.184	**31.242**	31.098	0.144
0.25	**34.119**	34.058	0.060	**33.237**	33.007	0.230	**34.310**	34.154	0.156
0.5	**37.172**	37.091	0.080	**36.124**	35.818	0.306	**37.341**	37.189	0.152
1	**40.274**	40.224	0.050	**39.112**	38.807	0.305	**40.358**	40.198	0.160
2	**44.835**	44.830	0.005	**43.188**	43.062	0.126	**44.723**	44.547	0.176
Café (2560×2048)									
0.125	**20.755**	20.724	0.031	**20.255**	20.209	0.046	**20.732**	20.664	0.069
0.25	**23.012**	22.942	0.070	**22.417**	22.273	0.144	**23.081**	22.955	0.126
0.5	**26.333**	26.232	0.101	**25.495**	25.239	0.256	**26.452**	26.240	0.212
1	**31.192**	30.994	0.197	**29.936**	29.514	0.422	**31.346**	31.003	0.343
2	**38.084**	37.867	0.217	**36.431**	35.868	0.563	**38.033**	37.583	0.449
Bike (2560×2048)									
0.125	**26.010**	25.952	0.059	**25.269**	25.104	0.165	**26.137**	26.016	0.121
0.25	**29.054**	28.948	0.106	**28.080**	27.852	0.227	**29.202**	29.040	0.162
0.5	**32.701**	32.570	0.131	**31.486**	31.195	0.291	**32.856**	32.641	0.215
1	**37.192**	37.017	0.175	**35.792**	35.398	0.394	**37.320**	37.040	0.280
2	**43.226**	43.020	0.206	**41.728**	41.269	0.458	**43.168**	42.789	0.379
Woman (2560×2048)									
0.125	**27.390**	27.353	0.037	**26.699**	26.553	0.145	**27.479**	27.360	0.120
0.25	**29.996**	29.868	0.128	**29.002**	28.774	0.228	**30.052**	29.879	0.174
0.5	**33.597**	33.404	0.193	**32.424**	32.025	0.399	**33.678**	33.419	0.258
1	**38.209**	38.007	0.202	**37.214**	36.641	0.573	**38.254**	37.956	0.298
2	**43.978**	43.821	0.157	**42.391**	41.785	0.606	**43.880**	43.526	0.354
Finger Print (448×448)									
0.125	**19.878**	19.740	0.138	**18.785**	18.452	0.333	**20.176**	19.995	0.180
0.25	**23.006**	22.846	0.160	**21.902**	21.503	0.398	**23.405**	23.181	0.224
0.5	**26.771**	26.498	0.273	**25.168**	24.760	0.408	**27.251**	26.984	0.266
1	**30.898**	30.603	0.295	**29.103**	28.583	0.519	**31.367**	31.073	0.294
2	**36.764**	36.300	0.464	**34.732**	34.018	0.713	**37.082**	36.546	0.536

Table 2. Coding results for comparison. Bit rate (bpp)/PSNR(dB)

suppose an arbitrary frequency component $\cos(\omega n+\varphi_0)$ is multiplied by $(-1)^n$:

$$\begin{aligned}(-1)^n\cos(\omega n+\varphi_0) &= \cos(\pi n)\cos(\omega n+\varphi_0)\\ &= \frac{1}{2}\left[\cos(\omega n+\varphi_0+\pi n)+\cos(\omega n+\varphi_0-\pi n)\right]\\ &= \cos\left[(\omega-\pi)n+\varphi_0\right]\\ &= \cos\left[(\pi-\omega)n-\varphi_0\right]\end{aligned} \tag{4}$$

Obviously, $\cos\left[(\pi-\omega)n-\varphi_0\right]$ is the spectrum flipped version of $\cos(\omega n+\varphi_0)$ in the frequency band [0, π] of discrete signals. Care needs to be taken to flip the two-dimensional wavelet coefficients. The high-high (HH) frequency band coefficients $C_{\mathrm{HH}}(n_x, n_y)$ needs to be flipped in both vertical and horizontal axes, which is accomplished by multiplying with $(-1)^{n_x+n_y}$. The low-high (LH) frequency band coefficients $C_{\mathrm{LH}}(n_x, n_y)$ needs to be flipped only in the *y* direction by multiplying with $(-1)^{n_y}$, and similarly the high-low (HL) frequency band needs to be multiplied with $(-1)^{n_x}$.

Table 2 shows the test coding results using SPIHT [1] for six popular test images: Barbara, Goldhill, Lena, Café, Bike, Woman, and a finger print test image. The decomposition structure used was a 3-level uniform wavelet packet decomposition plus further 3-level dyadic wavelet decomposition on the DC band (the lowest frequency band), called the 3+3 structure in [6]. To implement the SPIHT coder with the 3+3 decomposition structure, spatial orientation trees are formed using the method described in [6].

Three most popular filters were chosen for the coding test: the 9/7 filters [3] [9] [2], the 5/3 filters [9] [2] and the 10/18 filters [5]. Table 2 presents the coding results from wavelet packet decomposition using spectrum flipping and the results from the conventional wavelet packet decomposition (i.e., without spectrum flipping). As shown, for all filters, all test images and all tested bit rates, there is a positive gain when spectrum flipping is used. The maximum gain in this test is 0.72 dB, which is not negligible. Also, the gain tends to be larger for higher bit rates. This is because at extremely low bit rates, most coefficients in the high-frequency bands, where spectrum flipping is necessary, are thresholded to be zeros after quantization.

Flipping of the spectrum is equivalent to swapping the analysis and synthesis filters, which is simpler to implement than the change of signs described by equation (4). Actually, swapping the analysis and synthesis filters for further decomposing high-frequency bands requires no extra computations.

Finally, we would like to point out that although the gain from spectrum flipping is generally small as shown in table 2, but it is extremely robust. We have tested more than 50 natural images, there is not a signal image that failed to achieve a gain at any rate. And remember, it is a free gain (i.e. with no extra computations). This means that whenever wavelet packet decomposition is performed, spectrum flipping is always necessary. It is this point that makes the main contribution of the paper. And this discovery is incorporated in another work on wavelet packet decomposition [11].

4 Conclusion

In this paper, it is shown that the asymmetric frequency response of biorthogonal filters leads to higher compression performance over orthogonal wavelet filters for low-frequency-rich signals. Based upon this observation, we introduced an improved method for wavelet packet decomposition that involves swapping the analysis and synthesis filters for further decomposition of the high-frequency bands, which is equivalent to spectrum flipping for the high-frequency bands. Our experimental results using SPIHT show quantitative improvements over the conventional wavelet packet decomposition.

References:

[1] 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, pp.243-250, May 1996

[2] David S. Taubman, Michael W. Marcellin, JPEG2000 : image compression fundamentals, standards, and practice, Kluwer Academic Publishers, 2002

[3] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Trans. Image processing,* vol. 1, pp.205-220, Apr. 1992

[4] I. Daubechies, *Ten Lectures on Wavelets,* Capital City Press, Montpelier, Vermont, 1992

[5] M. J. Tsai, J. D. Villasenor, and F. Chen, "Stack-run image coding," *IEEE Trans. Circuits Syst. Video Technol.,* vol. 6, no. 5, pp.519-512, Oct. 1996

[6] Z. Xiong and X. Wu, "Wavelet image coding using trellis coded space-frequency quantization," *IEEE Signal Processing Let.,* vol. 6, no. 7, pp.158-161, July 1999

[7] T. D. Tran and T. Q. Nguyen, "A progressive transmission image coder using linear phase uniform filterbanks as block transforms," *IEEE Trans. Image Processing,* vol. 8, no. 11, pp.1493-1507, Nov. 1999

[8] Stephane Mallat, *A Wavelet Tour of Signal Processing,* Academic Press, 1999

[9] M. Unser and T. Blu, "Mathematical properties of the JPEG2000 wavelet filters," *IEEE Trans. Image Processing,* vol. 12, no. 9, pp.1080-1090, Sept. 2003

[10] W. H. Yap, A. A. Kassim and W.S. Lee, "Residual Image Coding using A SPIHT-Based Video Coder". In Proceedings of International Conference on Communication Systems 2004(ICCS'2004,) pp. 225-229

[11] J. Lin and M. T. J. Smith, "Cyclic filter bank implementation of symmetric extension for subband/wavelet image compression," submitted to ICIP 2007