

On the Higher Subbands of the DGHM Multiwavelet Transform

溝畑 潔

同志社大学理工学数理システム学科

kmizoha@mail.doshisha.ac.jp

1 Introduction

画像処理や信号処理においてはコンパクトサポートを持ち対称なウェーブレットが最も望ましいとされているが、スケーリング関数が 1 個の場合、このようなウェーブレットは Haar のウェーブレットしかない事が理論的に証明されている [1]. 1996 年, Donovan, Geronimo, Hardin 及び Massopust はスケーリング関数が複数の場合においてコンパクトサポートを持つ対称なマルチウェーブレットを構成した [2]. これを DGHM マルチウェーブレットと言う.

離散データに対するマルチウェーブレット変換において問題になるのがデータの前処理についてである. 画像データなどの離散データに対して Mallat 変換を行う時にスケーリング関数の係数を決めるのを前処理と呼ぶが, いかなる方法が良いのかまだ未解決である.

もう一つ重要なのが高周波成分の解析である. DGHM マルチウェーブレットはウェーブレット関数が 2 個あるので, 一回 Mallat 変換すると 2 個の高周波成分が現れる. 2 次元の場合は 12 個の高周波成分が現れる. この高周波成分の特徴については何も知られていない.

この論文では DGHM マルチウェーブレットによる画像処理における前処理と高周波成分について解析を行い, 新しい圧縮法の提案を行う.

2 DGHM マルチウェーブレット

DGHM マルチウェーブレットは 2 個のスケーリング関数 $\phi_1(x), \phi_2(x)$ とウェーブレット関数 $\psi_1(x), \psi_2(x)$ から構成される. それぞれ次の dilation equation と wavelet equation を満たす.

$$\Phi(x) = \begin{bmatrix} \phi_1(x) \\ \phi_2(x) \end{bmatrix} = \sum_{k=0}^3 H_k 2^{1/2} \Phi(2x - k),$$

但し

$$H_0 = \frac{1}{10\sqrt{2}} \begin{bmatrix} 6 & 8\sqrt{2} \\ 1/\sqrt{2} & -3 \end{bmatrix}, \quad H_1 = \frac{1}{10\sqrt{2}} \begin{bmatrix} 6 & 0 \\ 9/\sqrt{2} & 10 \end{bmatrix}, \\ H_2 = \frac{1}{10\sqrt{2}} \begin{bmatrix} 0 & 0 \\ 9/\sqrt{2} & -3 \end{bmatrix}, \quad H_3 = \frac{1}{10\sqrt{2}} \begin{bmatrix} 0 & 0 \\ -1/\sqrt{2} & 0 \end{bmatrix};$$

$$\Psi(x) = \begin{bmatrix} \psi_1(x) \\ \psi_2(x) \end{bmatrix} = \sum_{k=0}^3 G_k 2^{1/2} \Phi(2x - k),$$

但し

$$G_0 = \frac{1}{10\sqrt{2}} \begin{bmatrix} -1/\sqrt{2} & -3 \\ -1 & 3\sqrt{2} \end{bmatrix}, \quad G_1 = \frac{1}{10\sqrt{2}} \begin{bmatrix} 9/\sqrt{2} & -10 \\ -9 & 0 \end{bmatrix},$$

$$G_2 = \frac{1}{10\sqrt{2}} \begin{bmatrix} 9/\sqrt{2} & -3 \\ 9 & -3\sqrt{2} \end{bmatrix}, \quad G_3 = \frac{1}{10\sqrt{2}} \begin{bmatrix} -1/\sqrt{2} & 0 \\ -1 & 0 \end{bmatrix}.$$

これをグラフで表すと次のようになる。

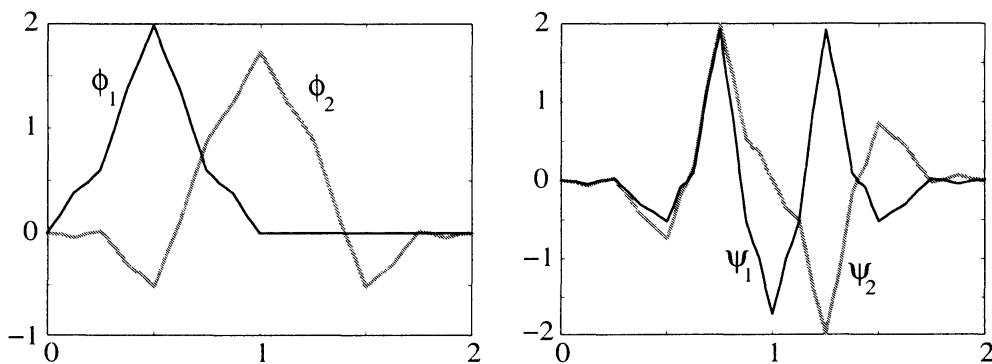


Fig 1: スケーリング関数 $\phi_1(x), \phi_2(x)$ とウェーブレット関数 $\psi_1(x), \psi_2(x)$.

全てコンパクトサポートかつ対称性を持つが、スケーリング関数のサポート長が異なり、しかもウェーブレット関数は $\psi_1(x)$ が対称であるのに対して $\psi_2(x)$ は反対称である。

3 DGHM マルチウェーブレット変換における前処理

Mallat 変換について簡単に紹介しておこう。 $\phi(x)$ と $\psi(x)$ はそれぞれスケーリング関数とウェーブレット関数とする。離散データ $\{x[k]\}_{k \in \mathbb{Z}}$ に対して V_0 の関数

$$f(x) = \sum_{k \in \mathbb{Z}} x[k] \phi(x - k)$$

とおく。Mallat 変換は

$$f(x) = \sum_{k \in \mathbb{Z}} p[k] \phi\left(\frac{x}{2} - k\right) + q[k] \psi\left(\frac{x}{2} - k\right)$$

と分解する事により、離散データ $\{x[k]\}_{k \in \mathbb{Z}}$ を低周波成分 $p[k]$ と高周波成分 $q[k]$ に分解する。マルチウェーブレットで問題になるのは離散データ $\{x[k]\}_{k \in \mathbb{Z}}$ に対して V_0 の関数

$$f(x) = \sum_{k \in \mathbb{Z}} a_1[k] \phi_1(x - k) + a_2[k] \phi_2(x - k)$$

の係数 $a_1[k]$ と $a_2[k]$ をどのように決定するかである。この処理を前処理と言う。最も単純なのは Mallat 変換を単純に拡張して $a_1[k] = a_2[k] = x[k]$ とする方法があるが、データ量が 2 倍になるため画像処理等では困難が生じる。データ量の変わない方法を 2 つ紹介する。

前処理 I: Strela 達 [4] は係数 $a_1[k]$ と $a_2[k]$ を

$$\begin{aligned} a_1[k] &= \frac{\phi_2(1)x[2k+1] - \phi_2(\frac{1}{2})(x[2k] + x[2k+2])}{\phi_2(1)\phi_1(\frac{1}{2})}, \\ a_2[k] &= \frac{x[2k+2]}{\phi_2(1)}, \end{aligned}$$

と定めた。これは V_0 の関数 $f(x)$ が

$$f(n) = x[2n], \quad f(n + 1/2) = x[2n + 1],$$

を満たすように係数を決める方法である。

前処理 II: これは溝畑 [3] が提案した方法で

$$a_1[k] = \frac{\sqrt{2}}{\sqrt{3}}x[2k+1], \quad a_2[k] = \frac{1}{\sqrt{3}}x[2k],$$

とおく方法である。この場合、 $f(x)$ は $f(n) = x[2n]$ を満たすが $f(n + 1/2) = x[2n + 1]$ は満たさない。しかし画像圧縮においてはこちらの方がいい結果を出している ([3])。

この前処理によって得られる V_0 の関数を比較しよう。関数 $f(x)$ に対して離散データ $x[n]$ を

$$x[2n] = f(n), \quad x[2n + 1] = f(n + 1/2),$$

で定める。前処理 I によって定まる係数を $\alpha_1[k]$ と $\beta_1[k]$ とおき、前処理 II によって定まる係数を $\alpha_2[k]$ と $\beta_2[k]$ 。これより定まる V_0 の関数

$$\begin{cases} f_1(x) = \sum \alpha_1[k]\phi_1(x-k) + \beta_1[k]\phi_2(x-k), \\ f_2(x) = \sum \alpha_2[k]\phi_1(x-k) + \beta_2[k]\phi_2(x-k). \end{cases}$$

とおく。一般に離散データはある関数のサンプリングとして得られる場合が多いので、元の関数 $f(x)$ と V_0 の関数 $f_1(x)$ と $f_2(x)$ の差を解析しよう。

$f(x)$ が一次関数の場合 $f(x) \equiv f_1(x) \equiv f_2(x)$ が成立する。2 次以上の多項式の場合、 $\int_0^5 |f(x) - f_1(x)|^2 dx$ と $\int_0^5 |f(x) - f_2(x)|^2 dx$ を求めると次のようになる。

$f(x)$	前処理 I	前処理 II
x^2	0.019724	0.014310
x^3	0.938513	0.680829
x^4	35.471229	25.711597

表より前処理 II の方がいい結果を出している。不連続関数の場合をグラフに描くと次のようになる。

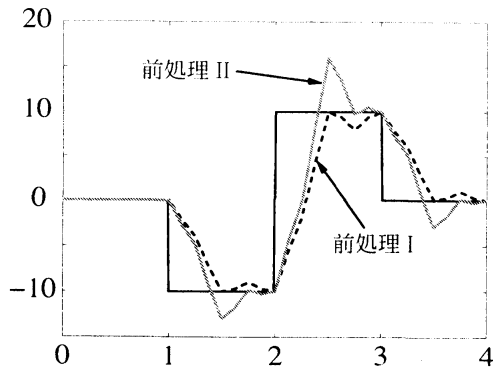


Fig 2: 不連続関数の場合.

グラフでわかるように前処理 I の場合は $f(n) = f_1(n)$, $f(n+1/2) = f_1(n+1/2)$ を満たすのに対して前処理 II の場合は $f(n+1/2) = f_2(n+1/2)$ を満たさない. しかし誤差の二乗の積分値は前処理 I が 38.04 なのに対して前処理 II が 33.36 と前処理 II の方がよい結果が出た. 他の不連続関数についても同様の結果が得られた.

4 2次元マルチウェーブレット変換による高周波成分

2次元マルチウェーブレットをテンソル積で構成し, 画像データを処理するには, まず画像データを前処理で4個の部分に分割する. これをマルチウェーブレット変換すると4個の低周波成分と12個の高周波成分が現れる.

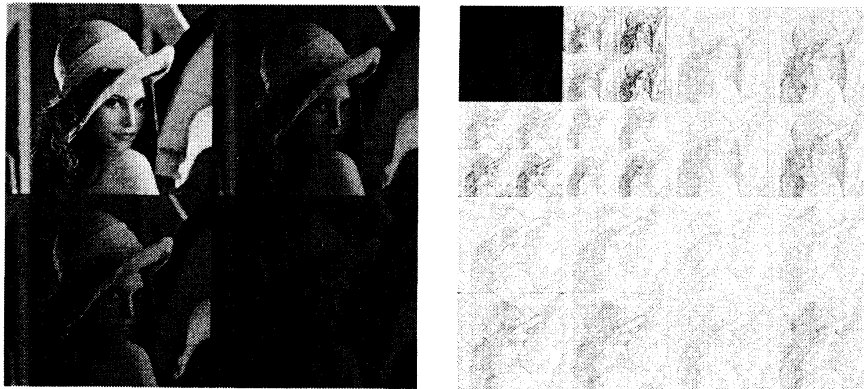


Fig 3: 前処理後の画像とマルチウェーブレット変換された画像 (two levels). 右画像の左上の4個が低周波成分.

画像をマルチウェーブレット変換すると12個の高周波成分が現れ, これらが画像のどのような特徴をあらわしているかは全く知られていない. そこで4個の低周波成分領域と12個の高周波成分領域に次の図のように名前を付け, 縦, 横方向のエッジについて詳しい解析を行った.

	ϕ_1	ϕ_2	ψ_1	ψ_2
ϕ_1	LL1	LL2	HL1	HL2
ϕ_2	LL3	LL4	HL3	HL4
ψ_1	LH1	LH2	HH1	HH2
ψ_2	LH3	LH4	HH3	HH4

上の図の LL1 から LL4 までは低周波成分, LH1 から LH4 までは主として縦方向のエッジ, HL1 から HL4 までは主として横方向のエッジで HH1 から HH4 までは主として斜め方向のエッジの領域である。まず縦方向のエッジについてシミュレーションを行う。上が黒, 下が白の画像 (行列で表すとある行より上が 0 で, 下が 255) を変換する。画像を縦方向に移動すると前処理とサホートの違い等で 4 パターンのデータが現れる。これのエッジ付近のデータを表にしたのが次の図である。

前処理I(縦方向に(-2)移動)	前処理I(縦方向に(-1)移動)	前処理I(縦方向に0移動)	前処理I(縦方向に1移動)
HL1 -111 -2 -2 -2	HL1 -13 -2 -2 -2	HL1 -49 -2 -2 -2	HL1 -27 -2 -2 -2
HL2 -35 22 -2 -2	HL2 1 -26 -2 -2	HL2 6 -107 -2 -2	HL2 -10 5 -2 -2
HL3 0 0 0 0	HL3 0 0 0 0	HL3 0 0 0 0	HL3 0 0 0 0
HL4 0 0 0 0	HL4 0 0 0 0	HL4 0 0 0 0	HL4 0 0 0 0
LH1 25 23 -2 -2	LH1 -13 49 -2 -2	LH1 -8 54 -2 -2	LH1 52 5 -2 -2
LH2 35 -35 0 0	LH2 -53 206 0 0	LH2 -11 7 0 0	LH2 74 -10 0 0
LH3 9 22 -2 -2	LH3 -9 35 -2 -2	LH3 -5 40 -2 -2	LH3 35 5 -2 -2
LH4 13 -33 0 0	LH4 -12 -34 0 0	LH4 -6 22 0 0	LH4 49 -9 0 0
HH1 -35 22 0 0	HH1 1 1 0 0	HH1 6 6 0 0	HH1 -10 5 0 0
HH2 -49 -33 0 0	HH2 2 16 0 0	HH2 8 70 0 0	HH2 -13 -9 0 0
HH3 0 0 0 0	HH3 0 0 0 0	HH3 0 0 0 0	HH3 0 0 0 0
HH4 0 0 0 0	HH4 0 0 0 0	HH4 0 0 0 0	HH4 0 0 0 0

前処理II(縦方向に(-2)移動)	前処理II(縦方向に(-1)移動)	前処理II(縦方向に0移動)	前処理II(縦方向に1移動)
HL1 -406 0 0 0	HL1 0 0 0 0	HL1 -175 0 0 0	HL1 0 0 0 0
HL2 -124 82 0 0	HL2 -2 0 0 0	HL2 20 -390 0 0	HL2 0 0 0 0
HL3 0 0 0 0	HL3 0 0 0 0	HL3 0 0 0 0	HL3 0 0 0 0
HL4 0 0 2 0	HL4 0 0 0 0	HL4 0 0 0 0	HL4 0 0 0 0
LH1 50 87 0 0	LH1 -38 144 0 0	LH1 -16 166 0 0	LH1 180 0 0 0
LH2 71 -124 0 0	LH2 -53 206 0 0	LH2 -22 71 0 0	LH2 255 0 0 0
LH3 5 82 0 0	LH3 -28 102 0 0	LH3 -7 122 0 0	LH3 128 0 0 0
LH4 7 -117 0 0	LH4 -38 -146 0 0	LH4 -9 115 0 0	LH4 180 0 0 0
HH1 -124 82 0 0	HH1 -2 0 0 0	HH1 20 20 0 0	HH1 0 0 0 0
HH2 -175 -117 0 0	HH2 -2 -2 0 0	HH2 29 260 0 0	HH2 0 0 0 0
HH3 0 0 0 0	HH3 0 0 0 0	HH3 0 0 0 0	HH3 0 0 0 0
HH4 0 0 0 0	HH4 0 0 0 0	HH4 0 0 0 0	HH4 0 0 0 0

ここで注意が必要で画像データが上が黒, 下が白で横方向にエッジが無くても前処理をすると横方向にエッジが現れる場合があり, この場合には HL 成分が 0 でなくなる。前処理 I の場合にはどうしても HL 成分が 0 にはならないが, 前処理 II の場合には 0 になる場合がある。(下の段の左から 2 番目と 4 番目) この点については前処理 II の方が優れている。

このように縦方向のエッジの成分が HL の領域にも現れるが, HL3 と HL4 には現れないのは興味深い事である。

横方向のエッジについても同様の結果が得られた。横方向のエッジの成分が LH の領域にも現れるが, LH2 と LH4 には現れない。

以上の結果より, 縦方向のエッジの強さは LH2 と LH4 の領域で評価し, 横方向のエッジの強さは HL3 と HL4 の領域で評価すれば良いことがわかる。

5 画像圧縮への応用

前の結果を用いて次のような画像圧縮を行い従来の結果と比較した。目標の高周波成分の圧縮率 rate に対して LH2 と LH4 の領域の自乗平均が a , HL3 と HL4 の領域の自乗平均が b の場合,

LH 成分の圧縮率: $1.25 \times \text{rate} \times \frac{a}{a+b}$,

HL 成分の圧縮率: $1.25 \times \text{rate} \times \frac{b}{a+b}$,

HH 成分の圧縮率: $0.5 \times \text{rate}$,

として, 次の Lena, Barbara, Mandrill と Gold に対して前処理 II を用いて 2-Levels の分解を行い圧縮を行った。



Fig 4: Barbara, Lena, Mandrill と Gold images.

結果は次の通りである。

Barbara	preI	preII	new
MSE(0.01)	281.8	225.5	220.5
MSE(0.03)	209.4	156.7	151.0
MSE(0.05)	165.4	116.5	114.5
MSE(0.1)	98.4	63.3	60.5
PSNR(0.01)	23.6	24.6	24.7
PSNR(0.03)	24.9	26.2	26.3
PSNR(0.05)	25.9	27.5	27.5
PSNR(0.1)	28.2	30.1	30.3
Max(0.01)	115.8	121.0	121.0
Max(0.03)	106.3	99.3	84.7
Max(0.05)	90.1	80.6	80.6
Max(0.1)	62.7	55.5	62.2

Lena	preI	preII	new
MSE(0.01)	142.1	91.0	87.9
MSE(0.03)	97.8	60.2	55.7
MSE(0.05)	74.4	42.2	40.5
MSE(0.1)	38.7	21.8	20.5
PSNR(0.01)	26.6	28.5	28.7
PSNR(0.03)	28.2	30.3	30.7
PSNR(0.05)	29.4	31.9	32.1
PSNR(0.1)	32.3	34.7	35.0
Max(0.01)	115.3	98.3	107.3
Max(0.03)	80.0	63.6	75.9
Max(0.05)	65.7	54.9	48.3
Max(0.1)	37.2	32.6	34.7

Mandrill	preI	preII	new
MSE(0.01)	538.1	455.5	453.1
MSE(0.03)	454.6	369.3	361.9
MSE(0.05)	393.6	309.1	304.1
MSE(0.1)	279.5	205.4	209.8
PSNR(0.01)	20.8	21.5	21.6
PSNR(0.03)	21.6	22.5	22.5
PSNR(0.05)	22.2	23.2	23.3
PSNR(0.1)	23.7	25.0	24.9
Max(0.01)	150.7	142.2	130.7
Max(0.03)	150.8	133.3	131.5
Max(0.05)	111.5	139.6	113.3
Max(0.1)	93.7	93.0	94.7

Gold	preI	preII	new
MSE(0.01)	161.0	115.1	111.6
MSE(0.03)	125.2	84.1	81.5
MSE(0.05)	102.4	68.2	64.7
MSE(0.1)	66.0	42.8	40.8
PSNR(0.01)	26.1	27.5	27.7
PSNR(0.03)	27.2	28.9	29.0
PSNR(0.05)	28.0	29.8	30.0
PSNR(0.1)	29.9	31.8	32.0
Max(0.01)	103.0	97.8	102.7
Max(0.03)	81.1	76.9	80.2
Max(0.05)	65.9	60.9	60.1
Max(0.1)	48.2	37.3	42.7

但し preI と preII はそれぞれ前処理 I と前処理 II を用いて高周波成分を一様に圧縮した場合で、new は前処理 II と今回提案した方法で圧縮した場合である。誤差の評価は代表的な誤差評価関数、MSE(平均自乗誤差)、PSNR(ピーク SN 比) と Max(最大誤差) を用いた。カッコの中の数値は圧縮率である。

表を見たらわかるが、今回提案した方法の方が従来の方がいい結果が出た。画像の質を比較するために Barbara の同じ圧縮率 (0.01) の画像を D4(DGHM マルチウェーブレットと同じ精度) の場合、前処理 I で一様に圧縮した場合と今回提案した方法で圧縮した場合で比較する。ズボンの縞に注目すると、今回提案した方法が最も画質が良いのがわかる。

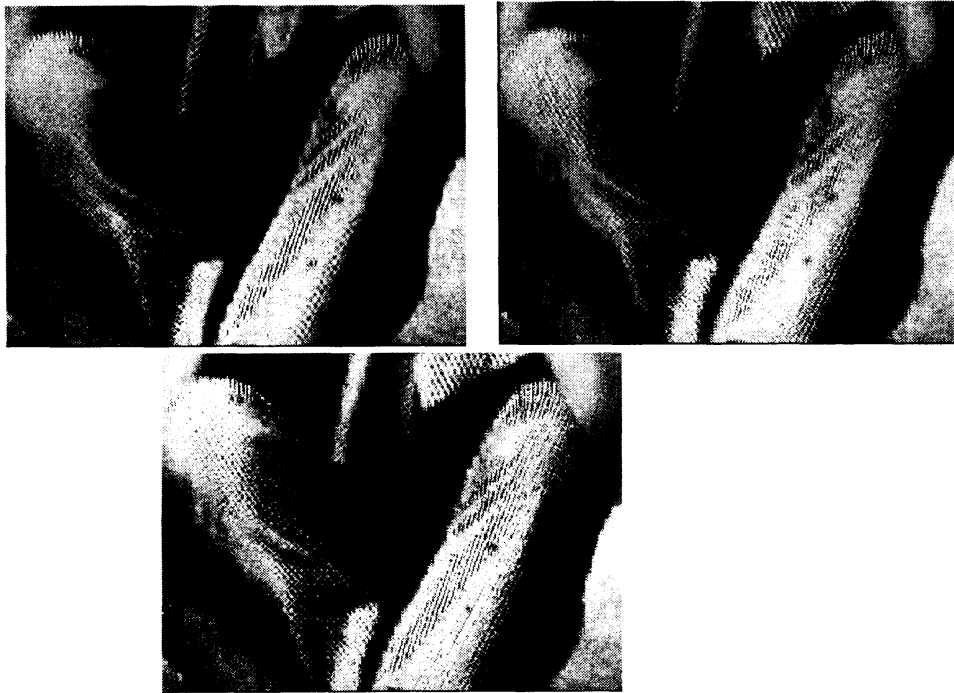


Fig 5: 左上が D4, 右上が preI で下の画像が今回提案した新しい方法による圧縮画像。圧縮率は全て 0.01。

6 結論

今回、マルチウェーブレット変換で現れる高周波成分を縦方向と横方向のエッジを調べることにより、前処理により予想外の領域にエッジが現れることがわかった。また縦方向と横方向のエッジのみが現れる領域が特定でき、これを用いて画像圧縮を行うと従来の方がいい結果が出る事が数値シミュレーションにより示された。12 個の高周波成分を更にくわしく解析することにより、更により良い画像処理が可能になると思われる。

References

- [1] I. Daubechies, *Ten Lectures on Wavelets*, CBMS 61, SIAM. (1992).
- [2] G. Donavan, J. Geronimo, D. Hardin and P. Massopust, *Construction of orthogonal wavelet using fractal interpolation functions*, SIAM J. Math. Anal. **27** (1996), 1158–1192.
- [3] K. Mizohata, *The Application of the GHM Multiwavelet Transform to Image Processing*, 1–8, Research Reports of Research Institute for Mathematical Sciences, Kyoto University, **1385**, 2004.
<http://hdl.handle.net/2433/25760>
- [4] V. Strela, P. Heller, G. Strang, P. Topiwala and C. Heil, *The application of multiwavelet filter banks to signal and image processing*, IEEE Trans. on Image Proc., **8**(4) (1999), 548–563.