

博 士 論 文

直交変換によるデジタル画像の
高圧縮・高精細化に関する研究

2007 年

後 藤 富 朗

目次

第 1 章	序論	1
1.1	高圧縮符号化技術	2
1.2	高精細化技術	3
1.3	本論文の構成	6
第 2 章	デジタル画像処理	7
2.1	離散コサイン変換 (DCT)	8
2.2	サブバンド符号化	11
2.2.1	帯域分割フィルタバンク	11
2.2.2	帯域合成フィルタバンク	12
2.2.3	フィルタバンクの構成要素	12
2.2.4	2 分割フィルタバンク	15
2.2.5	フィルタバンクの設計	16
2.3	ウェーブレット変換	18
2.3.1	ウェーブレット関数とウェーブレット基底関数	18
2.3.2	ウェーブレット変換	19
2.3.3	ウェーブレット級数	20
2.3.4	離散ウェーブレット変換	21
2.4	量子化	22
2.4.1	線形量子化	22
2.4.2	非線形量子化	22
2.5	可変長符号化	24
2.5.1	エントロピー	24
2.5.2	可変長符号化の原理	24
2.5.3	ランレンゲス符号化	24
2.6	カラー画像成分と色変換	25

目次

2.7	評価指標	28
2.7.1	客観評価	28
2.7.2	主観評価	30
第 3 章	高圧縮符号化技術	33
3.1	マルチ走査によるウェーブレット変換符号化	34
3.1.1	マルチ走査	34
3.1.2	量子化器	35
3.1.3	シミュレーション	36
3.1.4	ウェーブレット変換係数の分布	37
3.1.5	復号画像の画質評価	39
3.1.6	むすび	42
3.2	ウェーブレット – DCT 符号化	43
3.2.1	ウェーブレット – DCT	43
3.2.2	実験結果	45
3.2.3	むすび	53
3.3	カラー成分間の相関を利用したカラー画像符号化	55
3.3.1	成分間の関係を利用した符号化方法	55
3.3.2	提案するウェーブレット変換符号化器の構成	55
3.3.3	フィルタバンク	56
3.3.4	RGB から $l\ C'_r\ C'_b$ への色変換	58
3.3.5	量子化	60
3.3.6	符号化	61
3.3.7	実験結果	64
3.3.8	むすび	84
第 4 章	解像度変換技術	87
4.1	画像空間	87
4.1.1	整数分の 1 倍の縮小	87
4.1.2	整数倍の拡大	88
4.1.3	任意の倍率のサイズ変換	93
4.2	DCT 空間	97
4.2.1	ブロックサイズ変更	97
4.2.2	ブロック数変更	99
4.2.3	提案法	107

4.2.4	提案法の高速化	117
4.3	実験結果	119
4.3.1	解像度変換画像の比較	119
4.3.2	ブロックひずみ混入画像の比較	134
4.3.3	演算回数の比較	135
4.3.4	高速手法の演算回数の比較	143
4.4	むすび	144
第 5 章	高精細化技術	147
5.1	Deblocking Edge Filter	148
5.2	ウェーブレット変換を用いた高精度化	148
5.2.1	DCT 係数のサブバンド表現	148
5.2.2	DCT 係数のウェーブレット表現	149
5.2.3	逆ウェーブレット変換を用いたブロックひずみ低減手法	152
5.3	実験結果	161
5.3.1	実験条件	164
5.3.2	客観評価	164
5.3.3	主観評価	165
5.3.4	予備実験	166
5.3.5	閾値 τ の設計	166
5.3.6	実験結果	167
5.4	むすび	200
第 6 章	結論	203
6.1	高圧縮符号化技術	203
6.1.1	マルチ走査によるウェーブレット変換符号化	204
6.1.2	ウェーブレット – DCT 符号化	205
6.1.3	カラー成分間の相関を考慮したウェーブレット変換符号化	207
6.2	高精細化技術	208
6.2.1	隣接ブロックを用いた DCT 係数上での解像度変換	208
6.2.2	逆ウェーブレット変換を用いたブロックひずみ低減手法	209
6.3	今後の課題および展望	210
参考文献		213
発表論文一覧		219

目次

査読論文	219
国際学会論文	219
国内学会発表等	220

第 1 章

序論

近年の LSI 技術の飛躍的な発展に伴って、デジタル信号処理技術はその応用範囲を急速に広げている。また、コンピュータの普及により音声や画像信号をデジタル的に取り扱いデジタルデータとして保存するのみならず、アナログでは不可能だった様々な処理がデジタルかつリアルタイムで実現されつつある。また、インターネットの普及により、デジタル画像を目にする機会が増え、デジタル画像は、身近なものとなっている。しかし、音声信号や画像信号ではデータ量が膨大なものとなるため、一般に符号化や量子化を行いデータの冗長を削除することによってデータを圧縮して保存・伝送する。

静止画像の圧縮・保存方式である国際標準規格として JPEG (Joint Photographic Experts Group) [1] が、動画画像の圧縮・保存方式である国際標準規格として MPEG (Moving Picture Experts Group) [2] や H.264 [3] があり、これらの国際標準規格では、変換符号化に離散コサイン変換 (DCT) が用いられている。この DCT は圧縮効率が良いため古くから用いられており今日の画像符号化技術の主流になっている。

しかし DCT は、画像を任意の変換ブロックに分割し、そのブロックごとに量子化・符号化等の処理が行われるため、圧縮率を上げるにつれてブロックひずみやモスキート雑音などの雑音が復元画像に混入し画質の劣化が生じてしまう。そこでこれらの雑音が混入しにくい符号化方式としてフィルタバンク符号化が注目されている。フィルタバンク符号化

第1章 序論

には、サブバンド符号化やウェーブレット変換符号化といったものがある。サブバンド符号化は、ローパスフィルタおよびハイパスフィルタによって信号を帯域制限しその出力信号全体をさらにフィルタリングする方法であり、ウェーブレット変換符号化は、ローパスフィルタおよびハイパスフィルタによって信号を帯域制限し情報量の多い低周波数帯域のみを階層的にフィルタリングする方法である。これらの符号化は、フィルタバンクやデシメーション・インタポレーションといった技術を用いて実現する。ウェーブレット変換符号化を採用した新たな国際標準規格として JPEG2000 や Motion-JPEG2000 があり、変換符号化にはウェーブレット変換が採用されており、高い圧縮性能を実現している。

前出の国際標準規格では、変換符号化に DCT やウェーブレット変換を用いており、これらの変換は、変換前および変換後の座標系がそれぞれ直交していることから、直交変換と呼ばれている。直交変換による画像圧縮は前出の国際標準規格により、画像を低ビットレートで伝送・蓄積が可能となってきた。

また、液晶ディスプレイやプラズマディスプレイ等、ディスプレイの大画面化に伴い、蓄積された画像データを再利用する際に様々な問題が指摘されている。

本論文では、高圧縮符号化技術および画像の高精細化技術について論じる。

1.1 高圧縮符号化技術

一般的にフィルタバンク符号化は DCT 符号化に比べ処理速度の点で劣る。画像圧縮において最も考慮される点が圧縮率であるため多くの研究者が圧縮効率を改善するべく様々な研究を行ってきており、DCT 符号化での圧縮率に比べ、フィルタバンク符号化での圧縮率の方が高くなってきている。また、リアルタイム処理において処理速度は圧縮率同様重要なテーマである。にもかかわらず処理速度を向上させるための研究は、それほど盛んに行われていない。

そこで本論文では、高速化の観点から斜め走査およびマルチ走査によるフィルタバンク符号化を提案し、それらを適用したウェーブレット変換符号化を設計する。従来の走査と

提案走査を用いたウェーブレット変換符号化の復号画像を視覚的に比較することにより提案走査法が主観的に画像に影響を及ぼさないことを示し、復元画像のピーク SN 比や平均ビットレートの比較によって提案走査法が画像に対して客観的にも影響を及ぼさないことを示す。得られた結果から提案走査法の有用性について検討する。

また、ウェーブレット変換符号化は、新しい国際標準規格である JPEG2000 [4] に採用されており、広く用いられ始めている。ウェーブレット変換符号化によりカラー画像信号を圧縮処理する場合には、従来は赤緑青の RGB 信号を色変換を行い輝度信号 Y と 2 種類の色差信号 C_r , C_b に変換し、各成分を独立に符号化する。よってカラー画像を符号化する場合には Y , C_r , C_b 各成分の 3 枚のグレースケール画像を符号化すると同様に符号化を行う。そのため 3 成分の関係を考慮した符号化を行うことで符号化効率の改善が期待できる。

そこで、本論文ではカラー画像に対するウェーブレット変換符号化の符号化効率の改善を目的として新たな色変換を提案し、変換係数において 3 成分の関係を考慮した符号化を行う。また、提案したウェーブレット変換符号化器および従来型のウェーブレット変換符号化器を設計し、ビットレートや PSNR による客観評価および視覚的に評価することで、提案手法の有効性について検討する。

1.2 高精細化技術

デジタル画像処理における研究分野は、圧縮・保存だけでなく様々な応用が考えられる。その 1 つに解像度変換がある。これは現在、デジタルカメラやディスプレイ、携帯電話等デジタル画像を扱う製品が幅広く利用されており、これらの各種製品にはその用途、性能に応じて様々な種類の解像度が存在することに起因する。これらの互換性を保つためには、画像のサイズを変換する解像度変換技術が必要となる。また、プラズマテレビや大型液晶テレビに代表されるような大型表示装置の発展により、低解像度画像を高解像度で表示あるいは保存する機会が増えている。しかし、カメラ付き携帯電話で撮影されたよう

第1章 序論

な低解像度画像をテレビなどの大型表示装置で表示する場合、ブロック状のひずみが混入した画像をしばしば目にし、そのブロックひずみは拡大率を上げるにつれてより顕著なものとなる。携帯電話のカメラで撮影された災害時や事件発生直後の映像をインターネットやテレビのニュース番組で目にする機会が増えてきており、それらの画像はブロックひずみが混入していることが多い。そこでブロックひずみの発生しない、より鮮明な画像を生成できる解像度変換技術が求められている [5]–[7]。また、現在静止画像や動画の圧縮・保存形式の国際標準規格として、JPEG や MPEG が広く普及していることから、これらの符号化データである DCT 係数上での処理に関する研究が行われている [8]。一般に解像度変換は画像空間上で実行される。しかし、JPEG や MPEG の符号化データである DCT 係数に対し画像空間上での解像度変換を適用すると、符号化や復号化といった処理が必要となり、処理量が増大し使用メモリも増加してしまう。そこで、DCT 係数上でそのまま解像度変換する方法が注目されている [9]–[16]。従来の DCT 係数上での解像度変換は、DCT のブロックサイズを変更して実現している [9]。しかし、この手法は DCT のブロックサイズが変更されるため JPEG や MPEG の汎用 8×8 点 DCT をそのまま利用することができない。そこで、DCT のブロックサイズではなくブロック数を変更することで解像度変換する手法が提案されている [10]。この方法は、DCT のブロックサイズが固定されるため JPEG 等の 8×8 点汎用 DCT がそのまま利用できる。しかし、ブロックごとに独立して処理を行うため解像度変換後の画像にブロックひずみが発生する問題がある。

そこで本研究では、隣接ブロックを用いた DCT 係数上での解像度変換法を提案する。提案法は、DCT 係数のブロックサイズではなくブロック数を変更することで JPEG や MPEG の 8×8 点汎用 DCT をそのまま利用できる。また、ブロック毎に処理を行うのではなく周辺ブロックの DCT 係数を参照することで、解像度変換後の画像に現れるブロックひずみの解消が期待できる。

近年では JPEG2000 や MPEG4 [17] といった技術が登場し、静止画像や動画がさら

にきれいに扱われるようになってきた。しかし、ひと昔前まではハードディスクの容量が小さく伝送路も狭かったため、画像は小さなファイルサイズで保存・伝送されてきた。そのため、JPEG 形式で高圧縮し雑音が混入した画像が多く存在している。この保存容量の制限等で小さくせざるを得なかったデジタルカメラ画像や Web 上に存在する既存の JPEG 画像を利用することも多い。また近年、高解像度の表示装置が開発されており、低解像度の JPEG 画像を表示する際には解像度変換が必要となるが、拡大時にブロック状のひずみが混入することも問題となる。

これらの画像中のブロックひずみを軽減する方法として最も簡単なものに LPF によるノイズ除去 [18] がある。しかし、画像全体に LPF をかけることから、エッジまでも不鮮明となってしまう適切ではない。また、POCS (the Projection Onto Convex Sets) を用いた繰り返しによってブロックひずみを軽減する試みが提案されている [19]。しかし、繰り返しの操作を必要とすることから計算コストがかかる。符号化側でブロックをオーバーラップさせることによりブロックひずみを軽減する LOT [20] や、プレ・ポストフィルタリングを行う方法 [21] が提案されているが既に符号化されてしまっている画像に対してはプレフィルタリングを行うことはできない。つまり、既に符号化されてしまっている画像に対しては、復号化側で画像の視覚的特性を改善させる方法が適切である。

そこで本研究では、逆ウェーブレット変換による DCT 係数の高画質化復号法を提案する。提案法では DCT 係数表現はウェーブレット変換係数表現と等価であることを示し、さらに、画像信号が周波数帯域に分割されていることを利用しブロックひずみを軽減しつつエッジは残す復号化を行う。以上により解像度が低くブロックひずみの発生した既存の JPEG 画像からウェーブレット変換係数へと変換し、ブロックひずみがなく原画像より高解像度の画像が得られると期待でき、実験により提案手法の有効性について検討する。

第 1 章 序論

1.3 本論文の構成

本論文では, 第 2 章において JPEG や MPEG に用いられている画像処理の基礎的な技術について述べ, さらに画像圧縮における評価指標について述べる.

次に第 3 章では, 高速処理の観点から斜め走査およびマルチ走査を提案し, 符号化効率改善の観点からカラー成分間の相関を利用したカラー画像符号化を提案する. そして実験結果より, 提案手法の有効性について論じる.

第 4 章では, 解像度変換によるノイズ混入を低減させるため, 隣接ブロックを用いた DCT 係数上での解像度変換を提案し, 実験により, 解像度変換画像の画質および処理時間の観点から提案手法の有効性について論じる.

第 5 章では, ブロックひずみ低減の観点から逆ウェーブレット変換を用いたブロックひずみ低減手法を提案し, 提案手法の有効性について論じる.

最後に結論において各提案手法についてまとめおよび今後の課題について論じる.

第 2 章

ディジタル画像処理

ディジタル化した画像信号を効率的に伝送あるいは蓄積するためには、画像の圧縮によってできるだけ画像の冗長性を除去することが重要となる。そこで一般的な画像圧縮では、まず符号化によって画素信号の発生頻度を偏らせ、次に量子化によって効率的に冗長を削除し、そして量子化レベルに応じた符号語を割り当てる、といった流れで処理が行われる。

画像の符号化方式として代表的なものに変換符号化方式と予測符号化方式がある。特に変換符号化方式の代表的なものとして 2 次元 DCT (2 次元離散コサイン変換) があり、エネルギー圧縮効率が非常に良いため画像の圧縮で広く利用されている。しかし、DCT を用いた符号化方式では、画像を任意のブロックに分割するため、圧縮率を上げるにつれブロックひずみやモスキート雑音が復元画像に混入してしまい、復元画像の画質が劣化してしまう。

そこでこれらの雑音が混入しない符号化方式としてフィルタバンク符号化方式が注目されている。フィルタバンク符号化方式には、サブバンド符号化やウェーブレット変換符号化などがあり、これらの符号化では、画像を任意のブロックに分割することなく符号化するため、DCT に比べ圧縮効率や処理速度が問題になる。

符号化によって得られた信号は、逆の操作をすれば元どおりの信号にもどる。つまり、符

第2章 デジタル画像処理

号化する前と符号化した後では情報に偏りがあるものの情報量に違いはない。そこで情報を削除するために量子化を行い、得られた量子化レベル数に応じて符号語を割り当てることによって画像圧縮は達成される。

本章では、デジタル画像処理の基礎的な理論について論じる。

2.1 離散コサイン変換 (DCT)

離散コサイン変換は、フーリエ変換の実数項である \cos の項のみを用いる直交変換であり、自然画像の国際標準である JPEG や 動画画像の国際標準である MPEG に採用されている。符号化効率の点でカルーネン・レーベ (KL) 変換に最も似ており、符号化効率が高い。また、高速算法が存在するため LSI 化が容易に行える。

N 次の 1 次元離散コサイン変換、逆変換はそれぞれ式 (2.1), (2.2) のように表される。

$$F(u) = \sqrt{\frac{2}{N}} \sum_{x=0}^{N-1} f(x) \cos \frac{(2x+1)\pi}{2N} u \quad (u = 0, 1, \dots, N-1) \quad (2.1)$$

$$f(x) = \sqrt{\frac{2}{N}} \sum_{u=0}^{N-1} F(u) \cos \frac{(2x+1)\pi}{2N} u \quad (x = 0, 1, \dots, N-1) \quad (2.2)$$

図 2.1 に 8 次の場合の離散コサイン変換 (DCT) の基底ベクトルを示す。ブロックサイズ $N \times N$ の 2 次元離散コサイン変換、逆変換はそれぞれ式 (2.3), (2.4) のように表される。

$$F(u, v) = \frac{2C(u)C(v)}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cos \left[\frac{(2x+1)\pi}{2N} u \right] \cos \left[\frac{(2y+1)\pi}{2N} v \right] \quad (2.3)$$

$(u, v = 0, 1, \dots, N-1)$

$$f(x, y) = \frac{2}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} C(u)C(v)F(u, v) \cos \left[\frac{(2x+1)\pi}{2N} u \right] \cos \left[\frac{(2y+1)\pi}{2N} v \right] \quad (2.4)$$

$(x, y = 0, 1, \dots, N-1)$

ただし、

$$C(u), C(v) = \begin{cases} \frac{1}{\sqrt{2}} & u = 0, v = 0 \\ 1 & u \neq 0, v \neq 0 \end{cases}$$

である。

2.1 離散コサイン変換 (DCT)

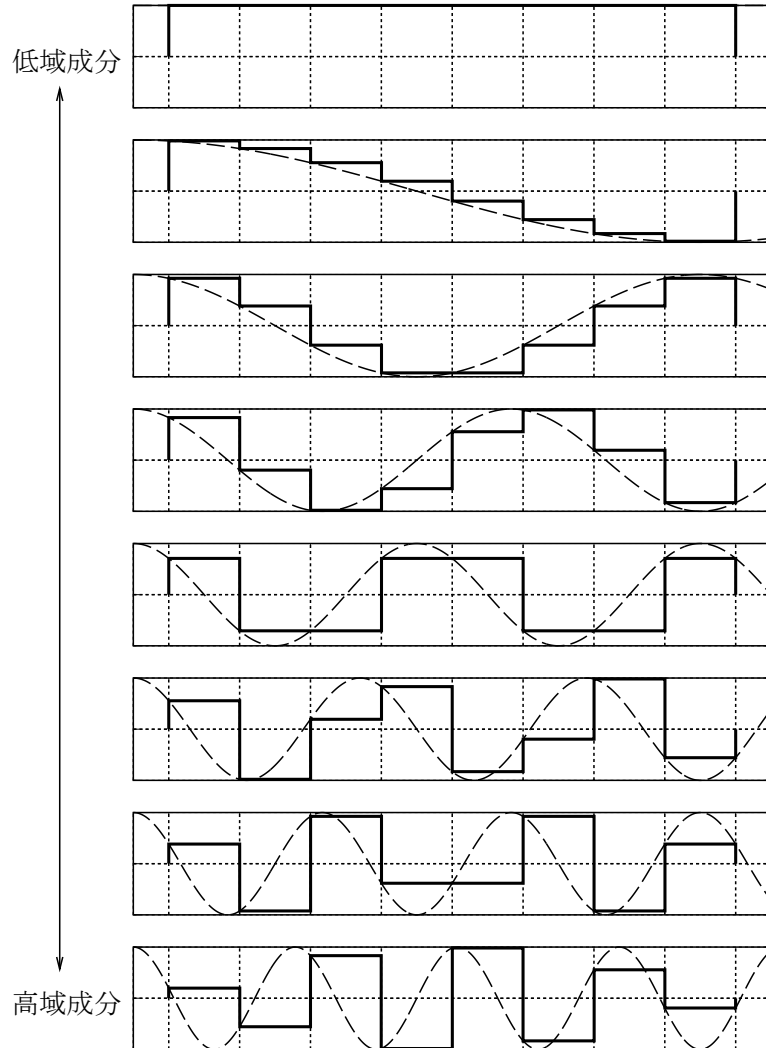


図 2.1 8 次の離散コサイン変換の基底ベクトル

2次元離散コサイン変換は、水平方向、垂直方向それぞれに1次元離散コサイン変換を行うことにより実現することができる。また、離散コサイン変換では一般に直流成分の近辺にエネルギーが集中することが知られており、直流成分に多くのビットを割り当て、高域成分になるにしたがって少ないビットで表現することによって高い圧縮率において高画質を実現している。

しかし、圧縮率を上げるにつれ、ブロックひずみやモスキート雑音と呼ばれる雑音が復元画像に混入する。式 (2.3) において $u = 0, v = 0$ のとき、つまり DCT の出力信号 $F(u, v)$ の直流成分は以下のように表される。

第2章 デジタル画像処理



図 2.2 ブロックひずみの例

$$\begin{aligned} F(0,0) &= \frac{2C(0)C(0)}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \cos \left[\frac{(2x+1)\pi}{2N} \cdot 0 \right] \cos \left[\frac{(2y+1)\pi}{2N} \cdot 0 \right] \\ &= N \cdot \frac{\sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y)}{N^2} \end{aligned} \quad (2.5)$$

この式 (2.5) は, $N \times N$ の画素値の平均をとることを意味している. 圧縮率が上がるにつれてこの直流成分の量子化幅が大きくなり量子化誤差が大きくなる. 量子化誤差が無視できないほど大きくなると復元画像にブロック状の模様ができてしまう. このブロック状の模様をブロックひずみと呼んでいる. 言い替えるとブロックひずみは, それぞれの変換ブロックの直流成分間の相関がなくなることによって混入してしまうひずみである. ブロックひずみの例を図 2.2 に示す.

画像のエッジ部分は, 高周波数成分を多く含んでいる. そのためエッジ部分では高周波

2.2 サブバンド符号化

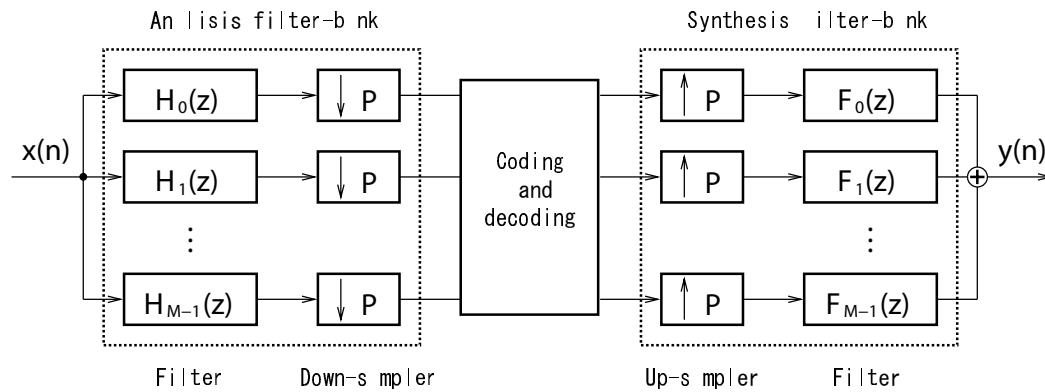


図 2.3 M 分割フィルタバンク

係数の量子化雑音が出やすくなる。さらに、この雑音は変換ブロック全体に広がる。このため、エッジから離れたところまで雑音の影響が及んでしまう。これがモスキート雑音発生 の概略である。

ブロックひずみやモスキート雑音は、画像を任意の変換ブロックに分割して符号化や量子化を行うことによって発生する。ブロックひずみやモスキート雑音が発生しない符号化方式として注目されているのがサブバンド符号化やウェーブレット変換符号化である。これらの符号化では画像全体を任意のブロックに分割することなく符号化する。

2.2 サブバンド符号化

1 次元信号に対するサブバンド符号化の原理図を図 2.3 に示す。この図からわかるように符号化器および復号化器のほかにフィルタバンクと呼ばれるシステムを用いる。

フィルタバンクは、帯域分割フィルタバンクと帯域合成フィルタバンクという二つの部分からなる。図 2.3 にあるようにそれらのフィルタバンクはデジタルフィルタ、ダウンサンプラおよびアップサンプラと呼ばれる要素から構成される。

2.2.1 帯域分割フィルタバンク

図 2.3 の構成では、帯域を M 個に分割しその後 $P : 1$ にダウンサンプリングを行う。 $M = P$ の場合、このフィルタバンクを最大間引きフィルタバンクという。サブバンド符

第2章 デジタル画像処理

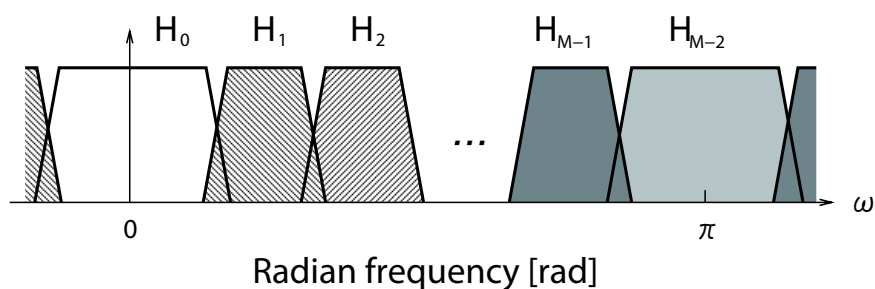


図 2.4 M 分割フィルタバンクの振幅特性

号化では、情報圧縮の観点から最大間引きフィルタバンクを用いる。

図 2.4 にフィルタバンクに用いる典型的なフィルタ特性を示す。情報損失が生じないようにフィルタ特性が互いに重複し全帯域をカバーしていることがわかる。DCT 符号化が DCT 係数の電力分布の偏りを利用するように、サブバンド符号化では各帯域間の電力分布に応じて符号化ビット数を割り当てる。

2.2.2 帯域合成フィルタバンク

帯域分割され、ダウンサンプリング処理された信号を再びもとの信号に復元するシステムが帯域合成フィルタバンクである。したがって信号に量子化を施さない場合、理想的にはフィルタバンクの入力信号 $x(n)$ と出力信号 $y(n)$ は遅延を除き一致する。つまりフィルタバンクの設計は再構成を保证するものでなければならない。

2.2.3 フィルタバンクの構成要素

デジタルフィルタ

線形時不変システムであり、図 2.5 のようにブロック表現される。したがってその入出力関係式は、時間領域、 z 領域、周波数領域それぞれ以下のようにになる。

$$y(n) = \sum_{k=-\infty}^{\infty} h(k)x(n-k) \quad (2.6)$$

$$Y(z) = H(z)X(z) \quad (2.7)$$

$$Y(e^{j\omega}) = H(e^{j\omega})X(e^{j\omega}) \quad (2.8)$$

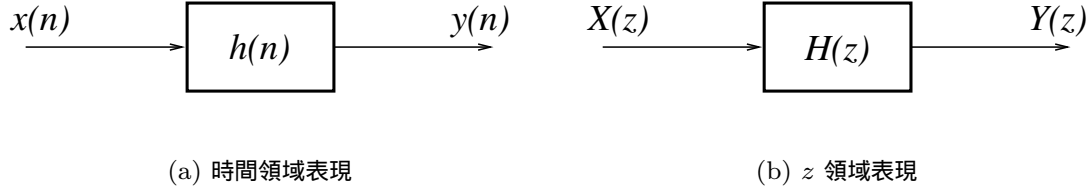


図 2.5 デジタルフィルタのブロック表現

アップサンブラ

信号に零値を挿入しサンプリングレートを U 倍にするシステムであり, 図 2.6 のように表される. したがってその入出力関係式は, 時間領域では,

$$y(m) = \begin{cases} x(m/U) & m = 0, \pm U, \pm 2U, \dots \\ 0 & \text{その他} \end{cases} \quad (2.9)$$

となる. この式の両辺を z 変換すると,

$$Y(z) = \sum_{m=-\infty}^{\infty} y(m)z^{-m} \quad (2.10)$$

$$= \sum_{m=-\infty}^{\infty} x(m)z^{-mU} = X(z^U) \quad (2.11)$$

となる. 入力 $x(n)$ およびその周波数特性 $X(e^{j\omega_x})$ を図 2.6 (b) とすると, 出力 $y(n)$ およびその周波数特性 $Y(e^{j\omega_y})$ は図 2.6 (c) となる. このアップサンプリングにより新たに生じるスペクトル成分 (図 2.6 (c) 参照) をイメージング成分と呼ぶ.

ダウンサンブラ

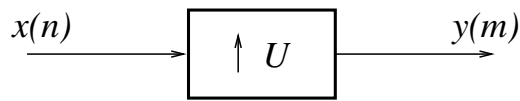
信号を間引き, サンプリングレートを $1/D$ にするシステムであり, 図 2.7 のように表される. また出力信号の時間領域表現 $y(m)$ は, 図 2.6 (b) と (c) の数列の積

$$v(n) = x(n)p(n) \quad (2.12)$$

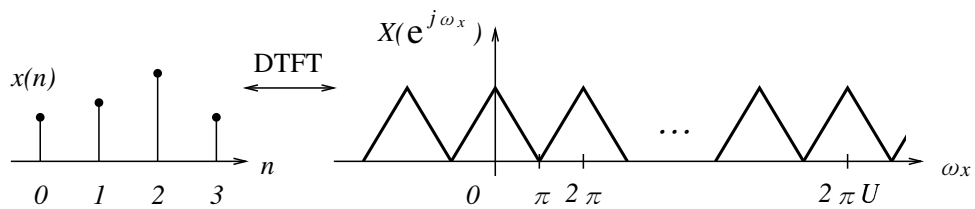
を用いて

$$y(m) = v(Dm) = x(Dm)p(Dm) \quad (2.13)$$

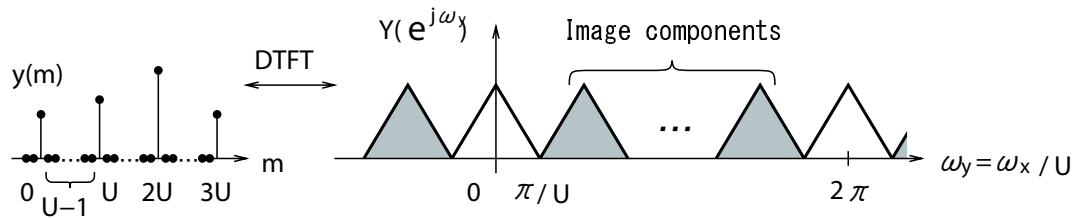
第2章 デジタル画像処理



(a) ブロック図



(b) 処理前の振幅・周波数特性



(c) 処理後の振幅・周波数特性

図 2.6 アップサンブラ

と表現できる．また出力信号の z 領域表現 $Y(z)$ は上式の両辺を z 変換することで求めることができる．つまり,

$$Y(z) = \sum_{m=-\infty}^{\infty} y(m)z^{-m} \quad (2.14)$$

$$= \sum_{m=-\infty}^{\infty} v(Dm)z^{-m} \quad (2.15)$$

$$= \sum_{k=-\infty}^{\infty} v(k)z^{-k/D} (k = Dm) \quad (2.16)$$

$$= \sum_{k=-\infty}^{\infty} x(k)p(k)z^{-k/D} \quad (2.17)$$

2.2 サブバンド符号化

となる．さらに上式に $p(n)$ の離散フーリエ級数展開

$$p(n) = \frac{1}{D} \sum_{r=0}^{D-1} e^{j2\pi rn/D} \quad (2.18)$$

を代入すると,

$$Y(z) = \sum_{k=-\infty}^{\infty} x(k) \left[\frac{1}{D} \sum_{r=0}^{D-1} e^{j2\pi rk/D} \right] z^{-k/D} \quad (2.19)$$

$$= \frac{1}{D} \sum_{r=0}^{D-1} \sum_{k=-\infty}^{\infty} x(k) \left(e^{-j2\pi r/D} z^{1/D} \right)^{-k} \quad (2.20)$$

$$= \frac{1}{D} \sum_{r=0}^{D-1} X(W^r z^{1/D}) \quad (2.21)$$

$$\text{ただし, } W = e^{-j2\pi/D} \quad (2.22)$$

上式において $r = 0$ 以外の成分 $X(W^r z^{1/D})$ は, ダウンサンプリングにより新たに生じたものであり, エイリアジング成分に相当する．

また, 出力信号の周波数領域表現 $Y(e^{j\omega})$ は式 (2.21) において $z = e^{j\omega}$ を代入すると

$$Y(e^{j\omega}) = \frac{1}{D} \sum_{p=0}^{D-1} X(e^{j(\omega-2\pi p)/D}) \quad (2.23)$$

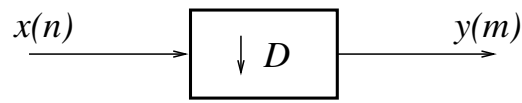
となる．

2.2.4 2 分割フィルタバンク

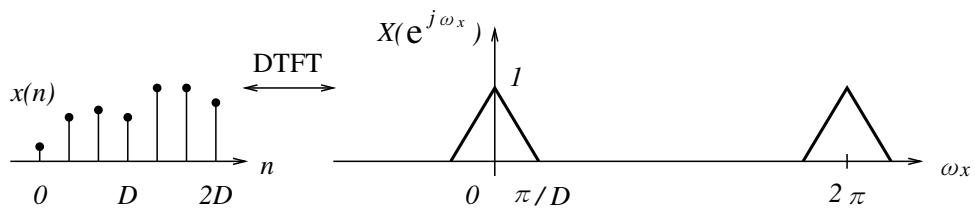
もっともよく使用されるフィルタバンクは, 帯域を 2 分割し再び元の信号に復元する 2 分割フィルタバンクである (図 2.8 (a)).

画像信号の場合には, 図 2.8 (b) のようにフィルタバンクを縦続接続した 4 分割フィルタバンクが用いられる．ただし, 2 次元信号のフィルタリングは, 図 2.8 (b) にあるように画像の水平方向と垂直方向に対してそれぞれ行う．またダウンサンプラとアップサンプラの処理も水平・垂直方向それぞれに対して施される．

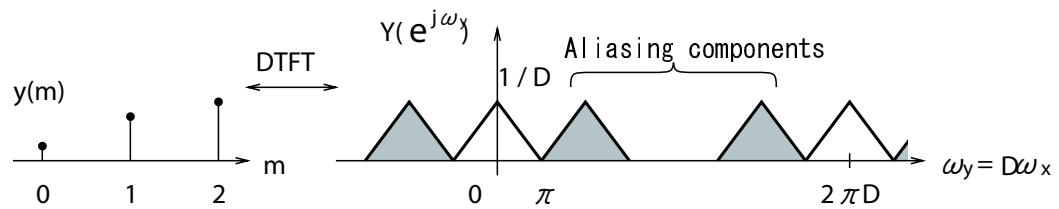
第2章 デジタル画像処理



(a) ブロック図



(b) 処理前の振幅・周波数特性



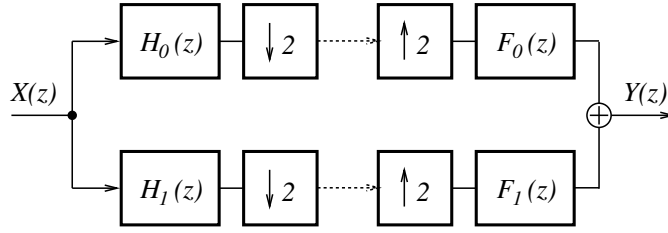
(c) 処理後の振幅・周波数特性

図 2.7 ダウンサンプリング

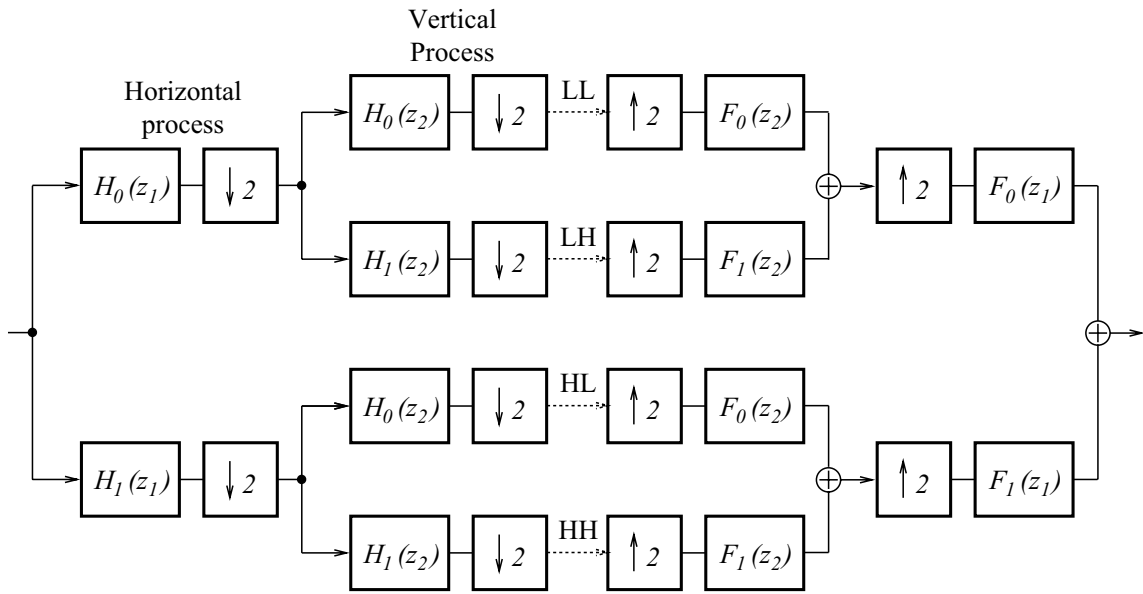
2.2.5 フィルタバンクの設計

フィルタバンクは、信号を帯域分割すると同時に帯域分割された信号からもとの信号を再構成する必要がある。つまり、フィルタバンクを構成する各フィルタは、信号を再構成できるように設計しなければならない。

2.2 サブバンド符号化



(a) 2 分割 (1 次元処理)



(b) 4 分割 (2 次元処理)

図 2.8 フィルタバンク

フィルタバンクの入出力関係

図 2.8 (a) のような構成で式 (2.6) ~ (2.21) の関係を用いると z 領域において入力 $X(z)$ と出力 $Y(z)$ の関係は,

$$Y(z) = \frac{1}{2} \{ H_0(z) F_0(z) + H_1(z) F_1(z) \} X(z) + \frac{1}{2} \{ H_0(-z) F_0(z) + H_1(-z) F_1(z) \} X(-z) \quad (2.24)$$

となる. 上式の右辺第二項は, ダウンサンプリングおよびアップサンプリングにより生じたエイリアジング成分とイメージ成分である.

第2章 デジタル画像処理

フィルタバンクの設計条件

フィルタバンクは、信号を再構成できる必要がある。この条件をいいかえると式 (2.24) が、

$$Y(z) = z^{-L} X(z) \quad (2.25)$$

と整理されるようなフィルタの設計問題に帰着する。ただし、 z^{-L} は L クロック遅延を意味する。つまり上式は、入力 $X(z)$ が L クロックだけ遅延して出力 $Y(z)$ に現われることの数式表現である。

式 (2.24) と (2.25) より以下の式が導ける。

$$H_0(-z)F_0(z) + H_1(-z)F_1(z) = 0 \quad (2.26)$$

$$H_0(z)F_0(z) + H_1(z)F_1(z) = 2z^{-L} \quad (2.27)$$

上式は完全再構成条件と呼ばれ、フィルタバンクの設計では、この条件を満足するように設計するのが望ましい。

2.3 ウェーブレット変換

2.3.1 ウェーブレット関数とウェーブレット基底関数

図 2.9 (a) に示すように平均値が零で原点 $t = 0$ のまわりに Δt の幅で局在する関数 $\psi(t)$ を考える。この $\psi(t)$ のフーリエ変換 $\Psi(\omega)$ は、図 2.9 (b) に示すように $\Omega_c = (\Omega_L + \Omega_H)/2$ を中心に帯域 $\Delta\Omega = \Omega_H - \Omega_L$ で局在しているとする。このとき $\psi(t)$ をウェーブレット関数、マザーウェーブレットあるいは基本ウェーブレットという。

また、この $\psi(t)$ を正の実数 a と実数 b を用いて以下のように関数 $\psi_{a,b}(t)$ を定義する。

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) \quad (2.28)$$

ここで、 a は $\psi(t)$ を拡大・縮小するスケール係数であり、 b は時間的な移動量を決定するシフト係数である。また、 \sqrt{a} は正規化のための係数である。この $\psi_{a,b}(t)$ をウェーブレット基底関数という。

2.3 ウェーブレット変換

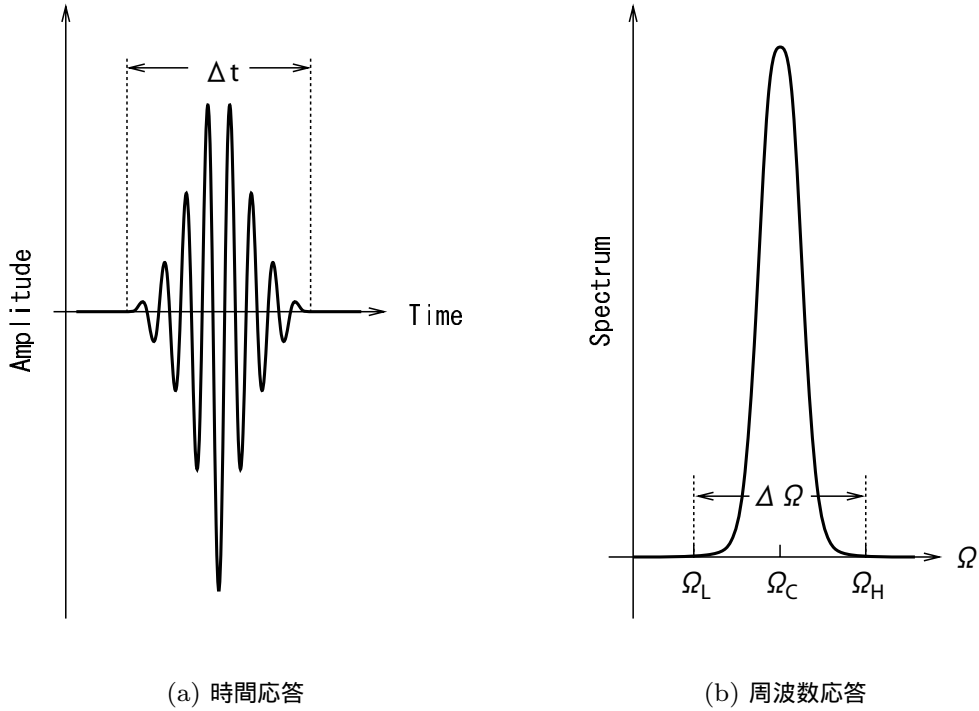


図 2.9 ウェーブレット関数

基底関数 $\psi_{a,b}(t)$ は, $a\Delta t$ の時間幅で $b - a\Delta t/2 \leq t \leq b + a\Delta t/2$ の範囲に局在する. また, $\psi_{a,b}(t)$ のフーリエ変換 $\Psi_{a,b}(\Omega)$ は $\Psi_{a,b}(\Omega) = a^{1/2}\Psi(a\Omega)e^{-j\omega b}$ であり, 中心周波数 Ω_c/a , 帯域 $\Delta\omega/a$ となる. したがって, 中心周波数と帯域幅の比 $(\Omega_c/a)/(\Delta\Omega/a)$ が a の変化に対して一定となる.

2.3.2 ウェーブレット変換

有限なエネルギーをもつ信号 $f(t)$ とウェーブレット関数 $\psi_{a,b}(t)$ の内積として

$$\begin{aligned} W_f(a, b) &= \langle f(t), \psi_{a,b}(t) \rangle \\ &= \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} f(t) \bar{\psi}\left(\frac{t-b}{a}\right) dt \end{aligned} \quad (2.29)$$

を定義する. ここで, $\bar{\psi}(\cdot)$ は $\psi(\cdot)$ の複素共役である. この $W_f(a, b)$ を $f(t)$ のウェーブレット変換あるいは連続ウェーブレット変換という. $\Psi_{a,b}(\Omega)$ および $\psi_{a,b}(t)$ の性質から a が小さい場合, $\psi_{a,b}(t)$ の時間幅は短く $W_f(a, b)$ は高い周波数成分に対応する. 一方, a

第2章 デジタル画像処理

が大きい場合には, $\psi_{a,b}(t)$ の時間幅は長く $W_f(a, b)$ は低い周波数成分に対応する.

ウェーブレット関数 $\psi(t)$ が以下のアドミッシブル条件を満足すると仮定する.

$$C_\psi = \int_{-\infty}^{\infty} \frac{|\Psi(\Omega)|}{|\Omega|} d\Omega < \infty \quad (2.30)$$

この条件は, $\psi(t)$ が大きな t において十分な減衰をもち, 時間平均が零であることに相当する. このとき, 次式の逆変換が成立する.

$$f(t) = \frac{1}{C_\psi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} W_f(a, b) \psi_{a,b}(t) \frac{da}{a^2} db \quad (2.31)$$

2.3.3 ウェーブレット級数

ウェーブレット変換 $W_f(a, b)$ では係数 a および b を連続値とした. 今, この係数を

$$a = a_0^j, b = kb_0 a_0^j \quad (2.32)$$

と離散化する. a, b を離散的にとったものをウェーブレット級数という. 上式を式 (2.28) に代入すると,

$$\psi_{j,k}(t) = a_0^{-j/2} \psi(a_0^{-jt} - kb_0) \quad (2.33)$$

となる. よって式 (2.29) からウェーブレット変換式は,

$$\begin{aligned} W_f(j, k) &= \langle f(t), \psi_{j,k}(t) \rangle \\ &= a_0^{-j/2} \int_{-\infty}^{\infty} f(t) \bar{\psi}(a_0^{-jt} - kb_0) dt \end{aligned} \quad (2.34)$$

と表される. このとき,

$$\begin{aligned} A \langle f(t), f(t) \rangle &\leq \sum_{j,k} |W_f(j, k)|^2 \leq B \langle f(t), f(t) \rangle \quad (2.35) \\ (0 < A \leq B < \infty) \end{aligned}$$

を満たす A, B をフレームバンドといい, $\{\psi_{j,k}(t)\}$ をフレームという. このとき, 次式により信号 $f(t)$ をウェーブレット変換 $W_f(j, k)$ から再構成できる.

$$f(t) \approx \frac{2}{A+B} \sum_{j,k} W_f(j, k) \psi_{j,k}(t) \quad (2.36)$$

2.3 ウェーブレット変換

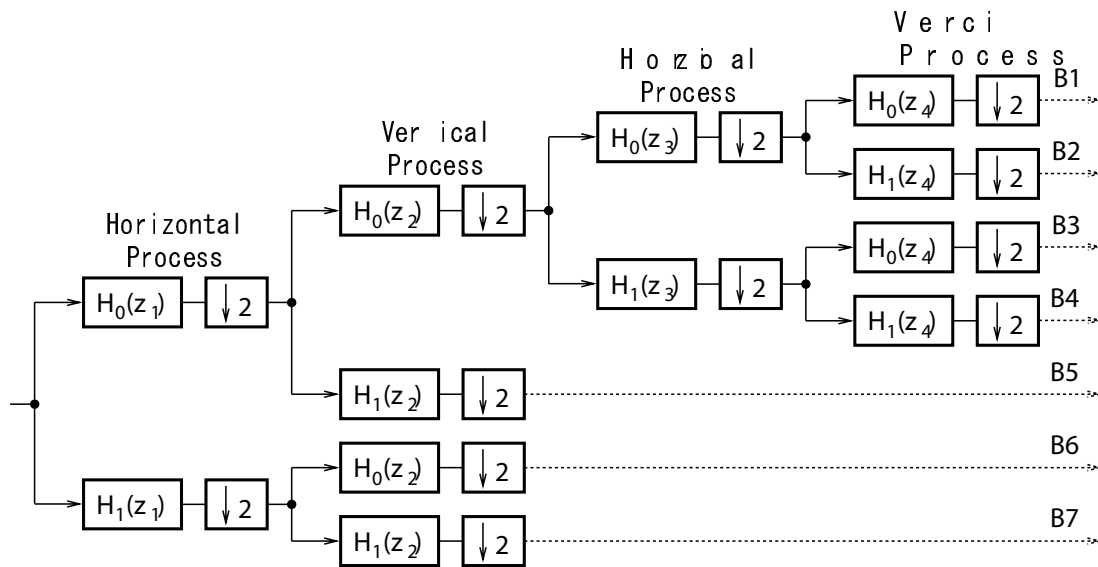


図 2.10 ウェーブレット変換の構成

とくに $A = B$ のとき $\psi_{j,k}(t)$ は正規直交基底となり、信号 $f(t)$ は、

$$f(t) = \sum_{j,k} W_f(j,k) \psi_{j,k}(t) \quad (2.37)$$

により完全に再構成される。また、この場合のフレーム $\{\psi_{j,k}(t)\}$ をタイトフレームという。

正規直交基底を導出する場合には、式 (2.32) において係数 a_0 および b_0 は、 $a_0 = 2, b_0 = 1$ と選ぶことが多い。また、正規直交基底をなさないウェーブレット関数でも信号を再構成するものは数多く知られている。

2.3.4 離散ウェーブレット変換

係数 a および b を式 (2.32) のように離散化し、連続時間信号 $f(t)$ ではなく離散時間信号 $f(n)$ に対するウェーブレット変換を離散ウェーブレット変換という。この変換の 1 つの実現法を図 2.10 に示す。

図 2.10 は、サブバンド符号化で説明した 2 分割フィルタバンクを低域信号に対して階層的にしたものである。つまり、ウェーブレット変換の設計は、2 分割フィルタバンクの設計問題に帰着する。よって式 (2.26), (2.27) の完全再構成条件を満足するフィルタを用い

第2章 デジタル画像処理

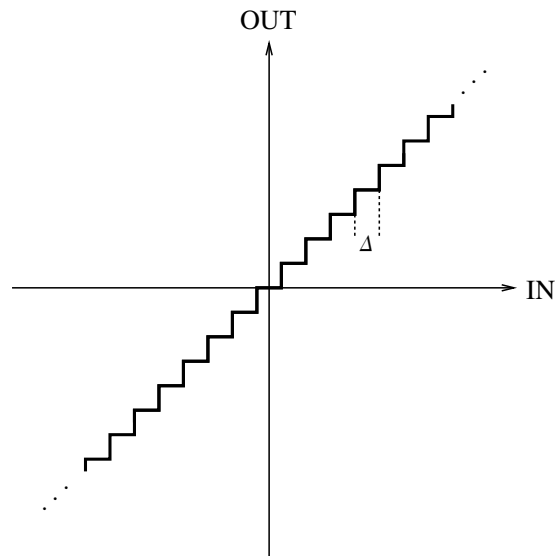


図 2.11 線形量子化

ればよいことになる。

2.4 量子化

ここでは、画像信号処理においてよく用いられる量子化について説明する。量子化とは、入力データを区切り代表値（量子化レベル）を決める処理である。量子化雑音は入力データと代表値の平均的な誤差で計算される。

2.4.1 線形量子化

入力データがそのダイナミックレンジ内で一様に分布しているときには、代表値を等間隔に設定することで量子化雑音が最小になる。これは線形量子化（一様量子化）と呼ばれる。図 2.11 に線形量子化の概念図を示す。

2.4.2 非線形量子化

図 2.12 に非線形量子化の例を示す。図 2.12(a) ではスレッシュヨルド $-d \sim d$ の間にある入力を零にスレッシュヨルド t 以上あるいは $-t$ 以下の入力をそれぞれ $t, -t$ に丸め

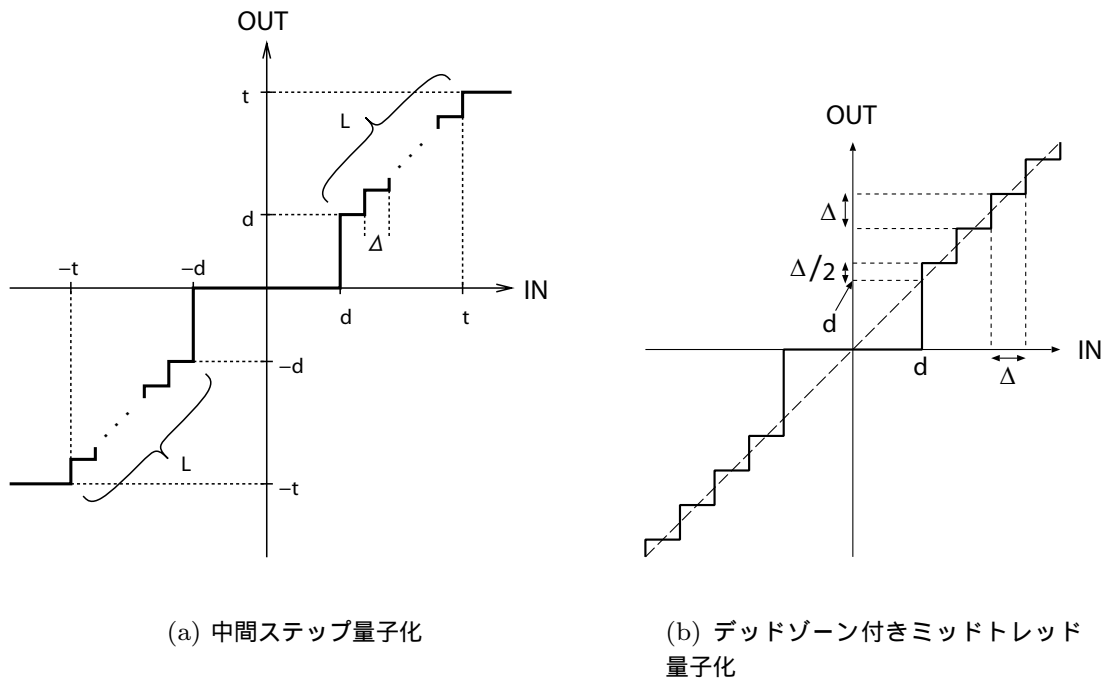


図 2.12 非線形量子化

$d \sim t$ あるいは $-t \sim -d$ の間にある入力に対しては線形量子化する. このような量子化を中間ステップ量子化という.

また, 図 2.12(b) では, 下限スレッシュヨルド d より入力データの絶対値が小さい場合には零に, 下限スレッシュヨルド d より大きな場合には量子化幅 Δ で線形量子化を行う. デッドゾーン付きミッドトレッド量子化により絶対値の大きなデータに重点をおいた量子化ができる.

非線形量子化により平均的な量子化雑音を低減させることができるが, これは統計的にみた最適値にすぎない.

2.5 可変長符号化

2.5.1 エントロピー

画像信号の振幅に関する情報は統計的なものである。ある値 x_i の発生する確率が p_i であり、 x_i のとり得る全レベル数が N とする。この時ある画素のもつ情報量の期待値を H とすると、

$$H = - \sum_{i=1}^N p(x_i) \log_2 p(x_i) \quad (2.38)$$

となる。この H は画像信号の平均情報量を表し、エントロピーと呼ばれる。画像符号化処理後のエントロピーが原信号のエントロピーに比べ小さくなるほど情報削減率が高いことになる。式 (2.38) より分かるようにエントロピーの値は発生確率 p_i の偏りが大きいほど小さく、また x_i の取り得る全レベル数 N が少ないほど小さい。

DCT 符号化、サブバンド符号化およびウェーブレット変換符号化は、画像信号のもつエネルギーを集中化することにより発生確率 p_i の偏りを大きくすることが目的である。また、量子化は x_i の取り得る全レベル数を少なくすることを目的とする処理である。

2.5.2 可変長符号化の原理

可変長符号化は量子化出力に対して施される。非常に頻繁に発生する信号に対しては短い符号を割り当て、たまにしか発生しない符号に対しては長い符号を割り当てることにより、平均符号長を短くすることができる。入力される信号に対して最短の平均符号長を割り当てるのが望ましくそのような符号をコンパクト符号という。ハフマン符号もコンパクト符号の一つである。

2.5.3 ランレングス符号化

同じシンボルが連続して発生する場合に効率の良い符号化である。2 値に対するランレングス符号化の一つに MH 符号化がある。この方式による符号の構成は次式のように表

される。

$$RL = 64 \times M + T \quad (2.39)$$

ここで、 RL はラン長、64 の整数倍である $64 \times M$ を表す符号をメイクアップ符号、0 から 63 までの長さ T を表す符号をターミネーティング符号という。この符号を用いることにより、“0”あるいは“1”が連続するビット数だけ同じ符号を続けて伝送するのに比べ、 M と T を各 1 ワード伝送する方がラン長が長いほど効率が良くなる。

また 2 値ではなく多値におけるランレングス符号では一般に、ランが大きいときにはひき続くレベルの値は小さいことが多い。またランが小さいときには大きいレベルが現われることが多い。そこで零係数のランと非零係数のレベルを組み合わせた情報源に対して可変長符号を割り当てる。これは 2 次元可変長符号化と呼ばれている。

2.6 カラー画像成分と色変換

テレビジョンやディスプレイでは赤 (R)、緑 (G)、青 (B) の 3 原色の混色により色を表現している。そのため、カラー画像は RGB の 3 種類の信号成分を合成することで表現できる。図 2.13(a) に RGB それぞれ 256 階調 (最大表現色数約 1677 万色以上) のカラー画像を示し、図 2.13(b) ~ (d) に RGB 成分のみの画像を示す。カラー画像は RGB の 3 成分の信号により構成される。グレースケールでは白から黒の輝度情報だけであるが、カラー画像を表示するには RGB 3 成分の情報が必要となるため、RGB 成分をそのまま符号化した場合にはグレースケールの約 3 倍のデータになってしまう。この問題を解消するため、RGB 信号を明るさの情報を表す輝度信号と 2 種類の色差信号に変換する。これを色変換という。色変換により、輝度信号に情報を集中させ、色差信号の情報は少なくなり、符号化した場合には RGB で符号化した時よりもデータ量を削減できる。

静止画像圧縮において広く用いられている JPEG では RGB から輝度信号 Y と 2 種類の色差信号 C_r 、 C_b に色変換して処理を行い、符号化する。 YC_rC_b への変換式は式 (2.40)

第2章 デジタル画像処理



(a) plane 画像



(b) R 成分のみの画像



(c) G 成分のみの画像



(d) B 成分のみの画像

図 2.13 カラー画像と RGB 成分

で定義される.

$$\begin{aligned}
 \begin{pmatrix} Y \\ C_r \\ C_b \end{pmatrix} &= \begin{pmatrix} 0.299R + 0.587G + 0.114B \\ 0.713 \times (R - Y) \\ 0.564 \times (B - Y) \end{pmatrix} \\
 &= \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ 0.500 & -0.419 & -0.081 \\ -0.169 & -0.331 & 0.500 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}
 \end{aligned} \tag{2.40}$$

2.6 カラー画像成分と色変換



(a) Y 成分の画像



(b) C_r 成分の画像



(c) C_b 成分の画像

図 2.14 図 2.13(a) の YC_rC_b 成分

図 2.13(a) の画像信号を YC_rC_b に変換した時の輝度信号 Y を図 2.14(a) に、色差信号 C_r 、 C_b を図 2.14(b), (c) にそれぞれ示す。図 2.14(a) ~ (c) では、白い部分ほど情報が多く、黒い部分ほど情報が少ない。そのため、輝度信号に情報が集中し、色差信号の情報が少なくなっていることが確認できる。つまり、RGB 信号をそのまま符号化した場合よりも YC_rC_b に色変換してから符号化した方がデータ量を削減できる。また、人間の視覚特性は

第2章 デジタル画像処理

色差に対する感度が悪く、輝度に対する感度が良いことが知られている。そのため、色差信号を輝度信号よりも高圧縮することにより視覚的な劣化を少なくし、データ量の削減することができる。

2.7 評価指標

本論文で用いる評価指標を以下に示す。

2.7.1 客観評価

PSNR

画質の客観評価指標には、PSNR (Peak Signal to Noise Ratio) が広く用いられている [22]。PSNR は 2 枚の画像の差を定量化した指標である。画像サイズを $X \times Y$ pixel, 違いを測定する 2 枚の画像の画素値をそれぞれ $f_1(x, y), f_2(x, y)$ とすると、平均 2 乗誤差 (MSE: Mean Squared Error) と最大画素値 S_{\max} より、次式で表される。

$$\text{MSE} = \frac{1}{XY} \sum_{x=1}^X \sum_{y=1}^Y \{f_1(x, y) - f_2(x, y)\}^2 \quad (2.41)$$

$$\text{PSNR} = 10 \log_{10} \left(\frac{S_{\max}^2}{\text{MSE}} \right) \quad (2.42)$$

また、カラー画像の PSNR は RGB の各成分毎に MSE を求め RGB の平均 MSE より式 (2.42) で求められる。例えば、 $f_1(x, y)$ を原画像、 $f_2(x, y)$ を復号画像とすると、原画像と復号画像の違いが数値として現れ、復号画像の画質評価を行うことができる。PSNR の値が高いほど 2 枚の画像の差が小さいことを示す。

GBIM

PSNR は簡単に客観評価を行うことができるが、実際の画像の視覚的評価とは必ずしも一致しない。そこでブロックひずみを定量化した指標である GBIM (Generalized Block-edge Impariment Metric) が提案されている [23], [24]。GBIM は隣接する 2 ブロックを基本単位とし、各ブロック内に含まれるブロックひずみ量を定量化する。GBIM

は画像中に含まれるブロックひずみのサイズが 8×8 点で発生するとして定義されている。しかし、本論文第 4 章において得られる拡大画像のブロックひずみのサイズは拡大率に応じて変化する。そこで第 4 章では、 $B \times B$ 点で発生するブロックひずみ量を測定できるよう GBIM を変更した GBIM_B を用いる。 GBIM_B は次式より求められる。

$$\text{GBIM}_B = \frac{(M_h + M_v)}{2} \quad (2.43)$$

M_v は画像全体の大きさを $BN_I \times BN_J$ 画素として次のように表される。

$$M_v = \frac{\sqrt{\sum_{i=0}^{N_I-2} \sum_{j=0}^{N_J-1} \sum_{l=0}^{B-1} \omega_{i,j}^2(l) \{f_{i,j}(B-1, l) - f_{i+1,j}(0, l)\}^2}}{E} \quad (2.44)$$

ここで $f_{i,j}(k, l)$ は j 行 i 列目のブロックの l 行 k 列目の画素値を表す。また、

$$E = \frac{1}{B-1} \sum_{k=0}^{B-2} \sqrt{\sum_{i=1}^{N_I-1} \sum_{j=0}^{N_J-1} \sum_{l=0}^{B-1} \omega_{i-1,j}^2(l) \{f_{i,j}(k, l) - f_{i,j}(k+1, l)\}^2} \quad (2.45)$$

$$\omega_{i,j}(l) = \begin{cases} \lambda \ln \left(1 + \frac{\sqrt{\mu_{i,j}(l)}}{1 + \sigma_{i,j}(l)} \right) & (\mu_{i,j}(l) \leq \zeta) \\ \ln \left(1 + \frac{\sqrt{255 - \mu_{i,j}(l)}}{1 + \sigma_{i,j}(l)} \right) & (\mu_{i,j}(l) > \zeta) \end{cases} \quad (2.46)$$

$$\lambda = \frac{\ln(1 + \sqrt{255 - \zeta})}{\ln(1 + \sqrt{\zeta})} \quad (2.47)$$

である。ただし

$$\mu_{i,j}(l) = \frac{\mu_{i,j}^L(l) + \mu_{i,j}^R(l)}{2} \quad (2.48)$$

$$\mu_{i,j}^L(l) = \frac{1}{B} \sum_{k=0}^{B-1} f_{i,j}(k, l)$$

$$\mu_{i,j}^R(l) = \frac{1}{B} \sum_{k=0}^{B-1} f_{i+1,j}(k, l)$$

第2章 デジタル画像処理

$$\begin{aligned}\sigma_{i,j}(l) &= \frac{\sigma_{i,j}^L(l) + \sigma_{i,j}^R(l)}{2} \\ \sigma_{i,j}^L(l) &= \sqrt{\frac{1}{B} \sum_{k=0}^{B-1} \{f_{i,j}(k, l) - \mu_{i,j}^L(l)\}^2} \\ \sigma_{i,j}^R(l) &= \sqrt{\frac{1}{B} \sum_{k=0}^{B-1} \{f_{i+1,j}(k, l) - \mu_{i,j}^R(l)\}^2}\end{aligned}\tag{2.49}$$

である。ここで、定数 ζ はブロック境界画素値の差が最大となる位置で重み $\omega_{i,j}$ が最大となる値を設定する。また、式 (2.43) の M_h は水平方向で M_v と同様の計算をすることにより導出できる。GBIM は値が小さいほど含まれるブロックひずみ量が少ないことを示す。また、カラー画像の GBIM は RGB の各成分毎の GBIM 値の平均より求められる。本論文では 2 倍の拡大つまり、 $B = 16$ とした実験を行う。

2.7.2 主観評価

PSNR および GBIM による客観評価に加え、解像度変換画像を視覚的に評価を行う。PSNR が高い画像が必ずしも画質が良好であるとは限らない。そのため、本研究では主観評価による画質評価を行う。

対比較試験

画像の主観評価として対比較試験を行う。被験者に対し従来法と提案法の 2 枚の拡大画像を同時に見せ、その優劣を判断してもらう。本実験で用いる対比較試験の実験条件を表 2.1 に示す。本実験では画像全体の比較に加え、拡大画像から 256×256 pixel を 4 ヶ所切り出し、画像の一部の比較を行う。

表 2.1 対比較試験の実験条件

使用ディスプレイ	19 inch CRT カラーモニタ (EIZO FlexScan T761)
ディスプレイからの距離	画像の高さ $\times 4$
部屋の明るさ	やや暗い (ディスプレイ上部の照明を消灯)
実験画像	1 画像につき 5 種類 (画像全体 1 枚 + 画像の一部 4 枚)
画像表示時間	10 秒/枚

第 3 章

高圧縮符号化技術

ウェーブレット変換符号化における問題点は, DCT 符号化と比べ, 主に処理速度および符号化効率にある. そこで本章では, これらの問題点を解決するための一手法について述べる.

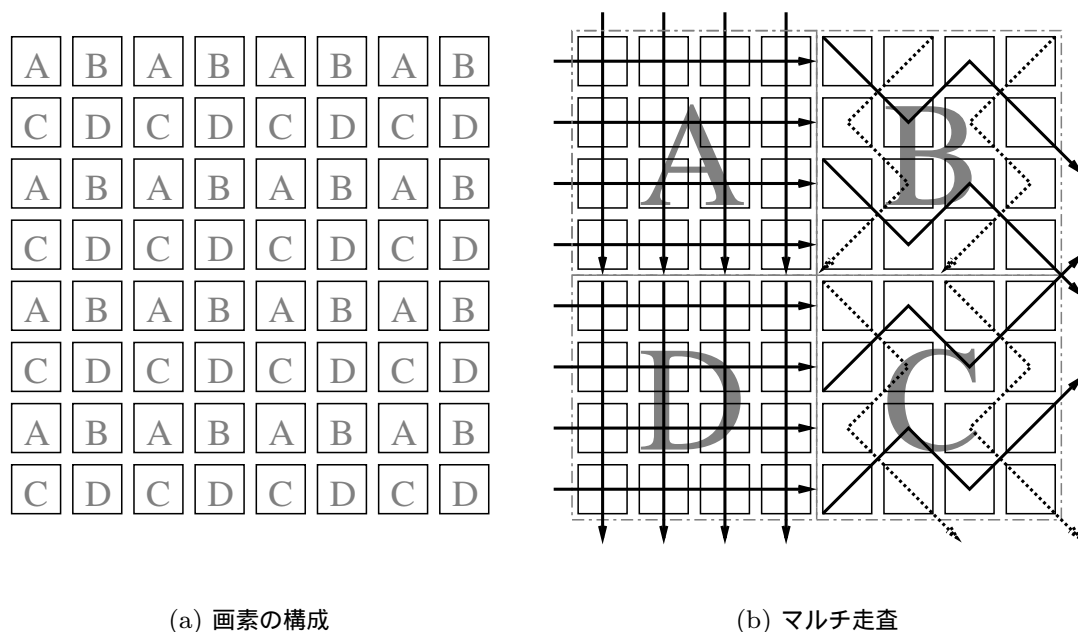


図 3.1 マルチ走査の概念図

3.1 マルチ走査によるウェーブレット変換符号化

3.1.1 マルチ走査

本研究で提案したマルチ走査の概念図を図 3.1 に示す。マルチ走査では、まず図 3.1 (a) のように原画像に対して一画素毎スキップした画素を領域 A ～ D にグループ分けする。ここで、領域 A ～ D のうち従来の走査と斜め走査との組合せは複数考えられるが、もし A・B あるいは A・C に対して従来走査を行ったとすると、A・B の組合せでは一画素毎に水平方向に走査することに相当し、A・C の組合せでは一画素毎に垂直方向に走査することと同等である。つまり水平あるいは垂直方向に対する画像の再現性のみが増すことになる。そこで図 3.1 (b) のように領域 A・D に対しては従来走査を適用し、領域 B・C に対しては斜め走査を適用する。同様に、領域 B・C においても一画素毎に右下あるいは右上方向に走査しないように同図 (b) 中では、領域 B の水平 (あるいは垂直) 走査と領域 C の水平 (あるいは垂直) 走査の方向を変えている。

3.1 マルチ走査によるウェーブレット変換符号化

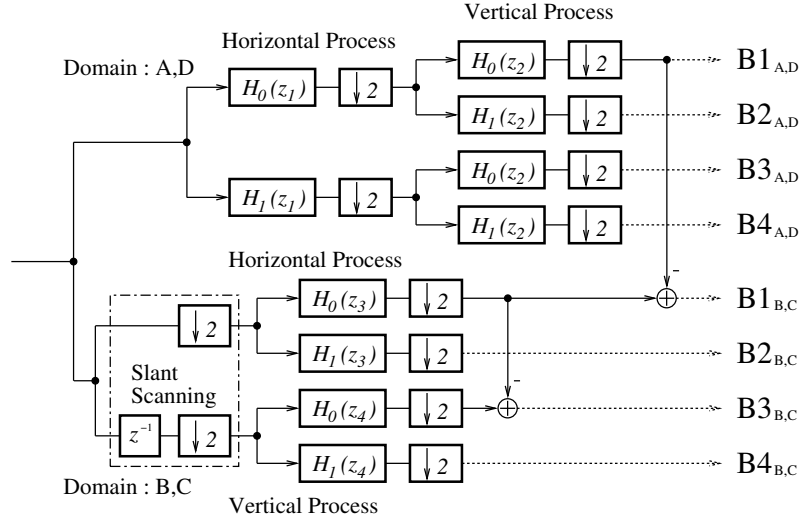


図 3.2 低域信号の差分を考慮したマルチ走査の構成

このように走査することにより、従来の走査と斜め走査の特徴をあわせもつこととなり、水平・垂直・斜め方向のエッジに対する再現性も向上し、復号画像の画質が良好なものになると考えられる。

また、マルチ走査においても斜め走査同様、低域信号の重複が考えられるが、図 3.2 に示すように低域成分の差分により解決できる。処理ステップ総数は、低域成分の差分を考慮すると従来の走査方法に比べ約 0.84 倍となる。

3.1.2 量子化器

斜め走査およびマルチ走査が復号画像の画質に及ぼす影響を調べるためウェーブレット変換係数を帯域成分毎に量子化する。一般に自然画像の低域成分の振幅値は一様に分布しており、高域成分にはエッジ信号の他に画像ノイズと呼ばれる雑音が混入していることが知られている。そこで、低域成分に対しては、図 3.3 (a) の線形量子化を用いて、高域成分に対しては同図 (b) の中間ステップ量子化 [25] を用いて量子化を行う。

第 3 章 高圧縮符号化技術

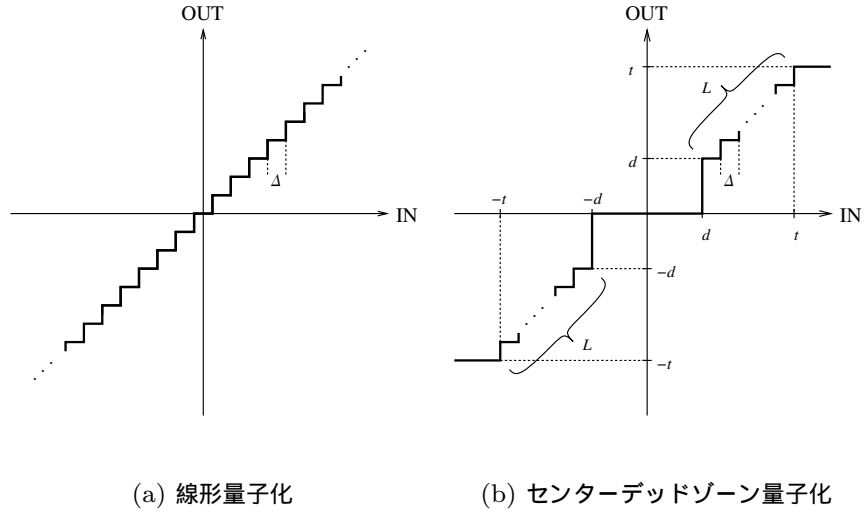


図 3.3 量子化

3.1.3 シミュレーション

本研究で提案した斜め走査およびマルチ走査の有効性を示すために、提案走査を適用したウェーブレット変換符号化器及び復号化器を設計し、復号画像の画質に及ぼす影響について評価を行う。実験画像として、ここでは縦・横 512×512 ピクセル、256 階調、グレイスケールの“Lena”画像を用いてウェーブレット変換を 2 階層行った場合について示す。

また、本研究でフィルタバンクに用いたローパスフィルタおよびハイパスフィルタの伝達関数を式 (3.1) に示す。

$$\begin{cases} H_0(z) = \frac{1 + z^{-1}}{\sqrt{2}} \\ H_1(z) = \frac{1 - z^{-1}}{\sqrt{2}} \end{cases} \quad (3.1)$$

これらの式はハール基底と呼ばれ、直線位相特性を有する。これらのフィルタの利得は 3 [dB] であるため、フィルタの出力信号の振幅値は、入力信号に対して $\sqrt{2}$ 倍となる。また、シミュレーションにおいて用いた図 3.3 に示す量子化器の各係数は、文献 [25] のものを用いている。ただし、斜め走査およびマルチ走査における斜め走査を適用している部分 (図 3.2 中の領域 B および C) においては、フィルタを通過する回数が従来走査のものと異な

3.1 マルチ走査によるウェーブレット変換符号化



(a) Lenna



(b) 拡大画像

図 3.4 原画像

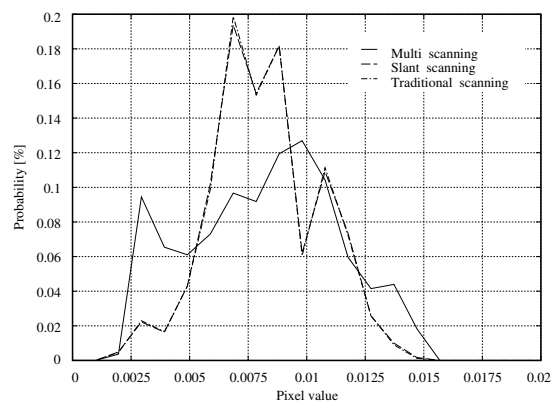


図 3.5 変換係数の分布

るためフィルタによる利得分だけ量子化係数を調整している。

3.1.4 ウェーブレット変換係数の分布

ウェーブレット変換に用いているフィルタの利得によってそれぞれの走査方法において出力されるウェーブレット変換係数の最大値は異なるため、各走査方法において量子化係数を変えており、ウェーブレット変換係数に対する量子化が効率良く行うことができているかを確認する必要がある。そこで、各走査方法におけるウェーブレット変換係数の量子化後の係数分布の比較を行った。図 3.5 に各走査方法における低域成分のウェーブレット

第3章 高圧縮符号化技術



(a) 従来法 (30.12 [dB])



(b) 斜め走査 (29.05 [dB])

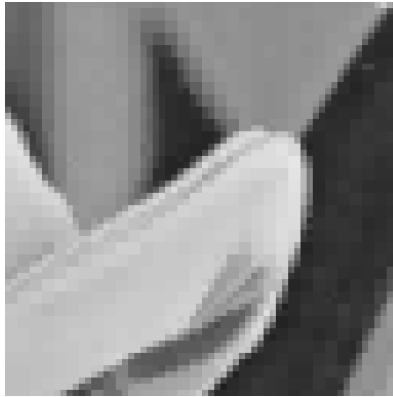


(c) マルチ走査 (29.90 [dB])

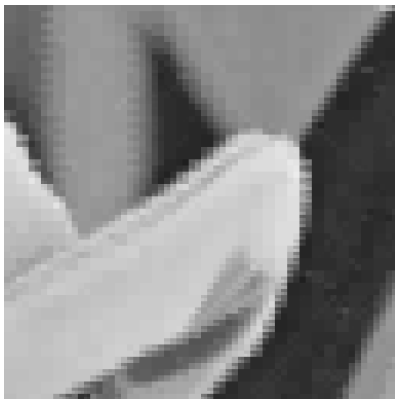
図 3.6 復号画像

変換係数の分布を示す．ここで，図 3.5 の横軸は，量子化後の係数であり，走査方法によってフィルタ処理を行う回数が異なり，変換係数の理論上の最大値が異なるため，それぞれの走査方法での変換係数の理論上の最大値で規格化を行っている．また縦軸は，各成分の出現確率である．図 3.5 より各走査方法の低域成分は，ほぼ同じような分布を示しており，これにより各走査方法での量子化係数が妥当な値であることが確認できる．

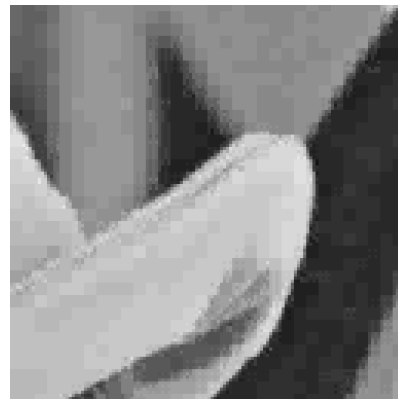
3.1 マルチ走査によるウェーブレット変換符号化



(a) 従来法



(b) 斜め走査



(c) マルチ走査

図 3.7 復号画像の拡大図

3.1.5 復号画像の画質評価

従来の走査方法を適用した場合の復号画像を図 3.6 (a) に, 斜め走査を適用した場合の復号画像を同図 (b) に, マルチ走査を適用した場合の復号画像を同図 (c) に示す. 図 3.6 における平均ビットレートは, 約 1.5 [bit/pixel] である. 図 3.6 より, 従来の走査に比べ斜め走査およびマルチ走査による復元画像の PSNR が若干劣っている理由として, 斜め走査やマルチ走査では低域成分間の差分を行っており, この差分による冗長性の削除が従

第3章 高圧縮符号化技術



(a) 従来法 (27.10 [dB])



(b) 斜め走査 (27.06 [dB])



(c) マルチ走査 (27.14 [dB])

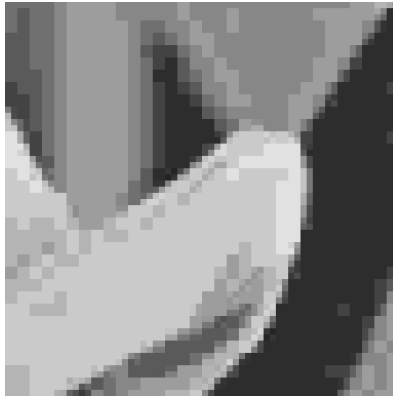
図 3.8 復号画像

来法に比べ完全ではないためと考えられる。

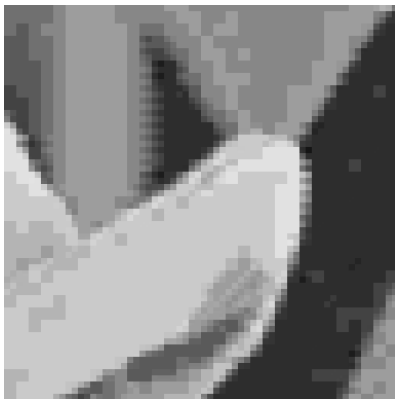
また、走査方法のエッジへの影響を調べるため、それぞれの復号画像においてエッジを多く含む部分の拡大図を図 3.7 (a) ~ (c) に示す。ただし、ここでは各走査方法が画像の細部に与える影響を見るため、復元画像の拡大には零次ホールド法を用いている。

図 3.6 (a) ~ (c) より画像の滑らかな部分 (低域信号部分) に対してはどの走査方法においても同じような結果が得られていることが分かる。次にエッジ付近については、図 3.7 (a) の従来走査では、水平・垂直方向のエッジに対する再現性が良くなっている。それに対

3.1 マルチ走査によるウェーブレット変換符号化



(a) 従来法



(b) 斜め走査



(c) マルチ走査

図 3.9 復号画像の拡大図

し同図 (b) の斜め走査では斜め方向のエッジに対する再現性は向上しているが、水平・垂直方向のエッジに対する再現性はあまり良くない。しかし、同図 (c) のマルチ走査では、水平・垂直・斜め方向のエッジに対する再現性が他の走査方法に比べ、改善されていることが確認できる。つまり、従来走査および斜め走査では、走査方向以外のエッジ付近に低域成分の量子化によるブロック状のひずみが生じているのに対し、マルチ走査ではこのブロック状のひずみが低減されている。

次に、圧縮率を上げた場合においても提案した走査方法の有効性を示すため、圧縮率を

第 3 章 高圧縮符号化技術

上げ、同様の実験を行った。各走査方法における復号画像を図 3.8 に、拡大図を図 3.9 に示す。図 3.8 における平均ビットレートは、約 1.0 [bit/pixel] である。

図 3.8 より、先程同様滑らかな部分に対しては各走査方法ともほぼ同等の画質が得られていることが確認できる。また、図 3.9 より、エッジ成分は、マルチ走査を用いた復号画像が他の走査手法による復号画像より画質が優れていることが確認できる。

3.1.6 むすび

本研究では画像の再現性および復号画質改善の観点から各画素を重複することなく走査できる斜め走査によるフィルタバンク符号化を提案し、提案した斜め走査を適用したウェーブレット変換符号化器の設計を行った。ウェーブレット変換には 2 タップのハール基底を用いてフィルタバンクで構成した。

斜め走査では、斜め方向のエッジに対する画像の再現性は良いが、水平および垂直方向のエッジに対する画像の再現性が良くなかった。

そこで両者の特徴をあわせもつ手法としてマルチ走査を提案し、その動作の確認を行った。マルチ走査を適用したウェーブレット変換の復号画像は、水平・垂直および斜め方向のエッジ成分を再現できることが確認できた。またエッジ付近によく見られるブロック状のひずみが低減されており、従来走査および斜め走査に比べて視覚的に良好な画像が得られることが確認できた。

さらに処理ステップ数は、斜め走査では従来の走査方法に比べ 0.6 倍となり、マルチ走査では 0.84 倍で処理できることが確認できた。

ところで本論文で提案した斜め走査およびマルチ走査は、ウェーブレット変換を行う際のフィルタリング処理における走査方向を変更することで実現できるため、近年提案されているウェーブレット変換による画像符号化方式 (Embedded Zerotree Wavelet : EZW) [26]–[28] においても適用可能である。その場合、図 3.2 において領域 A・D は従来走査を行っているため、EZW をそのまま適用可能であるのに対し、領域 B・C は、斜め走査を

3.2 ウェーブレット – DCT 符号化

行っており、さらに低域成分の差分を行っているため、EZW での親と子の関係が成り立たなくなる。しかし、領域 $B \cdot C$ の各画素は、領域 $A \cdot D$ の隣接画素であるため、

非常に強い相関があり、領域 $A \cdot D$ の低域成分を親、領域 $B \cdot C$ を子とみなし、EZW を構成することで EZW の効果が期待できる。

本論文では、フィルタバンクにハール基底を用いてウェーブレット変換を行っており、マルチ走査によるウェーブレット変換係数は、各階層での帯域の縦・横の大きさが等しく出力されるため問題とはならない。しかし、より高次の基底をフィルタバンクに用いた場合、各階層での帯域の縦・横の大きさが等しくならず、そのまま低域成分間の差分をとることができない。

そこで今後の課題としてマルチ走査の EZW への適用による符号化効率の改善およびフィルタバンクに用いるフィルタの高次化等が挙げられる。

3.2 ウェーブレット – DCT 符号化

3.2.1 ウェーブレット – DCT

本研究で提案するウェーブレット – DCT の基本構成を図 3.10 に示す。ウェーブレット – DCT では図 3.10 のように、まず原画像の各画素を水平方向に走査し、ローパスフィルタ ($H_0(z)$) およびハイパスフィルタ ($H_1(z)$) によってフィルタリングした後、 $1/2$ ダウンサンプリングを行う。次にその出力信号を垂直方向に走査し、水平方向と同様の処理を行う。以上のような 2 分割フィルタバンクを用いてウェーブレット変換を行い、各帯域 ($B1 \sim B7$) に周波数分割を行う。そして各帯域ごとに DCT を行うことで変換係数を得る。

ブロックサイズ

ウェーブレット変換係数に対して DCT を行う際、DCT のブロックサイズを考慮する必要がある。図 3.10 において 2 階層のウェーブレット変換係数である帯域 $B1 \sim B4$ での 1 画素は、復号の際、縦・横それぞれ 2 回ずつアップサンプリングをするため、復号画

第 3 章 高圧縮符号化技術

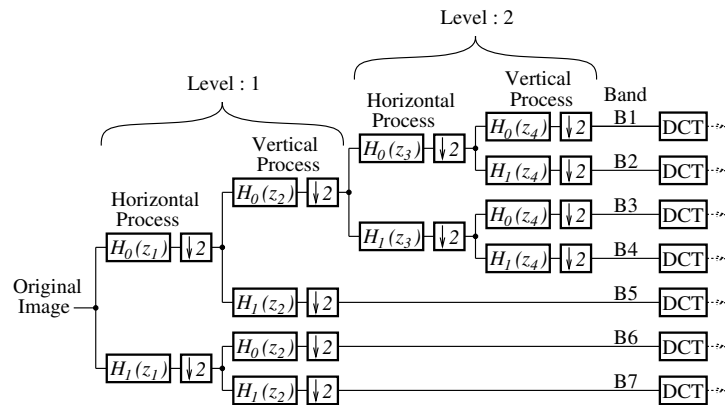


図 3.10 ウェーブレット - DCT の構成

表 3.1 帯域とブロックサイズの関係

条件	帯域 B1 ~ B4	帯域 B5 ~ B7
(1)	4×4	8×8
(2)	8×8	8×8
(3)	8×8	16×16
(4)	8×8	4×4
(5)	4×4	4×4
(6)	16×16	16×16

像の 4×4 画素に対応している。また、1 階層のウェーブレット変換係数である帯域 B5 ~ B7 での 1 画素は、復号の際、縦・横それぞれ 1 回づつアップサンプリングをするため、復号画像の 2×2 画素に対応している。つまり、ウェーブレット変換の階層数に応じて復号画像に与える影響が異なるため、DCT のブロックサイズを変更することで復元画像の画質を改善することができる。そこで本報告では、各帯域とブロックサイズを表 3.1 に示すように設定し、それぞれの条件における復号画像に現れるブロック歪みの影響を比較することで帯域とブロックサイズの関係を決める。表 3.1 において条件 (1), (3) は、帯域 B1 ~ B4 と帯域 B5 ~ B7 のブロックサイズの比が 1:4 となっており、復号後の画像に与える影響が等しくなる場合である。また、条件 (2), (5), (6) は、各帯域とも同じブロックサイズに設定した場合であり、条件 (4) は、条件 (1) とは逆に設定した場合である。

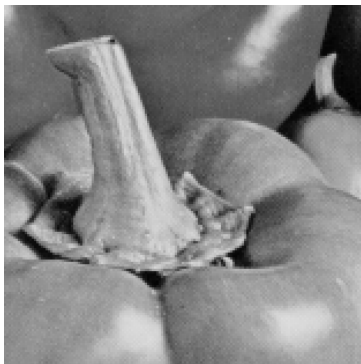
3.2 ウェーブレット – DCT 符号化



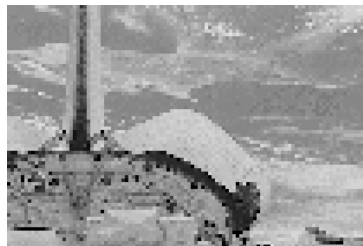
(a) Lenna



(b) Barbara



(c) peppers



(d) shuttle

図 3.11 原画像の拡大図

3.2.2 実験結果

本研究で提案したウェーブレット – DCT の動作を確認するため C 言語によるシミュレーションを行った。実験画像には、256 階調、グレースケールの Lenna, Barbara, peppers, shuttle 画像を用いた。Lenna, Barbara, peppers 画像は、 512×512 [pixel] であり、shuttle は 384×256 [pixel] である。原画像の一部分の拡大図を図 3.11 に示す。また、ウェーブレット変換基底には、ハール基底を用い、量子化係数は図 3.12 のものを用いた。さらに、符号化および復号化は JPEG の符号化アルゴリズムを用いた。

第 3 章 高圧縮符号化技術

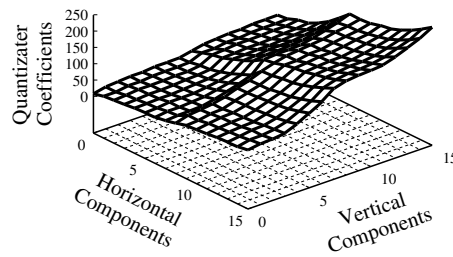


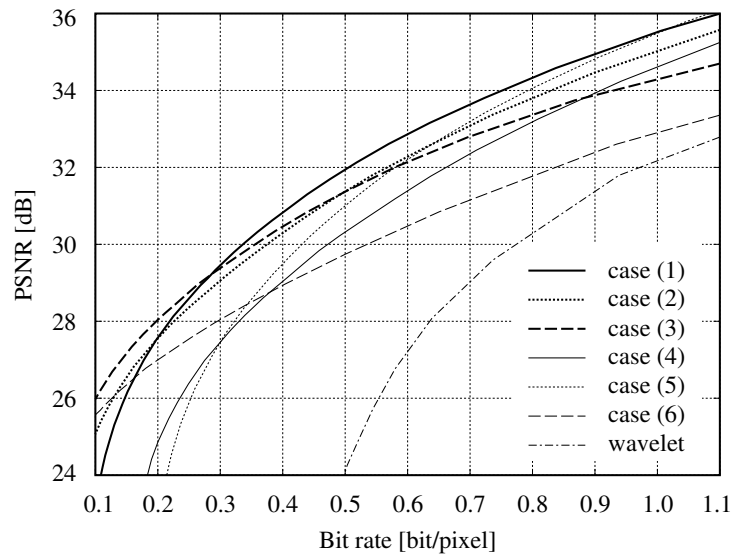
図 3.12 量子化係数

帯域とブロックサイズの客観評価

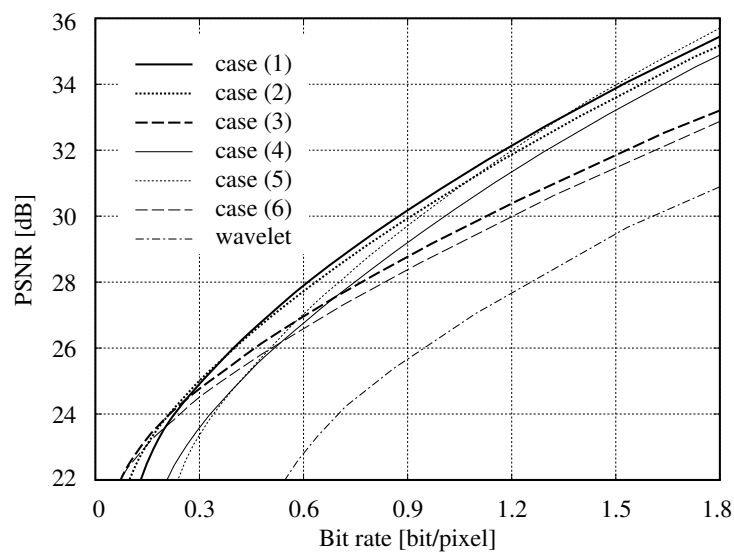
表 3.1 のように各帯域においてブロックサイズを変更することによる復号画像への影響を調べるため、各条件におけるビットレートとピーク SN 比の関係を調べた。実験結果を図 3.13 に示す。ここで、各帯域によってブロックサイズを変更しているため、量子化係数のブロックサイズも各帯域によって変更する必要がある。そこで本実験では、ブロックサイズ 4×4 , 8×8 , 16×16 の量子化には、図 3.12 の水平・垂直 $0 \sim 3$, $0 \sim 7$, $0 \sim 15$ の量子化係数を基本とし、これらの係数に重み係数を乗算することでビットレートを変更した。

図 3.13 中の (1) ～ (6) は、表 3.1 の条件である。また、比較のためウェーブレット変換のみを用いた場合の実験結果もあわせて示す。図 3.13 (a) より Lenna 画像においては、ビットレートが 0.3 [bit/pixel] 以上では、条件 (1) が最も良い結果となっており、 0.25 [bit/pixel] 以下では条件 (3) が最も良い結果となっている。また、同図 (b) より Barbara 画像においては、 0.4 [bit/pixel] 以上では、条件 (1) が最良となっており、 0.2 [bit/pixel] 以上 0.4 [bit/pixel] 未満では条件 (2) が 0.2 [bit/pixel] 未満では条件 (3) が良い結果となった。同図 (c) より peppers 画像においては、ビットレートが 0.25 [bit/pixel] 以上 0.85 [bit/pixel] 以下では、条件 (1) が最も良い結果となっており、 0.25 [bit/pixel] 以下では、条件 (3) が最も良い結果となっている。さらに同図 (d) より shuttle 画像においては、ビットレートが 0.7 [bit/pixel] 以上では、条件 (5) が最も良い結果となっており、 0.7 [bit/pixel] 以下では、条件 (1) が最も良い結果となっている。

3.2 ウェーブレット – DCT 符号化



(a) Lenna



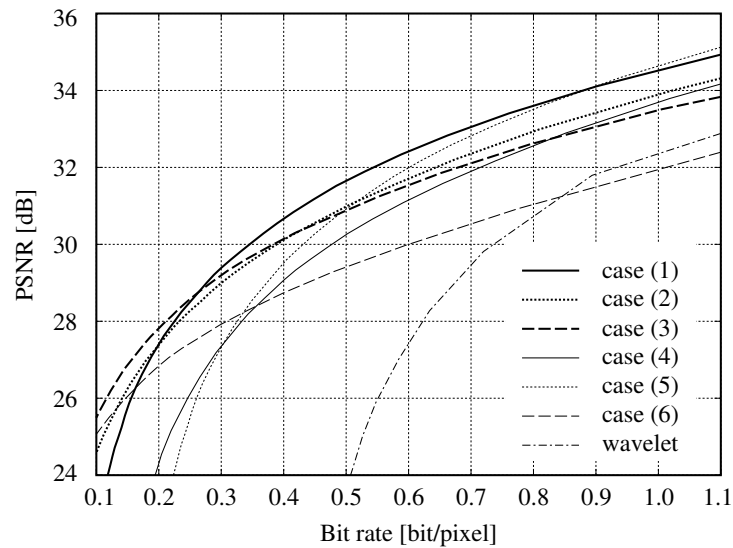
(b) Barbara

図 3.13 ビットレートとピーク SN 比の関係

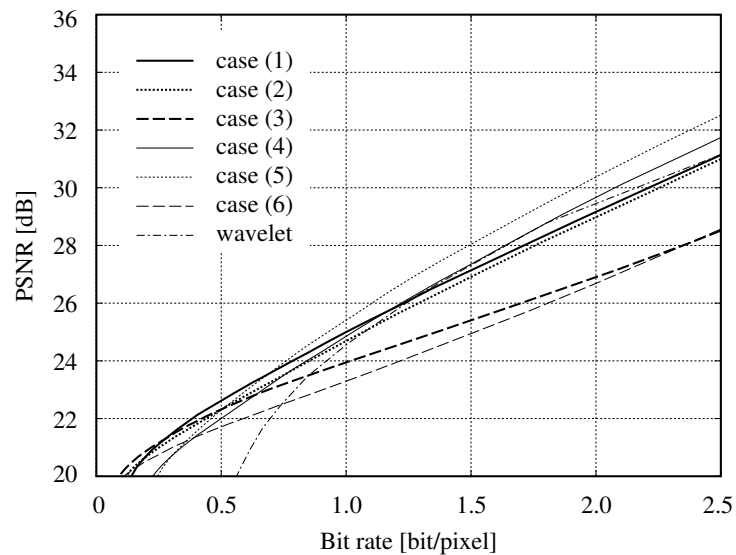
図 3.13 において 0.3 [bit/pixel] 以下ではいずれの場合でも条件 (1) の場合の PSNR が急激に悪くなっている。これは、量子化後の DCT 係数において低域成分以外はすべて 0 となっており、変換ブロックが小さいほど符号化効率が悪くなるためである。

また、図 3.13 において従来のウェーブレット変換より圧縮率が改善されていることが

第 3 章 高圧縮符号化技術



(c) peppers



(d) shuttle

図 3.13 ビットレートとピーク SN 比の関係

確認できる.

以上より, 客観評価であるビットレートと PSNR との関係において, 圧縮率が比較的低い時には, 帯域 B1 ~ B4 はブロックサイズを 4×4 , 帯域 B5 ~ B7 は 8×8 とした場合が最も効率良く符号化できており, 圧縮率が高くなるにつれ, 帯域 B1 ~ B4 はブロックサイズを 8×8 , 帯域 B5 ~ B7 は 16×16 とした場合が最も効率良く符号化できている

ことが確認できる。

帯域とブロックサイズの主観評価

次に、提案した変換手法において表 3.1 の各条件における復号画像への視覚的影響を確認するため、ビットレートを固定した時の各条件での復号画像の比較を行う。Lenna 画像においてはビットレートを約 0.40 [bit/pixel] とした場合、Barbara 画像においては約 0.60 [bit/pixel] とした場合、peppers 画像においては約 0.30 [bit/pixel] とした場合、shuttle 画像においては約 1.00 [bit/pixel] とした場合のそれぞれの条件における復号画像を図 3.14 ~ 3.17 に示す。図 3.14 より、Lenna 画像においては条件により若干の変化はあるもののピーク SN 比は、約 30 [dB] となった。同図 (a), (c) より、DCT でのブロックサイズを大きくしたことによるモスキート雑音の広がりが大きくなっていることが確認できる。また、同図 (b), (d) においてもモスキート雑音の影響により視覚的にちらついた画像となっている。同図 (e) においては、ブロック歪みが確認できる。さらに同図 (f) は、ブロックサイズが大きいため、エッジ部分がぼやけてしまっている。

以上より、Lenna 画像を用いた実験においては、復号画像でのブロックサイズの影響を等しくするようにレベル 2 とレベル 1 でのブロックサイズをそれぞれ 4×4 , 8×8 とした条件 (1) 場合の復号画像の画質が最も良くなることが確認できる。

図 3.15 より、Barbara 画像ではピーク SN 比は、約 27 [dB] となった。同図 (d), (e) では顕著にブロック歪みが、(b) では若干のブロック歪みが現れており、復号画像の画質が劣化している。また、同図 (b), (c), (d) には、顎の部分にモスキート雑音が現れており、ちらついた画像となっている。さらに、同図 (f) はエッジ部分がぼやけており、復号画像が劣化している。同図 (a) においては、ブロック歪み・モスキート雑音ともに発生しておらず、視覚的に良好な画像となっている。

以上より、Barbara 画像においては、条件 (1) での復号画像が視覚的に良好な画像であることが確認できる。

図 3.16 より、peppers 画像ではピーク SN 比は、約 27 ~ 29 [dB] となった。同図 (b),

第3章 高圧縮符号化技術



(a) 条件 (1) : 30.83 [dB]



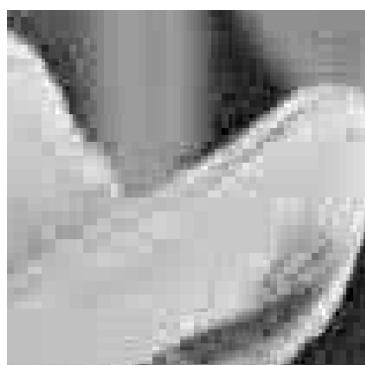
(b) 条件 (2) : 30.31 [dB]



(c) 条件 (3) : 30.49 [dB]



(d) 条件 (4) : 29.07 [dB]



(e) 条件 (5) : 29.05 [dB]



(f) 条件 (6) : 28.97 [dB]

図 3.14 復号画像の拡大図 (Lenna)

(d), (e) には、顕著にブロック歪みが現れており、復号画像の画質が劣化している。また、同図 (f) はエッジ部分がぼやけている上、モスキート雑音も混入しており、ちらついた画像となっている。さらに同図 (a), (c) は、ピーク SN 比もほぼ同等となっており、視覚的にも良好な画像となっている。

3.2 ウェーブレット – DCT 符号化



(a) 条件 (1) : 27.90 [dB]



(b) 条件 (2) : 27.74 [dB]



(c) 条件 (3) : 26.96 [dB]



(d) 条件 (4) : 26.75 [dB]



(e) 条件 (5) : 27.02 [dB]



(f) 条件 (6) : 26.59 [dB]

図 3.15 復号画像の拡大図 (Barbara)

以上より, peppers 画像においては, 条件 (1) および (3) での復号画像が視覚的に良好な画像であることが確認できる.

図 3.17 より, shuttle 画像ではピーク SN 比は, 約 23 ~ 25 [dB] となり, どの条件においてもブロック歪みは確認できない. また, 同図 (d), (e) には若干, 同図 (f) には顕著に

第3章 高圧縮符号化技術



(a) 条件 (1) : 29.40 [dB]



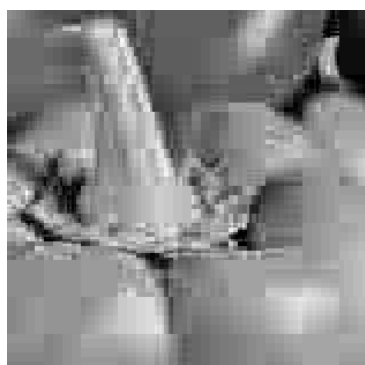
(b) 条件 (2) : 29.00 [dB]



(c) 条件 (3) : 29.21 [dB]



(d) 条件 (4) : 27.37 [dB]



(e) 条件 (5) : 27.33 [dB]



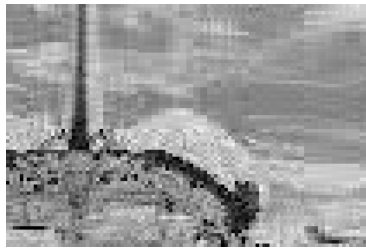
(f) 条件 (6) : 27.93 [dB]

図 3.16 復号画像の拡大図 (peppers)

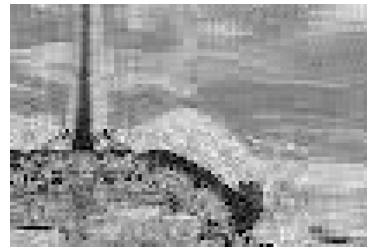
モスキート雑音が現れており、ちらついた画像となっている。さらに同図 (a), (b), (c) は、ピーク SN 比に若干の差があるものの視覚的には、ほぼ同等の復元画像となっている。

以上より, shuttle 画像においては, 条件 (1), (2), (3) での復号画像が視覚的に良好な画像であることが確認できる。

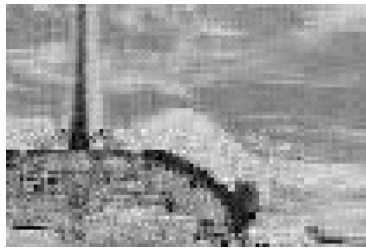
3.2 ウェーブレット – DCT 符号化



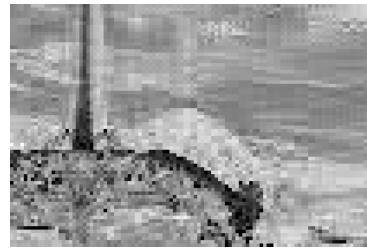
(a) 条件 (1) : 25.01 [dB]



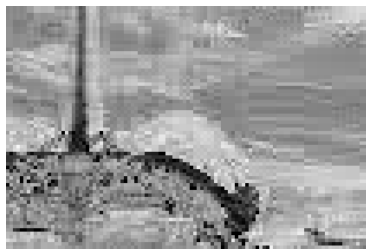
(b) 条件 (2) : 24.68 [dB]



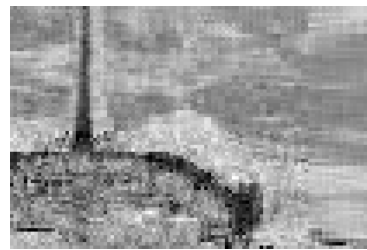
(c) 条件 (3) : 23.95 [dB]



(d) 条件 (4) : 24.86 [dB]



(e) 条件 (5) : 25.41 [dB]



(f) 条件 (6) : 23.31 [dB]

図 3.17 復号画像の拡大図 (shuttle)

以上の考察より、主観評価では、帯域 B1 ~ B4 でのブロックサイズを 4×4 とし、帯域 B5 ~ B7 でのブロックサイズを 8×8 とした条件 (1) の場合が最良の結果を示しており、視覚的に良好な画像を生成できることが確認できた。

3.2.3 むすび

本節では、ウェーブレット変換係数に対して DCT を行うウェーブレット – DCT 符号化を提案し、ウェーブレット変換の各帯域において DCT のブロックサイズを変更することで、符号化性能が改善されることおよびブロック歪みやモスキート雑音等の復号画像へ混入する雑音の影響を小さくできることを示すために C 言語を用いてシミュレーション

第3章 高圧縮符号化技術

を行った。その結果、客観評価であるビットレートとピーク SN 比の関係においては、圧縮率が比較的低い場合、帯域 B1 ~ B4, 帯域 B5 ~ B7 でのブロックサイズをそれぞれ 4×4 , 8×8 とした場合が最良の符号化性能となり、圧縮率が高くなるにつれ、帯域 B1 ~ B4, 帯域 B5 ~ B7 でのブロックサイズをそれぞれ 8×8 , 16×16 とした場合が最良の符号化性能となった。また、主観評価である復号画像の比較においては、いずれの画像においても帯域 B1 ~ B4, 帯域 B5 ~ B7 でのブロックサイズをそれぞれ 4×4 , 8×8 とした場合、視覚的に良好な復号画像が得られた。以上より、帯域 B1 ~ B4, 帯域 B5 ~ B7 でのブロックサイズをそれぞれ 4×4 , 8×8 とすることで主観的にも客観的にも良好な復号画像が得られることとなり、本報告で提案したウェーブレット - DCT 符号化の有用性について示した。

本節では、各帯域での量子化係数を一定としていたが、帯域毎に量子化係数を変化させることで復号画像のさらなる画質改善が期待できる。つまり、低域信号に対する量子化では高域成分を削除し、高域成分に対する量子化では低域成分を削除するように量子化することで復号画像へのブロック歪みやモスキート雑音の影響をさらに軽減できると考えられる。そこで今後の課題として、各帯域における量子化係数の最適化があげられる。また、提案したウェーブレット - DCT では、最終的な符号化データは DCT 係数であるため、動き保証等を容易に行えと考えられる。そこでウェーブレット - DCT の動画像への適用も今後の課題としてあげられる。

3.3 カラー成分間の相関を利用したカラー画像符号化

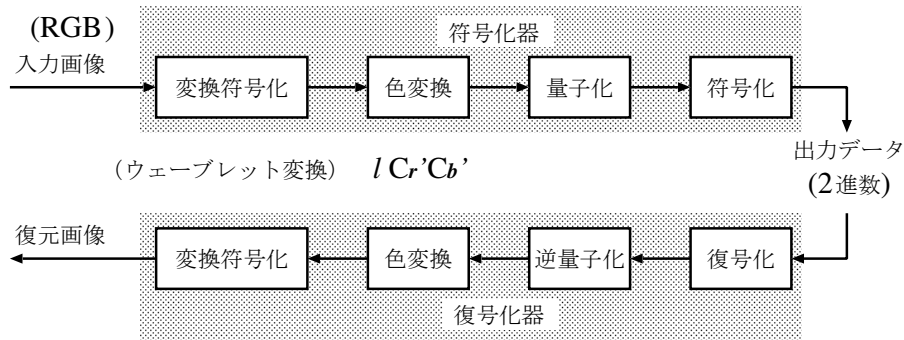


図 3.18 $lCrCb'$ を用いたウェーブレット変換の手順

3.3 カラー成分間の相関を利用したカラー画像符号化

一般的なカラー画像のウェーブレット変換符号化では符号化部分において各成分を独立に符号化しており、3 成分の関係をを用いた符号化は行われていない。そのため、3 成分の関係をを用いた符号化を行うことにより冗長性を排除し、データ量を削減することができ、符号化効率を改善することができると考えられる。

3.3.1 成分間の関係を利用した符号化方法

本研究では 3 成分の関係を利用した符号化方法として、RGB 成分から $lCrCb'$ 成分に変換し、3 成分の関係をを用いた符号化を行うウェーブレット変換符号化器を提案する。提案した符号化器を C 言語を用いて設計し、その性能を確認する。また、比較の対象として前章で述べた従来法である YC_rC_b に変換し、成分毎に独立して符号化するウェーブレット変換符号化器の設計も併せて行う。

3.3.2 提案するウェーブレット変換符号化器の構成

本研究で提案するウェーブレット変換符号化器の構成を図 3.18 に示す。提案法では、RGB に対してウェーブレット変換を行い、その後色変換を行い $lCrCb'$ に変換する。そして量子化、符号化を行う。

第3章 高圧縮符号化技術

従来法では $YC_r C_b$ に変換後にウェーブレット変換を行っていたが、提案法ではウェーブレット変換後に $l C'_r C'_b$ への変換を行うことが従来法 $YC_r C_b$ との相違である。

3.3.3 フィルタバンク

ウェーブレット変換が用いられている静止画像の国際標準規格である JPEG2000 にはフィルタバンクとして 9/7 タップフィルタと 5/3 タップフィルタが用いられている [29]。9/7 タップフィルタと 5/3 タップフィルタの帯域分割フィルタのローパスフィルタ $H_0(z)$ 、ハイパスフィルタ $H_1(z)$ および帯域合成フィルタのローパスフィルタ $F_0(z)$ 、 $F_1(z)$ を式 (3.2) から (3.9) に示す。

● 9/7 タップフィルタ

$$H_0(z) = E - Dz^{-1} - Cz^{-2} + Bz^{-3} + Az^{-4} + Bz^{-5} - Cz^{-6} - Dz^{-7} + Ez^{-8} \quad (3.2)$$

$$H_1(z) = I - Hz^{-1} - Gz^{-2} + Fz^{-3} - Gz^{-4} - Hz^{-5} + Iz^{-6} \quad (3.3)$$

$$F_0(z) = -I - Hz^{-1} + Gz^{-2} + Fz^{-3} + Gz^{-4} - Hz^{-5} - Iz^{-6} \quad (3.4)$$

$$F_1(z) = E + Dz^{-1} - Cz^{-2} - Bz^{-3} + Az^{-4} - Bz^{-5} - Cz^{-6} - Dz^{-7} + Ez^{-8} \quad (3.5)$$

ただし、

$$\begin{aligned} A &= 0.6029490182363579 & B &= 0.2668641184428723 & C &= 0.07822326652898785 \\ D &= 0.01686411844287495 & E &= 0.02674875741080976 & F &= 1.115087052456994 \\ G &= 0.5912717631142470 & H &= 0.05754352622849957 & I &= 0.09127176311424948 \end{aligned}$$

● 5/3 タップフィルタ

$$H_0(z) = -\frac{1}{8} + \frac{1}{4}z^{-1} + \frac{3}{4}z^{-2} + \frac{1}{4}z^{-3} - \frac{1}{8}z^{-4} \quad (3.6)$$

$$H_1(z) = -\frac{1}{2} + z^{-1} - \frac{1}{2}z^{-2} \quad (3.7)$$

$$F_0(z) = +\frac{1}{2} + z^{-1} + \frac{1}{2}z^{-2} \quad (3.8)$$

$$F_1(z) = -\frac{1}{8} - \frac{1}{4}z^{-1} + \frac{3}{4}z^{-2} - \frac{1}{4}z^{-3} - \frac{1}{8}z^{-4} \quad (3.9)$$

図 3.19 に 9/7 タップフィルタと 5/3 タップフィルタのローパスフィルタ $H_0(z)$ および

3.3 カラー成分間の相関を利用したカラー画像符号化

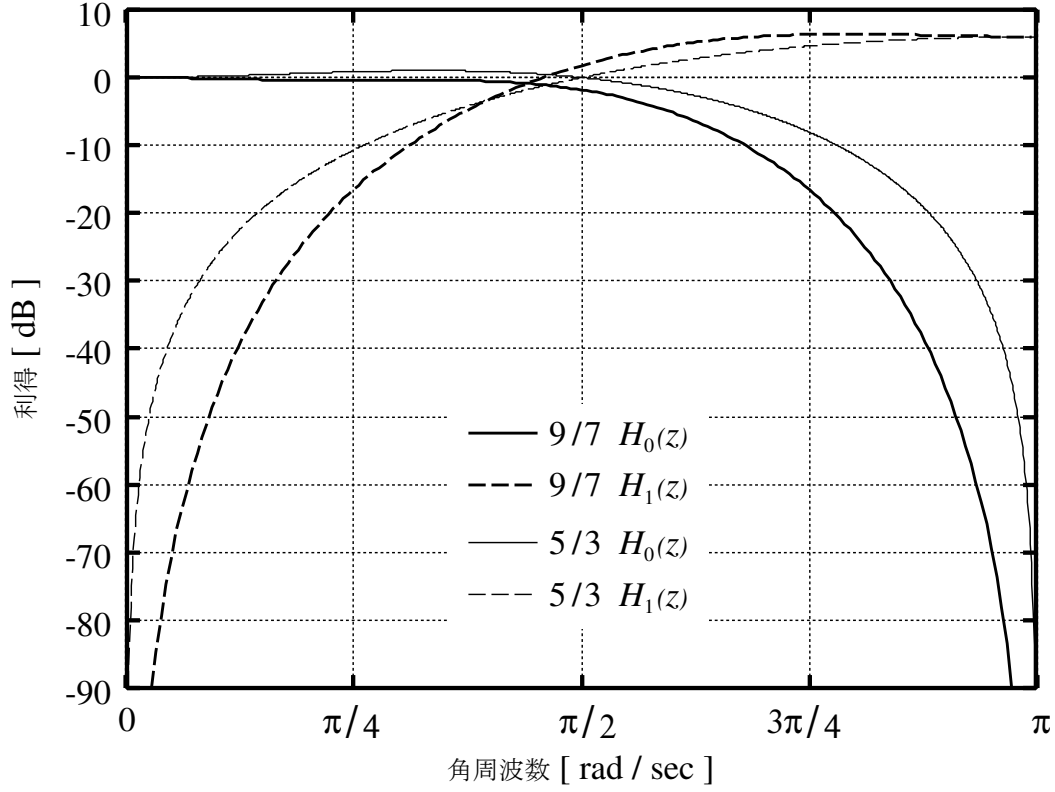


図 3.19 フィルタバンクの振幅特性

ハイパスフィルタ $H_1(z)$ の振幅特性を示す. 図 3.19 より 2 種類のフィルタは通過域では大きな違いはなく, 阻止域で 9/7 タップフィルタの方が大きな減衰が得られている. そのため, 9/7 タップフィルタの方がハイパスフィルタ通過後の出力である高域成分が少なくなり, 高圧縮できると考えられる. そこで本研究では, 提案法および従来法に 9/7 タップフィルタを用いてウェーブレット変換する.

9/7 フィルタの完全再構成条件

式 (3.2) から (3.5) の伝達関数より

$$\begin{aligned}
 H_0(z)F_0(z) + H_1(z)F_1(z) &= 4.90 \times 10^{-17}z^{-1} - 5.27 \times 10^{-16}z^{-3} \\
 &\quad + 3.55 \times 10^{-15}z^{-5} + 2.00z^{-7} \\
 &\quad + 3.55 \times 10^{-15}z^{-9} \\
 &\quad - 5.27 \times 10^{-16}z^{-11} + 4.90 \times 10^{-17}z^{-13} \\
 &\simeq 2z^{-7}
 \end{aligned} \tag{3.10}$$

$$H_0(-z)F_0(z) + H_1(-z)F_1(z) = 0 \tag{3.11}$$

第3章 高圧縮符号化技術

となり、式 (2.26), (2.27) の完全再構成条件をほぼ満たしており、遅延を生じるものの入力と同一の画像信号を復元できる。

3.3.4 RGB から l $C'_r C'_b$ への色変換

RGB から l $C'_r C'_b$ への変換、逆変換を式 (3.12), (3.13) に示す。

$$\begin{pmatrix} l \\ C'_r \\ C'_b \end{pmatrix} = \begin{pmatrix} \sqrt{R^2 + G^2 + B^2} \\ |R| - l/\sqrt{3} \\ |B| - l/\sqrt{3} \end{pmatrix} \quad (3.12)$$

$$\begin{pmatrix} |R| \\ |G| \\ |B| \end{pmatrix} = \begin{pmatrix} C'_r + l/\sqrt{3} \\ \sqrt{R^2 + G^2 + B^2} \\ C'_b + l/\sqrt{3} \end{pmatrix} \quad (3.13)$$

図 3.20(a) から (d) に変換後の各成分の画像を示す。 l は RGB の 3 次元空間で考えた時のベクトルの長さに対応する。 よって、 l 成分を図示した図 3.20(a) は、 RGB の重み付け平均である Y 成分を図示した図 2.14(a) よりも信号の画素値は大きくなる。 R, G, B が同じ値の場合には l は RGB の各成分や Y 成分の $\sqrt{3}$ 倍の大きさの信号になる。 そのため、 l を $1/\sqrt{3}$ 倍した図 3.20(b) の画像信号は、 Y と同じような画像信号になる。

C'_r, C'_b は C_r, C_b を参考にし、 Y と l の関係を考慮して定義している。 そのため、 C_r, C_b と同様に図 3.20(c), (d) C'_r, C'_b 成分には情報が少なくなっており、低データ量で符号化できると考えられる。

式 (3.13) より l $C'_r C'_b$ から RGB への逆変換では RGB の絶対値しか求めることはできない。 そのため、ウェーブレット変換係数のように負の値をもつ信号を逆変換する場合には、 l $C'_r C'_b$ 以外にも RGB の正負が付加情報として必要となる。

l $C'_r C'_b$ の成分間の関係

式 (3.12) より l 成分と RGB 成分には、式 (3.14) の関係がある。

$$0 \leq |r|, |g|, |b| \leq l \quad (3.14)$$

そのため、式 (3.12), (3.14) より l と C'_r の関係は、

$$-1/\sqrt{3} \times l \leq C'_r \leq (1 - 1/\sqrt{3}) \times l \quad (3.15)$$

3.3 カラー成分間の相関を利用したカラー画像符号化



(a) l 成分



(b) $1/\sqrt{3}$ 倍した l 成分



(c) C'_r 成分



(d) C'_b 成分

図 3.20 l C'_r C'_b に変換後の画像成分

となる. ここで, l と $|C'_r|$ の関係は,

$$\begin{aligned} 0 \leq |C'_r| &\leq 1/\sqrt{3} \times l \\ &= l \times 0.577... \end{aligned} \quad (3.16)$$

となる. よって, 常に $l > |C'_r|$ という関係が成立する. また, C'_b と l についても同様の関係があり, $l > |C'_b|$ となる.

第 3 章 高圧縮符号化技術

B10 (固定)	B9 Q	B6 2・Q	B3 2・Q
B8 Q	B7 2・Q		
B5 2・Q		B4 2・Q	B1 2・Q
B2 2・Q			

図 3.21 量子化係数の重み付け

以上のことから, 常に $l > |C'_r|, |C'_b|$ となる.

3.3.5 量子化

ウェーブレット変換後の信号成分はすべてローパスフィルタを通過した低域成分と一箇所でもハイパスフィルタを通った高域成分で出力分布が異なっている. そのため, 低域成分と高域成分で異なる量子化器を用いる.

ローパスフィルタは, 信号の重み付けした平均値を出力する. そのため, 低域成分は画像信号の平均的な信号となり画像によって分布が異なる. しかし偏りが少なく, 比較的一様な分布となる. そのため, 低域成分には図 2.11 の線形量子化を行う. また, 人間の視覚特性は低域成分の量子化誤差に対して敏感であるため, 量子化幅 Δ を小さくして低域成分の量子化誤差を少なくした方が視覚的に良好な画像が得られる. 本研究では低域成分の量子化幅 Δ を 2 で固定してシミュレーションを行う. 自然画像において高域成分は輪郭部分に多く含まれるが, 輪郭部分以外では少なく, 振幅値は 0 付近に集中する. この 0 付近に集中する高域成分を取り除いて復元しても復元画像への視覚的な影響が少ない. その

3.3 カラー成分間の相関を利用したカラー画像符号化

ため、高域成分は 0 付近の信号は取り除き、輪郭部分の量子化に重点を置いた図 2.12 に示すデッドゾーン付きミッドトレッド量子化を行う。高域成分は低域成分よりも量子化誤差に対する感度が低い。よって、低域成分よりも量子化幅 Δ を大きくする。また、高域成分の中でも階層の高い帯域ほど感度が高く、低い帯域ほど感度が低い。そこで帯域によって量子化幅 Δ や下限スレッシュールド d の重み付けを行ってシミュレーションを行う。図 3.21 に各帯域での重み付けを示す [30]。B8, B9 の量子化幅は B1 から B7 の量子化幅の $1/2$ にする。また、下限スレッシュールド d と量子化幅 Δ の比が常に $d : \Delta = 2 : 1$ になるように量子化幅と下限スレッシュールドを変化させシミュレーションを行う。具体例として $Q = 5$ の場合には B1 から B7 までの帯域の量子化幅 $\Delta = 10$ とし、下限スレッシュールド $d = 20$ とする。B8, B9 では量子化幅 $\Delta = 5$ 、下限スレッシュールド $d = 10$ とする。B10 の領域は Q の値に関わらず 2 とする。

$l C'_r C'_b$ や $Y C_r C_b$ は成分により復元画像への影響が異なるため、成分毎に重み付けすることも考えられるが、本研究ではどの成分においても同じ量子化幅で量子化する。 $l C'_r C'_b$ を同じ量子化幅で量子化した場合には、成分間で $l > |C'_r|, |C'_b|$ という関係が成立するので量子化後の係数が $l = 0$ ならば常に $C'_r = 0, C'_b = 0$ となる。

3.3.6 符号化

信号の分布が低域成分と高域成分により異なるので量子化と同様に低域成分と高域成分で異なった符号化を行う。

自然画像では隣接する画素間の相関が強いため近隣画素との重みづけ平均を出力するローパスフィルタの出力である低域成分も相関が強く、隣接する信号との差分を取れば差分値は 0 付近に集中する。そのため、隣接する画素間において差分をとる DPCM 符号化を行い、冗長性を排除する。DPCM 符号化後の差分信号は 0 付近に分布が集中するため、絶対値の小さいものほど少ない符号長で符号化する可変長符号化を用いる。表 3.2 に本研究で使用した可変長符号を示す。

第 3 章 高圧縮符号化技術

表 3.2 可変長符号列

グループ番号	差分值	識別符号	付加ビット数
0	0	00	0
1	-1 , 1	010	1
2	-3,-2 , 2,3	011	2
3	-7,...,-4 , 4,...,7	100	3
4	-15,...,-8 , 8,...,15	101	4
5	-31,...,-16 , 16,...,31	110	5
6	-63,...,-32 , 32,...,63	1110	6
7	-127,...,-64 , 64,...,127	11110	7
8	-255,...,-128 , 128,...,255	111110	8
9	-511,...,-356 , 256,...,511	1111110	9
10	-1023,...,-512 , 512,...,1023	11111110	10
:	:	:	:

表 3.3 ゼロツリー符号化の符号列

シンボル	識別符号
zt(zerotree)	00
iz(isolated zero)	01
pos(positive)	10
neg(negative)	11

高域成分は 0 付近に信号が集中している。そのため、低域成分の差分值と同様に表 3.2 を用いて値を符号化する。また、量子化後には輪郭部分を除き 0 の信号が多くなる。そのため、ゼロツリー符号化を用いて帯域間の相関を利用し符号化を行う。表 3.3 にゼロツリー符号化に用いた識別符号を示す。

l, C'_r, C'_b の場合には、 l, C'_r, C'_b の 3 成分以外にも色変換前の RGB のプラス・マイナスの符号が必要である。R,G,B の正負を調べると R,G,B すべてが正または負になる確率が

3.3 カラー成分間の相関を利用したカラー画像符号化

表 3.4 RGB の正負の符号列

識別符号	R の正負	G の正負	B の正負
10	－	－	－
11	＋	＋	＋
010	－	＋	－
011	＋	－	－
0000	－	－	＋
0001	＋	＋	－
0010	－	＋	－
0011	＋	－	＋

高い。そこで R,G,B の正負の符号は 3 つの符号をまとめて表 3.4 に示した識別符号を付加情報として符号化する。

成分間の関係を用いた符号化

l $C'_r C'_b$ では色変換後の信号は $l > |C'_r|, |C'_b|$ となるため、量子化後に $l = 0$ の場合には $C'_r = C'_b = 0$ となる。そのため、符号化時に $l = 0$ の場合には、 $l = 0$ の情報さえあれば C'_r, C'_b の情報は必要ない。そこで、本研究では成分間の関係を利用し $l = 0$ の場合には、符号化側では l だけを符号化し、 C'_r, C'_b を符号化しない。そして、復号化側では $l = 0$ であれば、 $C'_r = C'_b = 0$ とする。成分間の関係を利用することにより付加情報なしで 3 成分をまとめて符号化することができる。また、 $l = C'_r = C'_b = 0$ の場合には逆変換後に $R = G = B = 0$ になるため、RGB の正負の符号も必要なくなり、 $l = 0$ を符号化するだけでよい。

高域成分は $l = 0$ となる確率が高くなるため、提案した符号化方法をゼロツリー符号化と組み合わせることにより図 3.22 の HH のように帯域間と成分間の関係を用いて対応する画素をまとめて符号化することができる。 l 成分 3 階層の 1 画素と原画像で対応する 2 階層の 2×2 , 1 階層の 4×4 の画素がすべて 0 であり、zt の場合には成分間の関

第3章 高圧縮符号化技術

係から原画像で対応する C'_r, C'_b もすべて 0 であり z_t となる。そのため、 l 成分が z_t という情報だけで C'_r, C'_b も z_t であることが表現できる。従来法では z_t である場合には、 $w \ 1 + 2 \times 2 + 4 \times 4 = 21$ 個の画素しかまとめて表現することができないが、提案法では $21 \times 3 = 63$ 個の画素をまとめて表すことができる。また、圧縮率を高くし量子化幅を大きくした場合には l 成分が z_t となる確率が高くなるため、特に低ビットレート時に符号化効率を改善できると考えられる。図 3.22 では、HH 帯域での成分間の関係を用いた符号化の例を示しており、HL, LH 帯域でも同様に成分間の関係を用いてまとめて符号化を行う。このように成分間の関係を用いることにより多くの画素をまとめて符号化することができ、特に $l = 0$ となる確率が高くなる低ビットレート時に符号化効率の劇的な改善が期待できる。

3.3.7 実験結果

提案手法である $l \ C'_r C'_b$ に変換し、成分間の関係を用いた符号化方法の性能を調べるため、提案法を用いたウェーブレット変換符号化器を C 言語を用いて設計し、カラー画像を用いて実験を行う。また、従来法である $YC_r C_b$ に変換し、符号化を行うウェーブレット変換符号化器も同様に C 言語を用いて作成し、提案法との比較を行う。

本研究に用いた実験画像とその拡大画像を図 3.23 から図 3.30 に示す。これらの実験画像は RGB それぞれ 256 階調 (最大表現色数約 1677 万色, 24 bit/pixel) である。図 3.23 の Lenna 画像は大きさが 512×512 ピクセルの画像であり、画像圧縮の文献によく用いられている。また、RGB 成分の分布による影響を調べるため、各成分が均等に含まれている図 3.24 の 800×600 ピクセルの home 画像や成分の偏りが大きい図 3.25 の 512×512 ピクセルの milkdrop 画像を使用する。さらに、高域成分の多い画像への影響を調べるため、図 3.26 の 512×512 ピクセルの mandrill 画像を用いる。

3.3 カラー成分間の相関を利用したカラー画像符号化

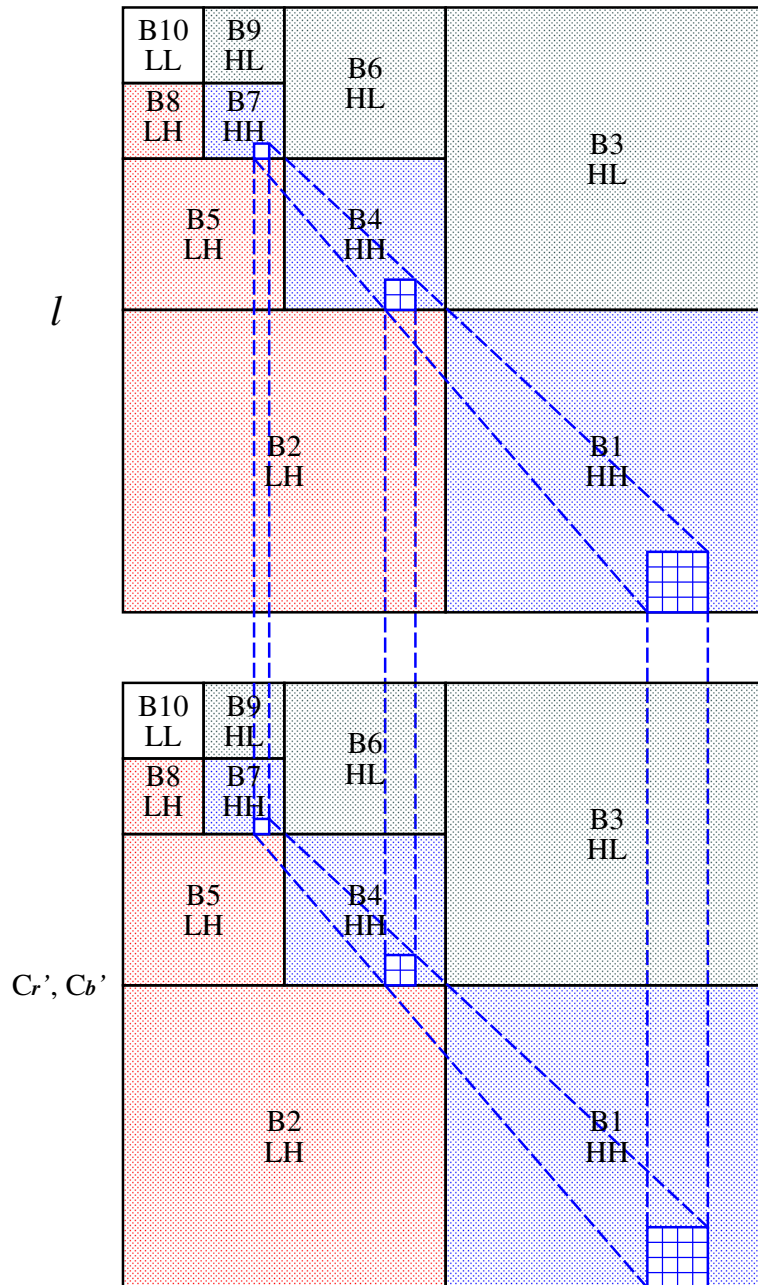


図 3.22 成分間の関係を利用した符号化方法の概念図

復元画像の客観評価

客観評価の指標として復元画像の画質を表すピーク SN 比とデータ量を表すビットレートとの関係を用いる。それぞれの定義を以下に示す。

第 3 章 高圧縮符号化技術



図 3.23 Lenna 画像



図 3.24 home 画像

PSNR カラー画像のピーク SN 比 (PSNR) の定義式を式 (3.17) に示す [31].

$$\text{PSNR} = 10 \log_{10} \frac{(255\sqrt{3})^2}{\sum_{x=1}^X \sum_{y=1}^Y \{F(x, y)\}^2} \quad (3.17)$$

$$\{F(x, y)\}^2 = \{f_R(x, y) - g_R(x, y)\}^2 + \{f_G(x, y) - g_G(x, y)\}^2 + \{f_B(x, y) - g_B(x, y)\}^2$$

ただし、画像の大きさは $X \times Y$ ピクセルであり、 $f_R(x, y)$, $f_G(x, y)$, $f_B(x, y)$ は原画像の R, G, B 成分の画素値であり、 $g_R(x, y)$, $g_G(x, y)$, $g_B(x, y)$ は復元画像の R, G, B 成分の

3.3 カラー成分間の相関を利用したカラー画像符号化

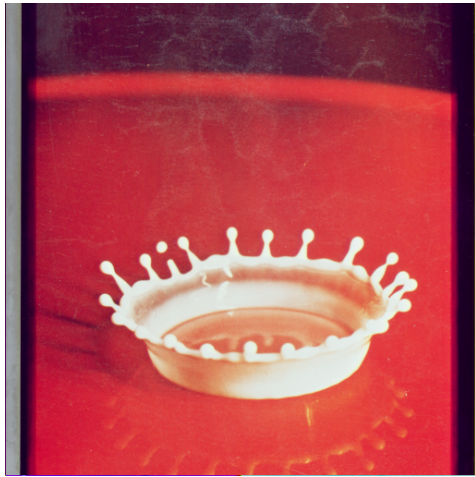


図 3.25 milkdrop 画像

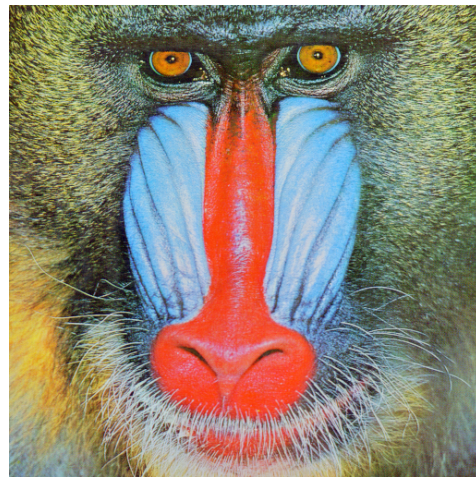


図 3.26 mandrill 画像

画素値である.

式 (3.17) より, 原画像と復元画像との差が大きいほどピーク SN 比は小さくなり, 原画像と復元画像との差が小さいほどピーク SN 比は大きくなる. そのため, ピーク SN 比が高いほど原画像を忠実に再現していることを示している.

ビットレート ビットレートは 1 画素当たりのビット数であり, 値が小さいほどデータ量が少なく, 値が大きいほどデータ量が多いことを意味する. 例えば図 3.23 の原画像のビットレートは 24 bit/pixel であり, 画像の大きさが 512×512 ピクセルのため, ファイルサイズは,

$$24 \times 512 \times 512 = 6291456 \text{ [bit]} = 786432 \text{ [Byte]} = 768 \text{ [kB]} \quad (3.18)$$

と計算できる. もし, 1 bit/pixel で画像の大きさが 512×512 ピクセルならファイルサイズは,

$$1 \times 512 \times 512 = 262144 \text{ [bit]} = 32768 \text{ [Byte]} = 32 \text{ [kB]} \quad (3.19)$$

となる. よって, ビットレートが低いほどファイルサイズは小さくなる.

第3章 高圧縮符号化技術

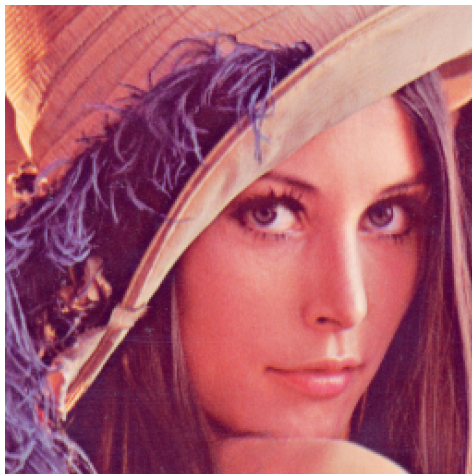


図 3.27 図 3.23 の拡大画像



図 3.28 図 3.24 の拡大画像

客観評価

提案法である $l C'_r C'_b$ の性能を確認するため、図 3.23 から図 3.26 の画像に対して量子化係数を変化させピーク SN 比とビットレートの関係性を調べ、従来法である $YC_r C_b$ と比較する。量子化係数は線形量子化を行う低域成分の量子化幅 Δ を固定し、デッドゾーン付きミッドトレッド量子化を行う高域成分の下限スレッシュホールド d と量子化幅 Δ の比が一定となるように d および Δ を変化させる。

3.3 カラー成分間の相関を利用したカラー画像符号化

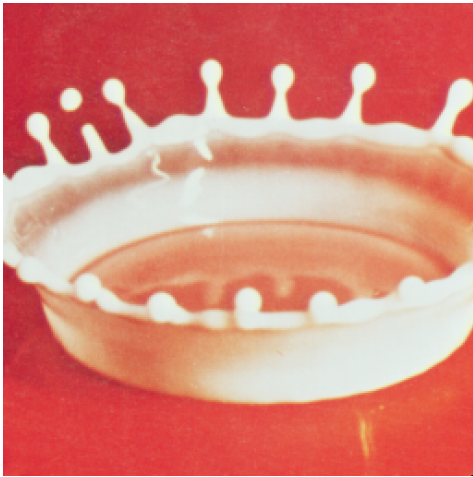


図 3.29 図 3.25 の拡大画像

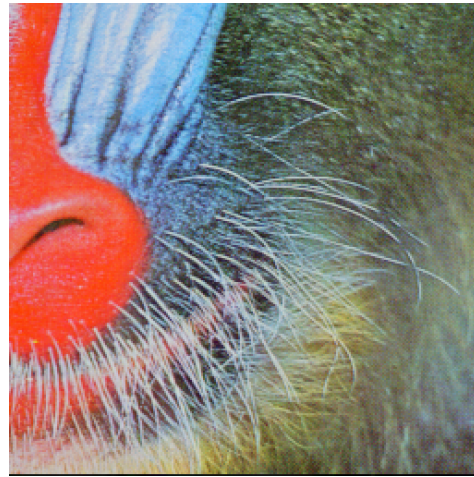


図 3.30 図 3.26 の拡大画像

図 3.31 から図 3.34 に図 3.23 から図 3.26 の画像に対する提案法 $l C'_r C'_b$ および従来法 $YC_r C_b$ の符号化性能を示す. 図 3.31 から図 3.34 では同一ビットレートにおいてピーク SN 比が高いことは性能の高いことを意味する. すなわち, 上側に曲線があるほど客観評価により性能が高いことを表している.

Lenna 画像の性能である図 3.31 では提案法の性能を示す青い実線と従来法の性能を示す赤い破線が 1 bit/pixel 付近で交点を持つ. 交点より高いビットレートでは, 同一ビットレートにおいて提案法の方が若干ピーク SN 比が低く, 性能が若干低いことがわかる. 一方, 交点よりも低いビットレートでは提案法 $l C'_r C'_b$ の方が同一ビットレートでピーク SN 比が高くなる. また, ビットレートが低くなればなるほどピーク SN 比の差が大きくなる. つまり, 低ビットレートでは提案法の性能が非常に高い. このような結果となった理由として, 高ビットレートにおいて高域成分は $l = 0$ となる確率が低くなり, 成分間の関係を利用した符号化をあまり行うことができないため従来法よりも性能が若干低くなるが, 低ビットレートでは $l = 0$ となる確率が高くなり, 成分間の関係を利用して効率良く符号化できるため, 性能が非常に高くなると考えられる.

home 画像や milkdrop 画像に対する性能である図 3.32 や図 3.33 でも図 3.31 と同様に提案法と従来法の性能を示す曲線が交点を持ち, 交点よりも高ビットレートでは提案法

第3章 高圧縮符号化技術

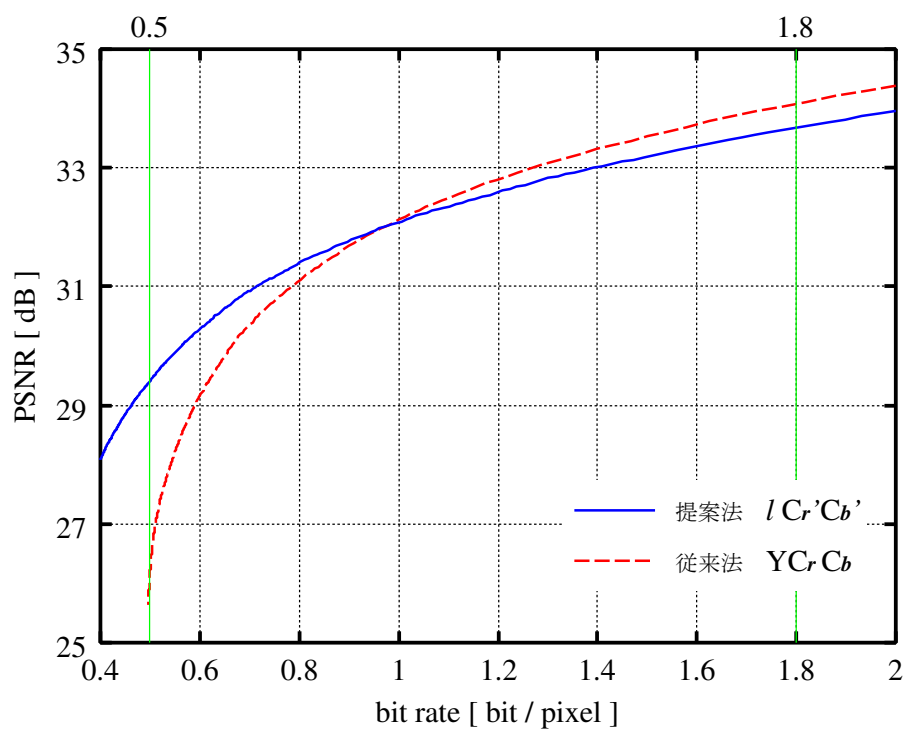


図 3.31 Lenna に対する性能

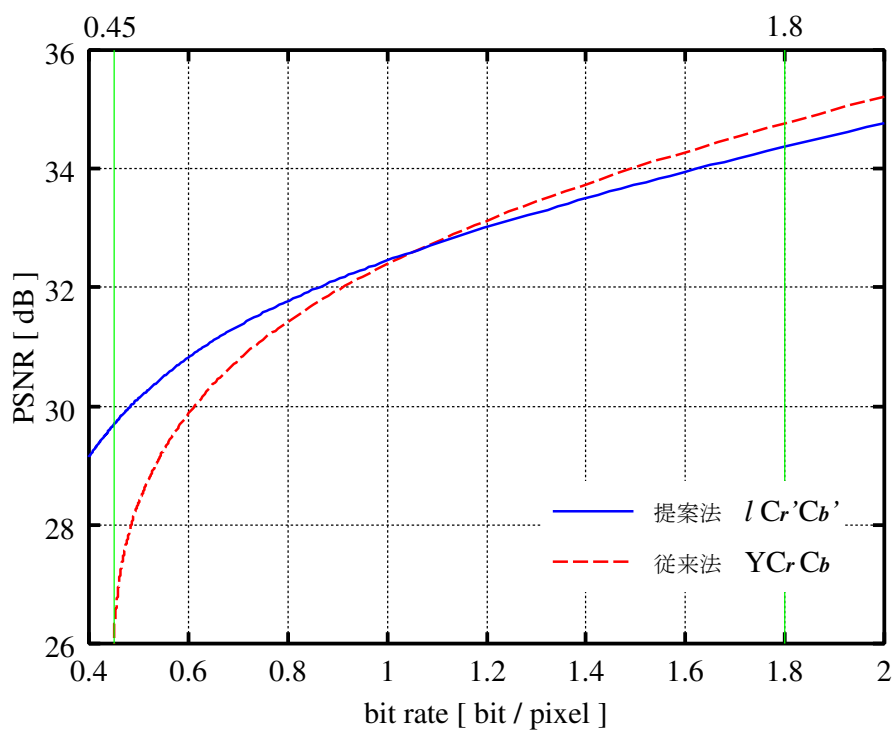


図 3.32 home に対する性能

3.3 カラー成分間の相関を利用したカラー画像符号化

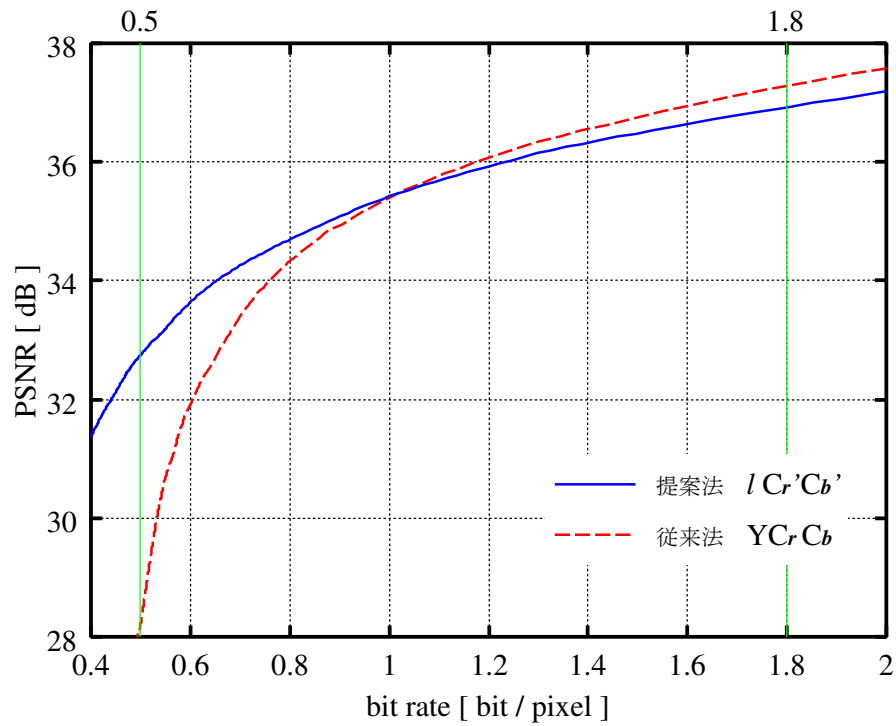


図 3.33 milkdrop に対する性能

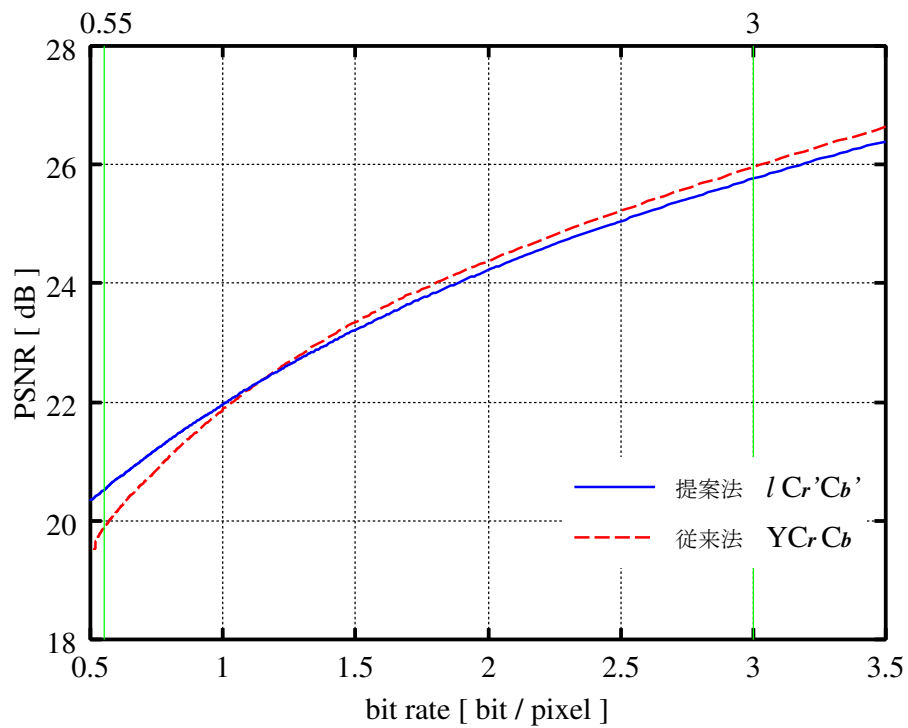


図 3.34 mandrill に対する性能

第 3 章 高圧縮符号化技術

の性能が若干低くなるものの、低ビットレートでは提案法の性能が非常に高くなる。よって、RGB 成分の分布によらず、高ビットレートでは提案法の性能が若干低く、低ビットレートでは性能が非常に高くなることが確認できる。

mandrill 画像の性能である図 3.34 でも図 3.31 から図 3.33 と同様に 2 曲線が交点を持ち、交点よりも高ビットレートでは提案法の性能が低くなり、低ビットレートでは高くなっている。しかし、図 3.31 から図 3.33 の他の画像と異なり、低ビットレート時のピーク SN 比の差が小さくなっており、提案法があまり有効でないことがわかる。このようになった理由として mandrill 画像のように高域成分を多く含む画像では、低ビットレートの場合でも高ビットレート時と同様に高域成分が $l = 0$ となる確率が低くなるため成分間の関係を用いて効率良く符号化できないことが挙げられる。

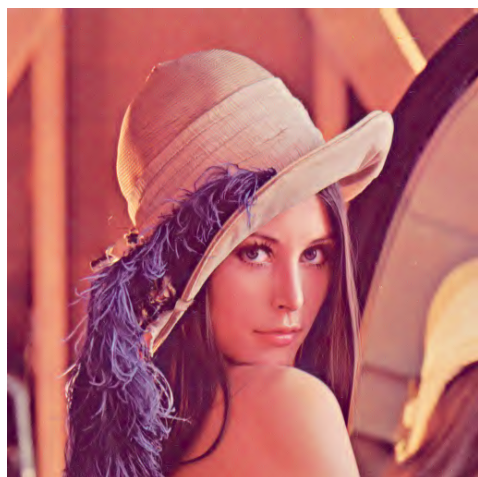
よって客観評価により以下の事が言える。

- 高ビットレートでは提案法の符号化性能は若干低い。
- 低ビットレートでは提案法の符号化性能は高い。また、ビットレートが低くなるにつれ提案法の符号化性能は非常に高くなる。
- mandrill 画像のように高域成分を多く含む画像では低ビットレートにおいて提案法の符号化性能は高くなる。

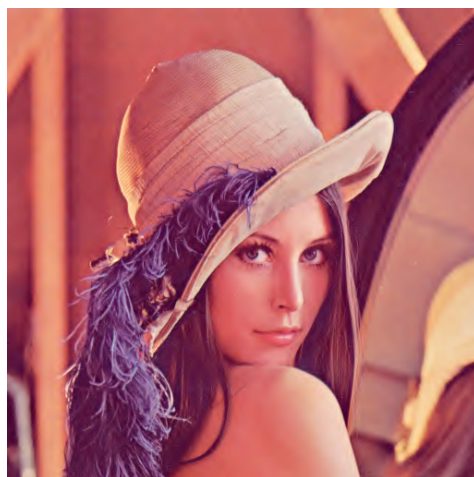
主観評価

ピーク SN 比は客観的な指標であり、ピーク SN 比が高い復元画像の画質が良好であるとは必ずしも言えない。そのため、ここでは実際に復元画像を比較する。3.3.7 の客観評価から交点よりも低ビットレートおよび高ビットレートでの提案法 $l C'_r C'_b$ と従来法 $Y C_r C_b$ とのピーク SN 比の関係が異なるため、低ビットレート時と高ビットレート時の同一ビットレートの復元画像を比較する。図 3.35 から図 3.42 に提案法および従来法の低ビットレートおよび高ビットレート時の復元画像を示す。ただし、それぞれの図で (a) は提案法の復元画像を示し、(b) は従来法の復元画像を示す。(c), (d) はそれぞれ提案法 (a),

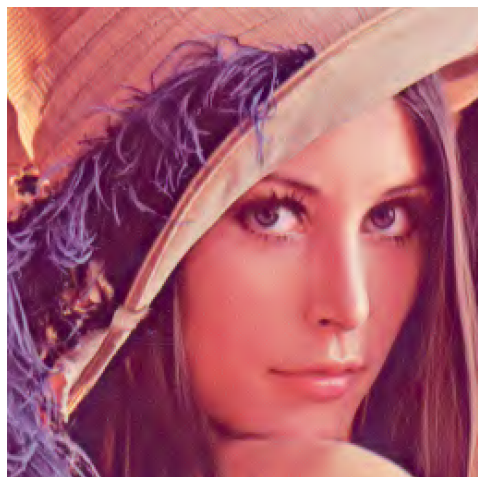
3.3 カラー成分間の相関を利用したカラー画像符号化



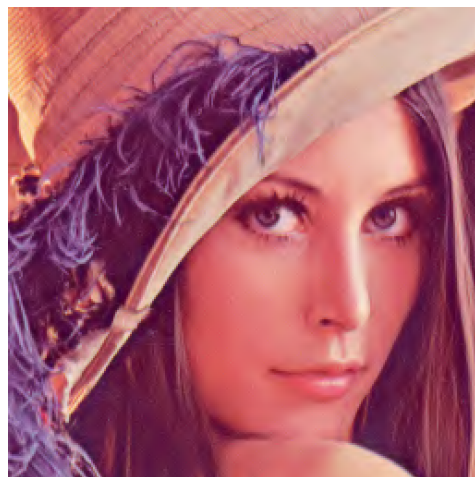
(a) $l \ C'_r C'_b$ (33.61 [dB], 1.803 [bit/pixel])



(b) $YC_r C_b$ (34.09 [dB], 1.803 [bit/pixel])



(c) (a) の拡大図



(d) (b) の拡大図

図 3.35 Lenna 画像の高ビットレート時の復元画像

従来法 (b) の拡大図である。まず, Lenna 画像の高ビットレート時と低ビットレート時の復元画像を比較する。Lenna 画像の高ビットレート時として 1.8 bit/pixel 付近の提案法と従来法の復元画像を図 3.35 に示す。同図 (a), (b) を比較すると従来法の復元画像の方がピーク SN 比が約 0.5 dB 高くなっているが, 復元画像を視覚的に比較した場合には (a) と (b) に大きな違いは見られない。さらに, 細部への影響を確認するために拡大

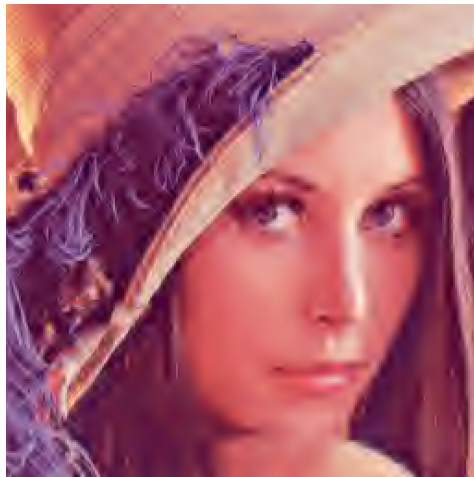
第 3 章 高圧縮符号化技術



(a)
 $l\ C'_r C'_b$ (29.40 [dB], 0.500 [bit/pixel])



(b) $YC_r C_b$ (26.20 [dB], 0.500 [bit/pixel])



(c) (a) の拡大図



(d) (b) の拡大図

図 3.36 lena 画像の低ビットレート時の復元画像

画像である (c), (d) を比較した場合でも大きな違いが見られない。つまり、客観評価において高ビットレート時の提案法の性能は従来法よりも低いため欠点だと考えられたが、主観評価では差異はなく同等の画質が得られている。次に Lenna 画像の低ビットレート時として 0.5 bit/pixel 付近の復元画像を図 3.36 に示す。同図 (a), (b) を比較すると提案法 $l\ C'_r C'_b$ の復元画像の方がピーク SN 比が 3 dB 以上高くなっており、主観的に比較した

3.3 カラー成分間の相関を利用したカラー画像符号化



(a) $l C'_r C'_b$ (34.38 [dB], 1.803 [bit/pixel])



(b) $YC_r C_b$ (34.78 [dB], 1.803 [bit/pixel])

図 3.37 home 画像の高ビットレート時の復元画像

第 3 章 高圧縮符号化技術



(c) (a) の拡大図



(d) (b) の拡大図

図 3.37 home 画像の高ビットレート時の復元画像

3.3 カラー成分間の相関を利用したカラー画像符号化



(a) $l C'_r C'_b$ (29.70 [dB], 0.451 [bit/pixel])



(b) $YC_r C_b$ (26.37 [dB], 0.450 [bit/pixel])

図 3.38 home 画像の低ビットレート時の復元画像

第 3 章 高圧縮符号化技術



(c) (a) の拡大図



(d) (b) の拡大図

図 3.38 home 画像の低ビットレート時の復元画像

3.3 カラー成分間の相関を利用したカラー画像符号化



(a) $l \ C'_r C'_b$ (36.93 [dB], 1.801 [bit/pixel])



(b) $YC_r C_b$ (37.29 [dB], 1.806 [bit/pixel])



(c) (a) の拡大図



(d) (b) の拡大図

図 3.39 milkdrop 画像の高ビットレート時の復元画像

場合でも従来法の復元画像は全体的にぼやけて不鮮明になっているのに対して、提案法の復元画像では鮮明になっている。また、拡大画像である (c), (d) を比較した場、2 つの画像の違いがより顕著になり、提案法の復元画像の画質が高いことがわかる。よって低ビットレートでは、客観評価が示すように主観評価でも提案法の性能が高いことを確認した。

次に RGB 成分の分布の違いによる影響を調べるため home 画像および milkdrop 画

第3章 高圧縮符号化技術



(a) $l C'_r C'_b$ (32.75 [dB], 0.500 [bit/pixel])



(b) $YC_r C_b$ (28.23 [dB], 0.500 [bit/pixel])



(c) (a) の拡大図

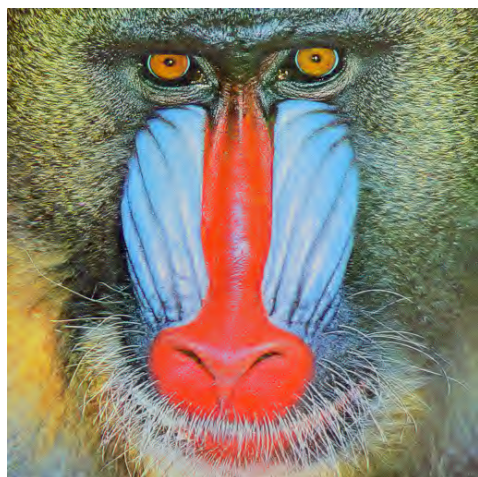


(d) (b) の拡大図

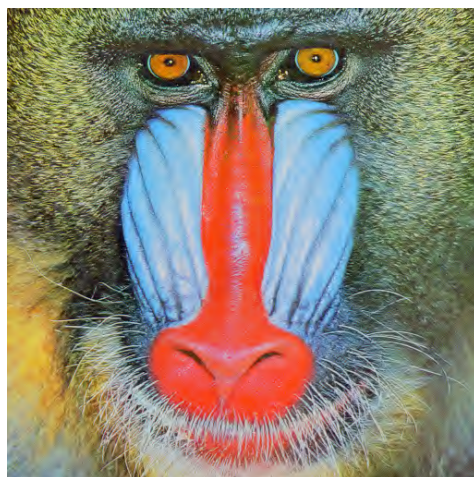
図 3.40 milkdrop 画像の低ビットレート時の復元画像

像の復元画像を比較する. 高ビットレート時の home と milkdrop の復元画像を図 3.37 および図 3.39 に示す. 図 3.37, 3.39 より, ピーク SN 比がそれぞれ 0.4 dB ほど提案法 $l C'_r C'_b$ の復元画像が低くなっているが, 復元画像を比較した場合には Lenna と同様に同等の画質が得られているおり, 高ビットレート時の提案法の性能に問題はない. 次に低ビットレート時の home と milkdrop の復元画像を図 3.38 および 3.40 に示す. 図 3.38, 3.40

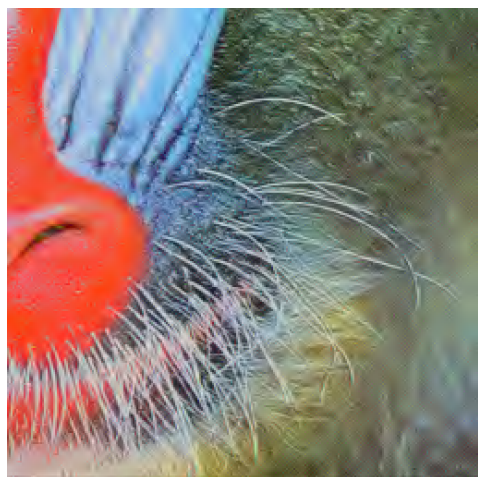
3.3 カラー成分間の相関を利用したカラー画像符号化



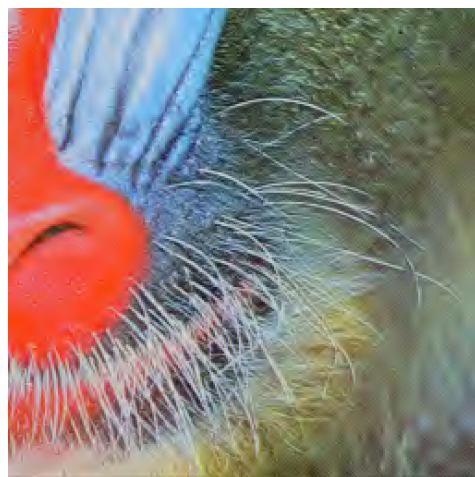
(a)
 $l C'_r C'_b$ (25.77 [dB], 3.003 [bit/pixel])



(b)
 $YC_r C_b$ (26.00 [dB], 3.003 [bit/pixel])



(c) (a) の拡大図



(d) (b) の拡大図

図 3.41 mandrill 画像の高ビットレート時の復元画像

より復元画像のピーク SN 比は提案法の方が 3 dB 以上高くなっており、復元画像を比較した場合にも良好な画質が得られている。

最後に高域成分の多い画像への影響を調べるため、mandrill 画像の復元画像を比較する。高ビットレート時の mandrill の復元画像を図 3.41 に示す。図 3.41 より提案法の復元画像のピーク SN 比は 0.23 dB 低くなっているが、主観的に比較した場合には同等の

第3章 高圧縮符号化技術



(a)
 $l C'_r C'_b$ (20.51 [dB], 0.550 [bit/pixel])



(b)
 $Y C_r C_b$ (19.88 [dB], 0.550 [bit/pixel])



(c) (a) の拡大図



(d) (b) の拡大図

図 3.42 mandrill 画像の低ビットレート時の復元画像

性能が得られている。また、低ビットレートでは提案法の復元画像のピーク SN 比は 0.63 dB 高くなっているが、主観的に評価した場合には同等の性能しか得られていない。

よって提案法と従来法の比較により以下の事を確認した。

- 高ビットレートにおいてピーク SN 比が若干低くなるが主観評価では同等の符号化

3.3 カラー成分間の相関を利用したカラー画像符号化

性能が得られる

- 低ビットレートにおける主観評価では符号化性能が高い
- 低ビットレートでも高域成分が多い画像に対してはピーク SN 比が若干高くなるが主観評価では従来法と同等の符号化性能しか得られない

そのため、主観評価により高域成分が多い画像に対して提案法 $l C'_r C'_b$ は従来法 $YC_r C_b$ と同等の性能しか得られないが、多くの自然画像のように低域部分が多い画像に対しては低ビットレートでは高い性能が得られており、提案法の有効性を確認した。

第3章 高圧縮符号化技術

3.3.8 むすび

静止画像の圧縮符号化方式として広く用いられている JPEG には離散コサイン変換 (DCT) が用いられている。DCT 符号化効率が高い反面、圧縮率を上げるにつれてブロックひずみやモスキート雑音が復元画像に混入する。そのため、これらの雑音の影響を軽減する符号化方式としてウェーブレット変換が注目を集めている。カラー画像をウェーブレット変換する場合には RGB 成分を YC_rC_b に色変換し、各成分を 3 枚のグレースケールの画像を符号化するように YC_rC_b を成分毎に個別に符号化する。そのため、成分間の関係を用い、まとめて符号化することにより符号化効率の改善が期待できる。

そこで本研究では、 $lC'_rC'_b$ に変換する色変換を提案し、色変換後に $lC'_rC'_b$ の関係を用いた符号化を行うウェーブレット変換符号化器を提案した。提案したウェーブレット変換符号化器を C 言語により作成し、様々な画像に対して復元画像に与える影響を調べた。また、比較のため従来型のウェーブレット変換符号化器でも同様に復元画像に与える影響を調べた。

実験結果の客観評価より提案法の符号化性能は従来法と比べ

- 高ビットレートにおいてピーク SN 比が若干低くなるが、符号化性能はほぼ同等
- 低ビットレートにおいてピーク SN 比は非常に高くなり、符号化性能を劇的に改善

となり、また主観評価より以下のことを確認した。

- 高ビットレートにおける復元画像の画質はほぼ同等
- 低ビットレートにおける復元画像の画質は飛躍的に改善

以上により提案法の有効性を確認した。

本研究では、ウェーブレット変換符号化器を C 言語により作成した。そのため、ウェーブレット変換を用いた新しい静止画像の国際標準規格である JPEG2000 に $lC'_rC'_b$ を

3.3 カラー成分間の相関を利用したカラー画像符号化

適用させ、符号化効率を改善することが今後の課題として挙げられる。その際、提案した $l\ C'_r\ C'_b$ では、3 成分以外にも色変換前の RGB の正負の情報が必要となり、メモリ量が従来法よりも多く必要となる。そのため、メモリ量を削減することが課題となる。また、ウェーブレット変換符号化を用いた動画像符号化の新たな国際標準規格である Motion-JPEG2000 に対して $l\ C'_r\ C'_b$ への変換を適用させることも考えられるが、動画像符号化では静止画像よりもデータ量やメモリ量が膨大となるため、静止画像以上にメモリ量の増加が問題となり、演算時のメモリ量を減らすことが今後の課題となる。以上の課題を克服して $l\ C'_r\ C'_b$ の色変換を JPEG2000 や Motion-JPEG2000 に適用することが今後の課題として挙げられる。

第 4 章

解像度変換技術

解像度変換とは、画像の拡大または縮小を行い画像のサイズを変換することである。画像の解像度変換はアプリケーション毎に異なるため、その互換性を保つために必要な技術である。また、画像の一部を切り取り拡大する、メモリの節約のために縮小して保存するなど様々な用途で解像度変換は用いられる。

4.1 画像空間

ここでは、一般的に広く使われている画像空間上での解像度変換法について紹介する。

4.1.1 整数分の 1 倍の縮小

画像空間上での画像の縮小について示す。ここでは、画像を縦に $1/H$ 倍、横に $1/W$ 倍するように縮小する。ただし、 H, W は正の整数である。画像空間上での縮小法は、余分な画素値の捨て方によりダウンサンプリング法、平均操作法、メディアン法などがある。

ダウンサンプリング法

最も単純な縮小方法は図 4.1 のように原画像を $H \times W$ のブロックに分け各ブロックから 1 画素を単純に抜き取り縮小画像を得る方法である。この方法をダウンサンプリング法（サブサンプリング法）という。図 4.1 は、ダウンサンプリング法により縦横 $1/3$ 倍

第 4 章 解像度変換技術

に縮小する例を示している。Barbara 画像を用いた場合の画像の縮小を図 4.4 に示す。図 4.4(a) は原画像であり、図 4.4(b) はダウンサンプリング法により縦横 1/2 倍に縮小した画像である。

平均操作法

図 4.2 のように原画像の $H \times W$ のブロック毎に平均値を求め、その値を縮小画像の画素値とする方法を平均操作法 という。図 4.4(c) に Barbara 画像を平均操作法により縦横 1/2 倍に縮小した画像を示す。

メディアン法

図 4.3 のように原画像の $H \times W$ のブロック毎にメディアン値 (中央値) を求め、その値を縮小画像の画素値とする方法をメディアン法 という。メディアン値とは、ブロック内の n 個 ($H \times W$ 個) の画素値を大きさ順に並べたとき、中央に位置する値である。 n が奇数ならば $(n + 1)/2$ 番目の値、 n が偶数ならば $n/2$ 番目と $n/2 + 1$ 番目の値の平均値を縮小画像の画素値とする。図 4.4(d) に Barbara 画像をメディアン法により縦横 1/2 倍に縮小した画像を示す。

4.1.2 整数倍の拡大

画像空間上での画像の拡大について示す。まず、画像の拡大において重要な補間関数について説明する。1 次元の補間関数の働きを図 4.5 に示す。各画素値に相当する離散信号 $x(n)$ から補間関数を用いて補間する。補間値は補間関数を用いて連続信号により生成され、次式で計算される。

$$x(t) = \sum_{n=-\infty}^{\infty} x(n)h(t - n) \quad (4.1)$$

ここでは例として、画像を縦に H 倍、横に W 倍に拡大する場合について説明する。ただし、 H, W は正の整数である。画像空間上での拡大法は、補間関数の種類により 0 次ホールド法、直線補間法、B-spline 補間法、3 次畳み込み補間法などがある。

4.1 画像空間

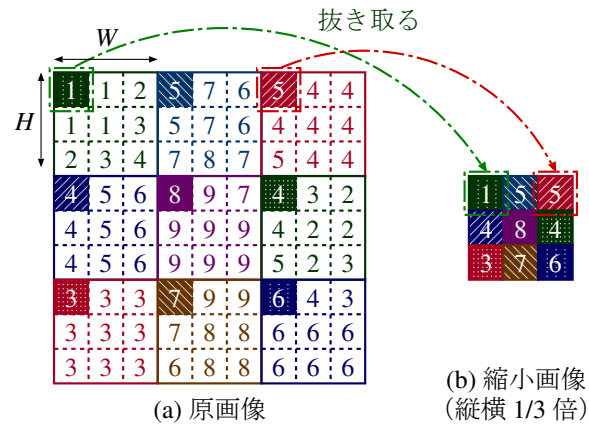


図 4.1 ダウンサンプリング法

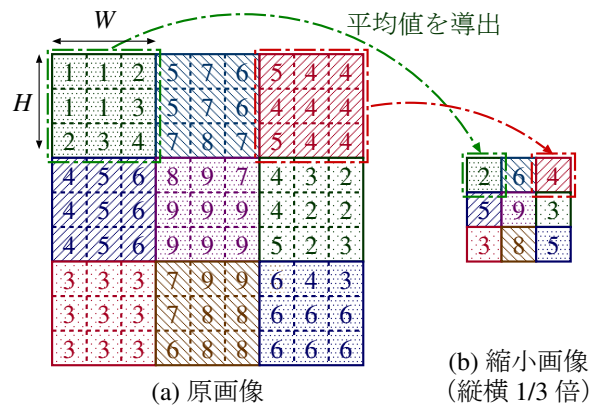


図 4.2 平均操作法

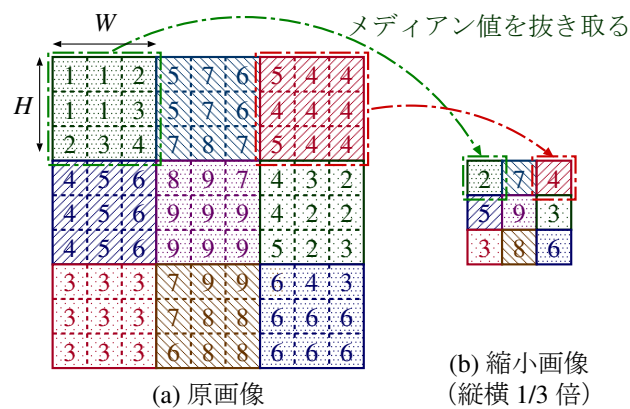
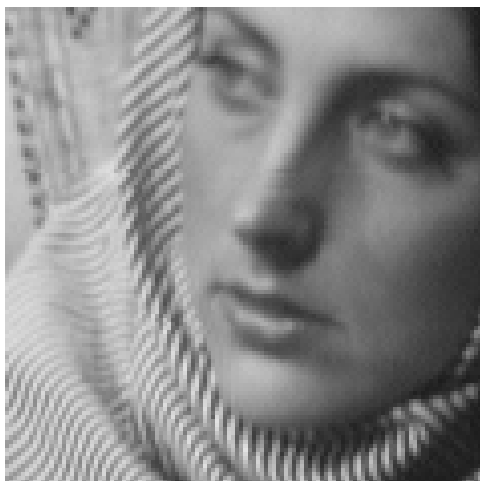


図 4.3 メディアン法

第 4 章 解像度変換技術



(a) 原画像



(b) ダウンサンプリング法



(c) 平均操作法

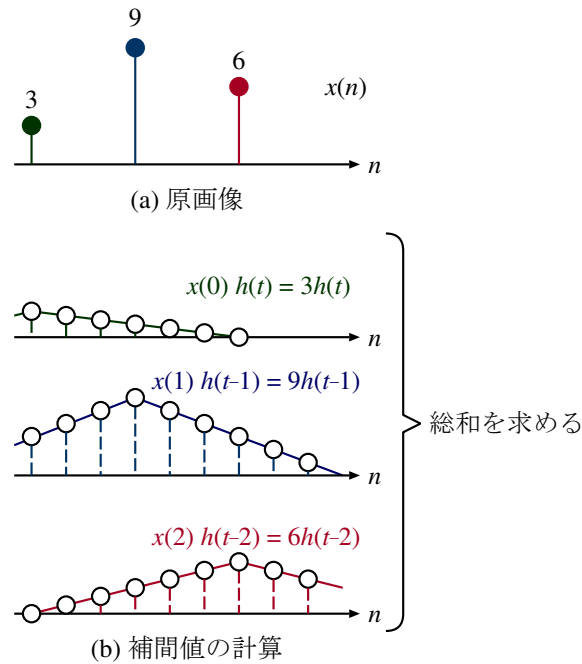


(d) メディアン法

図 4.4 縮小画像の比較 (縦横 1/2 倍)

0 次ホールド法

最も簡単な画像拡大方法は、図 4.6 のように原画像の各画素を $W \times H$ に並べる方法である。これを 0 次ホールド法 (最近傍法) という。図 4.6 は、0 次ホールド法により原画像を縦横 3 倍に拡大する例である。

図 4.5 補間関数 $h(t)$ による補間 (3 倍)

0 次ホールド法の補間関数は式 (4.2) および 図 4.8(a) で表される.

$$h(t) = \begin{cases} 1 & (0 \leq |t| < 0.5) \\ 0 & (0.5 \leq |t|) \end{cases} \quad (4.2)$$

cameraman 画像を用いた場合の画像拡大の例を図 4.9 に示す. 図 4.9(a) は原画像であり, 図 4.9(b) は 0 次ホールド法により縦横 2 倍に拡大した画像である. 0 次ホールド法は簡単に拡大できるが, 原画像の画素値を単純に繰り返すため拡大された画素が不連続となりエッジ部分がギザギザして見え, このギザギザは拡大率を上げるにつれてより顕著となる.

直線補間法

図 4.7 のように拡大に伴う新たな画素を隣接画素間の直線近似により求める方法を直線補間法 (線形補間法, 双一次補間法) という. 直線補間法の補間関数は式 (4.3) および図 4.8(b) で表される.

$$h(t) = \begin{cases} 1 - |t| & (0 \leq |t| < 1) \\ 0 & (1 \leq |t|) \end{cases} \quad (4.3)$$

第 4 章 解像度変換技術

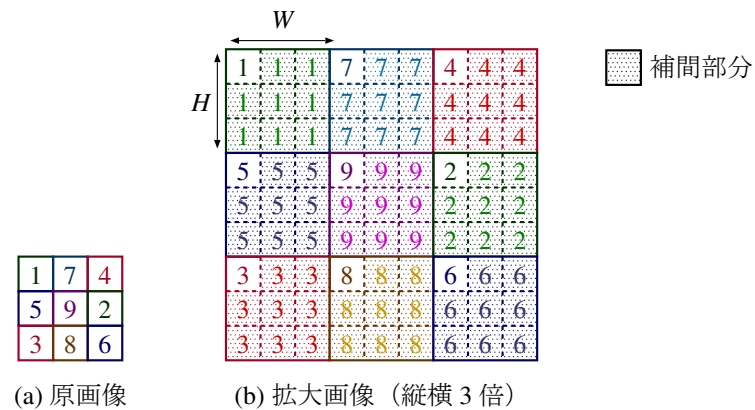


図 4.6 0 次ホールド法

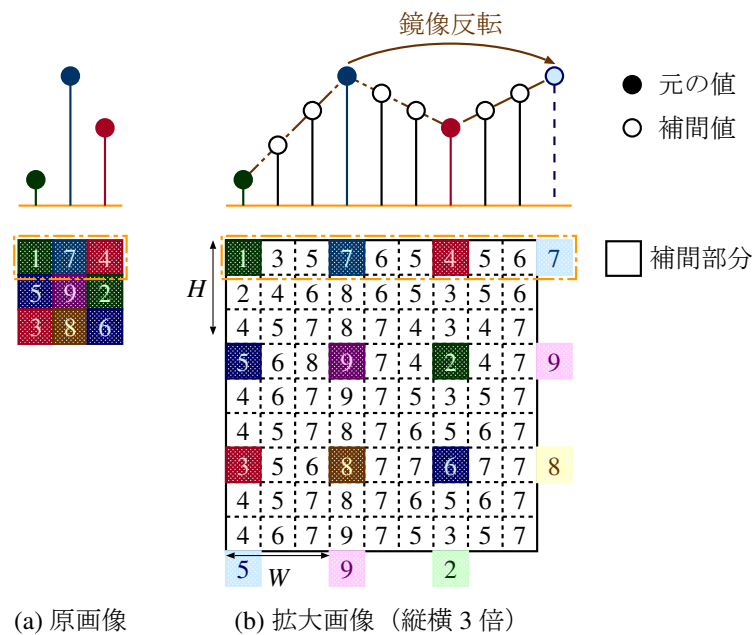


図 4.7 直線補間法

図 4.9(c) に直線補間法により cameraman 画像を縦横 2 倍に拡大した画像を示す。直線補間法は、0 次ホールド法に比べ滑らかに拡大することができる。実際の画像では、画素値は整数値であるため補間値を整数化する必要がある。

直線補間法を直接適用すると画像の端を 0 値と仮定するため画像の端で不連続が生じ、画像の端がきれいに拡大ができない。そこで、鏡像反転を行い画像の外側の画素値を仮定する。図 4.7 の例では、一番右端の画素値を補間する際、左隣の画素値より鏡像反転を行

い補間する.

B-spline 補間法

各補間値を原画像の 4 点より求めることでより精度の高い補間を行う方法がある.

B-spline 補間法の補間関数は式 (4.4) および 図 4.8(c) で表される.

$$h(t) = \begin{cases} \frac{1}{2}|t|^3 - |t|^2 + \frac{2}{3} & (0 \leq |t| < 1) \\ -\frac{1}{6}|t|^3 + |t|^2 - 2|t| + \frac{4}{3} & (1 \leq |t| < 2) \\ 0 & (2 \leq |t|) \end{cases} \quad (4.4)$$

図 4.9(d) に cameraman 画像を B-spline 補間法により縦横 2 倍に拡大した画像を示す.

B-spline 補間法は, 0 次ホールド法や直線補間法に比べ滑らかに拡大することができるが, エッジ部分が不鮮明となる.

3 次畳み込み補間法

3 次畳み込み補間 (Cubic Convolution) 法の補間関数は式 (4.5) および 図 4.8(d) で表される [32].

$$h(t) = \begin{cases} (a+2)|t|^3 - (a+3)|t|^2 + 1 & (0 \leq |t| < 1) \\ a|t|^3 - 5a|t|^2 + 8a|t| - 4a & (1 \leq |t| < 2) \\ 0 & (2 \leq |t|) \end{cases} \quad (4.5)$$

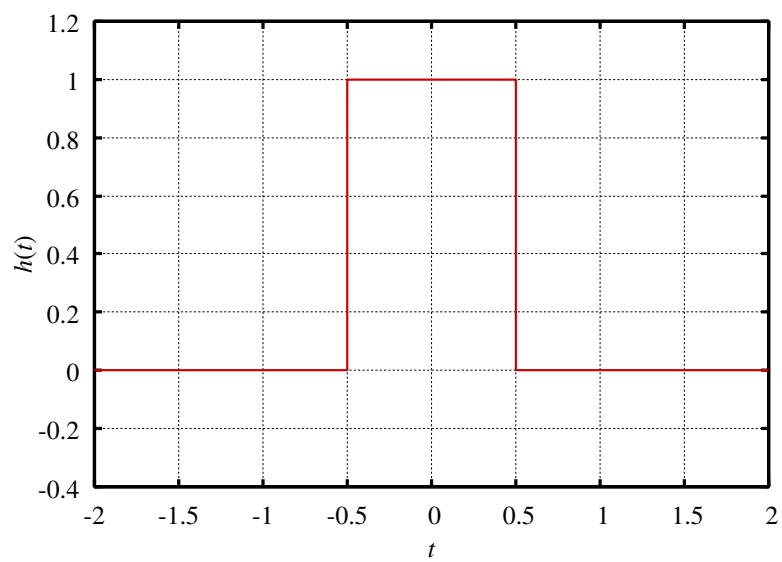
ここで a は定数であり, 定数 a の選択により補間特性を制御できる.

図 4.9(e) に $a = -0.5$ のときの 3 次畳み込み補間法での拡大画像を示す. 3 次畳み込み補間法は, B-spline 補間法と同様滑らかに拡大することができる. また, B-spline 補間法に比べエッジ部分を鮮明に拡大できる.

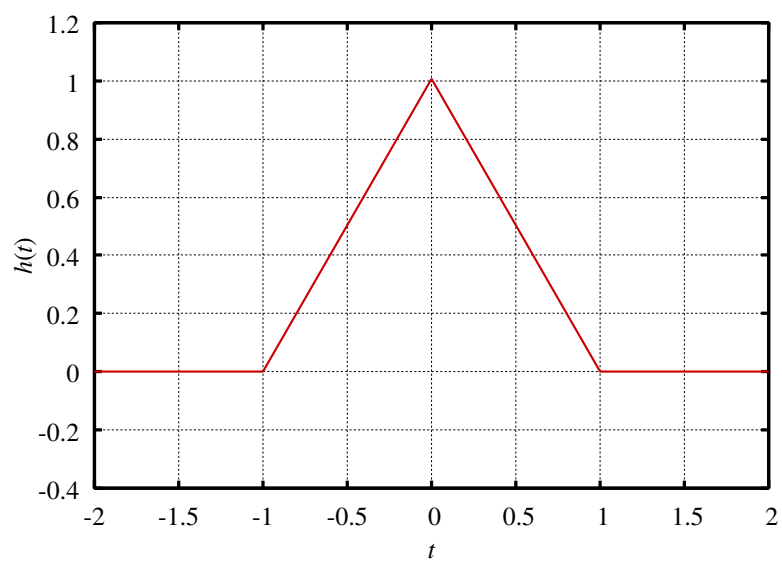
4.1.3 任意の倍率のサイズ変換

ここまでは, 整数倍の拡大, 整数分の 1 倍の縮小について述べてきたが, ここでは任意の倍率のサイズ変換について考える. つまり画像を横に W_1/W_2 倍, 縦に H_1/H_2 倍にする場合である. ただし, W_1, W_2, H_1, H_2 は正の整数である. ここで, $W_1 > W_2, H_1 > H_2$ ならば, 画像の拡大, $W_1 < W_2, H_1 < H_2$ ならば, 画像の縮小に相当する. この場合, 図

第 4 章 解像度変換技術

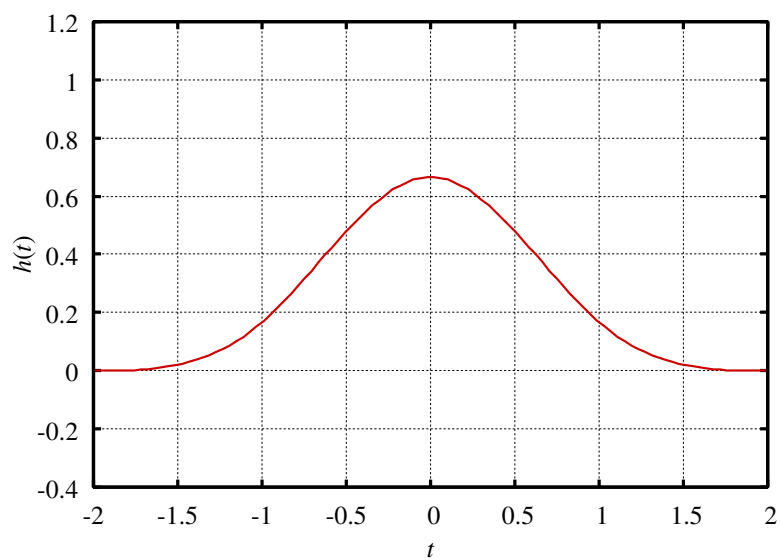


(a) 0 次ホールド法

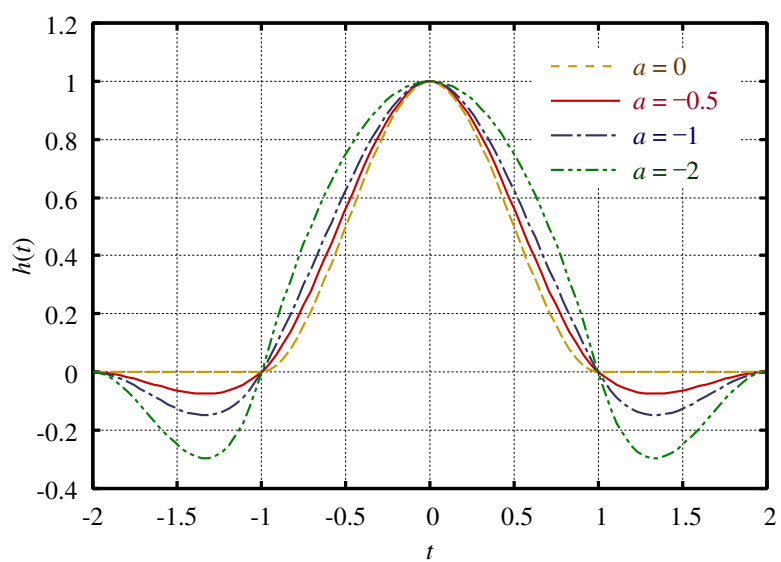


(b) 直線補間法

図 4.8 各拡大法の補間関数



(c) B-spline 補間法



(d) 3 次畳み込み補間法

図 4.8 各拡大法の補間関数 (続き)

第 4 章 解像度変換技術



(a) 原画像



(b) 0 次ホールド法



(c) 直線補間法



(d) B-spline 補間法



(e) 3 次畳み込み補間法 ($a = -0.5$)

図 4.9 拡大画像の比較 (縦横 2 倍)

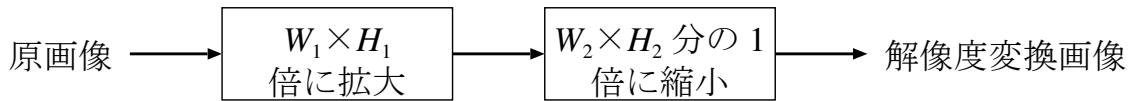


図 4.10 任意の倍率の解像度変換

4.10 のようにまず画像を横に W_1 倍、縦に H_1 倍に拡大し、その後画像を横に $1/W_2$ 倍、縦に $1/H_2$ 倍に縮小する。このように、画像の解像度変換は整数倍の拡大と整数分の 1 倍の縮小を組み合わせることで任意の倍率に解像度変換を行うことができる。

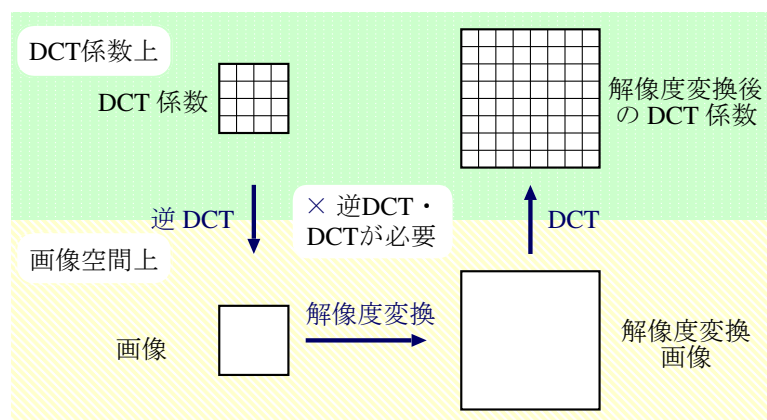
4.2 DCT 空間

解像度変換は、通常画像空間上で行われる。しかし、JPEG や MPEG の符号化データである DCT 係数を画像空間上で解像度変換するには、図 4.11(a) のように解像度変換前の DCT 係数を一度逆 DCT を行い画像に変換し、画像空間上で解像度変換を行い、その後再び DCT する操作が必要となり処理に無駄が生じる。そこで、DCT 係数上でそのまま解像度変換する方法が注目されている。DCT 係数上での解像度変換は、図 4.11(b) のように解像度変換前の DCT 係数から解像度変換後の DCT 係数を DCT 領域上でそのまま求めるため、処理に無駄が生じない。本章では、この DCT 係数上での解像度変換の手法について説明する。

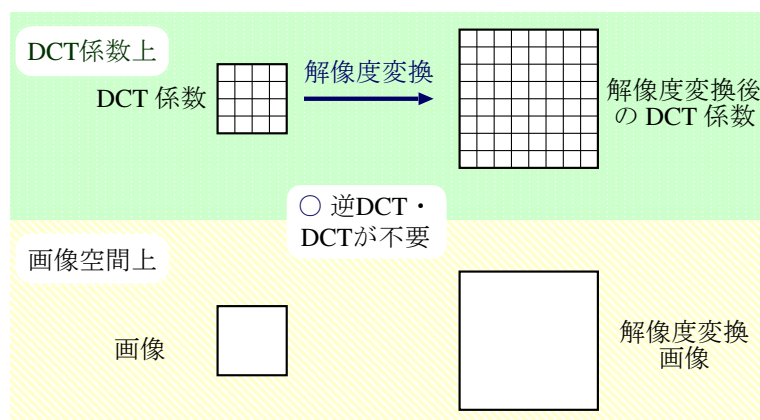
4.2.1 ブロックサイズ変更

DCT 係数上での最も一般的な解像度変換法は、解像度変換前の DCT のブロックサイズ ($N \times N$ 点) と解像度変換後の DCT のブロックサイズ ($P \times P$ 点) を変更することで解像度変換を行うものである [9]。図 4.12 に具体的な解像度変換の例を示す。この方法で画像を拡大する場合 ($N > P$ のとき)、 $N \times N$ 点の DCT 係数の高域側 (外側) に $P \times P$ 点になるように 0 値を付加する。また画像を縮小する場合 ($N < P$ のとき)、 $N \times N$ 点の DCT 係数から低域側 (内側) の $P \times P$ 点のデータだけを残し、それ以外の高域側のデータは削除する。ただし、 $1/N$ 倍以下の縮小は実現できない。高域側のデータを操作し

第4章 解像度変換技術



(a) 画像空間上での解像度変換

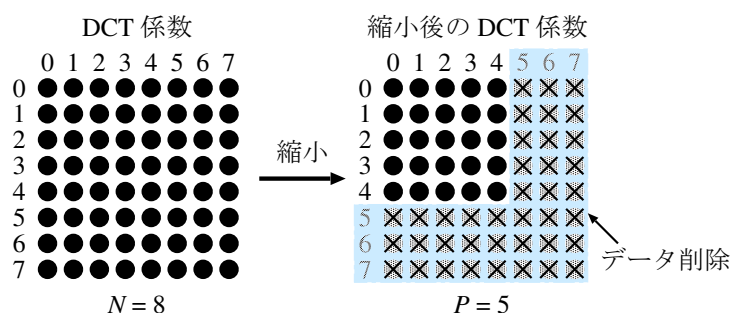


(b) DCT 係数上での解像度変換

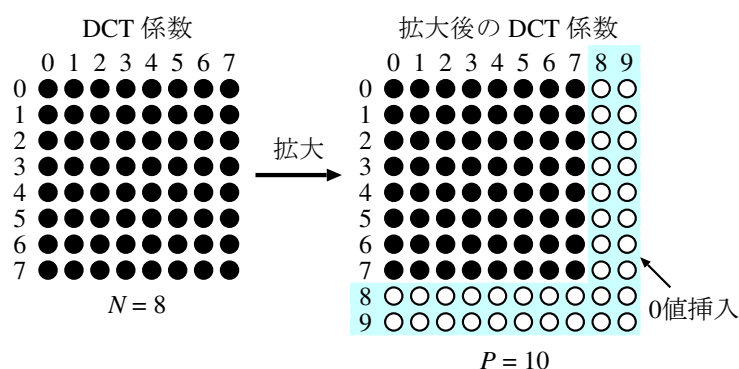
図 4.11 DCT 係数の解像度変換

得られた DCT 係数から解像度変換画像を再生するには、解像度変換後のデータ点に応じた $P \times P$ 点符号化器を用意する必要がある。また以上の操作を単純に実行すると、原画像と輝度値が異なる。そこで、解像度変換後の各画素値を P/N 倍して輝度調整を行う必要がある。

JPEG や MPEG では 8×8 点 DCT 符号化器が用いられている。しかし、この方法により得られた解像度変換後の DCT 係数は 8×8 点ではない。そのため、JPEG 等の 8×8 点の汎用符号化器を用いるためにはブロックサイズ変換を行う必要がある。ブロックサイ



(a) 縮小 (縦横 5/8 倍)



(b) 拡大 (縦横 10/8 倍)

図 4.12 DCT 係数上での解像度変換 (ブロックサイズ変更)

ズ変換は $1/2$, $1/4$ 倍, $1/8$ 倍の縮小時は, 複数ブロックを 1 つのブロックに変換することで実現できる. また, 整数倍の拡大時は 1 ブロックを 8×8 点の複数ブロックに変換することで実現できる. しかし, それ以外の倍率の場合は容易には実行できない.

4.2.2 ブロック数変更

4.2.1 節の手法では, 解像度変換後のブロックサイズが変更されてしまい, JPEG 等の汎用 DCT 符号化器にそのまま適用することはできない. そこで, ブロックサイズではなくブロック数を変更し解像度変換することで, JPEG 等の汎用符号化器にそのまま適用できる手法を示す [10]. 本論文では, この手法を従来法として扱う.

第 4 章 解像度変換技術

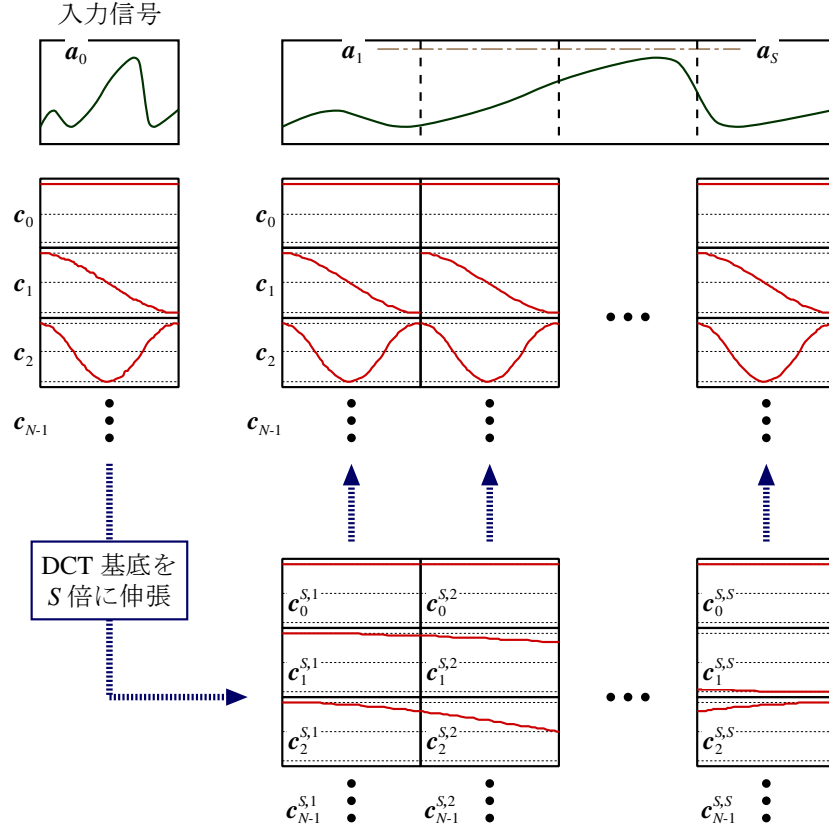


図 4.13 従来法の拡大原理

原理

画像サイズを P_1/P_2 倍に拡大または縮小する場合, $P_2 \times P_2$ ブロックの DCT 係数を演算処理し, $P_1 \times P_1$ ブロックの DCT 係数に変換する. ここでは簡単のため, N 点 1 次元 DCT を例にとり説明する.

まず, 拡大変換の原理について述べる. ある 1 次元ブロック領域の信号波形をベクトル a_0 で表し, これを S 倍に伸張する例を図 4.13 に示す. DCT の基底波形をベクトル c_0, c_1, \dots, c_{N-1} とし, a_0 の DCT 係数を $f_0^0, f_0^1, \dots, f_0^{N-1}$ とすると, 式 (4.6) で表される.

$$a_0 = \sum_{i=0}^{N-1} f_0^i c_i \quad (4.6)$$

ここで a_0 を S 等分に分割し, 波形ベクトル a_1, a_2, \dots, a_S を得るとする. これに対応さ

4.2 DCT 空間

せて DCT 基底波形も S 倍した後に S 等分に分割する. 分割後の k 番目のサブブロックの基底ベクトルを $c_0^{S,k}, c_1^{S,k}, \dots, c_{N-1}^{S,k}$ とすると, このサブブロックの信号波形ベクトル a_k は式 (4.7) で表される.

$$a_k = \sum_{i=0}^{N-1} f_0^i c_i^{S,k} \quad (4.7)$$

一方で, a_k を通常の DCT で表したときの DCT 係数を $f_k^0, f_k^1, \dots, f_k^{N-1}$ とすると, 式 (4.8) でも表される.

$$a_k = \sum_{j=0}^{N-1} f_k^j c_j \quad (4.8)$$

式 (4.7), (4.8) より拡大後のブロックの DCT 係数を求めると,

$$\sum_{i=0}^{N-1} f_0^i c_i^{S,k} = \sum_{j=0}^{N-1} f_k^j c_j \quad (4.9)$$

となり, DCT の原理より, 両辺と DCT 基底ベクトル c_0, c_1, \dots, c_{N-1} との内積をとり, $f_k^0, f_k^1, \dots, f_k^{N-1}$ を求める.

$$\sum_{l=0}^{N-1} \sum_{i=0}^{N-1} f_0^i c_l = \sum_{l=0}^{N-1} \sum_{j=0}^{N-1} f_k^j c_l \quad (4.10)$$

ここで, DCT の直交性

$$c_j \cdot c_l = \delta_{j,l} \quad (4.11)$$

より, 式 (4.10) は次式のように表現できる.

$$f_k^j = \sum_{i=0}^{N-1} u_k^{i,j} f_0^i \quad (4.12)$$

$$u_k^{i,j} = c_i^{S,k} \cdot c_j \quad (4.13)$$

つまり拡大変換フィルタ $U_k = [u_k^{i,j}]$ を用い, もとの DCT 係数 $f_0^0, f_0^1, \dots, f_0^{N-1}$ から拡大後の DCT 係数をブロック毎に求めることができる. この拡大変換はサブブロック単位での信号の表現形式を基底ベクトル $c_0^{S,k}, c_1^{S,k}, \dots, c_{N-1}^{S,k}$ による表現から通常の DCT 基底ベクトル c_0, c_1, \dots, c_{N-1} の表現に置換するものである. c_0, c_1, \dots, c_{N-1} は

第4章 解像度変換技術

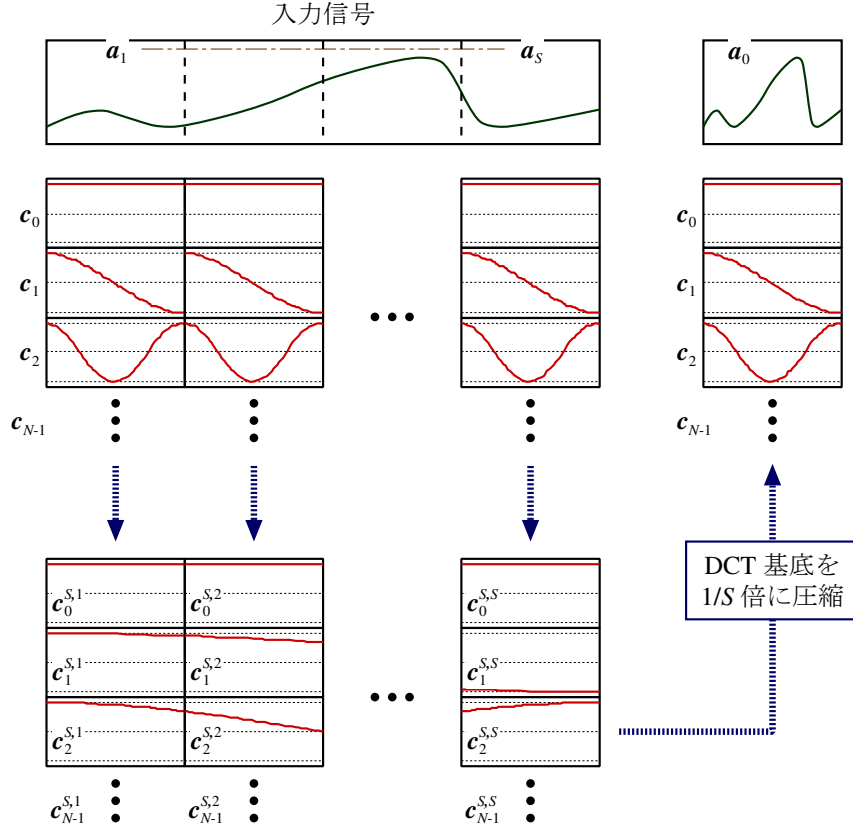


図 4.14 従来法の縮小原理

$c_0^{S,k}, c_1^{S,k}, \dots, c_{N-1}^{S,k}$ に対して完備であるため、この拡大変換はもとの信号波形 a_0 の帯域を保持する。

次に、縮小の原理について示す。図 4.14 は連続した S ブロックの波形ベクトルを a_1, a_2, \dots, a_S で表し、これを $1/S$ 倍に縮小して 1 個のブロック a_0 に対応づける例である。ここで拡大変換の場合と同様に、基底ベクトル c_0, c_1, \dots, c_{N-1} と $c_0^{S,k}, c_1^{S,k}, \dots, c_{N-1}^{S,k}$ を定義する。縮小前の k ブロック目の波形ベクトル a_k は式 (4.14) で表される。

$$a_k = \sum_{j=0}^{N-1} f_k^j c_j \quad (4.14)$$

縮小変換は a_k を $c_0^{S,k}, c_1^{S,k}, \dots, c_{N-1}^{S,k}$ で表現する形式に置換する。 a_k と $c_i^{S,k}$ との内積

をとり、その変換係数を $f_{0,k}^i$ とすると、

$$f_{0,k}^i = \mathbf{a}_k \cdot \mathbf{c}_i^{S,k} \quad (4.15)$$

となる。式 (4.14), (4.15) より、

$$f_{0,k}^i = \sum_{j=0}^{N-1} v_k^{i,j} f_k^j \quad (4.16)$$

となり、ここで、

$$v_k^{i,j} = \mathbf{c}_j \cdot \mathbf{c}_i^{S,k} \quad (4.17)$$

とおくと、 $\mathbf{c}_0^{S,k}, \mathbf{c}_1^{S,k}, \dots, \mathbf{c}_{N-1}^{S,k}$ は $\mathbf{c}_0, \mathbf{c}_1, \dots, \mathbf{c}_{N-1}$ に対して完備ではないため、 $f_k^0, f_k^1, \dots, f_k^{N-1}$ を $f_0^0, f_0^1, \dots, f_0^{N-1}$ に変換する過程で信号成分の帯域制限が行われる。

次に、縮小前の S ブロックの波形ベクトル $(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_S)$ を連結して 1 つの波形ベクトルとみなし、同様に連結した基底ベクトル $(\mathbf{c}_i^{S,1}, \mathbf{c}_i^{S,2}, \dots, \mathbf{c}_i^{S,S})$ との内積をとる。

$$(\mathbf{c}_i^{S,1}, \mathbf{c}_i^{S,2}, \dots, \mathbf{c}_i^{S,S}) \cdot (\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_S) = \sum_{k=1}^S f_{0,k}^i \quad (4.18)$$

ブロック長を正規化して考えると、 $(\mathbf{c}_i^{S,1}, \mathbf{c}_i^{S,2}, \dots, \mathbf{c}_i^{S,S})$ と通常の DCT 基底 \mathbf{c}_i が表す信号帯域は等しく、式 (4.18) と $(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_S)$ を縮小した波形ベクトル \mathbf{a}_0 を DCT 表現した式

$$f_0^i = \mathbf{c}_i \cdot \mathbf{a}_0 \quad (4.19)$$

は等価であり、式 (4.18), (4.19) より整理すると次の関係を得る。

$$\begin{aligned} f_0^i &= \sum_{k=1}^S f_{0,k}^i \\ &= \sum_{k=1}^S \sum_{j=0}^{N-1} v_k^{j,i} f_k^j \end{aligned} \quad (4.20)$$

サイズ変換

4.2.2 節の原理に基づき、整数倍の拡大および整数分の 1 倍の縮小が実現できる。ここでは、その中でも最も基本的な縦横 2 倍の拡大と縦横 1/2 倍の縮小について示す。図 4.15

第 4 章 解像度変換技術

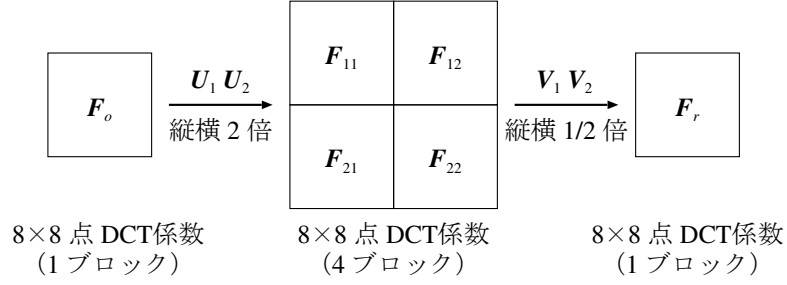


図 4.15 DCT 係数上での解像度変換 (ブロック数変更)

にブロック単位での縦横 2 倍の拡大および縦横 1/2 倍の縮小の処理の流れを示す。拡大前の DCT 係数 1 ブロック F_o を 4 ブロックの DCT 係数 $F_{11}, F_{12}, F_{21}, F_{22}$ に変換することで縦横 2 倍の拡大を実現する。実際には、式 (4.21) ~ (4.24) の行列演算により拡大を行う。

$$F_{11} = U_1 F_o U_1^t \quad (4.21)$$

$$F_{12} = U_2 F_o U_1^t \quad (4.22)$$

$$F_{21} = U_1 F_o U_2^t \quad (4.23)$$

$$F_{22} = U_2 F_o U_2^t \quad (4.24)$$

ここで、 U_1, U_2 は 2 倍の拡大に用いるフィルタで、それぞれ $(c_0^{2,1}, c_1^{2,1}, \dots, c_{N-1}^{2,1})$ と $(c_0^{2,2}, c_1^{2,2}, \dots, c_{N-1}^{2,2})$ とを 1 次元 DCT して得られ、式 (4.25), (4.26) で表される。

$$U_1 = \begin{bmatrix} 1 & 0.9018 & 0 & -0.3045 & 0 & 0.1875 & 0 & -0.1393 \\ 0 & 0.4223 & 1 & 0.7702 & 0 & -0.3138 & 0 & 0.2134 \\ 0 & -0.0827 & 0 & 0.5391 & 1 & 0.7185 & 0 & -0.2862 \\ 0 & 0.0341 & 0 & -0.1345 & 0 & 0.5668 & 1 & 0.7030 \\ 0 & -0.0177 & 0 & 0.0617 & 0 & -0.1500 & 0 & 0.5748 \\ 0 & 0.0100 & 0 & -0.0332 & 0 & 0.0698 & 0 & -0.1525 \\ 0 & 0.0056 & 0 & 0.0181 & 0 & -0.0357 & 0 & 0.0673 \\ 0 & 0.0025 & 0 & -0.0081 & 0 & 0.0155 & 0 & -0.0277 \end{bmatrix} \quad (4.25)$$

$$U_2 = \begin{bmatrix} 1 & -0.9018 & 0 & 0.3045 & 0 & -0.1875 & 0 & 0.1393 \\ 0 & 0.4223 & -1 & 0.7702 & 0 & -0.3138 & 0 & 0.2134 \\ 0 & 0.0827 & 0 & -0.5391 & 1 & -0.7185 & 0 & 0.2862 \\ 0 & 0.0341 & 0 & -0.1345 & 0 & 0.5668 & -1 & 0.7030 \\ 0 & 0.0177 & 0 & -0.0617 & 0 & 0.1500 & 0 & -0.5748 \\ 0 & 0.0100 & 0 & -0.0332 & 0 & 0.0698 & 0 & -0.1525 \\ 0 & 0.0056 & 0 & -0.0181 & 0 & 0.0357 & 0 & -0.0673 \\ 0 & 0.0025 & 0 & -0.0081 & 0 & 0.0155 & 0 & -0.0277 \end{bmatrix} \quad (4.26)$$

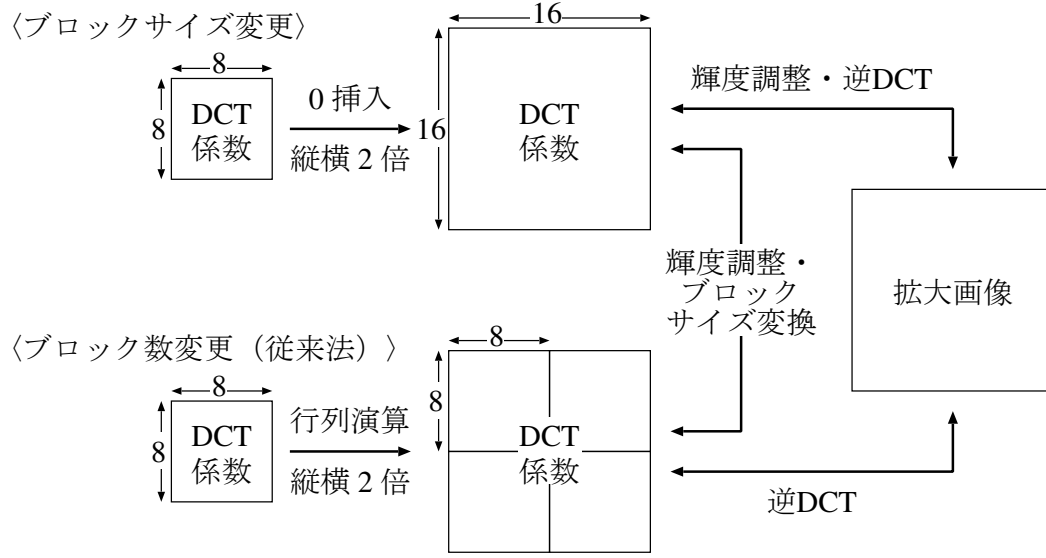


図 4.16 2 通りの拡大方法の対応関係

次に, 4 ブロックの DCT 係数 $F_{11}, F_{12}, F_{21}, F_{22}$ から 1 ブロックの DCT 係数 F_r を求めることで縦横 1/2 倍の縮小を実現する. 実際には, 式 (4.27) ~ (4.29) の行列演算により縮小を行う.

$$F_1 = V_1 F_{11} + V_2 F_{21} \quad (4.27)$$

$$F_2 = V_1 F_{12} + V_2 F_{22} \quad (4.28)$$

$$F_r = F_1 V_1^t + F_2 V_2^t \quad (4.29)$$

ここで, V_1, F_2 は, 一次元 DCT の基底 $(c_0, c_1, \dots, c_{N-1})$ と $(c_0^{2,1}, c_1^{2,1}, \dots, c_{N-1}^{2,1})$ および $(c_0^{2,1}, c_1^{2,1}, \dots, c_{N-1}^{2,1})$ の内積により求められ, 式 (4.30), (4.31) で表される.

従来法と 4.2.1 節 のブロックサイズを変更して解像度変換する手法の対応関係を 図 4.16 に示す. この 2 通りの手法により得られる解像度変換後の画像は等しくなる.

$$V_1 = \begin{bmatrix} 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.4509 & 0.2112 & -0.0414 & 0.0170 & -0.0088 & 0.0050 & -0.0028 & 0.0013 \\ 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.1522 & 0.3851 & 0.2696 & -0.0672 & 0.0309 & -0.0166 & 0.0090 & -0.0040 \\ 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 \\ 0.0938 & -0.1569 & 0.3592 & 0.2834 & -0.0750 & 0.0349 & -0.0179 & 0.0078 \\ 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 \\ -0.0697 & 0.1067 & -0.1431 & 0.3515 & 0.2874 & -0.0762 & 0.0336 & -0.0138 \end{bmatrix} \quad (4.30)$$

$$V_2 = \begin{bmatrix} 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.4509 & 0.2112 & 0.0414 & 0.0170 & 0.0088 & 0.0050 & 0.0028 & 0.0013 \\ 0 & -0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.1522 & 0.3851 & -0.2696 & -0.0672 & 0.0309 & -0.0166 & 0.0090 & -0.0040 \\ 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 \\ -0.0938 & -0.1569 & -0.3592 & 0.2834 & 0.0750 & 0.0349 & 0.0179 & 0.0078 \\ 0 & 0 & 0 & -0.5 & 0 & 0 & 0 & 0 \\ 0.0697 & 0.1067 & 0.1431 & 0.3515 & -0.2874 & -0.0762 & -0.0336 & -0.0138 \end{bmatrix} \quad (4.31)$$

4.2.3 提案法

4.2.2 節の手法は、ブロックごとに処理を行うため、解像度変換後の画像にブロックひずみが発生するという問題があった。そこで、本研究では解像度変換後の画像にブロックひずみが発生しない DCT 係数上での解像度変換を提案する。本手法は、ブロックごとに独立して解像度変換を行うのではなく、周辺の DCT 係数を用いブロック間の相関を考慮して解像度変換を行うため、解像度変換後の画像にブロックひずみが発生しない。具体的には、画像空間上での解像度変換と等価な処理を DCT 係数上で実現する。

行列演算の性質

提案法では、DCT 係数上での解像度変換を実現するのに多くの演算を行う。そのため、提案法は行列演算により処理を行う。行列を用いるのはこれらの操作をまとめて行うためであり、行列演算により演算量を減らすことができる。

例えば、ある行列 X に右から行列 A, B, C の順でかけ行列 Y を求める演算は次式で表される。

$$Y = XABC \quad (4.32)$$

しかし、この式は行列 ABC をかけた行列を M とおくと次式のようにも表される。

$$M = ABC \quad (4.33)$$

$$Y = XM \quad (4.34)$$

このように、あらかじめ行列 ABC をかけた行列 M を求めておくことで、演算量を減らすことができる。

解像度変換の手順

縮小手順 提案法で縦横 $1/2$ 倍に縮小する流れを図 4.17 に示す。縮小前の 4 ブロックの 8×8 点 DCT 係数を逆 DCT して画像に変換し、画像空間上で縮小を行い、再び DCT して 8×8 点の DCT 係数 1 ブロックを得る操作を行列演算を用い、まとめて実行する。

第4章 解像度変換技術

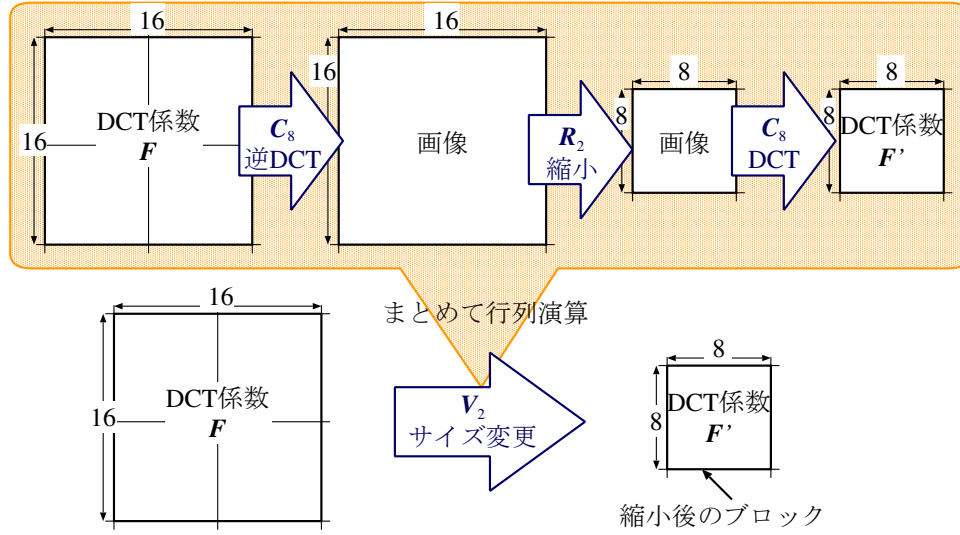


図 4.17 提案法での縮小 (縦横 1/2 倍)

画像空間上で画像を整数分の 1 倍に縮小する場合、一般にブロックひずみは発生しない。そこで、提案法での縮小は隣接のブロックの参照は行わない。提案法で縦横 $1/N$ 倍に縮小する場合、 $N \times N$ ブロックの 8×8 点の DCT 係数 (つまり、 $8N \times 8N$ 点の DCT 係数) F を式 (4.35) を用い、1 ブロックの 8×8 点の DCT 係数 F' に変換する。

$$F' = V_N F V_N^t \quad (4.35)$$

ここで、縮小行列 V_N は 8×8 点の DCT 基底行列 C_8 、縮小フィルタの行列 R_N と、 8×8 点の DCT 基底行列 C_8 を $N \times N$ 倍に対角化した行列 D_{8N} の積より導出できる。縮小行列 V_N は、次式により求められる。

$$V_N = C_8 R_N D_{8N} \quad (4.36)$$

ここで、 R_N は縮小フィルタの補間関数により求められる $8 \times 8N$ 点の行列で、次節で詳しく述べる。また D_{8N} は、

$$D_{8N} = \text{diag} \left\{ \underbrace{C_8^t, \dots, C_8^t}_N \right\} \quad (4.37)$$

である。

拡大手順 提案法での縦横 N 倍の拡大手順を以下に示す。

1. ブロックサイズ変換 (8×8 点 $\rightarrow 2 \times 2$ 点)
2. 隣接ブロックを用い縦横 N 倍に拡大
3. ブロックサイズ変換 (2×2 点 $\rightarrow 8 \times 8$ 点)

図 4.18(a) に手順 1. の流れを示す. 手順 1. では, 8×8 点の DCT 係数を逆 DCT して画像に変換し, その後 2×2 点 DCT を行い, 再び DCT 係数を得る操作を行列演算を用い, まとめて実行する.

手順 1. では, 8×8 点 DCT 係数 F_a 1 ブロックから式 (4.38) を用いて 16 ブロック (4×4 ブロック) の 2×2 点 DCT 係数 F'_a へブロックサイズを線形変換する.

$$F'_a = A F_a A^t \quad (4.38)$$

ここで, 基底変換行列 A は, 2×2 点の DCT 基底行列 C_2 を 8×8 点に対角化した行列と 8×8 点の DCT 基底行列 C_8 の積で表され, 式 (4.39) で表される.

$$A = \begin{bmatrix} C_2 & O_2 & O_2 & O_2 \\ O_2 & C_2 & O_2 & O_2 \\ O_2 & O_2 & C_2 & O_2 \\ O_2 & O_2 & O_2 & C_2 \end{bmatrix} \cdot C_8^t \quad (4.39)$$

ただし,

$$C_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (4.40)$$

$$O_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (4.41)$$

$$C_8 = \frac{1}{2} \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ C_{16}^1 & C_{16}^3 & C_{16}^5 & C_{16}^7 & C_{16}^9 & C_{16}^{11} & C_{16}^{13} & C_{16}^{15} \\ C_{16}^2 & C_{16}^6 & C_{16}^{10} & C_{16}^{14} & C_{16}^{18} & C_{16}^{22} & C_{16}^{26} & C_{16}^{30} \\ C_{16}^3 & C_{16}^9 & C_{16}^{15} & C_{16}^{21} & C_{16}^{27} & C_{16}^{33} & C_{16}^{39} & C_{16}^{45} \\ C_{16}^4 & C_{16}^{12} & C_{16}^{20} & C_{16}^{28} & C_{16}^{36} & C_{16}^{44} & C_{16}^{52} & C_{16}^{60} \\ C_{16}^5 & C_{16}^{15} & C_{16}^{25} & C_{16}^{35} & C_{16}^{45} & C_{16}^{55} & C_{16}^{65} & C_{16}^{75} \\ C_{16}^6 & C_{16}^{18} & C_{16}^{30} & C_{16}^{42} & C_{16}^{54} & C_{16}^{66} & C_{16}^{78} & C_{16}^{90} \\ C_{16}^7 & C_{16}^{21} & C_{16}^{35} & C_{16}^{49} & C_{16}^{63} & C_{16}^{77} & C_{16}^{91} & C_{16}^{105} \end{bmatrix} \quad (4.42)$$

第4章 解像度変換技術

である. ここで, $C_l^i = \cos(i\pi/l)$ であり, よって, 基底変換行列 A は, 式 (4.43) のようになる.

$$A = \begin{bmatrix} 0.5 & 0.6407 & 0.4619 & 0.2250 & 0 & -0.1503 & -0.1913 & -0.1274 \\ 0 & 0.0528 & 0.1913 & 0.3629 & 0.5 & 0.5432 & 0.4619 & 0.2654 \\ 0.5 & 0.2654 & -0.4619 & -0.5432 & 0 & 0.3629 & 0.1913 & -0.0528 \\ 0 & 0.1274 & 0.1913 & -0.1503 & -0.5 & -0.2250 & 0.4619 & 0.6407 \\ 0.5 & -0.2654 & -0.4619 & 0.5432 & 0 & -0.3629 & 0.1913 & 0.0528 \\ 0 & 0.1274 & -0.1913 & -0.1503 & 0.5 & -0.2250 & -0.4619 & 0.6407 \\ 0.5 & -0.6407 & 0.4619 & -0.2250 & 0 & 0.1503 & -0.1913 & 0.1274 \\ 0 & 0.0528 & -0.1913 & 0.3629 & -0.5 & 0.5432 & -0.4619 & 0.2654 \end{bmatrix} \quad (4.43)$$

8×8 点から 2×2 点へのブロックサイズ変換は, 次の手順 2. で扱う行列のサイズを小さくするために実行する. これにより演算回数が減少できる.

次に手順 2. では, 2×2 点の DCT 係数を拡大したいブロック (1 ブロック) に隣接ブロック (8 ブロック) を加えた計 9 ブロック (計 6×6 点) を用い拡大する.

$$G' = U_N G U_N^t \quad (4.44)$$

ここで, G, G' はそれぞれ拡大前, 拡大後の DCT 係数である. また拡大行列 U_N は, 2×2 点の DCT 基底行列 C_2 を $N \times N$ 倍に対角化した行列, 拡大フィルタの行列 M_N , 2×2 点の DCT 基底行列 C_2 を $3 \times N$ 倍に対角化した行列の積より導出できる $2N \times 6$ 点の行列である. 例えば縦横 2 倍に拡大する場合, 拡大行列 U_2 は次式により求められる.

$$U_2 = \begin{bmatrix} C_2 & O_2 \\ O_2 & C_2 \end{bmatrix} M_2 \begin{bmatrix} C_2 & O_2 \\ O_2 & C_2 \\ O_2 & O_2 \end{bmatrix} \quad (4.45)$$

ここで, 拡大フィルタの行列 M_N は N 倍の拡大を実現する演算を表した $2N \times 6$ 点の行列であり, 次節で詳しく述べる. この拡大フィルタを変更することで, 様々な補間法での拡大画像を得ることができる.

図 4.18(b) に手順 2. の流れを示す. 手順 2. では, 2×2 点の DCT 係数を逆 DCT し て画像に変換し, 画像空間上で解像度変換を行い, その後 DCT を行い再び DCT 係数を得る操作を行列演算を用いまとめて実行する.

最後に手順 3. では, 16 ブロック (4×4 ブロック) の 2×2 点 DCT 係数 F_c から式 (4.46) を用いて 1 ブロックの 8×8 点 DCT 係数 F'_c へブロックサイズを変換する.

$$F'_c = A^t F_c A \quad (4.46)$$

図 4.18(c) に手順 3. の流れを示す. 手順 3. では, 2×2 点の DCT 係数を逆 DCT して画像に変換し, その後 8×8 点 DCT を行い, 再び DCT 係数を得る. 以上の操作を行列演算を用いまとめて実行する.

DCT 係数上での鏡像反転 提案法では, 隣接ブロックを用いて拡大を行うが, 画像の端を拡大する場合, 隣接ブロックが存在しないため鏡像反転する必要がある. しかし, 本研究で扱うデータは DCT 係数となっているため画像空間上での鏡像反転と同様な処理はできない. そこで, DCT 係数上での鏡像反転処理について考える.

DCT 係数上での鏡像反転は, ブロック内の DCT 係数をコピーし一部分の符号を反転させることで実現できる [8]. 反転させる前の DCT 係数のブロックを F とし, 水平方向に反転させたブロックを F_H , 垂直方向に反転させたブロックを F_V , 水平・垂直方向に反転 (180° 回転) させたブロックを F_{HV} とすると

$$F_H = FV \quad (4.47)$$

$$F_V = VF \quad (4.48)$$

$$F_{HV} = VFV \quad (4.49)$$

ここで V は, $V = \text{diag}\{1, -1, 1, -1, \dots\}$ のように 1 と -1 を DCT のブロックサイズだけ対角化した行列であり, 提案法の場合, ブロックサイズ $N = 2$ より以下のようになる.

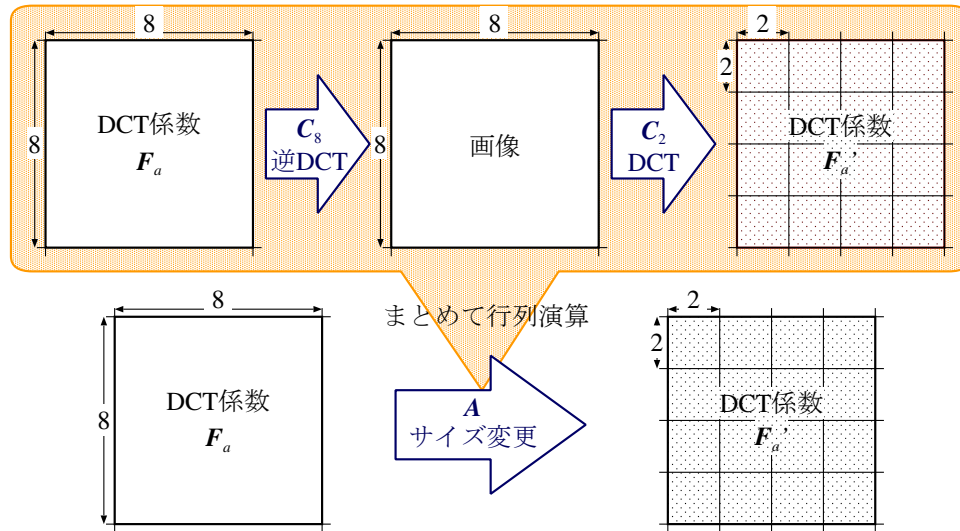
$$V = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad (4.50)$$

縮小行列・拡大行列

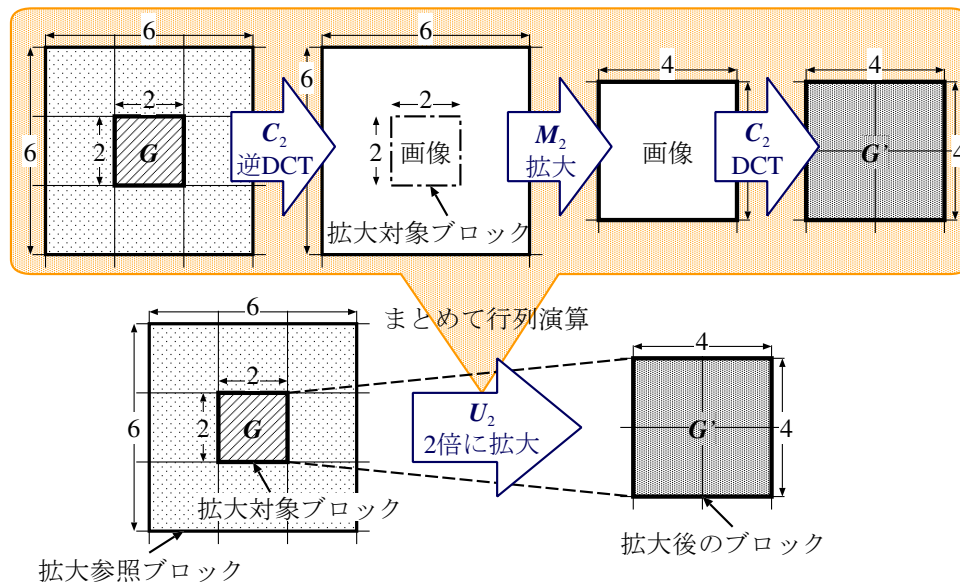
縮小行列 前節で用いた縮小行列 V_N の導出について示す. 簡単のため縦横 $1/2$ 倍に縮小する場合を考える. ダウンサンプリング法, 平均操作法の縮小フィルタの行列を R_{2D} , R_{2A} とすると, それぞれ式 (4.51), (4.52) で表される. これらは, 各補間法の補間関数より求めることができる.

式 (4.36) より各補間法の縮小行列を求める. ダウンサンプリング法, 平均操作法の縮小行列を V_{2D} , V_{2A} とすると, それぞれ式 (4.53), (4.54) で表される.

第4章 解像度変換技術



(a) DCT のブロックサイズ変更 (8×8 点 2×2 点)



(b) 縦横 2 倍に拡大

図 4.18 提案手法による拡大 (縦横 2 倍)

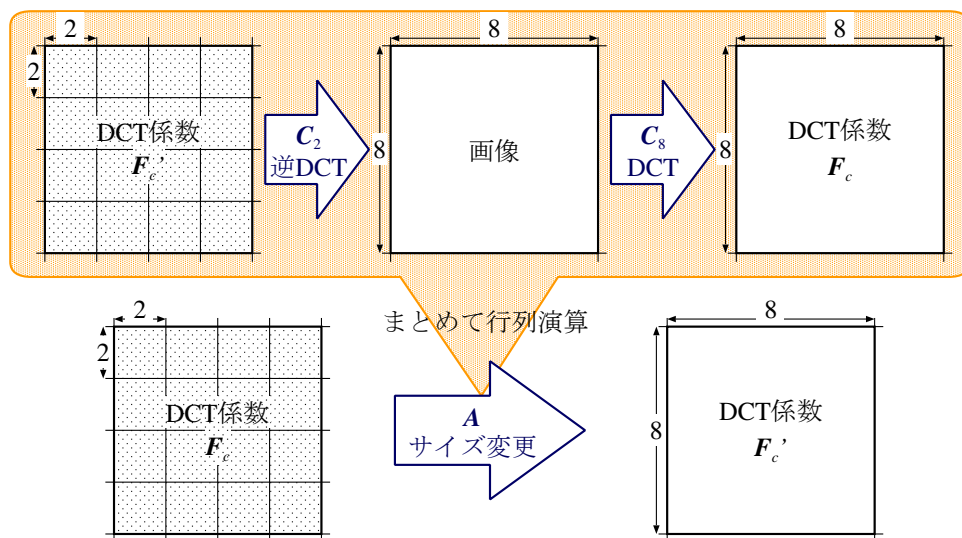
(c) DCT のブロックサイズ変更 (2×2 点 8×8 点)

図 4.18 提案手法による拡大 (縦横 2 倍) (続き)

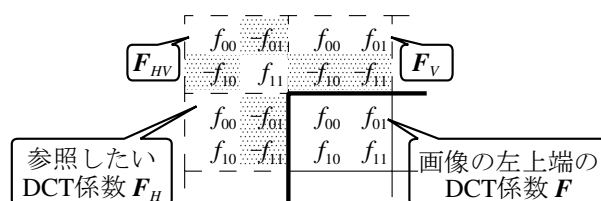


図 4.19 DCT 係数上での鏡像反転

[illegible]

[illegible]

$$V_{2D} = \begin{bmatrix} 0.5 & 0.09 & 0 & 0.11 & 0 & 0.16 & 0 & 0.45 & 0.5 & 0.09 & 0 & 0.11 & 0 & 0.16 & 0 & 0.45 \\ 0.45 & 0.29 & 0.04 & 0.09 & 0.09 & 0.11 & 0.19 & 0.38 & -0.45 & 0.12 & 0.11 & -0.07 & 0.09 & -0.12 & 0.16 & -0.47 \\ 0 & 0.49 & 0.18 & 0 & 0 & 0.19 & 0.43 & -0.10 & 0 & -0.49 & -0.18 & 0 & 0 & -0.19 & -0.43 & 0.10 \\ -0.16 & 0.34 & 0.38 & 0.06 & 0.11 & 0.17 & 0.26 & -0.32 & 0.16 & 0.44 & -0.09 & -0.14 & 0.11 & -0.12 & 0.45 & -0.16 \\ 0 & -0.04 & 0.46 & 0.26 & 0 & 0.38 & -0.19 & -0.19 & 0 & -0.04 & 0.46 & 0.26 & 0 & 0.38 & -0.19 & -0.19 \\ 0.11 & -0.18 & 0.26 & 0.42 & 0.16 & 0.19 & -0.38 & -0.01 & -0.11 & -0.16 & -0.45 & -0.02 & 0.16 & -0.47 & -0.09 & 0.08 \\ 0 & 0 & -0.07 & 0.42 & 0.46 & -0.28 & -0.18 & 0 & 0 & 0 & 0.07 & -0.42 & -0.46 & 0.28 & 0.18 & 0 \\ -0.09 & 0.13 & -0.19 & 0.24 & 0.45 & -0.41 & 0.04 & -0.04 & 0.09 & 0.14 & 0.16 & 0.48 & 0.45 & -0.07 & -0.11 & -0.02 \end{bmatrix} \quad (4.53)$$

$$V_{2A} = \begin{bmatrix} 0.5 & 0 & 0 & -0.03 & 0.01 & 0 & 0 & 0 & 0 \\ 0.45 & 0.20 & -0.03 & 0.01 & 0 & -0.01 & -0.01 & 0 & 0 \\ 0 & 0.49 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.16 & 0.39 & 0.24 & -0.04 & 0 & 0.03 & -0.10 & 0 & 0 \\ 0 & 0 & 0.46 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.11 & -0.17 & 0.35 & 0.20 & 0 & -0.14 & -0.15 & 0.03 & 0 \\ 0 & 0 & 0 & 0.42 & 0 & -0.28 & 0 & 0 & 0 \\ 0.09 & 0.14 & -0.17 & 0.36 & 0 & -0.24 & 0.07 & 0.14 & 0.17 \\ & & & & & & & & 0.5 \\ & & & & & & & & -0.45 \\ & & & & & & & & 0.20 \\ & & & & & & & & -0.49 \\ & & & & & & & & 0 \\ & & & & & & & & -0.24 \\ & & & & & & & & 0.39 \\ & & & & & & & & 0.16 \\ & & & & & & & & 0 \\ & & & & & & & & -0.17 \\ & & & & & & & & -0.35 \\ & & & & & & & & 0.20 \\ & & & & & & & & -0.42 \\ & & & & & & & & 0.28 \\ & & & & & & & & -0.07 \\ & & & & & & & & -0.03 \end{bmatrix}$$

(4.54)

拡大行列 前節では, 提案法の具体的な拡大手順について示した. ここでは, 前節の手順 2. で用いた拡大行列 U_N の導出について示す. 簡単のため縦横 2 倍に拡大する場合を考える. 0 次ホールド法, 直線補間法, B-spline 補間法, 3 次畳み込み補間法の拡大フィルタの行列をそれぞれ M_{2Z} , M_{2L} , M_{2B} , M_{2C} , とすると, 式 (4.55) ~ (4.58) で表される.

$$M_{2Z} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \quad (4.55)$$

$$M_{2L} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0.5 & 0.5 & 0 \end{bmatrix} \quad (4.56)$$

$$M_{2B} = \begin{bmatrix} 0 & 0.1667 & 0.6667 & 0.1667 & 0 & 0 \\ 0 & 0.0208 & 0.4792 & 0.4792 & 0.0208 & 0 \\ 0 & 0 & 0.1667 & 0.6667 & 0.1667 & 0 \\ 0 & 0 & 0.0208 & 0.4792 & 0.4792 & 0.0208 \end{bmatrix} \quad (4.57)$$

$$M_{2C} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0.125a & 0.5 - 0.125a & 0.5 - 0.125a & 0.125a & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0.125a & 0.5 - 0.125a & 0.5 - 0.125a & 0.125a \end{bmatrix} \quad (4.58)$$

これらの式は, 各補間法の補間関数より求めることができる.

式 (4.45) より各補間法の拡大行列を求める. 0 次ホールド法, 直線補間法, B-spline 補間法, 3 次畳み込み補間法の拡大行列をそれぞれ U_{2Z} , U_{2L} , U_{2B} , U_{2C} とすると, 式 (4.59) ~ (4.62) で表される.

$$\mathbf{U}_{2Z} = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (4.59)$$

$$\mathbf{U}_{2L} = \begin{bmatrix} 0 & 0 & 1 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0.75 & -0.75 & 0.25 & 0.25 \\ 0 & 0 & 0.25 & -0.25 & -0.25 & -0.25 \end{bmatrix} \quad (4.60)$$

$$\mathbf{U}_{2B} = \begin{bmatrix} 0.0938 & -0.0938 & 0.8958 & 0.25 & 0.0104 & 0.0104 \\ 0.0729 & -0.0729 & -0.0625 & 0.25 & -0.0104 & -0.0104 \\ 0 & 0 & 0.6667 & -0.4792 & 0.3333 & 0.3125 \\ 0 & 0 & 0.1667 & -0.0208 & -0.1667 & -0.1458 \end{bmatrix} \quad (4.61)$$

$$\mathbf{U}_{2C} = \begin{bmatrix} 0.0625a & -0.0625a & 1 - 0.125a & 0.5 & 0.0625a & 0.0625a \\ -0.0625a & 0.0625a & 0.125a & 0.5 & -0.0625a & -0.0625a \\ 0 & 0 & 0.75 & -0.75 + 0.125a & 0.25 & 0.25 - 0.125 \\ 0 & 0 & 0.25 & -0.25 - 0.125a & -0.25 & -0.25 + 0.125 \end{bmatrix} \quad (4.62)$$

4.2.4 提案法の高速度化

前節の方法では 3 段階の処理が行われているが, (2), (3) は冗長な処理を含んでいるため, 計算量が増加している. そこで最小限の隣接ブロックを用いることで, 画質の劣化を抑えつつ, より少ない計算量で解像度変換を行う方法を提案する. 提案法では, DCT 係数のブロック境界に対して拡大処理を行い, 拡大とブロックサイズ変換を同時に処理することで, 計算量を削減している. 提案法の詳細な処理手順を以下に示す.

1. 8 点 DCT 係数 1 ブロック X を式 (4.63) を用いて 4 点 DCT 係数 2 ブロック X_1, X_2 に等価変換する. 拡大時の処理として効率的に処理できることから, 4 点 DCT を用いる. ここで基底変換行列 A_4 は 4 点 DCT 行列 C_4 と 8 点 DCT 行列 C_8 の積より導出でき, 式 (4.64) で表される.

$$\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = A_4 X \quad (4.63)$$

$$A_4 = \begin{bmatrix} C_4 & O_4 \\ O_4 & C_4 \end{bmatrix} C_8^t$$

$$= \begin{bmatrix} 0.707 & 0.641 & 0 & -0.225 & 0 & 0.150 & 0 & -0.127 \\ 0 & 0.294 & 0.707 & 0.559 & 0 & -0.249 & 0 & 0.196 \\ 0 & -0.053 & 0 & 0.363 & 0.707 & 0.543 & 0 & -0.265 \\ 0 & 0.016 & 0 & -0.069 & 0 & 0.347 & 0.707 & 0.612 \\ 0.707 & -0.641 & 0 & 0.225 & 0 & -0.150 & 0 & 0.127 \\ 0 & 0.294 & -0.707 & 0.559 & 0 & -0.249 & 0 & 0.196 \\ 0 & 0.053 & 0 & -0.363 & 0.707 & -0.543 & 0 & 0.265 \\ 0 & 0.016 & 0 & -0.069 & 0 & 0.347 & -0.707 & 0.612 \end{bmatrix} \quad (4.64)$$

2. 4 点 DCT 係数 2 ブロックの間にまたがる領域を拡大する. ブロックの境界部分に相当する画像に対して拡大処理を行うことで, 必要なブロック数が減少する. また, 拡大処理と同時に出力する DCT 係数のブロックサイズを 8 点に変換することで, 前節の方法の (3) の処理が不要となり, 計算量を削減することができる. この処理は式 (4.65) で表される.

$$Y = B \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \quad (4.65)$$

第 4 章 解像度変換技術

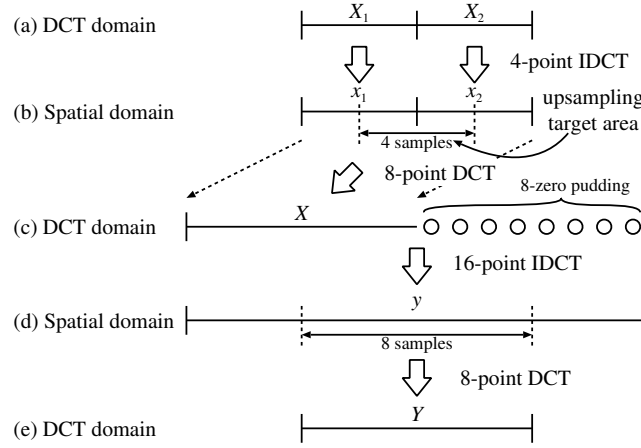


図 4.20 提案法の拡大フィルタ処理

ここで，拡大フィルタ B として 前節の方法と同様に DCT 係数の高域成分に 0 値を補間するフィルタを考えた場合，拡大フィルタは図 4.20 に示す処理を行うフィルタとなり，これは次式で表される．

$$\begin{aligned}
 B &= \sqrt{2} C_8 \begin{bmatrix} O_4 & O_4 \\ I_8 \\ O_4 & O_4 \end{bmatrix}^t C_{16}^t \begin{bmatrix} I_8 \\ O_8 \end{bmatrix} C_8 \begin{bmatrix} C_4 & O_4 \\ O_4 & C_4 \end{bmatrix}^t \\
 &= \begin{bmatrix} 0.707 & -0.641 & 0 & 0.225 & 0.707 & 0.641 & 0 & -0.225 \\ 0.656 & -0.485 & -0.266 & 0.424 & -0.656 & -0.485 & 0.266 & 0.424 \\ 0 & 0.294 & -0.707 & 0.559 & 0 & -0.294 & -0.707 & -0.559 \\ -0.259 & 0.516 & -0.633 & 0.318 & 0.259 & 0.516 & 0.633 & 0.318 \\ 0 & 0.053 & 0 & -0.363 & 0 & -0.053 & 0 & 0.363 \\ 0.011 & 0.013 & 0.020 & -0.169 & -0.011 & 0.013 & -0.020 & -0.169 \\ 0 & 0.016 & 0 & -0.069 & 0 & -0.016 & 0 & 0.069 \\ 0.001 & 0.005 & 0.002 & -0.032 & -0.001 & 0.005 & -0.002 & -0.032 \end{bmatrix} \quad (4.66)
 \end{aligned}$$

また， $B = [B_L \ B_R]$ (B_L ， B_R はそれぞれ B の左右の 8×4 の行列) とし，

$B_L = D + E$ ， $B_R = D - E$ と置くと，式 (4.65) は，

$$Y = (D + E)X_1 + (D - E)X_2 \quad (4.67)$$

$$= D(X_1 + X_2) + E(X_1 - X_2) \quad (4.68)$$

4.3 実験結果

$$D = \begin{bmatrix} 0.707 & 0 & 0 & 0 \\ 0 & -0.485 & 0 & 0.424 \\ 0 & 0 & -0.707 & 0 \\ 0 & 0.516 & 0 & 0.318 \\ 0 & 0 & 0 & 0 \\ 0 & 0.013 & 0 & -0.169 \\ 0 & 0 & 0 & 0 \\ 0 & 0.005 & 0 & -0.032 \end{bmatrix} \quad (4.69)$$

$$E = \begin{bmatrix} 0 & -0.641 & 0 & 0.225 \\ 0.656 & 0 & -0.266 & 0 \\ 0 & 0.294 & 0 & 0.559 \\ -0.259 & 0 & -0.633 & 0 \\ 0 & 0.053 & 0 & -0.363 \\ 0.011 & 0 & 0.020 & 0 \\ 0 & 0.016 & 0 & -0.069 \\ 0.001 & 0 & 0.002 & 0 \end{bmatrix} \quad (4.70)$$

となる. ここで, D , E の要素は 0 値が多く含まれるため, 行列演算を考えた場合, 式 (4.68) は式 (4.67) より計算量が少なくなる.

4.3 実験結果

提案法である隣接ブロックを用いた解像度変換法の性能を調べるため, 提案した解像度変換法を C 言語により実装した. また比較対象として, 画像空間上での解像度変換法, ブロックサイズ変更による DCT 係数上での解像度変換法, ブロック数変更による DCT 係数上での解像度変換法も併せて実装した. 本論文では, ブロック数変更による DCT 係数上での解像度変換法を従来法とする.

本研究に用いる実験画像を図 4.21 に示す. 図 4.21(a), (b) はグレースケール 256 階調, 512×512 pixel の Lenna 画像 および Barbara 画像であり, 図 4.21(c), (d) はグレースケール 256 階調, 512×768 pixel の Caps 画像 および Houses 画像であり, 図 4.21(e), (f) はカラー RGB 各 256 階調, 512×768 pixel の Motor 画像 および Parrots 画像である.

4.3.1 解像度変換画像の比較

縮小画像の比較

提案法の縮小に対する効果を調べるため, 縮小画像を比較した結果について示す. 図 4.22 に実験の流れを示す. 実験には 図 4.21 の原画像をそのまま使用する. 本実験では,



(a) Lenna



(b) Barbara

図 4.21 原画像

原画像を一度 DCT して求めた DCT 係数に対し、従来法および提案法で縮小を行い、その後、逆 DCT して得られる縮小画像を比較する。提案法は平均操作法の行列を用いて DCT 係数上で縮小を行う。また、カラー画像の縮小は Y, Cr, Cb の各成分毎に実行する。

図 4.23 ~ 4.28 に従来法および提案法でそれぞれ縦横 1/2 倍に縮小した結果を示す。図 4.23, 4.24 は 256×256 pixel, 図 4.25 ~ 4.28 は 256×384 pixel である。図 4.23 ~ 4.28 より、縮小画像を比較すると、どの画像を比較しても大きな違いは見られない。実際に従来法と提案法の縮小画像の PSNR をとると表 4.1 のようになり、全体的に高い PSNR 値をとり、低域成分を多く含む Lenna 画像や Caps 画像では 41dB を超えている。以上のことから従来法、提案法ともにほぼ同等の画質の縮小画像が得られていることが確認できる。

拡大画像の比較 画像拡大時における提案法の有効性を確認するため、拡大画像を比較した結果について示す。実験の流れを図 4.29 に示す。本実験では、図 4.21 の原画像を縦横 1/2 倍にダウンサンプリングした図 4.30 の画像を実験画像として用いる。実験画像を DCT して得られた DCT 係数を従来法および提案法でそれぞれ拡大し、その後、逆 DCT



(c) Caps



(d) Houses

図 4.21 原画像 (続き)

して得られる拡大画像を比較する. 提案法の拡大には 3 次畳み込み補間法 ($a = -0.5$) の行列を用いる. また, カラー画像の拡大は Y, Cr, Cb の各成分毎に拡大を行う. 拡大画像の評価のため, ダウンサンプリング前の原画像と拡大画像の PSNR および拡大画像の $GBIM_{16}$ を求める.

図 4.30 の実験画像を従来法および提案法でそれぞれ縦横 2 倍に拡大した画像の一部を

第 4 章 解像度変換技術



(e) Motor



(f) Parrots

図 4.21 原画像 (続き)

図 4.31 ~ 4.36 に示す.

図 4.31 ~ 4.36 より, 従来法および提案法の PSNR を比較すると, すべての拡大画像において従来法よりも提案法の方が優れており, 提案法の画質が客観的に良いことが確認できる. また, ブロックひずみ量を表す $GBIM_{16}$ を比較すると, すべての拡大画像において従来法よりも提案法の方が値が小さい. これは, 提案法の拡大画像に含まれるブロックひ

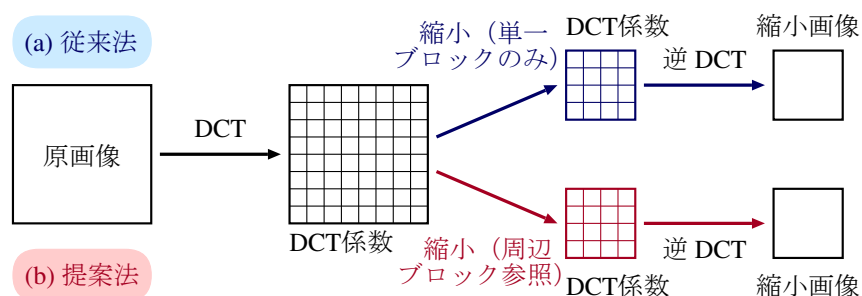


図 4.22 実験の概要 (縮小画像の比較)

表 4.1 縮小画像間の PSNR

原画像	PSNR[dB]
Lenna	41.53
Barbara	33.08
Caps	41.44
Houses	30.78
Motor	24.75
Parrots	32.37

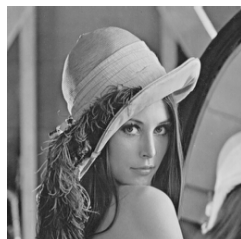
ずみ量が少なく、ブロックひずみがほとんど含まれていないことを示している。

図 4.31 より Lenna 画像による拡大画像を比較すると、従来法の拡大画像では髪や目の部分にブロックひずみが目立つ。それに対し、提案法ではブロックひずみが混入していないことがわかる。しかし、顔や髪の輪郭部分では従来法に比べ提案法の方が不鮮明となっている。これは、提案法で用いた補間法が原画像の複数点より補間する 3 次畳み込み補間法であるため、エッジ部分がぼやけたと考えられる。

次に図 4.32 より、滑らかな領域と振動の激しい領域が存在する Barbara 画像の拡大画像を比較する。従来法の拡大画像は、スカーフの部分にブロックひずみが確認できる。それに対し、提案法ではブロックひずみが認識できない。

次に図 4.33, 4.34 より Caps 画像, Houses 画像による拡大画像を比較する。従来法では、Caps 画像では壁や帽子の周辺, Houses 画像では、木や屋根の周辺にブロックひずみ

第 4 章 解像度変換技術



(a) 従来法



(b) 提案法

図 4.23 縮小画像の比較 (Lenna, 縦横 1/2 倍)



(a) 従来法



(b) 提案法

図 4.24 縮小画像の比較 (Barbara, 縦横 1/2 倍)



(a) 従来法



(b) 提案法

図 4.25 縮小画像の比較 (Caps, 縦横 1/2 倍)

4.3 実験結果



(a) 従来法



(b) 提案法

図 4.26 縮小画像の比較 (Houses, 縦横 1/2 倍)



(a) 従来法



(b) 提案法

図 4.27 縮小画像の比較 (Motor, 縦横 1/2 倍)



(a) 従来法



(b) 提案法

図 4.28 縮小画像の比較 (Parrots, 縦横 1/2 倍)

第 4 章 解像度変換技術

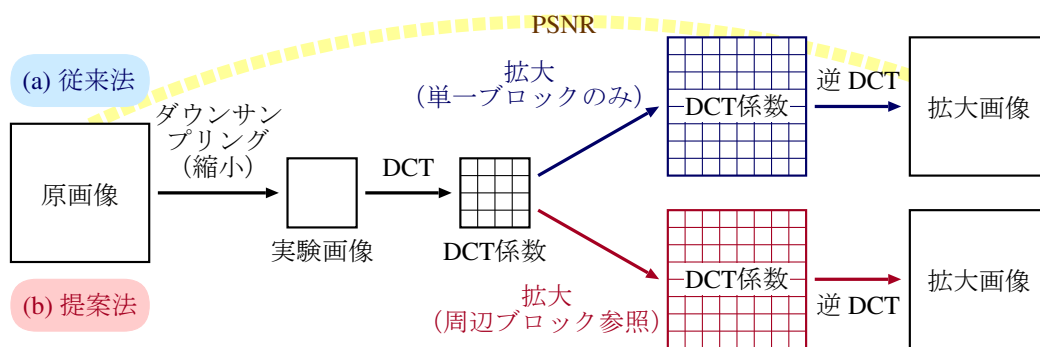


図 4.29 実験の概要 (拡大画像の比較)



(a) Lenna



(b) Barbara



(c) Caps



(d) Houses



(e) Motor



(f) Parrots

図 4.30 実験画像

4.3 実験結果



(a) 従来法 (PSNR: 29.38dB, GBIM₁₆: 2.21)



(b) 提案法 (PSNR: 33.93dB, GBIM₁₆: 1.11)

図 4.31 拡大画像の比較 (Lenna, 縦横 2 倍)

第 4 章 解像度変換技術



(a) 従来法 (PSNR: 22.12dB, GBIM₁₆: 2.17)



(b) 提案法 (PSNR: 24.64dB, GBIM₁₆: 1.04)

図 4.32 拡大画像の比較 (Barbara, 縦横 2 倍)

4.3 実験結果



(a) 従来法 (PSNR: 30.71dB, GBIM₁₆: 2.38)



(b) 提案法 (PSNR: 33.27dB, GBIM₁₆: 1.09)

図 4.33 拡大画像の比較 (Caps, 縦横 2 倍)

第 4 章 解像度変換技術



(a) 従来法 (PSNR: 19.68dB, GBIM₁₆: 2.28)



(b) 提案法 (PSNR: 22.36dB, GBIM₁₆: 1.06)

図 4.34 拡大画像の比較 (Houses, 縦横 2 倍)

4.3 実験結果



(a) 従来法 (PSNR: 22.70dB, GBIM₁₆: 2.15)



(b) 提案法 (PSNR: 25.66dB, GBIM₁₆: 1.05)

図 4.35 拡大画像の比較 (Motor, 縦横 2 倍)

第 4 章 解像度変換技術



(a) 従来法 (PSNR: 30.34dB, GBIM₁₆: 2.23)



(b) 提案法 (PSNR: 33.70dB, GBIM₁₆: 1.09)

図 4.36 拡大画像の比較 (Parrots, 縦横 2 倍)

表 4.2 拡大画像の対比較試験

画像	提案法が良いとした割合 [%]	
	画像全体	画像の一部
Lenna	60	77
Barbara	73	93
Caps	67	97
Houses	93	98
Motor	87	95
Parrots	80	92
平均	77	92

が確認できるが、提案法ではブロックひずみが含まれておらず、視覚的に良好な拡大画像となっていることが確認できる。

また、図 4.27, 4.28 にカラー画像である Motor 画像, Parrots 画像の拡大画像を示す。カラー画像の場合もグレイスケールと同様に従来法ではブロックひずみが見られるが、提案法では見られない。

次に従来法および提案法によるブロックひずみの視覚的影響を評価するため、20 代の男性 14 名、女性 1 名の計 15 名の被験者に対し、対比較試験を行った結果を表 4.2 に示す。表 4.2 は、従来法と提案法を比較し提案法の方がブロックひずみが少ないと答えた割合を示している。対比較試験より、全での画像において従来法よりも提案法の方がブロックひずみが少ないと答えた人の割合が多くなった。ただし、Lenna の拡大画像では従来法を選択する被験者も多かった。これは、Lenna 画像のような低域成分を多く含む画像ではブロックひずみが目立ちにくいためと考えられる。以上の結果より、提案法による解像度変換法は、従来法に比べブロックひずみが少なく良好な画質の拡大画像が得られることが示せたといえる。

4.3.2 ブロックひずみ混入画像の比較

実際の JPEG 画像は、圧縮されていて既にブロックひずみが混入している画像が多い。そこで、実際の使用時における提案法の有効性を確認するため、ブロックひずみの混入した画像を拡大した結果について示す。

実験の流れを図 4.37 に示す。実験には図 4.30 の実験画像に対し、JPEG 圧縮によりあらかじめブロックひずみを混入させた図 4.38 の画像を用いる。これらの画像は、JPEG の quality パラメータを 20 として圧縮したものである。図 4.38 の画像における Bit-Rate と GBIM の関係を表 4.3 に示す。本実験では、図 4.38 のブロックひずみ混入画像を DCT して得られた DCT 係数を従来法および提案法で拡大し、その後 逆 DCT して得られる拡大画像を比較する。拡大画像を客観的に評価するため、ダウンサンプリング前の原画像と拡大画像の PSNR および拡大画像の $GBIM_{16}$ を求める。

縦横 2 倍の拡大を行った結果を図 4.39 ~ 4.44 に示す。図 4.39 ~ 4.44 より拡大画像を比較すると、従来法の拡大画像ではブロックひずみがはっきりと現れているのに対し、提案法ではブロックひずみ量が抑えられていることが確認できる。次に PSNR を比較すると、すべての拡大画像において従来法よりも提案法の方が値が高く、提案法の画質が客観的に良いことが確認できる。また、ブロックひずみ量を表す $GBIM_{16}$ 値を比較すると、すべての拡大画像において従来法よりも提案法の方が値が小さい。これは、提案法の拡大画像に含まれるブロックひずみ量が従来法よりも少なくなっているためである。さらに、拡大する前の図 4.38 に比べ、従来法ではすべてにおいて $GBIM$ 値が増加しているが提案法ではすべてにおいて低下している。つまり、拡大時において従来法ではブロックひずみ量が増加しているのに対し、提案法では減少することを示している。

以上の実験結果より、既にブロックひずみが混入した画像の拡大において、拡大画像に含まれるブロックひずみ量は従来法よりも提案法の方が少なく、視覚的に良好な画像が得られていることが確認でき、提案法の有効性が示せたといえる。

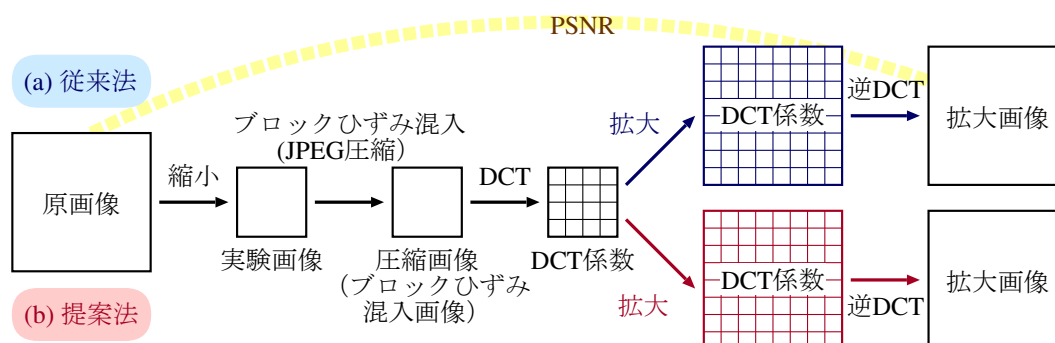


図 4.37 実験の概要 (ブロックひずみ混入画像の拡大)

表 4.3 ブロックひずみ混入画像の Bit-Rate と GBIM の関係

画像	Bit-Rate [bpp]	GBIM
Lenna	0.53	2.23
Barbara	0.69	1.50
Caps	0.38	2.46
Houses	0.99	1.55
Motor	0.93	1.64
Parrots	0.46	2.36

4.3.3 演算回数の比較

DCT 係数上での解像度変換に要する処理量を調べるため、拡大前の DCT 係数から拡大後の DCT 係数を求めるのに必要な演算回数を調べた。拡大前の 8×8 点の DCT 係数 1 ブロックを縦横 2 倍に拡大して 8×8 点の DCT 係数 4 ブロックを求めるのに必要な加算回数と乗算回数を測定した。結果を図 4.45 に示す。図 4.45 において画像空間上の演算回数は、解像度変換前の DCT 係数を逆 DCT し、画像空間上で解像度変換を行い、再び DCT して得られる DCT 係数を得るまでにかかる演算回数を示す。また画像空間上での拡大において DCT、逆 DCT には Chen の高速アルゴリズム [33] を用いた。さらに、画像空間上での拡大と提案法の拡大には 3 次畳み込み補間法 ($a = -0.5$) を用いた。図 4.45 より、従来法の演算回数が最も少なくなっていることが確認できる。これは、提案法では隣

第 4 章 解像度変換技術



(a) Lenna
(GBIM: 2.23)



(b) Barbara
(GBIM: 1.50)



(c) Caps (GBIM: 2.46)



(d) Houses (GBIM: 1.55)



(e) Motor (GBIM: 1.64)



(f) Parrots (GBIM: 2.36)

図 4.38 ブロックひずみ混入画像

接ブロックを参照する演算が必要であるのに対し、従来法ではその演算がないためであり、そのため従来法では拡大画像にブロックひずみが発生してしまう。また、提案法は画像空間上での拡大に比べ演算回数が少ない。これは、画像空間上での拡大は DCT や 逆 DCT といった操作が含まれるのに対し、提案法ではその操作がないためである。以上のことから、提案法は処理量の面で従来法には劣るが、画像空間上での拡大に比べ少ない処理量で演算できることが確認できた。

4.3 実験結果



(a) 従来法 (PSNR: 27.64dB, GBIM₁₆: 4.42)



(b) 提案法 (PSNR: 29.10dB, GBIM₁₆: 1.84)

図 4.39 ブロックひずみ混入画像の拡大 (Lenna, 縦横 2 倍)

第 4 章 解像度変換技術



(a) 従来法 (PSNR: 21.70dB, GBIM₁₆: 3.00)



(b) 提案法 (PSNR: 23.24dB, GBIM₁₆: 1.33)

図 4.40 ブロックひずみ混入画像の拡大 (Barbara, 縦横 2 倍)

4.3 実験結果



(a) 従来法 (PSNR: 29.06dB, GBIM₁₆: 4.88)



(b) 提案法 (PSNR: 29.68dB, GBIM₁₆: 1.91)

図 4.41 ブロックひずみ混入画像の拡大 (Caps, 縦横 2 倍)

第 4 章 解像度変換技術



(a) 従来法 (PSNR: 19.16dB, GBIM₁₆: 3.02)



(b) 提案法 (PSNR: 20.87dB, GBIM₁₆: 1.32)

図 4.42 ブロックひずみ混入画像の拡大 (Houses, 縦横 2 倍)

4.3 実験結果



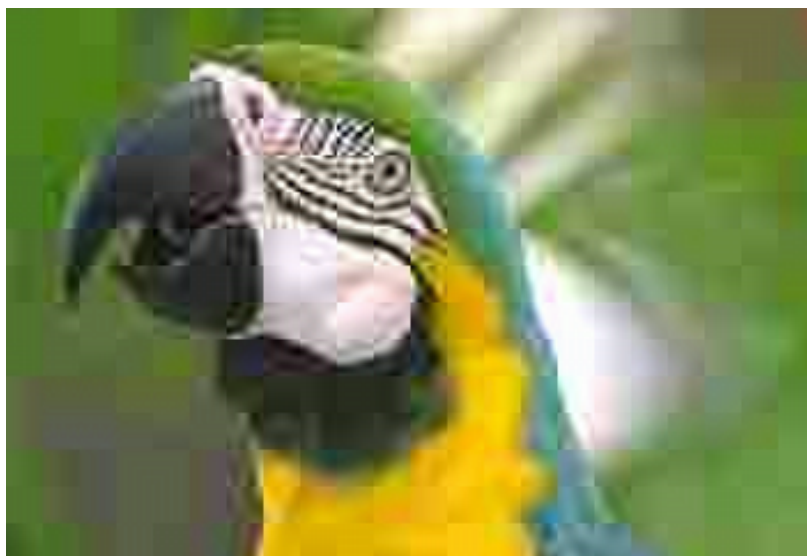
(a) 従来法 (PSNR: 21.11dB, GBIM₁₆: 3.24)



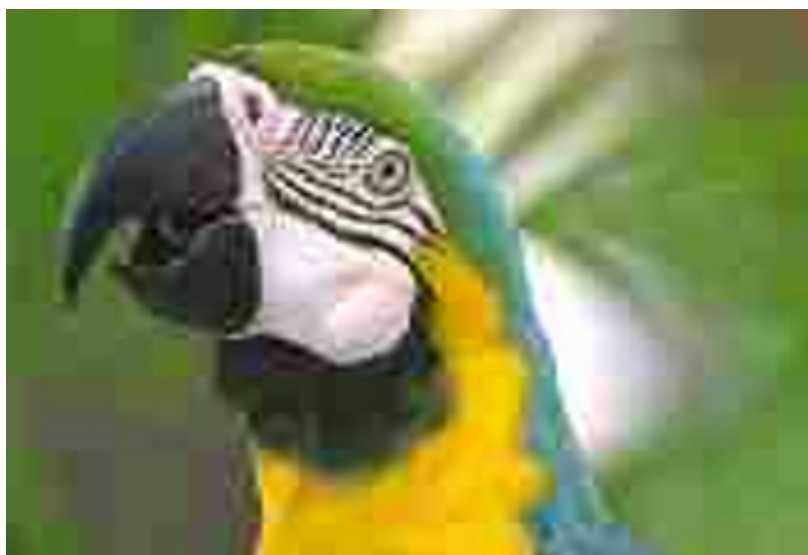
(b) 提案法 (PSNR: 21.99dB, GBIM₁₆: 1.46)

図 4.43 ブロックひずみ混入画像の拡大 (Motor, 縦横 2 倍)

第 4 章 解像度変換技術



(a) 従来法 (PSNR: 27.42dB, GBIM₁₆: 4.23)



(b) 提案法 (PSNR: 28.25dB, GBIM₁₆: 1.89)

図 4.44 ブロックひずみ混入画像の拡大 (Parrots, 縦横 2 倍)

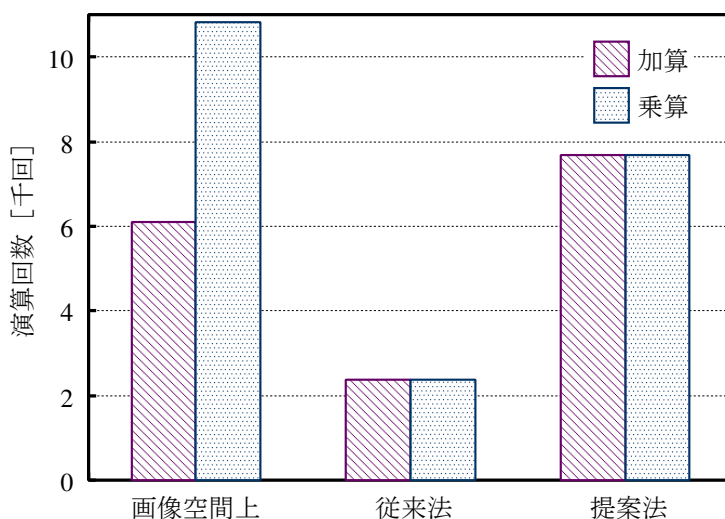


図 4.45 演算回数の比較 (縦横 2 倍)

表 4.4 1 ブロック拡大時の計算量

手法	乗算	加算	合計
Cubic Convolution	2416	3472	5888
MA 法	744	648	1392
HNG 法	3184	5328	8512
提案法	2032	2384	4416

4.3.4 高速手法の演算回数の比較

実験画像を CubicConvolution 補間法，論文 [10] の手法（以下，MA 法），提案法および提案法の高高速化手法でそれぞれ縦横 2 倍の拡大を行った際の計算量について比較を行う。各手法で 8×8 点 DCT 係数 1 ブロックを縦横 2 倍の 4 ブロックに拡大した場合の計算量を表 4.4 に示す。ただし画像空間上で解像度変換を行う CubicConvolution 補間法には高速逆 DCT，DCT [34] の計算量が含まれている。表 4.4 より，提案法は 1 ブロック毎に独立で処理する MA 法には劣るものの，画像空間上での手法や HNG 法に比べ，少ない計算量で処理が可能であることが確認できる。

第4章 解像度変換技術

表 4.5 DCT 係数に対する解像度変換法のまとめ

	ブロックサイズ	画質	演算回数
画像空間上	(固定)		×
DCT 係数上 (ブロックサイズ変更)	× (変化)	×	
DCT 係数上 (ブロック数変更・単一)	(固定)	×	
DCT 係数上 (ブロック数変更・周辺参照)	(固定)		

4.4 むすび

画像の解像度変換は一般に画像空間上で行われる。しかし、JPEG や MPEG の符号化方式には DCT が用いられていることから、DCT 係数上での解像度変換法が注目されている。

従来の DCT 係数上での解像度変換は、ブロックサイズを変更することで実現している。しかし、DCT のブロックサイズが変化してしまうため JPEG 等の汎用 DCT をそのまま用いることはできない。また、ブロックサイズを固定して処理できても解像度変換後の画像にブロックひずみが発生する問題がある。

そこで本研究では、隣接ブロックを用いた DCT 係数上での解像度変換法を提案し、その有効性について検討した。表 6.1 に DCT 係数に対する解像度変換法をまとめる。まず、単一ブロックのみを用い解像度変換を行う従来法と周辺ブロックを参照する提案法で解像度変換後の画像を比較した。縮小画像を比較した結果、従来法と提案法はどちらもほぼ同等の画質の縮小画像が得られた。また拡大画像を比較すると、従来法ではブロックひずみが発生するのに対し、提案法ではブロックひずみが発生しないことが確認できた。

次に実使用に対する実験として、既にブロックひずみが混入した拡大画像による比較実験を行った。その結果、従来法ではブロックひずみがはっきりとした拡大画像が得られるのに対し、提案法ではブロックひずみが軽減できていることが確認された。

最後に、画像拡大の際に必要な演算回数を比較した結果、提案法の演算回数は従来法に

比べ多くなっているが、画像空間上での拡大に比べ少なくなっていることが確認できた。

また、提案法は DCT のブロックサイズが変更されないため JPEG 等の汎用 DCT をそのまま用いることができる。

しかし、提案法は周辺ブロックを参照するため演算回数が増加してしまう。そこで、今後の課題としてアルゴリズムの単純化による演算の高速化が挙げられる。また提案法は、画像空間上での解像度変換と等価な処理を DCT 係数上に適用したものであった。そのため、解像度変換画像の画質は画像空間上での解像度変換画像と同等で、それを超えることはない。もし、DCT 係数の特徴（例えば、周波数情報など）を用い解像度変換できればさらなる画質の向上が期待できる。

現在、JPEG は画像の国際標準として広く普及している。しかし今後は、新しい画像の国際標準の規格である JPEG2000 が普及するであろう。そのため、JPEG2000 などの新しい画像規格の符号化データに対応した解像度変換も検討する必要がある。

第 5 章

高精細化技術

JPEG や MPEG ではブロック変換符号化が用いられている。これらの符号化方式では低ビットレートにおいてブロックひずみが混入することが問題となる。

このブロックひずみを軽減する方法として、最も簡単なものに復号画像全体に対してローパスフィルタをかける方法がある [18]。これによりブロック間の不連続性を解消することができ、滑らかな画像を得ることができる。しかし、ローパスフィルタをかけることから、エッジ成分までなだらかになり画像が不鮮明となり適切ではない。

また、POCS (the Projection Onto Convex Sets) を用いた繰り返しによってブロックひずみを軽減する試みが提案されている [19], [35]。しかし、繰り返しの操作を必要とすることから計算コストがかかり、この手法では動画像においてリアルタイムに画面への画像描画ができないという問題がある。

さらに、違うアプローチとして符号化側でブロックひずみが混入しないように変換を施す方法もある。この手法には、例えば、符号化側でブロックをオーバーラップさせることによりブロックひずみを軽減する LOT [20] があるが、この方法ではビットレートが増加し、情報圧縮の概念からはずれる。また、プレ・ポストフィルタリングを行う方法 [21], [36], [37] が提案されているが、本研究では既に符号化されている画像を対象としていることからプレフィルタリングを行うことはできないためこれらの方法は適さない。

第 5 章 高精細化技術

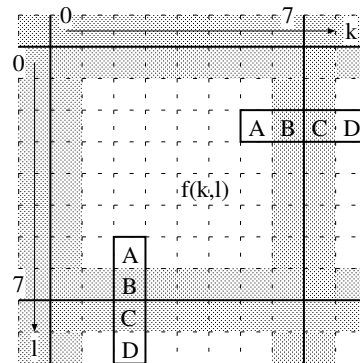


図 5.1 DEF の適用範囲

つまり、既に符号化されている画像に対しては、符号化データを基に復号化側で視覚的特性を改善させることが必要である。

そこで、本研究では逆ウェーブレット変換を用いた JPEG 画像のブロックひずみ軽減法を提案する。DCT 係数がウェーブレット係数と等価であることを示し、また、周波数成分に帯域分割された係数を利用しブロックひずみを軽減させる。

5.1 Deblocking Edge Filter

ブロックひずみを低減する手法の一つとして、ITU-T H.263 –ANNEX J– [38] で定義されている Deblocking Edge Filter (以下、DEF) を用いる手法がある。DEF は復号化後の画像データに対し、ブロックひずみの発生する 8×8 画素のブロック境界の両側 2 画素づつに平滑化フィルタを施す。DEF の適用範囲の例を図 5.1 に示す。DEF は画像全体の縦方向、横方向の全ブロック境界に圧縮率に応じたフィルタ処理を施すことで、ブロックひずみの低減を行う。

5.2 ウェーブレット変換を用いた高精度化

5.2.1 DCT 係数のサブバンド表現

DCT はブロック変換であるが、サブバンド符号化とも解釈することができる [39]。

5.2 ウェーブレット変換を用いた高精度化

1 次元 N 点 DCT の各基底は N 分割フィルタバンクとも解釈でき、その分割フィルタバンクの各伝達関数 $F_i(z)$ は、

$$F_i(z) = \sum_{n=0}^{N-1} \sqrt{\frac{2}{N}} C(i) \cos \frac{\{2(N-n-1)+1\}i\pi}{2N} z^{-n} \quad (5.1)$$

となる。また、帯域合成フィルタバンクの伝達関数 $\hat{F}_i(z)$ は、

$$\hat{F}_i(z) = \sum_{n=0}^{N-1} \sqrt{\frac{2}{N}} C(i) \cos \frac{\{2n+1\}i\pi}{2N} z^{-n} \quad (5.2)$$

で与えられる。

$N = 8$ における $F_i(z)$ のサブバンドフィルタバンク構成および周波数特性をそれぞれ図 5.2(a), 図 5.2(b) に示す。

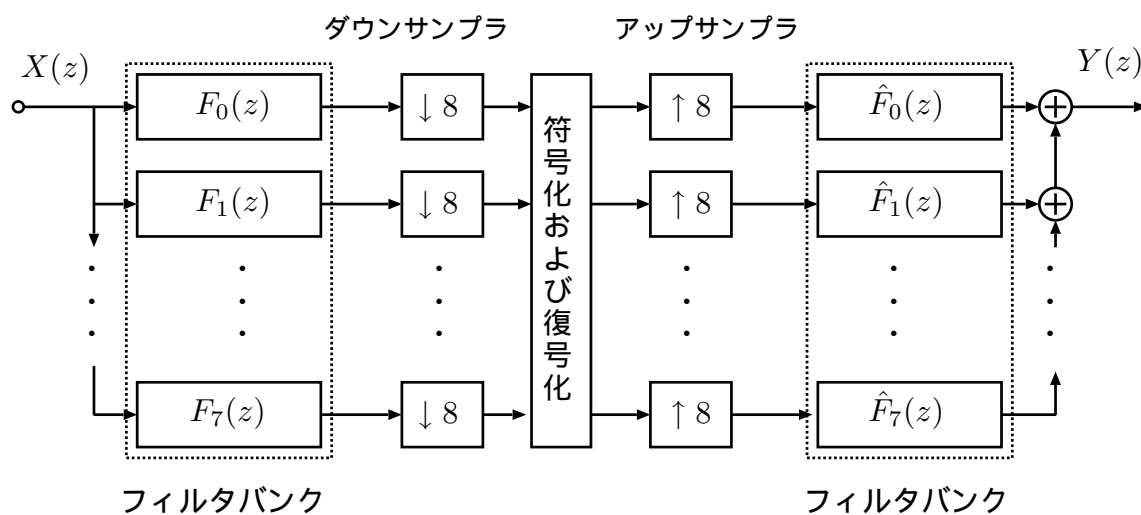
画像においても同様にブロックサイズ $N \times N$ の DCT によってブロック変換された画像をサブバンド表現に並び替えることができる。ブロック表現とサブバンド表現がどのような関係にあるか 4×4 ブロック DCT の場合を例に図 5.3 に示す。

図 5.3 において (a) で各ブロックの左上に存在する “1, 2, ..., 9” は周波数 $(u, v) = (0, 0)$ で直流成分を表しており、これらを集めることで (b) の左上の帯域が構成できる。また、(b) の “5 および, a, b, ..., o” は位置 $(x, y) = (2, 2)$ のブロック内の要素であり、(a) ではブロック表現される。このようにブロック変換された DCT 係数を再配置することによってサブバンド表現が実現できる。

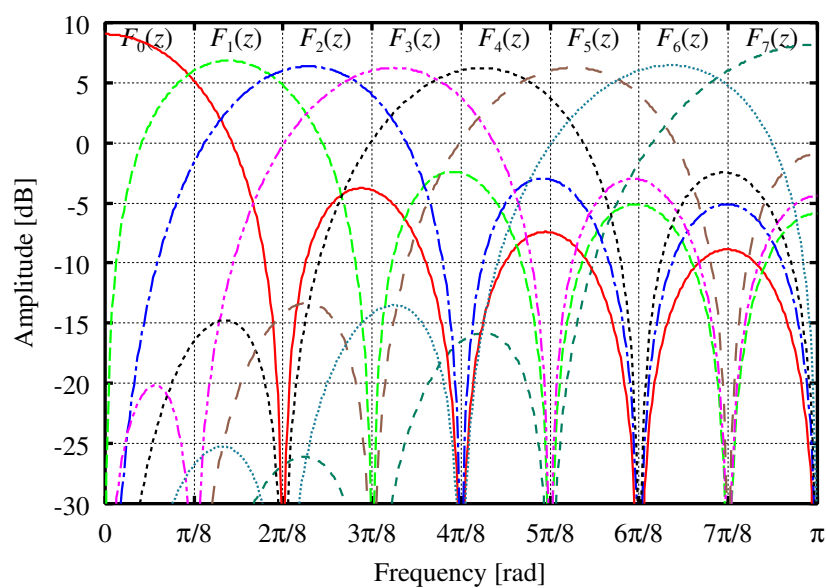
5.2.2 DCT 係数のウェーブレット表現

前節の考え方をもとに、DCT フィルタバンクにより分割し、サブバンド表現したと考える。

2.3 節においてウェーブレット変換もフィルタバンクによって構成されることを示した。図 2.10, 5.2(a) より、DCT もウェーブレット変換と同様にフィルタバンク、アップサンブラ、ダウンサンブラで構成することが可能である。両者の違いはフィルタバンクの伝達関数にある。したがってフィルタバンクの伝達関数、帯域分割数および帯域分割の構造 (ツ



(a) サブバンドフィルタバンクの構成



(b) フィルタバンクの振幅特性

図 5.2 1次元 DCT のサブバンド表現 ($N = 8$)

5.2 ウェーブレット変換を用いた高精度化

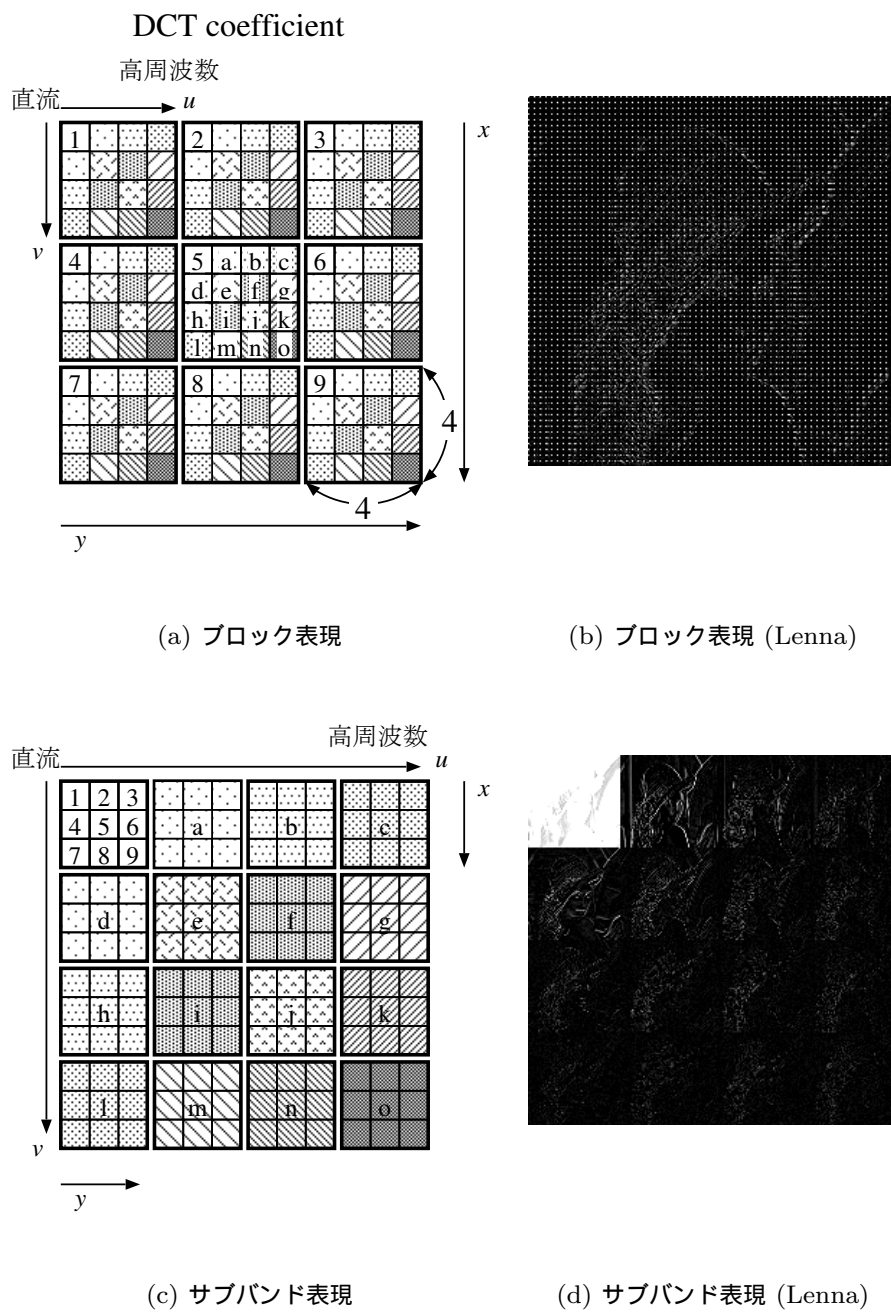


図 5.3 ブロック表現とサブバンド表現の関係 (4×4 DCT)

第 5 章 高精細化技術

リー構造・オクターブ構造) の構成しだいで等価な表現が可能である.

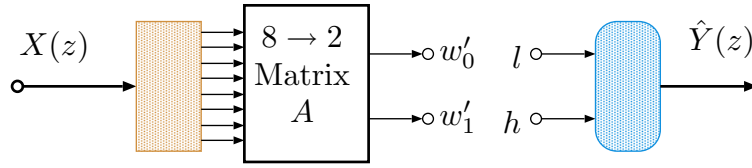
1 次元 DCT に対する 1 次元 1 階層逆ウェーブレット変換の構成を図 5.4(a) に示す. 1 次元の場合は DCT により信号は 8 分割され DCT 係数 w_0, \dots, w_7 となる. この 8 点 DCT 係数を行列 A を用い 2 点 DCT 係数 w'_0, w'_1 へと変換し, サブバンド表現に並び替える. その出力は低周波数成分 w'_0 と高周波数成分 w'_1 の 2 つの帯域から構成される. 通常 DCT された信号は DCT 係数となりその DCT 係数に対して逆 DCT を施すことにより元の信号に復号される. 一方, 1 次元 1 階層ウェーブレット変換係数も信号を 2 分割し, 低周波数成分 l , 高周波数成分 h へと分割する. この出力も通常は逆ウェーブレット変換を用いて復号される. ここで, 1 次元 2 点 DCT 係数の低周波数成分 w'_0 をウェーブレット変換係数の低周波数成分 l と 1 次元 2 点 DCT 係数の高周波数成分 w'_1 をウェーブレット変換係数の高周波数成分 h とすることで DCT で符号化された信号を逆ウェーブレット変換で復号することが可能である.

また, 図 5.4(b) に 2 次元の場合の構成を示す. 画像に 8×8 DCT を施すと縦方向に 8 分割され, その出力をさらに横方向へ 8 分割した計 64 分割の W_0, \dots, W_{64} からなるサブバンド符号化を行うことに相当する. その 64 分割された帯域を行列 A による変換により 2×2 DCT 係数 W'_0, \dots, W'_4 へと変換し, 縦方向 2 分割, 横方向 2 分割の計 4 分割を施したフィルタバンクの出力となる. ここで DCT 係数の縦方向, 横方向とも低周波数成分にあたる W'_0 をウェーブレット変換係数の低周波数成分にあたる LL とし, 同様に W'_1 を LH, W'_2 を HL, W'_3 を HH と解釈することで 2 次元 1 階層ウェーブレット変換を用いて復号することが可能となる.

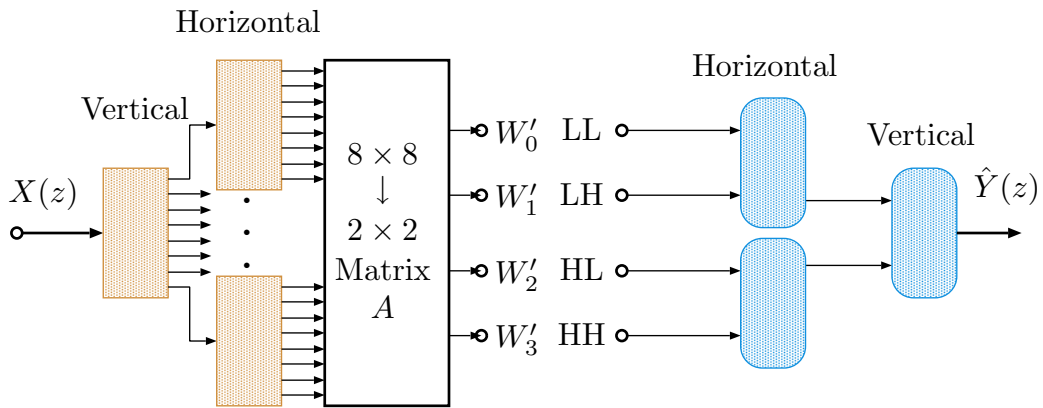
5.2.3 逆ウェーブレット変換を用いたブロックひずみ低減手法

前節において DCT 係数が変換を施すことでウェーブレット係数として解釈できることを示した. そこで, DCT で符号化された画像を逆ウェーブレット変換し, ブロックひずみを除去する方法を提案する.

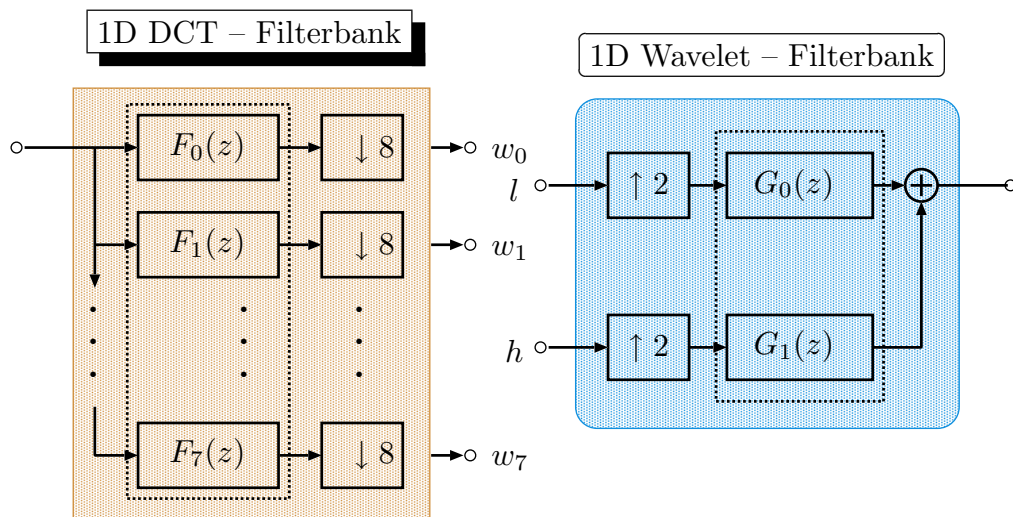
5.2 ウェーブレット変換を用いた高精度化



(a) 1次元 DCT \rightarrow 1次元 1階層ウェーブレット変換



(b) 2次元 DCT \rightarrow 2次元 1階層ウェーブレット変換



(c) 1次元 DCT および 1次元 1階層ウェーブレット変換の構成

図 5.4 DCT 係数のウェーブレット表現

第 5 章 高精細化技術

提案法の全体の流れを図 5.5 に、実際の画像変換の過程を図 5.6 に示す。JPEG において画像は図 5.6(a) のように 8×8 DCT 係数の形で表現される。この 8×8 DCT 係数を 2×2 DCT 係数へと変換し、サブバンド表現すると図 5.6(c) となる。図 5.5 に示すように 2×2 DCT 係数の各帯域を W_0, W_1, W_2, W_3 とすると、それらをウェーブレット変換における LL, LH, HL, HH と解釈し逆ウェーブレット変換を施す。つまり、サブバンド表現された図 5.6(c) に対して逆ウェーブレット変換することにより図 5.6(d) の復号画像を得ることができる。その際ウェーブレット変換のフィルタの選択により復号画像のブロックひずみは軽減できる。しかし、圧縮率を上げるにつれ、復号画像におけるブロックひずみが顕著に現れるため、ブロックひずみ軽減効果は十分でなくなる。そこで、 2×2 DCT 係数の低周波数成分にあたる W_0 に対して、ローパスフィルタをかけることにより直流成分および低周波数成分の不連続性を除去し、画像を滑らかにする。その際、エッジ成分を残すためマスキングという操作を行い、エッジ成分の減少を防ぐ。これらの処理によりブロックひずみを除去する。以下に個々の処理について説明する。

DCT のサイズ変換 まず、図 5.7 のように 1 つの 8×8 DCT 係数 F_l に対して逆 DCT を施し 8×8 の画像に戻す。その画像に対して 2×2 DCT を施すことにより 1 ブロックの 8×8 DCT 係数を 16 ブロックの 2×2 DCT 係数 F'_l へ変換する。実際には行列 A を用いて以上の処理をまとめて行う。変換は行列演算により行い変換式は、

$$F'_l = AF_l A^t \quad (5.3)$$

と表せる。ここで、行列 A は、

$$A = \begin{pmatrix} 0.5 & 0.64 & 0.46 & 0.22 & 0 & -0.15 & -0.19 & -0.13 \\ 0 & 0.05 & 0.19 & 0.36 & 0.5 & 0.54 & 0.46 & 0.27 \\ 0.5 & 0.27 & -0.46 & -0.54 & 0 & 0.36 & 0.19 & -0.05 \\ 0 & 0.13 & 0.19 & -0.15 & -0.5 & -0.22 & 0.46 & 0.64 \\ 0.5 & -0.27 & -0.46 & 0.54 & 0 & -0.36 & 0.19 & 0.05 \\ 0 & 0.13 & -0.19 & -0.15 & 0.5 & -0.22 & -0.46 & 0.64 \\ 0.5 & -0.64 & 0.46 & -0.22 & 0 & 0.15 & -0.19 & 0.13 \\ 0 & 0.05 & -0.19 & 0.36 & -0.5 & 0.54 & -0.46 & 0.27 \end{pmatrix} \quad (5.4)$$

5.2 ウェーブレット変換を用いた高精度化

であり, 8×8 DCT を 2×2 DCT に変換する行列である. この処理により図 5.6(a), 5.6(b) の 8×8 DCT 係数が図 5.6(c) の 2×2 DCT 係数へ変換される.

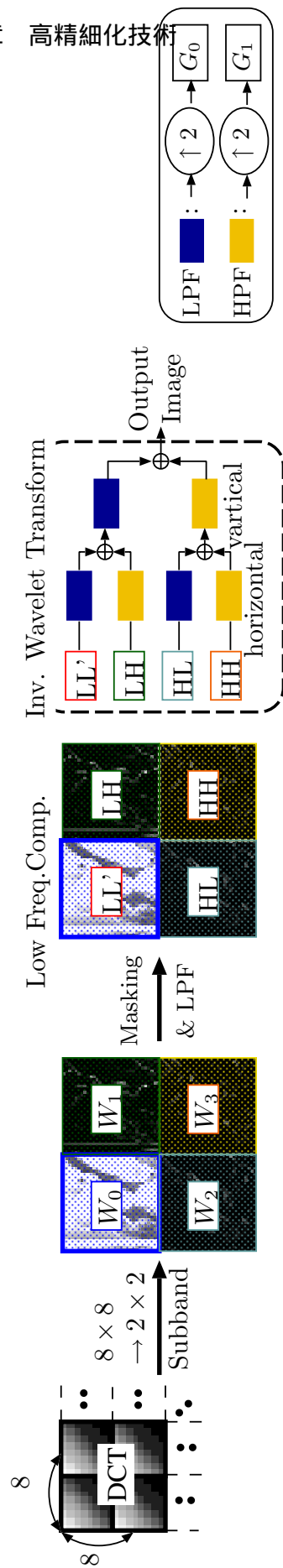
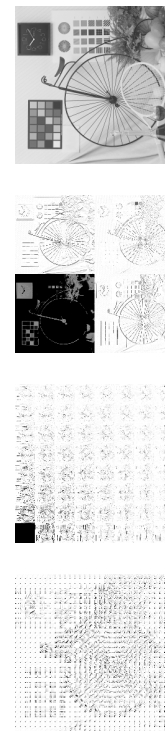


図 5.5 変換過程



(a) 8×8 DCT の結果
(b) 8×8 DCT のサブバンド表現
(c) 2×2 DCT への変換結果
(d) 逆ウェーブレット変換結果

図 5.6 変換の過程 (画像)

5.2 ウェーブレット変換を用いた高精度化

5/3 基底ウェーブレット変換 逆ウェーブレット変換に用いる 5/3 フィルタの伝達関数を式 (5.5) に、その周波数特性を図 5.8 に示す。DCT によって変換された帯域に対して適用すると低周波数成分に対してローパスフィルタをかけることに相当する。また、基底の長さが 3 タップであることからまわりの画素をみてローパスフィルタをかけることになりブロックひずみを減少できる。

ここで、5/3 フィルタの周波数特性 (図 5.8) より、直流成分に対する利得は 0 dB であり、DCT フィルタバンクと合わせるため輝度調整を必要とする。そのため、低周波数成分を 4 で割る処理を行う。

$$G_0(z) = \frac{1}{2} + z^{-1} + \frac{1}{2}z^{-2} \quad (5.5)$$

$$G_1(z) = -\frac{1}{8} - \frac{1}{4}z^{-1} + \frac{3}{4}z^{-2} - \frac{1}{4}z^{-3} - \frac{1}{8}z^{-4} \quad (5.6)$$

ローパスフィルタ 低周波数成分に対してローパスフィルタをかけることによりさらにブロックひずみを軽減する。2 × 2 DCT 係数に変換された信号は周波数分割されている。その低周波数成分にのみローパスフィルタをかけることとなるため高周波数成分のエッジ成分は残すことになる。ブロックひずみを除去するローパスフィルタとして、

$$L(z) = \frac{1}{3}z^{-1} + \frac{1}{3} + \frac{1}{3}z \quad (5.7)$$

を用いる。逆ウェーブレット変換にこのローパスフィルタを付加すると図 5.9 に示す構成で実現できる。通常の逆ウェーブレット変換は図 5.9(a) で表され、逆ウェーブレット変換を施す前に低周波数成分に対してローパスフィルタを施すため図 5.9(b) となる。システムの等価性を利用すると図 5.9(c) と変換することができる。逆ウェーブレット変換の基底として $G_0(z)$ として 5/3 フィルタを用いた場合、式 (5.5) および式 (5.7) よりアップサンプリングされた画像に対して

$$L'(z) = L(z^2)G_0(z) \quad (5.8)$$

$$= z^{-1} \left(\frac{1}{6}z^{-3} + \frac{1}{3}z^{-2} + \frac{1}{3}z^{-1} + \frac{1}{3} + \frac{1}{3}z + \frac{1}{3}z^2 + \frac{1}{3}z^3 \right) \quad (5.9)$$

第 5 章 高精細化技術

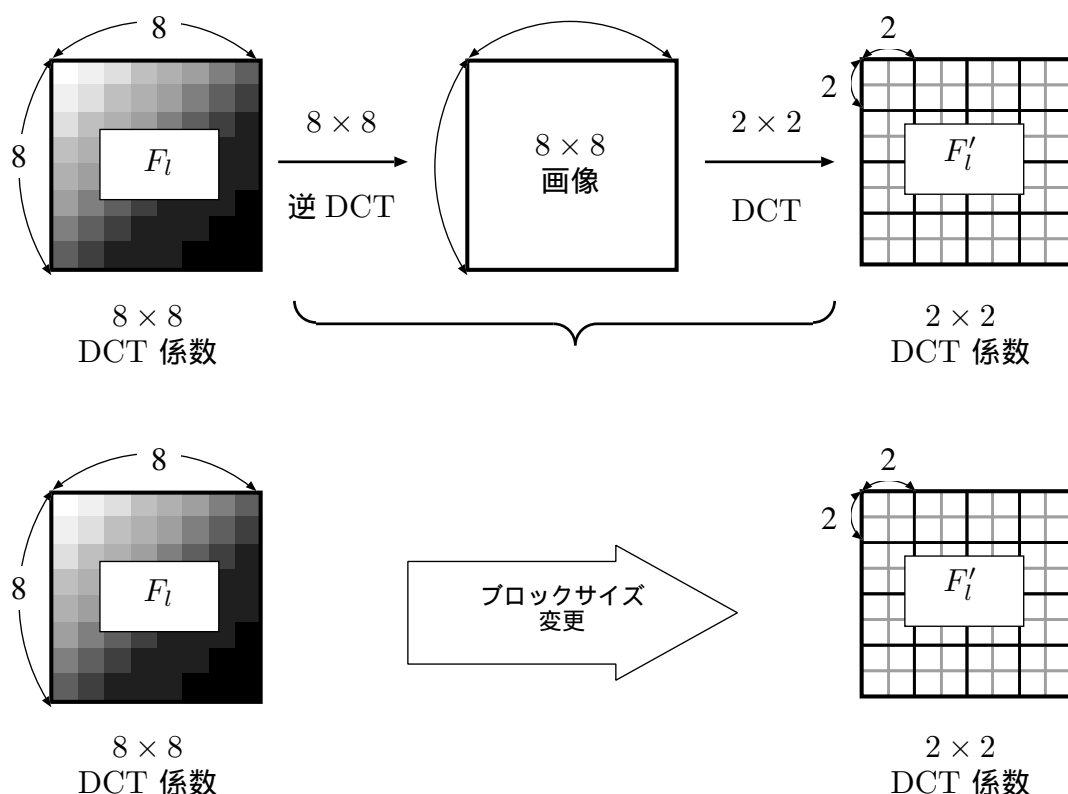


図 5.7 DCT サイズ変換

なる特性を持つフィルタをかけることとなり、周辺画素を考慮することに値する。 $L'(z)$ の周波数特性を図 5.10 に示す。

マスキング – 閾値 – 低周波数成分に対してローパスフィルタをかけることでブロックひずみは軽減できるが、図 5.10 の周波数特性からもわかるように縦・横それぞれ 2 分割しかしていないため低周波数成分にもエッジ成分が含まれている。つまり、低周波数成分に対してローパスフィルタをかけることで低周波数成分に含まれていたエッジ成分が削られ、画像が不鮮明となる。そこでエッジ成分にはローパスフィルタをかけない設計とする。

図 5.6(c) に示すように 2 次元 2×2 DCT により 4 つの帯域に周波数分割されている。ここで、縦方向、横方向それぞれ低周波数成分を LL、縦方向が低周波数成分で横方向が高周波数成分である帯域を LH、縦方向が高周波数成分で横方向が低周波数成分である帯域

5.2 ウェーブレット変換を用いた高精度化

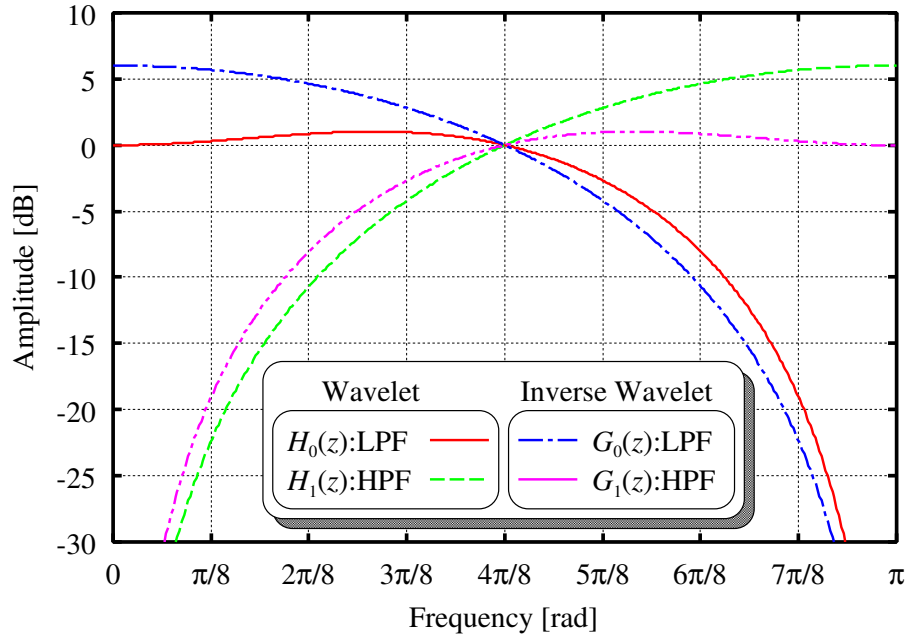


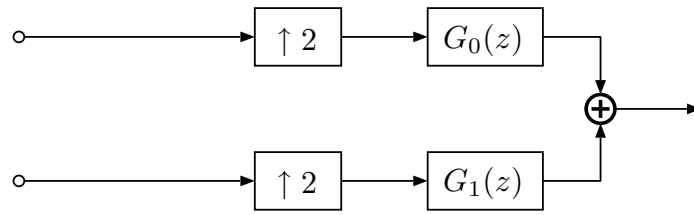
図 5.8 5/3 フィルタの周波数特性

を HL, 両方向とも高周波数成分の帯域を HH とする. それぞれの帯域には図 5.11 に示すように各帯域内での座標は他の帯域内でも同じ座標と関係があり, さらに復号画像の同じ座標の 4 画素に対応する. そこで, 高周波数成分 (LH, HL, HH) に値が存在する場合はその座標にエッジ成分が存在すると判断し, その座標に対応する低周波数成分 (LL) に対してローパスフィルタを施さないこととする.

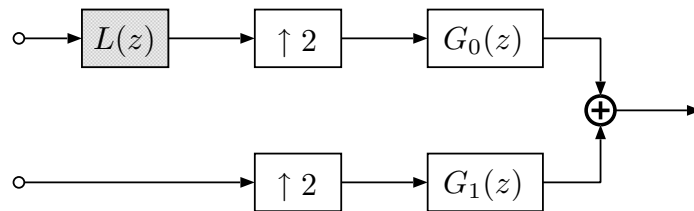
各高周波数成分に閾値 $\tau_{LH}, \tau_{HL}, \tau_{HH}$ を設定し絶対値が閾値以上の値があれば高周波数成分があると判断し, 対応する低周波数成分の座標にはローパスフィルタをかけない. 閾値 $\tau_{LH}, \tau_{HL}, \tau_{HH}$ 以下の値であれば, その座標に対応する低周波数成分の座標に対してローパスフィルタをかける. このパラメータ τ を変動させ, 滑らかさとエッジ成分のトレードオフを調節する.

τ がどのような効果を持つかその実験結果を図 5.12 に示す. 図 5.12 において低周波数成分 LL では, 白色で表示された座標にはローパスフィルタを施し, 黒で表示された座標はローパスフィルタをかけず値は変わらない. 一方高周波数成分 (LH, HL, HH) では, 灰色が閾値以下でありエッジ成分がないと判断された座標を表し, 黒はエッジ成分と判断

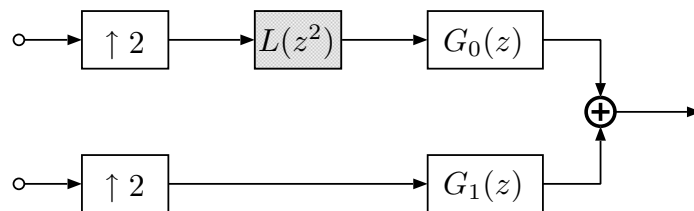
第 5 章 高精細化技術



(a) 通常の逆ウェーブレット変換



(b) 低周波数成分に対してローパスフィルタをかける



(c) (b) の等価表現

図 5.9 逆ウェーブレット変換と LPF の構成

された係数が $-$ であった場合、白はエッジ成分と判断された係数が $+$ であった場合を表す。図 5.12(a) より τ が小さい場合、エッジ成分と判断される座標が多く、低周波数成分に対してローパスフィルタを施さないこととなり、図 5.12(b) のように τ を大きく設定すると、エッジ成分と判断される座標が少なくなる。よって、 $\tau = 10$ である図 5.12(c) ではエッジ成分のはっきりした画像となるが、 $\tau = 100$ である図 5.12(b) ではエッジ成分が

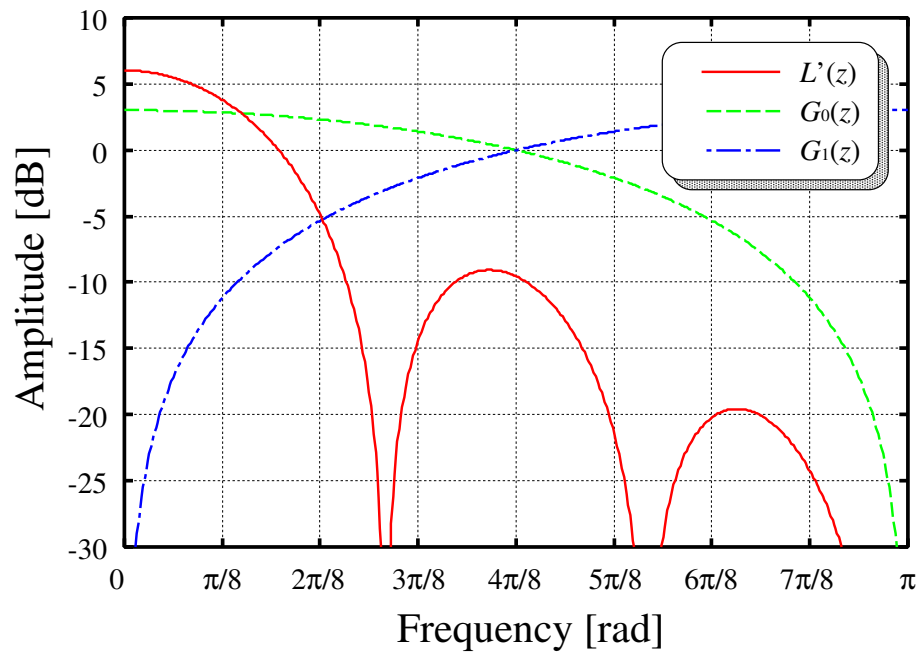
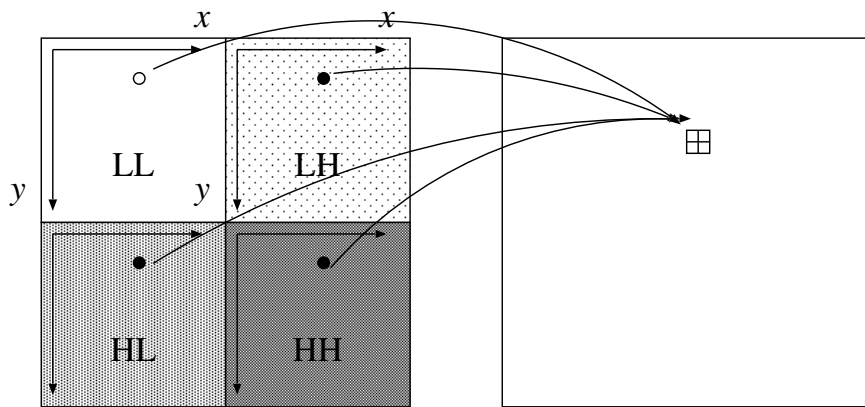
図 5.10 $L'(z)$ の周波数特性

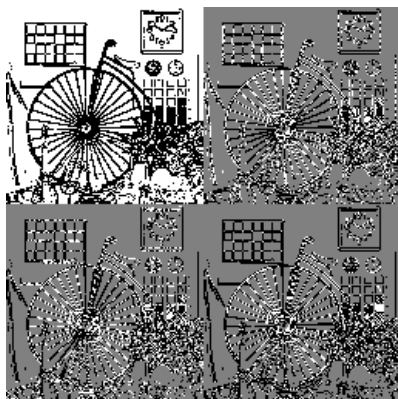
図 5.11 サブバンドにおける各帯域の対応関係

ローパスフィルタにより除去され画像が不鮮明となる。

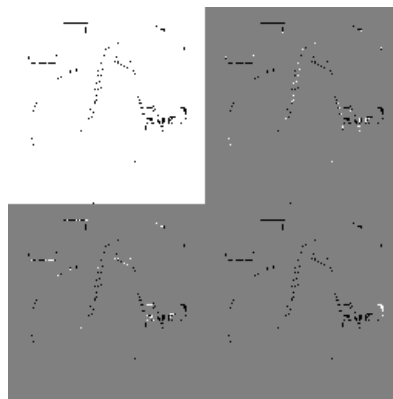
5.3 実験結果

提案手法である逆ウェーブレット変換を用いた低解像度 JPEG 画像のブロックひずみ軽減法の性能を調べるため、提案法を用いた復号化器を C 言語を用いて設計する。

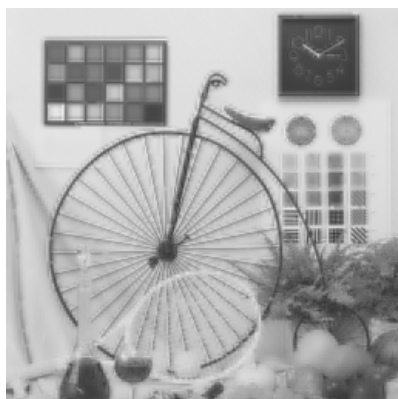
第5章 高精細化技術



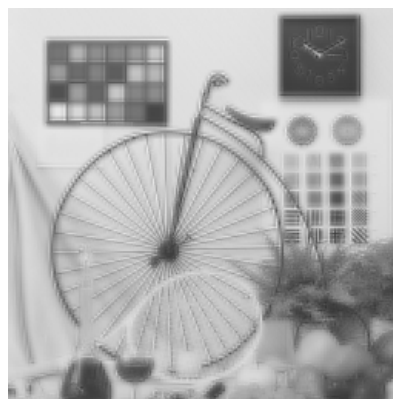
(a) $\tau = 10$



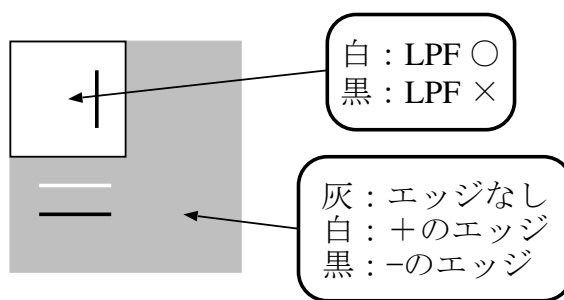
(b) $\tau = 100$



(c) $\tau = 10$ の場合 (高周波数成分ははっきり)



(d) $\tau = 100$ の場合 (高周波数成分のっぺり)



(e) 色の意味

図 5.12 マスキングの効果

まず、客観評価として画像符号化分野で一般的に使用されている PSNR により評価する。PSNR は復号画像と原画像との誤差を表しており、原画像に対して混入した誤差の量が定量的に示される。しかし、例えば全ての画素で画素値が 1 小さくなっただけでも PSNR は低くなるが視覚的には問題ない。本研究ではブロックひずみを軽減し、視覚的特性を向上することを目的としており原画像との誤差を評価することは本質でない。

そこで、ブロックひずみを定量化した評価法として GBD [40] が提案されている。GBD は PSNR と異なりブロックひずみを測定することが可能である。

まず、予備実験として提案法に用いる閾値 τ を設計する。閾値 τ によりブロックひずみの軽減とエッジ成分の保持のトレードオフを制御する。予備実験により求めた閾値 τ を用いて提案法と JPEG による復号と比較を行う。

JPEG では quality というパラメータで画質の制御を行っており、quality を制御することで JPEG 画像のビットレートを制御できる。原画像を JPEG により符号化し JPEG 画像を作成する。高圧縮して作成された JPEG 画像にはブロックひずみが混入する。その画像に対して JPEG を用いて復号した画像と提案法を用いて復号した画像について検討を行う。また、ブロックひずみ軽減法の従来法として画像全体に LPF をかける方法を用いる。ブロックひずみが混入した画像に対して従来法、JPEG および提案法を用いて復号し、復号画像の比較を行う。また、エッジ検出によりブロックひずみによるエッジ成分の比較を行う。

提案法の有効性を示すためグレイスケール 256 階調の様々な画像に対して検討を行う。例として以下の 6 つの画像を用いる。比較的低域成分の多い 512×512 の Lenna 画像および Barbara, peppers, 720×576 の boats 画像, 512×480 の pens, flower 画像について例として実験を示す。

第 5 章 高精細化技術

5.3.1 実験条件

5.3.2 客観評価

PSNR

画質を評価する方法として, PSNR や MAD などがよく知られている. 例えば, PSNR では平均ノイズ電力 (MSE) を原画像と復号画像の画素値の差の 2 乗平均とし, 最大信号電力を最大画素値の大きさの 2 乗としている. PSNR は平均ノイズ電力および最大画素値 (S_{max}^2) から

$$\text{MSE} = \frac{1}{X \times Y} \sum_{x=1}^X \sum_{y=1}^Y \{f(x, y) - F(x, y)\}^2 \quad (5.10)$$

$$\text{PSNR} = 10 \log_{10} \left(\frac{S_{max}^2}{\text{MSE}} \right) \quad (5.11)$$

となる. ただし, 画像サイズを $X \times Y$ ピクセルとし, $f(x, y)$, $F(x, y)$ はそれぞれ原画像と復号画像の画素値を表す. 上記の導出式より, PSNR は原画像と復号画像の差から導出されることが分かる.

本研究では無圧縮の画像を原画像とし, 原画像と JPEG による復号画像, 提案法による復号画像による PSNR を測定する.

GBD

PSNR では原画像と復号画像の誤差を定量化しているが, 実際に画像の視覚的評価とは必ずしも一致しない. そこでブロックひずみを定量化した指標である GBD が提案されている [40].

GBD は以下の式で定義される.

$$\text{GBD} = \alpha \text{GBD}_v + \beta \text{GBD}_h \quad (5.12)$$

$$\text{GBD}_v = \frac{1}{((N_h/8) - 1)N_v} [W \cdot D_h]^{\frac{1}{2}} \quad (5.13)$$

$$= \frac{1}{((N_h/8) - 1)N_v} \left[\sum_{k=1}^{(N_h/8)-1} |w_{i,k} \cdot d_{hk}|^2 \right]^{\frac{1}{2}} \quad (5.14)$$

ここで, N_v は画像の縦幅を示す. マトリクス W はブロック境界の誤差に対し重み付けを行うマトリクスであり, ブロック境界以外の画素間の誤差より算出される. i 行目の重み付けマトリクス W は $W = [w_{i,1}, w_{i,2}, w_{i,3}, \dots, w_{i,(N_h/8-1)}]$ となり, その要素は次式で導出できる.

$$w_{i,k} = \log_{10} \left(1 + \frac{\sqrt{\mu_{i,k}}}{1 + \sigma_{i,k}} \right) \text{ ただし } k = 1, 2, 3, \dots, (N_h/8 - 1)$$

ここで, $\mu_{i,k}$ と $\sigma_{i,k}$ は原画像と復号画像との画素値誤差の平均値と分散であり, マトリクス W を構成するパラメータである.

GBD を用いることで画像に混入するブロックひずみを定量的に測定することが可能である.

5.3.3 主観評価

PSNR および GBD によるに加え復号画像を用いて視覚的に評価を行う. PSNR は客観的な評価であり, PSNR の高い復号画像が必ずしも画質の良好な画像であるとは限らない. GBD はブロックひずみを定量的に測定しているが実際の画像への影響を調べるため実際に復号画像を用いて比較を行い, ブロックひずみが除去されていることを確認する. さらに画像に存在するエッジ成分を検出しブロックひずみが除去されており, 原画像に含まれるエッジ成分が保持されていることを確認する.

エッジ検出 画像を鮮明化する手法としてラプラシアンフィルタによる空間フィルタ法, 高域強調フィルタ法などが用いられる.

2 次微分 (ラプラシアン) を求めることでエッジを検出することができる.

$$\nabla^2 f = \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2} \quad (5.15)$$

$$= f_{xx}(x, y) + f_{yy}(x, y) \quad (5.16)$$

で定義される. 注目画素 (i, j) を中心とする差分で 2 次微分を表すと,

$$f_{xx}(i, j) = f(i+1, j) - 2f(i, j) + f(i-1, j) \quad (5.17)$$

$$f_{yy}(i, j) = f(i, j+1) - 2f(i, j) + f(i, j-1) \quad (5.18)$$

第 5 章 高精細化技術

となり、ラプラシアン差分近似は以下ようになる。

$$\nabla^2 f(i, j) = f(i+1, j) + f(i-1, j) + f(i, j+1) + f(i, j-1) - 4f(i, j) \quad (5.19)$$

このエッジ検出により、画像に混入するブロック状のエッジ成分の有無および原画像に含まれるエッジ成分の有無について比較検討を行う。

5.3.4 予備実験

逆ウェーブレット変換に用いるフィルタバンクの決定

ウェーブレット変換においてフィルタバンクには Haar 基底, 5/3 フィルタ, 9/7 フィルタがある。フィルタバンクの選び方で提案法の動作は変化する。Haar 基底を用いると入力画像をそのまま復号することができるが、本実験ではブロックひずみの軽減を目的としているので使用しない。9/7 フィルタを用いると高周波数成分に対するフィルタのタップ数が長くなり、エッジ成分が周辺の画素に拡がるため、本実験では 5/3 フィルタを用いる。

5.3.5 閾値 τ の設計

本研究で用いる閾値 τ について設計を行う。閾値 $\tau_{LH}, \tau_{HL}, \tau_{HH}$ を変化させその時の PSNR および GBD の変化を調べる。さまざまな画像で実験を行い、その中から比較的高周波数成分の少ない Lenna 画像および高周波数成分を多く含む Barbara 画像を例として示す。

図 5.13, 5.14 に Lenna および Barbara 画像における τ と PSNR の関係を示す。閾値は各帯域ごとに設定可能である。各帯域における閾値 $\tau_{LH}, \tau_{HL}, \tau_{HH}$ を変動させる。図において (a) は τ_{LH} を固定し、 τ_{HL} および τ_{HH} を変化させたときの PSNR の変動を示す。同様に (b) は τ_{HL} , (c) は τ_{HH} を固定し、残りのパラメータを変化させたときの PSNR の変動を示す。同様に、閾値 τ を変化させたときの GBD の変動を図 5.15, 5.16 に示す。また、さまざまな画像に対して閾値 τ を変化させたときの PSNR および GBD の変動を調べ、最も良い値となった τ を表 5.1 に示す。

図 5.13 および表 5.1 より Lenna 画像では、 $\tau_{LH} = 4.68, \tau_{HL} = 4.68, \tau_{HH} = 1.76$ に

5.3 実験結果

において GBD が最小となり、ブロックひずみが最も少ないことがわかる。一方 PSNR は、 $\tau_{LH} = 1.98, \tau_{HL} = 1.98, \tau_{HH} = 3.56$ で最大となる。また、図 5.14 および表 5.1 より高周波数成分が多い Barbara 画像では、 $\tau_{LH} = 3.48, \tau_{HL} = 3.48, \tau_{HH} = 3.78$ で GBD が最小となる。PSNR だけではブロックひずみは評価できないため GBD を用いて比較する。図 5.15, 5.16 より、GBD は閾値が低いときに良くなる。また、表 5.1 より、エッジ成分が多い画像ほど τ_{LH}, τ_{HL} が τ_{HH} にくらべ小さくなる傾向にあり、平坦な部分が多い画像では、 τ_{LH}, τ_{HL} が τ_{HH} にくらべ大きくなる傾向にある。その両方を持つ画像については平均的な値をとることがわかる。全画像の平均値をとると $\tau_{LH} = 2.58, \tau_{HL} = 2.58, \tau_{HH} = 4.25$ となる。そこで低周波数成分および高周波数成分がほぼ同じくらい存在すると各閾値が平均的な値をとることから四捨五入し $\tau_{LH} = 3, \tau_{HL} = 3, \tau_{HH} = 4$ を設計値とする。

実際に閾値 τ の効果を復号画像を用いて確認する。図 5.17 に Lenna 画像、図 5.18 に Barbara 画像の復号画像を示す。図 5.17(d), 5.18(d) の JPEG 画像に混入したブロックひずみが $\tau = 0$ の復号画像では除去できていないのに対し、 τ を設計値とすると除去できている。また、 $\tau = 50$ とするとブロックひずみは除去できているがエッジ成分が不鮮明となることが確認できる。

5.3.6 実験結果

ビットレートと PSNR および GBD の関係

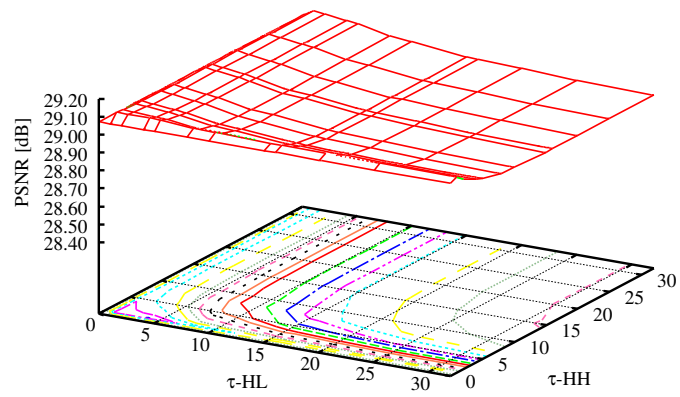
シミュレーション条件 JPEG との比較を行うため、圧縮率を変化させ PSNR および GBD を評価する。ブロックひずみを除去するための LPF として

$$L(z) = \frac{1}{3}z^{-1} + \frac{1}{3} + \frac{1}{3}z \quad (5.20)$$

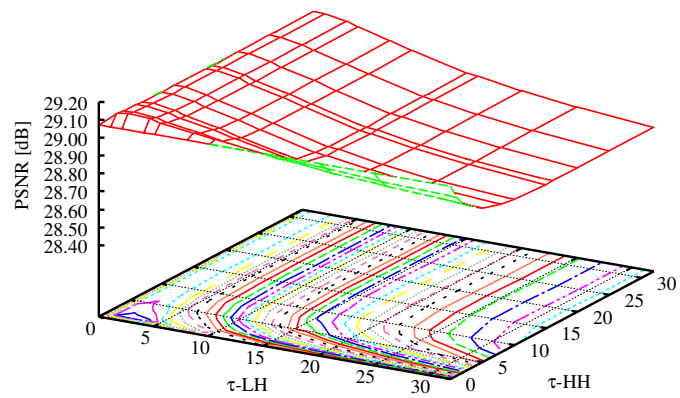
を用いる。また、ウェーブレット変換の基底として $5/3$ フィルタを用い、 $\tau_{LH} = 3, \tau_{HL} = 3, \tau_{HH} = 4$ とする。

考察 図 5.19 にビットレートを変動させたときの PSNR および GBD の推移を示す。JPEG によって圧縮されたビットレートを横軸にとり、原画像と JPEG による復号画像

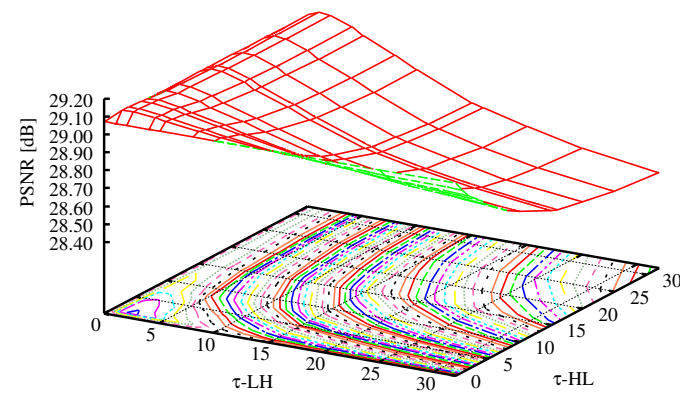
第 5 章 高精細化技術



(a) $\tau_{LH} = 5$ 固定



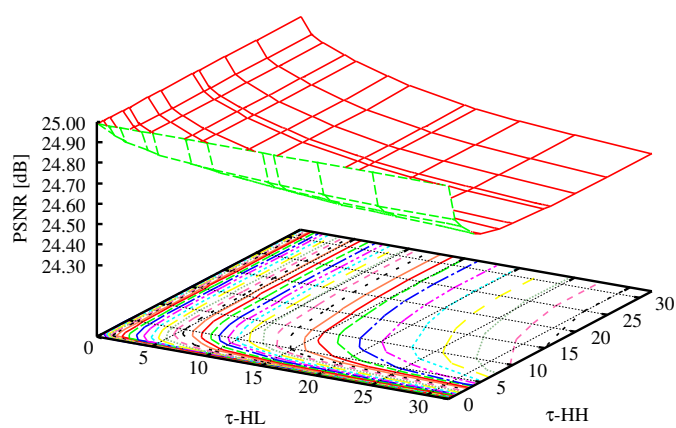
(b) $\tau_{HL} = 5$ 固定



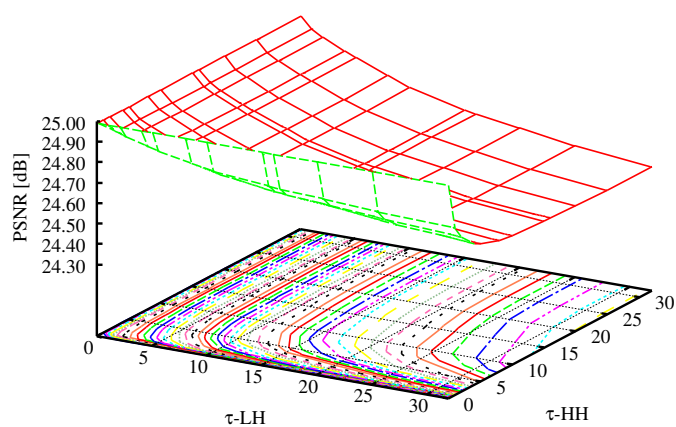
(c) $\tau_{HH} = 5$ 固定

図 5.13 Lenna 画像における τ の変化における PSNR の変化

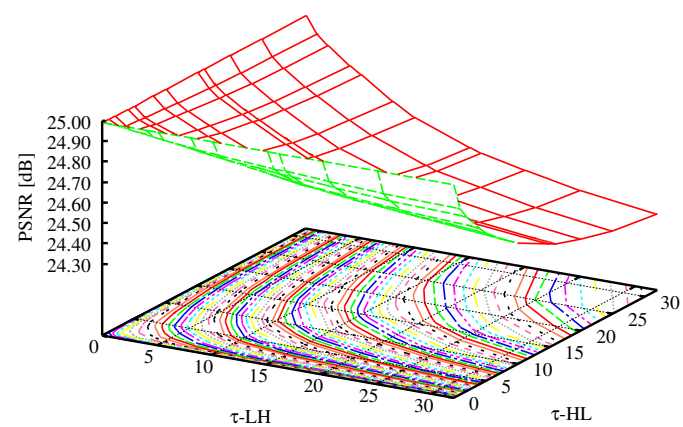
5.3 実験結果



(a) $\tau_{LH} = 5$ 固定



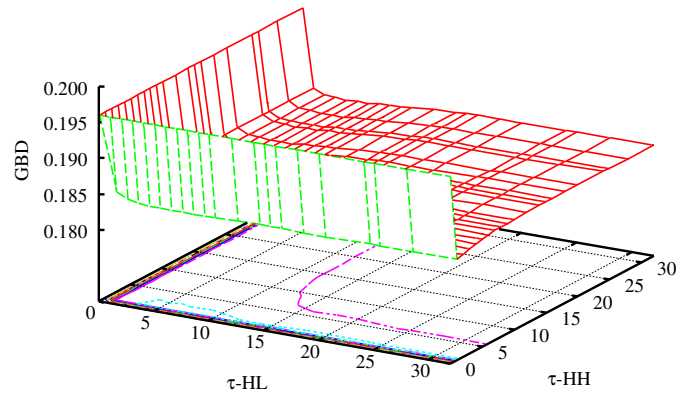
(b) $\tau_{HL} = 5$ 固定



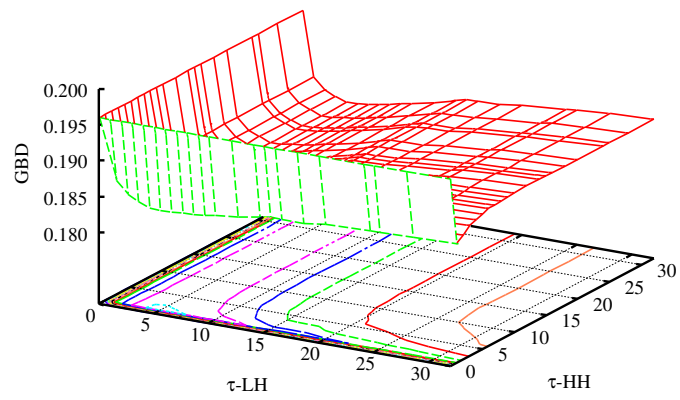
(c) $\tau_{HH} = 5$ 固定

図 5.14 Barbara 画像における τ の変化における PSNR の変化

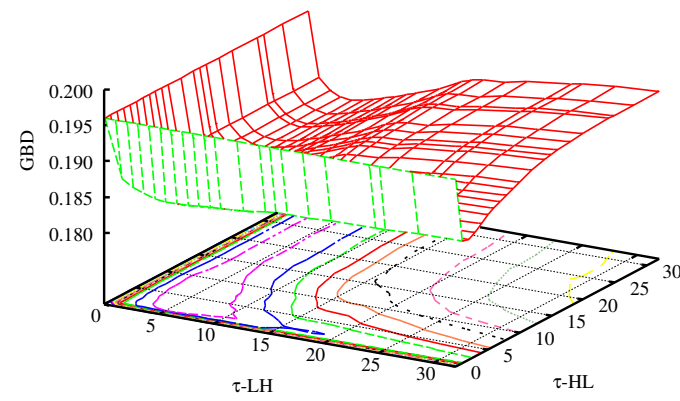
第 5 章 高精細化技術



(a) $\tau_{LH} = 5$ 固定



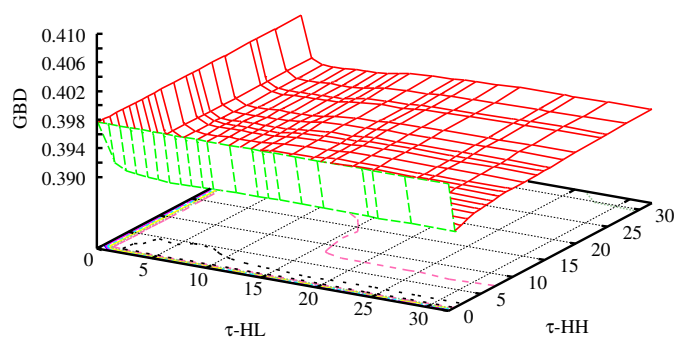
(b) $\tau_{HL} = 5$ 固定



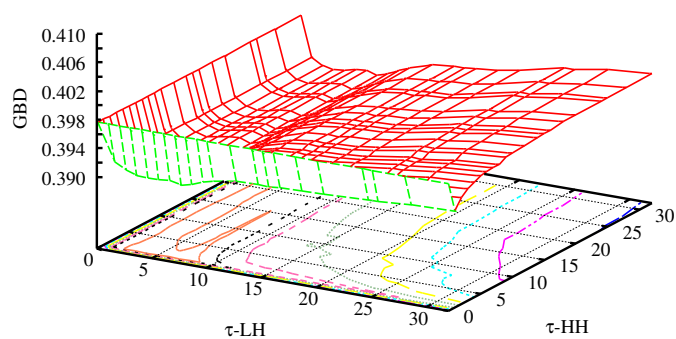
(c) $\tau_{HH} = 5$ 固定

図 5.15 Lenna 画像における τ の変化における GBD の変化

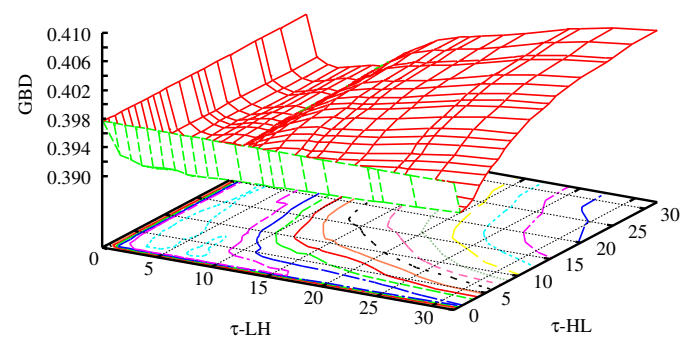
5.3 実験結果



(a) $\tau_{LH} = 5$ 固定



(b) $\tau_{HL} = 5$ 固定



(c) $\tau_{HH} = 5$ 固定

図 5.16 Barbara 画像における τ の変化における GBD の変化

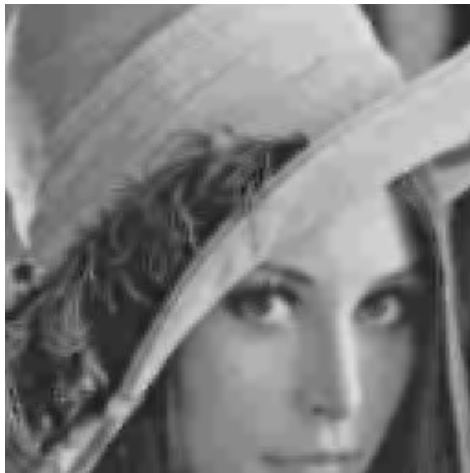
第 5 章 高精細化技術

表 5.1 各画像における閾値 τ の設計

画像	GBD			PSNR		
	τ_{LH}	τ_{HL}	τ_{HH}	τ_{LH}	τ_{HL}	τ_{HH}
Lenna	4.68	4.68	1.76	1.98	1.98	3.56
Barbara	3.48	3.48	3.78	1.60	1.60	3.68
peppers	4.66	4.66	3.30	8.26	8.26	2.20
boats	1.56	1.56	4.10	8.26	8.26	2.20
cap	2.74	2.74	5.24	5.24	5.24	3.62
window	3.92	3.92	5.12	8.26	8.26	2.20
island	2.40	2.40	4.76	3.04	3.04	4.90
parrot	4.00	4.00	4.02	3.66	3.66	3.12
article9	0.00	0.00	4.90	8.26	8.26	2.20
bicycle	1.94	1.94	3.82	8.26	8.26	2.20
bridge	2.32	2.32	4.98	8.26	8.26	2.20
cameraman	1.22	1.22	5.58	8.26	8.26	2.20
cablecar	2.06	2.06	3.56	8.26	8.26	2.20
flower	3.62	3.62	3.82	2.42	2.42	4.30
pens	4.66	4.66	1.70	4.06	4.06	3.54
yacht	1.66	1.66	5.42	8.26	8.26	2.20
goldhill	2.00	2.00	4.88	8.26	8.26	2.20
yacht2	1.26	1.26	4.38	8.26	8.26	2.20
kid	2.00	2.00	4.84	8.26	8.26	2.20
lighthouse	1.14	1.14	5.12	0.90	0.90	4.08
shuttle	2.82	2.82	4.08	8.26	8.26	2.20
平均	2.58	2.58	4.25	6.20	6.20	2.83

の PSNR を点線, GBD を 2 点破線で示し, 原画像と提案法による復号画像の PSNR を実線, GBD を 1 点破線で示す.

これらの画像より, 0.17 bpp 以下のビットレートでは PSNR は勝っているが 0.17 bpp 以上においては約 2 ~ 4 dB ほど劣っている. 一方 GBD で比較すると全ての画像, 全てのビットレートにおいて提案法が優れていることが確認できる. つまり, ブロックひずみの

(a) $\tau = 0$ (b) $\tau_{LH} = 3, \tau_{HL} = 3, \tau_{HH} = 4$ (設計値)(c) $\tau = 50$ 

(d) JPEG

図 5.17 復号画像 (Lenna)

が除去されていることがわかる。PSNR が示すように原画像と比べて雑音が混入したと考えられるが、ブロックひずみのような視覚的に目障りな雑音は GBD が示すように除去されており、提案法によりブロックひずみ混入画像の画質改善が可能となる。

実際にブロックひずみが除去されているかを確認するため主観的評価を行う。主観評価としてまず、復号画像を比較する。図 5.20 ~ 5.25 に 0.2 bpp, 0.25 bpp, 0.35 bpp にお

第 5 章 高精細化技術



(a) $\tau = 0$



(b) $\tau_{LH} = 3, \tau_{HL} = 3, \tau_{HH} = 4$ (設計値)



(c) $\tau = 50$



(d) JPEG

図 5.18 復号画像 (Barbara)

ける復号画像を示す. 左の列に JPEG による復号画像を右の列に提案法による復号画像を示す. 図 5.20 において 高圧縮された (a) の画像にはブロックひずみが混入し, 肌および帽子に階段状のブロックがはっきり認識できる. しかしこの画像に対して提案法を適用した (b) の画像では, ブロックひずみは確認できず, エッジ成分も鮮明である. 圧縮率を

下げ、ビットレートを 0.25 bpp にした場合、図 (c) においてもまだブロックひずみは混入しているが、提案法においてブロックひずみが軽減できていることが確認できる。さらにビットレートを 0.35 bpp とした場合には、JPEG および提案法において良好な画像が得られた。

次に滑らかな領域と振動の激しい領域が混在している Barbara 画像における実験結果を図 5.21 に示す。図 5.21 では、Lenna 画像のときと同様ブロックひずみが軽減できている。(a) では滑らかな領域はブロックひずみが軽減され、エッジ成分は保持されていることが確認できる。(b) において顔の部分は劣化がみられるが、これは (a) の画像において既に劣化がみられ、原画像の情報は所持していないため、このような画像になると考えられる。また、Barbara 画像では 0.35 bpp においてもブロックひずみが確認できる。これらの少量のブロックひずみが混入していても良好な復号画像が得られることが確認できる。

図 5.22 に boats 画像による実験結果を示す。(a) において船の横に認知できるブロックひずみは提案法を用いた (b) では、みられない。しかし、甲板の左上にある棒状のエッジがとぎれていることが確認できる。

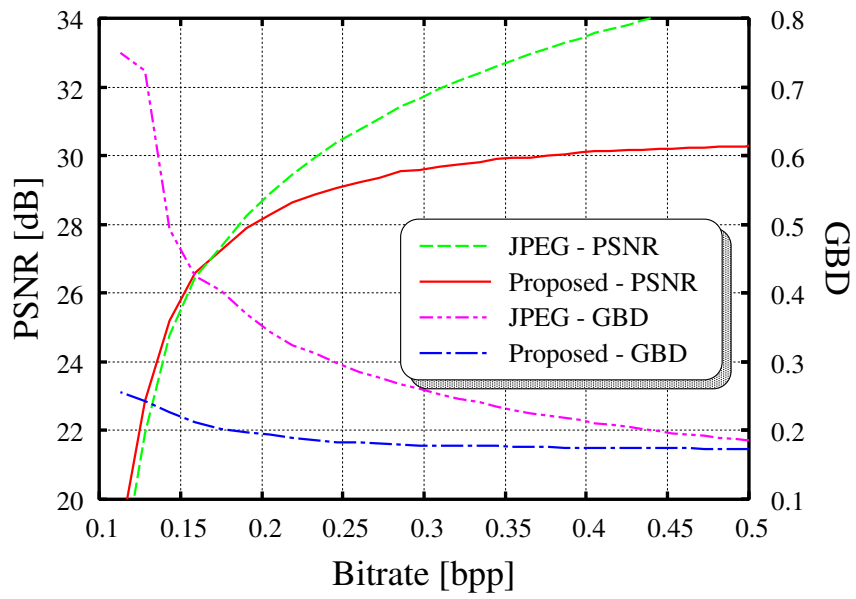
図 5.23, 5.24 に示す peppers 画像および pens 画像では、グラデーションであった領域に混入したブロックひずみを除去できていることが確認できる。

図 5.25 に示す flower 画像では、(a) において花のエッジ成分にもブロックひずみが混入しているが (b) では取り除かれていることが確認できる。

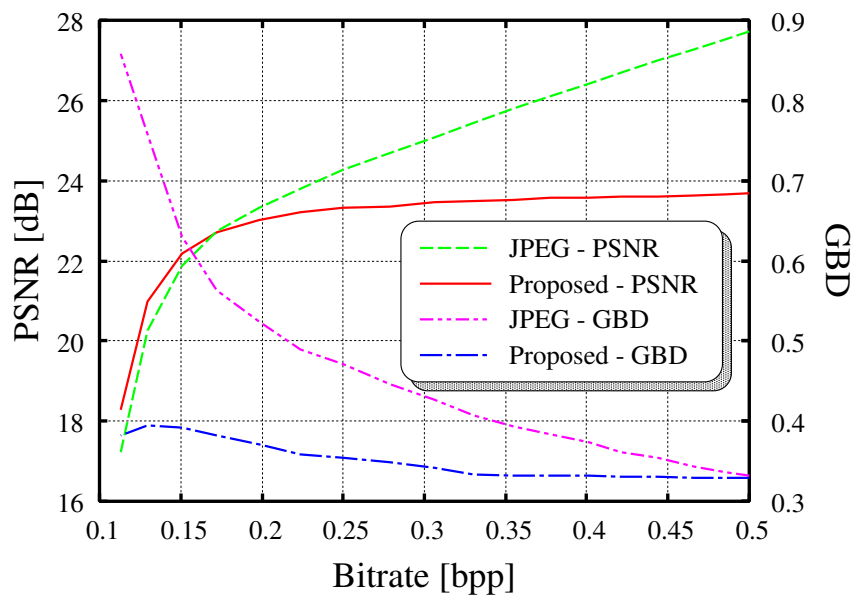
つまり、提案法では低周波数成分の多い領域やグラデーションの領域に対しては混入したブロックひずみを除去することが確認できた。また、エッジ成分が多い画像に対してもエッジ部分周辺に混入したブロックひずみを除去することができ、かつエッジ成分は保持できている。

画像全体 LPF との比較

提案法によりブロックひずみが除去できることを確認したため、ブロックひずみを軽減する従来法である画像全体に LPF をかける方法との比較を行う。

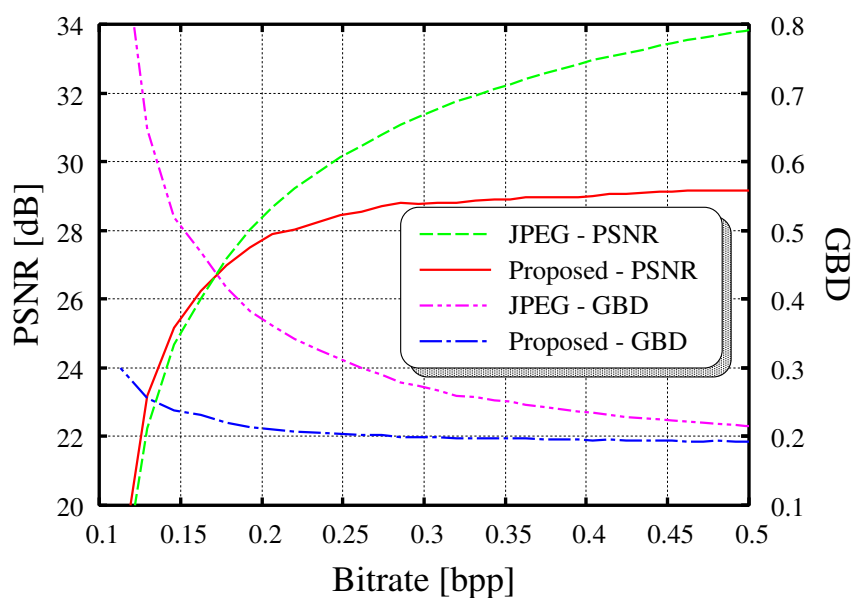


(a) Lenna

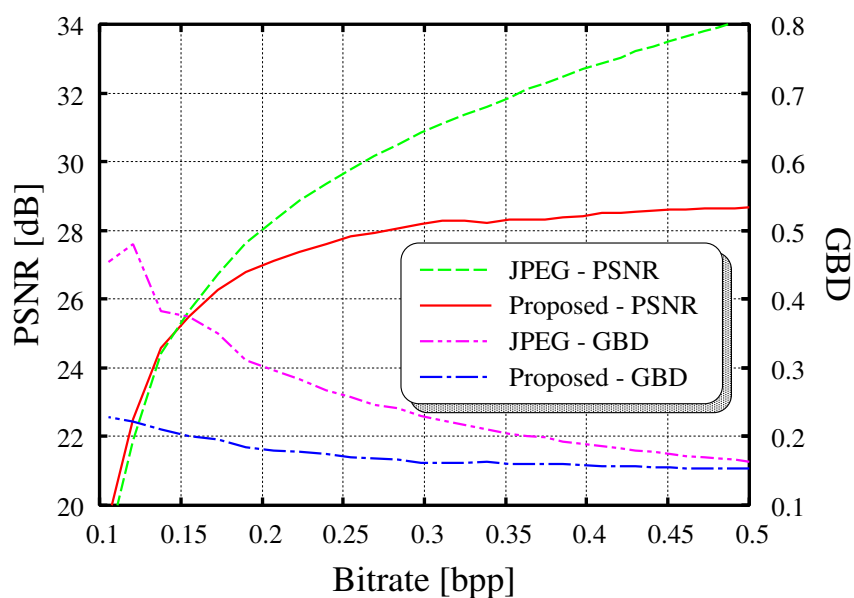


(b) Barbara

図 5.19 JPEG と提案法の比較 ($\tau_{LH} = 3, \tau_{HL} = 3, \tau_{HH} = 4$)

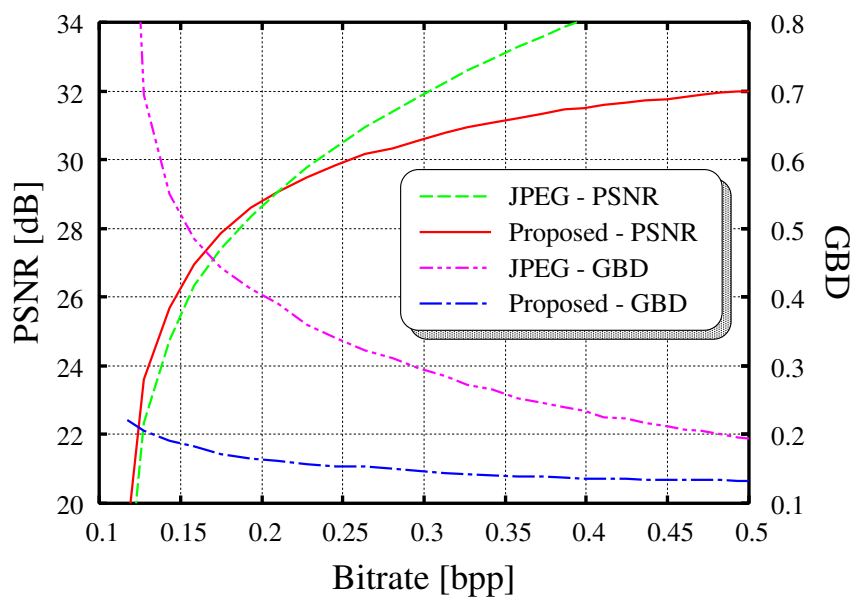


(c) peppers

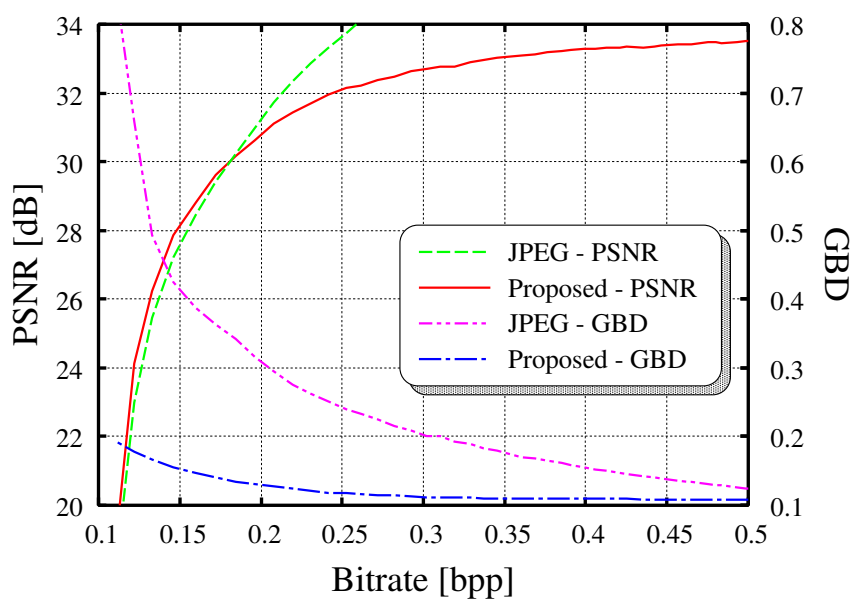


(d) boats

図 5.19 JPEG と提案法の比較 ($\tau_{LH} = 3, \tau_{HL} = 3, \tau_{HH} = 4$)



(e) pens



(f) flower

図 5.19 JPEG と提案法の比較 ($\tau_{LH} = 3, \tau_{HL} = 3, \tau_{HH} = 4$)

5.3 実験結果



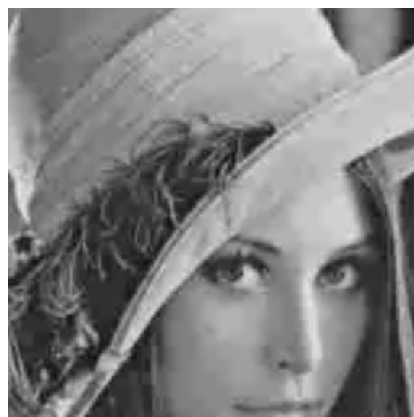
(a) JPEG (0.20[bpp], 28.2[dB], 0.370)



(b) 提案法 (0.20[bpp], 28.2[dB], 0.189)



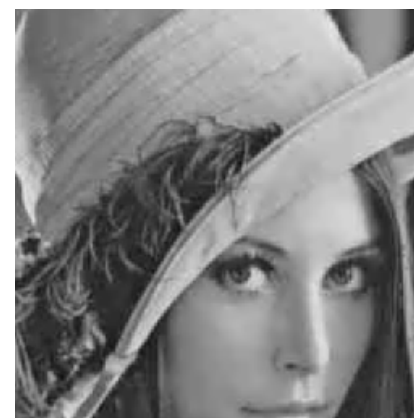
(c) JPEG (0.25[bpp], 30.4[dB], 0.2978)



(d) 提案法 (0.25[bpp], 29.1[dB], 0.182)



(e) JPEG (0.35[bpp], 32.8[dB], 0.230)



(f) 提案法 (0.35[bpp], 29.9[dB], 0.177)

図 5.20 復号画像による比較 (Lenna)

第 5 章 高精細化技術



(a) JPEG (0.20[bpp], 23.3[dB], 0.524)



(b) 提案法 (0.20[bpp], 23.1[dB], 0.372)



(c) JPEG (0.25[bpp], 24.3[dB], 0.471)



(d) 提案法 (0.25[bpp], 23.5[dB], 0.354)



(e) JPEG (0.35[bpp], 25.8[dB], 0.393)



(f) 提案法 (0.35[bpp], 24.1[dB], 0.332)

図 5.21 復号画像による比較 (Barbara)

5.3 実験結果



(a) JPEG (0.20[bpp], 28.3[dB], 0.296)



(b) 提案法 (0.20[bpp], 27.2[dB], 0.180)



(c) JPEG (0.25[bpp], 29.8[dB], 0.257)



(d) 提案法 (0.25[bpp], 27.9[dB], 0.170)



(e) JPEG (0.35[bpp], 31.9[dB], 0.205)



(f) 提案法 (0.35[bpp], 28.6[dB], 0.160)

図 5.22 復号画像による比較 (boats)

第 5 章 高精細化技術



(a) JPEG (0.20[bpp], 28.0[dB], 0.382)



(b) 提案法 (0.20[bpp], 27.5[dB], 0.212)



(c) JPEG (0.25[bpp], 30.1[dB], 0.313)



(d) 提案法 (0.25[bpp], 28.5[dB], 0.203)



(e) JPEG (0.35[bpp], 32.3[dB], 0.250)



(f) 提案法 (0.350[bpp], 29.0[dB], 0.197)

図 5.23 復号画像による比較 (peppers)

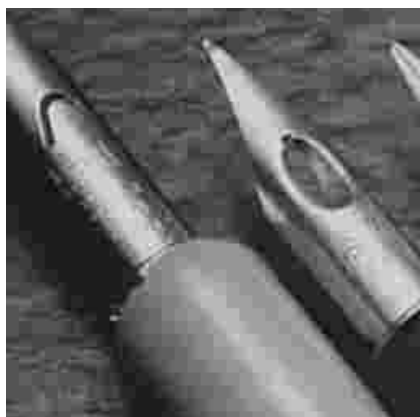
5.3 実験結果



(a) JPEG (0.20[bpp], 28.4[dB], 0.411)



(b) 提案法 (0.20[bpp], 28.6[dB], 0.158)



(c) JPEG (0.25[bpp], 30.4[dB], 0.341)



(d) 提案法 (0.25[bpp], 29.8[dB], 0.147)



(e) JPEG (0.35[bpp], 32.9[dB], 0.265)



(f) 提案法 (0.35[bpp], 31.1[dB], 0.133)

図 5.24 復号画像による比較 (pens)

第 5 章 高精細化技術



(a) JPEG (0.20[bpp], 31.7[dB], 0.295)



(b) 提案法 (0.20[bpp], 31.1[dB], 0.123)



(c) JPEG (0.25[bpp], 33.7[dB], 0.240)



(d) 提案法 (0.25[bpp], 32.1[dB], 0.114)



(e) JPEG (0.35[bpp], 36.8[dB], 0.175)



(f) 提案法 (0.35[bpp], 33.1[dB], 0.106)

図 5.25 復号画像による比較 (flower)

5.3 実験結果

従来法において 3 タップのフィルタでは、図 5.26(c) に示すようにブロックひずみ除去には不十分である。そこで、フィルタのタップ数を 5, 7, 9 と増やしたところ図 5.26(d) ~ 図 5.26(f) のようにエッジ部分が削られてしまい不鮮明な画像となる。

各画像において JPEG による復号画像および提案法による復号画像を (a)、提案法による復号画像を (b)、従来法においてフィルタのタップ数を 3 ~ 9 に変化させた場合をそれぞれ (c) ~ (f) に示す。どの画像においても 3 タップのフィルタではブロックひずみの除去が不十分であるが、ブロックひずみを減少させるためタップ数を増やすとそれに比例して画像が不鮮明となる。そこで最もエッジがはっきりしておりブロックひずみも軽減できている 3 タップの場合と比較を行う。

図 5.26(a), 5.26(b) に示すように提案法と従来法とを比較すると図 5.26(c) はブロックひずみが残っており雑音を除去しきれていない。また、図 5.26(d) ~ 5.26(f) は、ブロックひずみはなくなるがエッジが不鮮明となり、雑音を除去しても画質が低下している。それに比べ提案法ではブロックひずみがなく、かつエッジ部分が鮮明となっていることが確認できる。

エッジ検出画像での比較

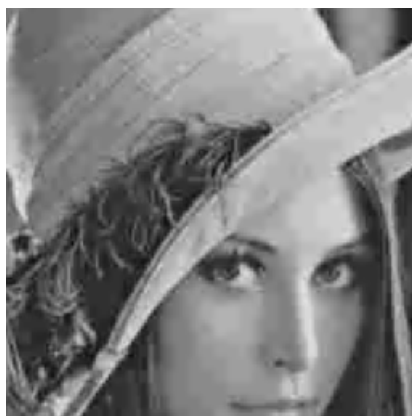
原画像、JPEG による復号画像および提案法による復号画像それぞれに対してエッジ検出フィルタを用いエッジ画像を作成する。原画像のエッジ成分を図 5.32 に示す。

図 5.33 ~ 5.38 に各実験画像においてエッジを検出した画像を示す。(a) に提案法、(b) に JPEG、(c) に 3 タップのフィルタを用いた従来法による復号画像を示す。このように JPEG を用いて復号した画像においては原画像に存在しないブロック状のエッジが多く検出されている。しかし、提案法においてはブロック状のエッジが検出されていないことが確認できる。

第 5 章 高精細化技術



(a) JPEG



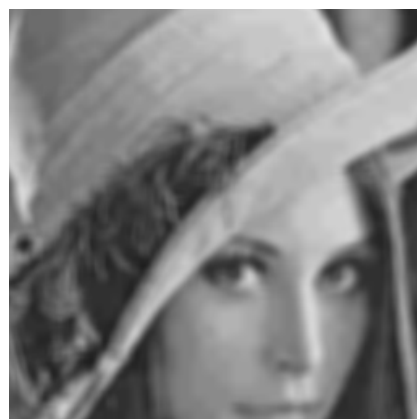
(b) 提案法



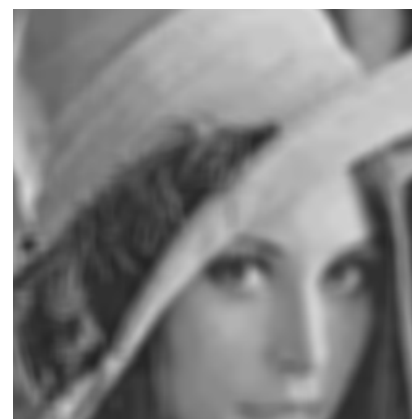
(c) 3 タップ



(d) 5 タップ



(e) 7 タップ



(f) 9 タップ

図 5.26 LPF によるブロックひずみ軽減 (Lenna)

5.3 実験結果



(a) JPEG



(b) 提案法



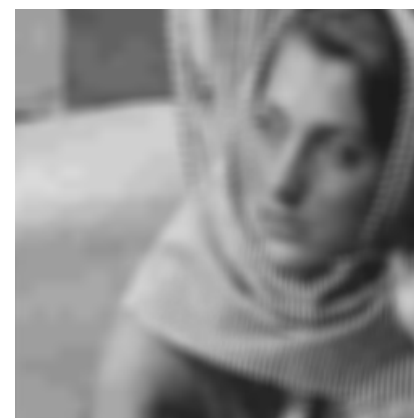
(c) 3 タップ



(d) 5 タップ



(e) 7 タップ



(f) 9 タップ

図 5.27 LPF によるブロックひずみ軽減 (Barbara)

第5章 高精細化技術



(a) JPEG



(b) 提案法



(c) 3 タップ



(d) 5 タップ



(e) 7 タップ



(f) 9 タップ

図 5.28 LPF によるブロックひずみ軽減 (boats)

5.3 実験結果



(a) JPEG



(b) 提案法



(c) 3 タップ



(d) 5 タップ



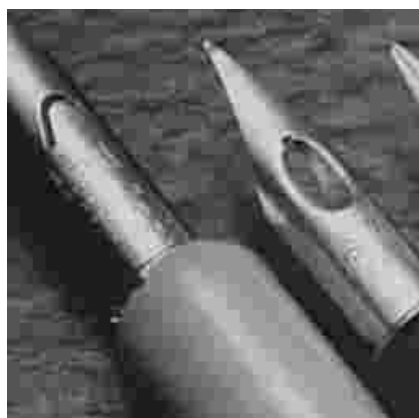
(e) 7 タップ



(f) 9 タップ

図 5.29 LPF によるブロックひずみ軽減 (peppers)

第 5 章 高精細化技術



(a) JPEG



(b) 提案法



(c) 3 タップ



(d) 5 タップ



(e) 7 タップ



(f) 9 タップ

図 5.30 LPF によるブロックひずみ軽減 (pens)

5.3 実験結果



(a) JPEG



(b) 提案法



(c) 3 タップ



(d) 5 タップ



(e) 7 タップ



(f) 9 タップ

図 5.31 LPF によるブロックひずみ軽減 (flower)

第 5 章 高精細化技術



(a) Lenna 画像



(b) Lenna 画像のエッジ成分



(c) Barbara 画像



(d) Barbara 画像のエッジ成分



(e) boats 画像



(f) boats 画像のエッジ成分

図 5.32 原画像におけるエッジ成分



(g) peppers 画像



(h) peppers 画像のエッジ成分



(i) pens 画像



(j) pens 画像のエッジ成分



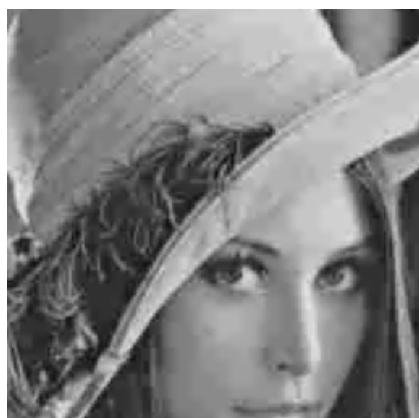
(k) flower 画像



(l) flower 画像のエッジ成分

図 5.32 原画像におけるエッジ成分

第 5 章 高精細化技術



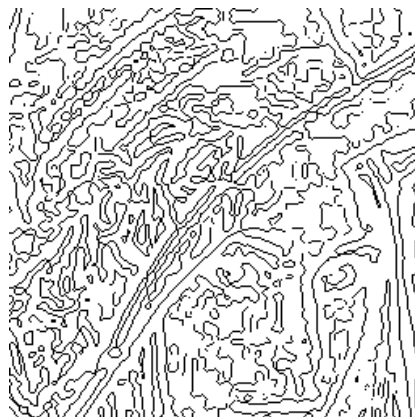
(a) 提案法



(b) 提案法のエッジ成分



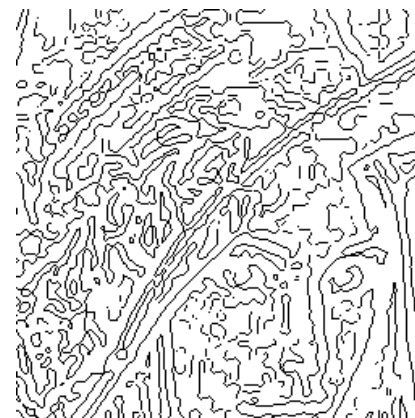
(c) JPEG



(d) JPEG 画像のエッジ成分



(e) LPF



(f) LPF 画像のエッジ成分

図 5.33 ブロックひずみ除去法による画像のエッジ成分 (Lenna)



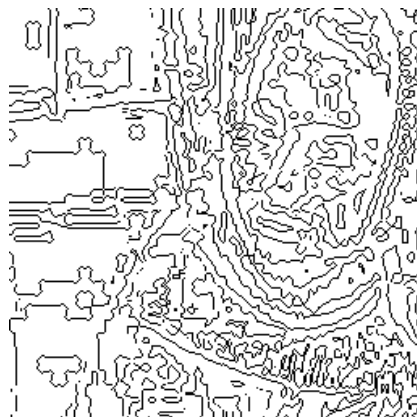
(a) 提案法



(b) 提案法のエッジ成分



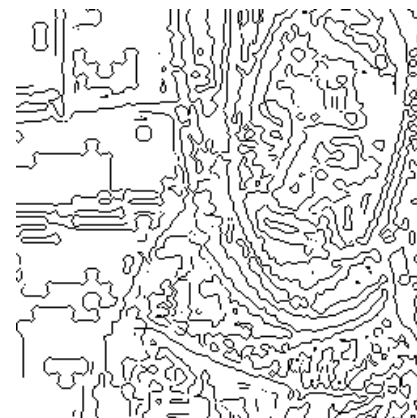
(c) JPEG



(d) JPEG 画像のエッジ成分



(e) LPF



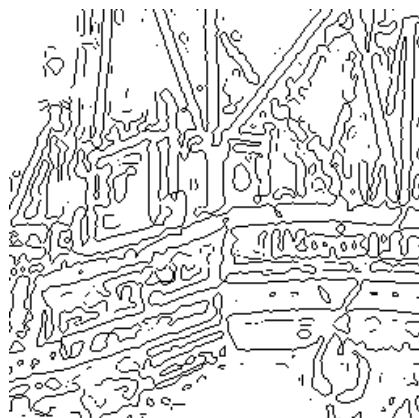
(f) LPF 画像のエッジ成分

図 5.34 ブロックひずみ除去法による画像のエッジ成分 (Barbara)

第5章 高精細化技術



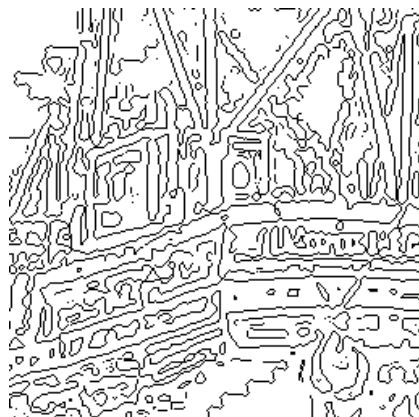
(a) 提案法



(b) 提案法のエッジ成分



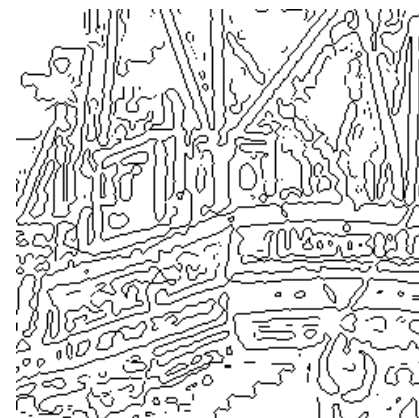
(c) JPEG



(d) JPEG 画像のエッジ成分



(e) LPF



(f) LPF 画像のエッジ成分

図 5.35 ブロックひずみ除去法による画像のエッジ成分 (boats)



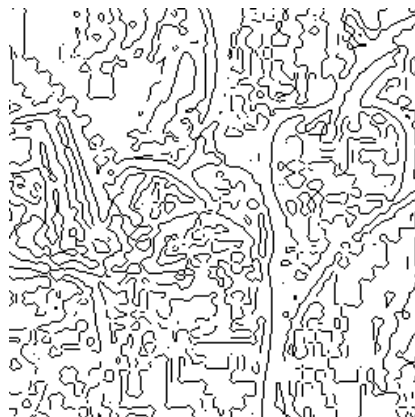
(a) 提案法



(b) 提案法のエッジ成分



(c) JPEG



(d) JPEG 画像のエッジ成分



(e) LPF



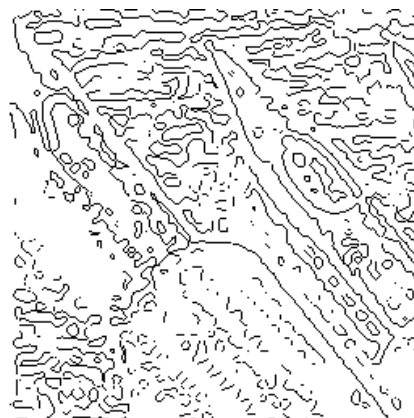
(f) LPF 画像のエッジ成分

図 5.36 ブロックひずみ除去法による画像のエッジ成分 (peppers)

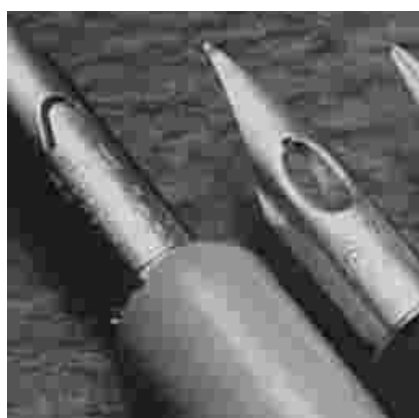
第 5 章 高精細化技術



(a) 提案法



(b) 提案法のエッジ成分



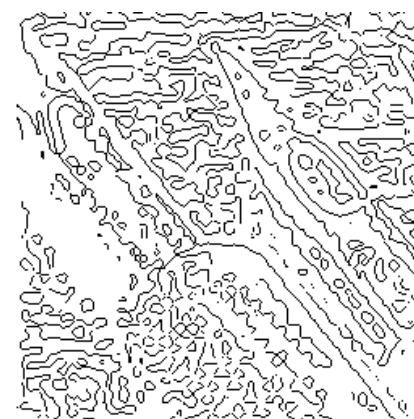
(c) JPEG



(d) JPEG 画像のエッジ成分



(e) LPF

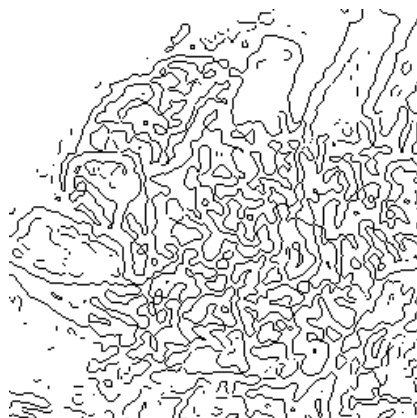


(f) LPF 画像のエッジ成分

図 5.37 ブロックひずみ除去法による画像のエッジ成分 (pens)



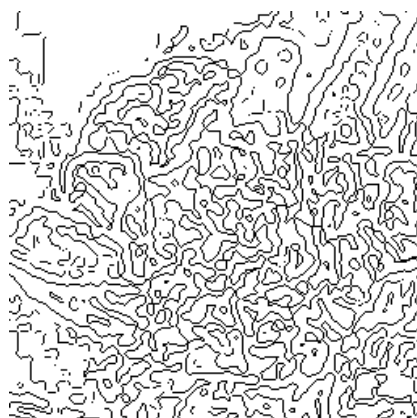
(a) 提案法



(b) 提案法のエッジ成分



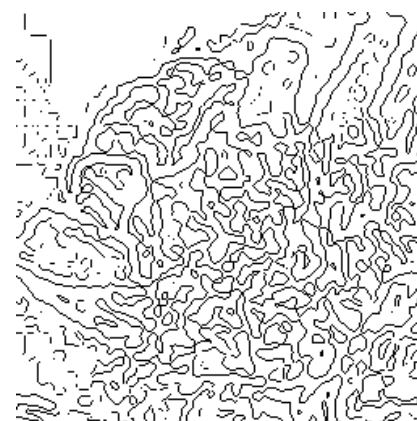
(c) JPEG



(d) JPEG 画像のエッジ成分



(e) LPF



(f) LPF 画像のエッジ成分

図 5.38 ブロックひずみ除去法による画像のエッジ成分 (flower)

5.4 むすび

JPEG や MPEG において符号化方式として採用されている DCT はブロックごとに量子化などの処理を行う。そのため、圧縮率を高めるにつれて主に低周波数成分の量子化誤差によりブロックひずみが生じ、復号画像の画質が劣化し再現性が著しく低下してしまうことが問題となる。

これらの問題を解決するものとしてウェーブレット変換が注目されており、新しい国際標準規格である JPEG2000 にも採用されている。

このように近年では JPEG2000 や MPEG4 といった技術が登場し、静止画像や、動画画像がより高画質で扱われるようになってきた。しかし、数年前まではハードディスクの容量が小さく伝送路も狭かったため、画像は小さなファイルサイズで保存・伝送されてきた。そのため、JPEG 形式で高圧縮し雑音が入った画像が多く存在している。しかし、JPEG2000 の登場により、従来保存容量の制限等で小さくせざるを得なかったデジタルカメラ画像や Web 上に存在する既存の JPEG 画像をそのまま捨ててしまうのは忍びない。

ブロックひずみを軽減する方法として POCS、プレポストフィルタリング法などさまざまな方法が提案されている。しかし、処理が複雑であったり既に符号化された画像に対して適用できないなどの問題がある。

そこで本研究では、逆ウェーブレット変換による DCT 係数の高画質化復号法を提案した。提案法では DCT 係数表現はウェーブレット変換係数表現と等価であることを示し、さらに、画像信号が周波数帯域に分割されていることを利用し、ブロックひずみを軽減しつつエッジ成分は保持する復号化である。提案法を既に符号化された画像に対して適用することにより、簡単な操作で画像に混入したブロックひずみを除去可能であることを客観評価の指標である PSNR およびブロックひずみの評価法である GBD を用いて示した。また、ブロックひずみの除去のみでなく高周波数成分であるエッジ成分の保持も実現でき

ることを主観評価により確認し、劣化画像に対する本提案法の有効性を示した。さらに、低周波数成分が多い画像、高周波数成分が多い画像および低周波数成分・高周波数成分が混在している画像など画像の特性が異なる場合でも提案法によりブロックひずみが除去できることを確認した。しかし、画像によっては符号化された時点で既に多くの情報を失っているものもあり、このような画像では復号画像の高画質化に限界があることもわかった。

近年高解像度の表示装置が開発されており、低解像度の JPEG 画像を表示する際には解像度変換が必要となる。その拡大時にブロック状のひずみが混入することが問題となるため、今後の課題として提案法に拡大プログラムを組み込むことにより、画像のブロックひずみを除去しつつ画像を拡大することが挙げられる。この処理では以下の 2 つの手法で実現可能である。

- DCT 係数上で拡大を行い、その出力に対して提案法を適用すること
- 逆ウェーブレット変換する直前のウェーブレット変換係数をウェーブレット変換係数上で拡大すること

また、逆ウェーブレット変換を 1 階層ではなくさらなる階層化を行うため、DCT 係数でオクターブ構造を構成することによりさらなる改善が期待できると考えられる。

第 6 章

結論

本論文では、デジタル画像処理技術の高圧縮化・高精細化を目的として近年デジタル画像圧縮の分野で注目を集めているウェーブレット変換符号化における問題点および JPEG や MPEG で用いられている DCT 符号化時の問題点に焦点を当て、それらの解決法の一手法を提案し、その有効性について検討を行った。

本章では、本研究の成果を高圧縮符号化および高精細化の各提案手法毎にまとめると共に今後の課題および展望について論じる。

6.1 高圧縮符号化技術

本論文では、高圧縮符号化技術に対する提案として

- マルチ走査によるウェーブレット変換符号化
- ウェーブレット – DCT 符号化
- カラー成分間の相関を利用した画像符号化

の提案を行い、その有効性について確認した。本節では、各提案手法についてまとめる。

第 6 章 結論

6.1.1 マルチ走査によるウェーブレット変換符号化

DCT 符号化方式では画像を任意のブロックに分割しそのブロックごとで処理するため並列処理が可能であり、FFT の高速アルゴリズムを使用することができるため、高速処理が可能となる。しかし、圧縮率を上げるにつれてブロック間の相関がなくなりブロックひずみやモスキート雑音が復号画像に混入してしまう。そこでこれらの雑音が混入しない符号化方式としてフィルタバンク符号化方式が注目されている。フィルタバンク符号化方式では、画像全体に対してフィルタリングをするため、フィルタの次数に応じた処理時間が必要になる。また、従来のフィルタバンクでは、各画素を水平方向および垂直方向に走査するため、画素を重複して走査することになる。

そこで本研究では高速化および斜め方向のエッジ成分の再現性向上の観点から各画素を重複することなく走査できるマルチ走査を提案し、提案走査を適用したウェーブレット変換符号化器の設計を行い、比較のために従来走査を適用したウェーブレット変換符号化器の設計も行った。

ウェーブレット変換は 2 タップのハール基底を用いてフィルタバンクで構成した。また、量子化器を設計し、ウェーブレット変換係数の低域成分に対しては

- 人間の視覚特性は輝度信号に対して感度が高い
- 自然画像の低周波数成分の振幅値は一様に分布している

などの理由により線形量子化を用い、高域成分に対しては

- エッジ成分を残す
- 画像ノイズを取り除く

などの理由により中間ステップ量子化を適用した。さらに符号化器を設計し、

- 低域成分は隣接画素間の相関が強い

- 高域成分は中間ステップ量子化により零値が多く出力される

といった理由により、低域成分に対しては DPCM 符号化を、高域成分に対してはランレングス符号および可変長符号を用いて符号化した。以上の仕様のウェーブレット変換符号化器を C 言語を用いてプログラミングし、その動作の確認を行った。

得られた符号データの平均ビットレートおよび復号画像のピーク SN 比よりマルチ走査は客観的な画質低下をひき起こすことを示した。しかし、主観評価においては、斜め方向のエッジの再現性が向上しており、視覚的に良好な画像が生成できることを示した。

また、従来の走査方法に比べマルチ走査を用いた符号および復号で約 1.67 倍高速に動作することが確認でき、以上の結果より提案したマルチ走査の有用性が示された。今回はフィルタバンクに 2 タップのハール基底を用いたが、例えば、4 タップのフィルタを用いた場合、従来の走査に比べ約 2.86 倍高速に動作し、8 タップのフィルタでは約 2.95 倍高速に動作する。つまり高次のフィルタを用いるとフィルタリングに要する時間が更に短縮され、提案したマルチ走査の有用性が増すことになる。

また、いずれの走査方法を用いた場合においても走査方向と同じ方向のエッジは滑らかに復元することができるが、走査方向とは違う方向のエッジは滑らかに復元することができないことも確認した。

6.1.2 ウェーブレット – DCT 符号化

現在、国際標準の画像圧縮符号化方式である JPEG に用いられている DCT は、画像を任意のブロックに分割しそのブロック毎に量子化、符号化などの処理を行うため、圧縮率を上げるにつれブロック歪みやモスキート雑音などの雑音が復号画像に混入し画質の劣化が生じてしまう。そこでこれらの雑音の混入を抑えられる符号化方式としてウェーブレット変換符号化が注目されている。しかし、ウェーブレット変換符号化を用いている JPEG2000 では JPEG に比べ、圧縮で 5～6 倍、伸長で 3～4 倍の処理能力が必要とされている。また、JPEG のようにジグザグ走査や EOB、動画に対する動き補償のような効

第 6 章 結論

果的な圧縮方法が確立していない等の問題点が残っている。

そこで本研究では新しい符号化手法として前段でウェーブレット変換によって各帯域に分割された係数に対して後段で DCT を行うウェーブレット – DCT 符号化を提案した。

提案する手法の有用性を示すためにウェーブレット – DCT 符号化器を C 言語を用いて設計し、比較のために DCT 符号化器の設計も行った。ウェーブレット変換のフィルタバンクには Haar, 5/3, 9/7 フィルタを用い、DCT は各帯域で任意に選択できるように様々なブロックサイズを用いた。また、ブロックサイズに合わせた量子化テーブルを用い線形量子化を行う量子化器を設計した。さらに、符号化器として DCT 係数の直流成分に対しては隣接のブロックの直流成分と相関が強いため DPCM を行った後にハフマン符号化を行った。また、交流成分に対してはブロックサイズに合わせたジグザグ走査を行った後にランレングス符号化とハフマン符号化を行った。以上の仕様のウェーブレット – DCT 符号化器を C 言語を用いてプログラミングし、その動作の確認を行った。

ウェーブレット – DCT 符号化の性能を評価するため、様々な条件において客観評価及び主観評価を用いて検討を行った。客観評価には平均ビットレートとピーク SN 比の特性を調べることによって行った。DCT に用いるブロックサイズは一般に自然画像に対して用いる場合、大きくするほど圧縮効率が良くなることが知られているが、実験からウェーブレット – DCT でのブロックサイズは大きくするにつれて符号化効率が劣化してしまう場合があることを確認し、最適なブロックサイズの組合せを示した。また、ウェーブレット変換で用いるフィルタバンクに高次のフィルタを用いることによってウェーブレット – DCT の符号化効率が改善することも示した。

さらに、主観評価のために復号画像の比較を行った。DCT 符号化と比べウェーブレット – DCT 符号化ではブロック歪みが抑えられていることが確認できた。また、ウェーブレット変換で用いるフィルタバンクを重複型にすることにより DCT を組み込むことによるブロック歪みの影響をさらに軽減できた。しかし、DCT のブロックサイズを大きくするにつれて復号画像においてブロック状の歪みが表れてしまうことも確認できた。

6.1.3 カラー成分間の相関を考慮したウェーブレット変換符号化

静止画像の圧縮符号化方式として広く用いられている JPEG には離散コサイン変換 (DCT) が用いられている。DCT は、符号化効率が高い反面、圧縮率を上げるにつれてブロック歪みやモスキート雑音が復号画像に混入する。そのため、これらの雑音の影響を軽減できる符号化方式としてウェーブレット変換が注目を集めている。カラー画像をウェーブレット変換する場合には、RGB 成分を YC_rC_b に色変換し、各成分を 3 枚のグレースケールの画像を符号化するように YC_rC_b を成分毎に個別に符号化する。そのため、成分間の関係を用い、まとめて符号化することにより符号化効率の改善が期待できる。

そこで本研究では $l C'_r C'_b$ に変換する色変換を提案し、色変換後に $l C'_r C'_b$ の関係を用いた符号化を行うウェーブレット変換符号化器を提案した。提案したウェーブレット変換符号化器を C 言語により作成し、様々な画像に対して復号画像に与える影響を調べた。また、比較のため従来型のウェーブレット変換符号化器でも同様に復号画像に与える影響を調べた。

実験結果の客観評価より提案法の符号化性能は従来法と比べ

- 高ビットレートにおいてピーク SN 比が若干低くなるが、符号化性能はほぼ同等
- 低ビットレートにおいてピーク SN 比は非常に高くなり、符号化性能を劇的に改善

となり、また主観評価より以下のことを確認した。

- 高ビットレートにおける復号画像の画質はほぼ同等
- 低ビットレートにおける復号画像の画質は飛躍的に改善

以上により提案法の有効性を確認した。

第 6 章 結論

表 6.1 DCT 係数に対する解像度変換法のまとめ

	ブロックサイズ	画質	演算回数
提案法（ブロック数変更・周辺参照）	（固定）		
従来法（ブロックサイズ変更）	×（変化）	×	
従来法（ブロック数変更・単一）	（固定）	×	
画像空間上	（固定）		×

6.2 高精細化技術

本論文では、デジタル画像の高精細化技術に対する提案として

- DCT 係数上での解像度変換
- ウェーブレット係数を用いた高画質化

の提案を行い、その有効性について確認した。本節では、各提案手法についてまとめる。

6.2.1 隣接ブロックを用いた DCT 係数上での解像度変換

画像の解像度変換は一般に画像空間上で行われる。しかし、JPEG や MPEG の符号化方式には DCT が用いられていることから、DCT 係数上での解像度変換法が注目されている。

従来の DCT 係数上での解像度変換は、ブロックサイズを変更することで実現している。しかし、DCT のブロックサイズが変化してしまうため JPEG 等の汎用 DCT をそのまま用いることはできない。また、ブロックサイズを固定して処理できても解像度変換後の画像にブロックひずみが発生する問題がある。

そこで本研究では、隣接ブロックを用いた DCT 係数上での解像度変換法を提案し、その有効性について検討した。

表 6.1 に DCT 係数に対する解像度変換法をまとめる。まず、単一ブロックのみを用い

解像度変換を行う従来法と周辺ブロックを参照する提案法で解像度変換後の画像を比較した。縮小画像を比較した結果、従来法と提案法はどちらもほぼ同等の画質の縮小画像が得られた。また拡大画像を比較すると、従来法ではブロックひずみが発生するのに対し、提案法ではブロックひずみが発生しないことが確認できた。

次に実使用に対する実験として、既にブロックひずみが混入した拡大画像による比較実験を行った。その結果、従来法ではブロックひずみがはっきりとした拡大画像が得られるのに対し、提案法ではブロックひずみが軽減できていることを確認した。

最後に、画像拡大の際に必要な演算回数を比較した結果、提案法の演算回数は従来法に比べ多くなっているが、画像空間上での拡大に比べ少なくなっていることを示した。

また、提案法は DCT のブロックサイズが変更されないため JPEG, MPEG 等の汎用 DCT をそのまま用いることができる。

6.2.2 逆ウェーブレット変換を用いたブロックひずみ低減手法

JPEG や MPEG において符号化方式として採用されている DCT はブロックごとに量子化などの処理を行う。そのため圧縮率を高めるにつれて主に低周波数成分の量子化誤差によりブロックひずみが生じ、復号画像の画質が劣化し再現性が著しく低下してしまうことが問題となる。

これらの問題を解決するものとしてウェーブレット変換が注目されており、新しい国際標準規格である JPEG2000 にも採用されている。

このように近年では JPEG2000 や MPEG4, H.264 といった技術が登場し、静止画像や動画画像がより高画質で扱われるようになってきた。しかし、数年前まではハードディスクの容量が小さく伝送路も狭かったため、画像は小さなファイルサイズで高圧縮され、保存・伝送されてきた。そのため、JPEG 形式で高圧縮し雑音が入った画像が多く存在している。しかし、JPEG2000 の登場により、従来保存容量の制限等で小さくせざるを得なかったデジタルカメラ画像や Web 上に存在する既存の JPEG 画像を利用することも

第 6 章 結論

多くなってきている。

ブロックひずみを軽減する方法として POCS, プレポストフィルタリング法などさまざまな方法が提案されている。しかし, 処理が複雑であったり既に符号化された画像に対して適用できないなどの問題がある。

そこで本研究では, 逆ウェーブレット変換による DCT 係数の高画質化復号法を提案した。提案法では DCT 係数表現はウェーブレット変換係数表現と等価であることを示し, さらに, 画像信号が周波数帯域に分割されていることを利用し, ブロックひずみを軽減しつつエッジ成分は保持する復号化である提案法を既に符号化された画像に対して適用することにより, 簡単な操作で画像に混入したブロックひずみを除去可能であることを客観評価指標である PSNR およびブロックひずみの評価法である GBIM を用いて示した。また, ブロックひずみの除去のみでなく高周波数成分であるエッジ成分の保持も実現できることを主観評価により確認し, 劣化画像に対する本提案法の有効性を示した。さらに, 低周波数成分が多い画像, 高周波数成分が多い画像および低周波数成分・高周波数成分が混在している画像など画像の特性が異なる場合でも提案法によりブロックひずみが除去できることを確認した。しかし, 画像によっては符号化された時点で既に多くの情報を失っているものもあり, このような画像では復号画像の高画質化に限界があることもわかった。

6.3 今後の課題および展望

今後の課題として, 本研究では JPEG で用いられている量子化テーブルを基本として量子化を行ったが, 各帯域の係数分布に対応した量子化テーブルの開発, 例えば高周波数帯域では DCT 係数の交流成分のうち低周波数係数を多く削減する等のウェーブレット-DCT の変換係数に適応した量子化器の開発, 動画像への適用, JPEG2000 への対応等が挙げられる。

しかし, 提案法は周辺ブロックを参照するため演算回数が増加してしまう。そこで, 今後の課題としてアルゴリズムの単純化による演算の高速化が挙げられる。また提案法は,

6.3 今後の課題および展望

画像空間上での解像度変換と等価な処理を DCT 係数上に適用したものであった。そのため、解像度変換画像の画質は画像空間上での解像度変換画像と同等で、それを越えることはない。もし、DCT 係数の特徴（例えば、周波数情報など）を用い解像度変換できればさらなる画質の向上が期待できる。現在、JPEG は画像の国際標準として広く普及している。

しかし今後は、新しい画像の国際標準の規格である JPEG2000 が普及するであろう。そのため、JPEG2000 などの新しい画像規格の符号化データに対応した解像度変換も検討する必要がある。

近年高解像度の表示装置が開発されており、低解像度の JPEG 画像を表示する際には解像度変換が必要となる。その拡大時にブロック状のひずみが混入することが問題となるため、今後の課題として提案法に拡大プログラムを組み込むことにより、画像のブロックひずみを除去しつつ画像を拡大することが挙げられる。この処理では以下の 2 つの手法で実現可能である。

- DCT 係数上で拡大を行い、その出力に対して提案法を適用すること
- 逆ウェーブレット変換する直前のウェーブレット変換係数をウェーブレット変換係数上で拡大すること

また、逆ウェーブレット変換を 1 階層ではなくさらなる階層化を行うため、DCT 係数でオクターブ構造を構成することによりさらなる改善が期待できると考えられる。

参考文献

- [1] “ISO/IEC 10918–1, Information technology – Digital compression and coding of continuous-tone still images: Requirements and guidelines”, International Standard, 1994.
- [2] ISO/IEC 11172–1 “Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1.5Mbit/s – Part 1 : Systems”, International Standard, 1993.
- [3] ITU-T Recommendation H.264, “Advanced Video coding for generic audiovisual services”, 2005.
- [4] “ISO/IEC 15444–1, “Information technology – JPEG 2000 image coding system – Part 1: Core coding system”, International Standard, Dec. 2000.
- [5] H.Greenspan, C.H. Anderson, S.Akber: “Image enhancement by nonlinear extrapolation in frequency space”, IEEE Trans. on Image Processing, Vol.9, No.6, pp.1035–1048, Jun. 2000.
- [6] X.Li, M.T.Orchard: “New edge-directed interpolation”, IEEE Trans. on Image Processing, Vol.10, No.10, pp.1521–1527, Oct. 2001.
- [7] C.A.Segall, R.Molina, A.K.Katsaggelos: “High-resolution images from low-resolution compressed video”, IEEE Signal Processing Magazine, pp.37–48, May 2003.
- [8] R.L.Queiroz: “Processing JPEG-compressed image and documents”, IEEE

参考文献

- Trans. on Image Processing, Vol.7, No.12, pp.1661–1672, Dec. 1998.
- [9] K.N. Ngan: “Experiments on two dimensional decimation in time and orthogonal transform domains”, IEEE Trans. on Signal Processing, Vol.11–3, pp.249–263, Oct. 1986.
- [10] 宮本 義弘, 赤岩 芳彦: “DCT 係数上での符号化画像のサイズ変換”, 電子情報通信学会論文誌 (D-II), Vol.J-82–D-II, No.1, pp.53–60, Jan. 1999.
- [11] 村松 正吾, 貴家 仁志: “変換符号化された画像に対する有理数倍率の解像度変換法”, 電子情報通信学会論文誌 (A), Vol.J-77-A, No.3, pp.396–378, Mar. 1994.
- [12] 鄒 曉霞, 貴家 仁志: “DCT 領域での画像解像度変換法の等価表現とその一般化処理”, 電子情報通信学会論文誌 (A), Vol.J-80-A, No.11, pp.1989–1998, Nov. 1997.
- [13] 小島 一浩, 貴家 仁志: “8 点逆 DCT を用いる画像解像度変換法の一般化”, 電子情報通信学会論文誌 (A), Vol.J-82-A, No.5, pp.599–608, May 1999.
- [14] 新堀 英二, 高木 幹雄: “DCT を用いた Gerchberg-Papoulis の反復法を適用した画像画質拡大”, 電子情報通信学会論文誌 (D-II), Vol.J-76–D-II, No.9, pp.1932–1940, Sep. 1993.
- [15] 花村 敏明, 根上 崇, 後藤 富朗, 北村 正: “DCT 係数上での解像度変換法に関する研究”, 情報理論とその応用シンポジウム (SITA2003), pp.229–232, Dec. 2003.
- [16] T.Goto, T.Hanamura, T.Negami, T.Kitamura: “A study on resolution conversion method using DCT coefficients”, 2004 International Symposium on Information Theory and its Applications (ISITA2004), pp.260–265, Oct. 2004.
- [17] “ISO/IEC 14496–1, “Information technology – Coding of audio-visual objects – Part 1: Systems”, International Standard, 2001.
- [18] H. B. Reeves and J. S. Lim: “Reduction of blocking effects in image coding”, IEEE Trans. on Optical Engineering, Vol.23, pp.34–37, 1984.
- [19] Y. Yang, N. P. Galatsanos, and A. K. Katsaggelos: “Regularized reconstruction

- to reduce blocking artifacts of block discrete cosine transform compressed images”, IEEE Trans. on Circuits and Systems for Video Technology, Vol.3, No.6, pp.421–432, Dec. 1993.
- [20] H. S. Malvar and D. H. Staelin: “The LOT: Transform coding without blocking effects”, IEEE Trans. on Acoustics, Speech and Signal Processing, Vol.37, No.4, pp.553–559, Apr. 1989.
- [21] Chengjie Tu and Trac D. Tran: “Context-based entropy coding of block transform coefficients for image compression”, IEEE Trans. on Image Processing, Vol.11, No.11, pp.1271–1283, Nov. 2002.
- [22] 的場 成浩, 青木 透, 池田 宏明: “PSNR によるビデオ客観評価測定装置と測定結果”, 画像電子学会 VMA 研究会, pp.1–9, Jul. 2001.
- [23] H.R.Wu, M.Yuen: “A generalized block-edge impairment metric for video coding”, IEEE Signal Processing Letters, Vol.4, No.11, pp.317–320, Nov. 1997.
- [24] 福田 光一, 川中 彰: “逐次的 DCT 係数推定に基づいた動画像符号化によるブロックひずみの適応的な制御”, 電子情報通信学会論文誌 (A), Vol.J-85-A, No.1, pp.112–123, Jan. 2002.
- [25] Hamid Gharavi and Ali Tabatabai: “Sub-Band Coding of Monochrome and Color Images”, IEEE Trans. on Circuits and Systems, Vol.35, No.2, pp.207–214, Feb. 1988.
- [26] Jerome M. Shapiro: “Embedded Image Coding Using Zerotrees of Wavelet Coefficients”, IEEE Trans. on Signal Processing, Vol.41, No.12, pp.3445–3462, Dec. 1993.
- [27] Charles D. Creusere: “A New Method of Robust Image Compression Based on the Embedded Zerotree Wavelet Algorithm”, IEEE Trans. on Image Processing, Vol.6, No.10, pp.1436–1442, Feb. 1997.

参考文献

- [28] Ohta M., Nogaki S.: “Hybrid Picture Coding with Wavelet Transform and Overlapped Motion Compensated Interframe Prediction Coding”, IEEE Trans. on Signal Processing, Vol.41, No.12, pp.3416–3424, Dec. 1993.
- [29] C.Christopoulos, et al.: “Image Coding System: An Overview”, IEEE Trans. on Consumer Electronics, Vol.46, No.4, pp.1103–1127, Dec. 2000.
- [30] Julien Reichel, et al.: “Integer Wavelet Transform for Embedded Lossy to Lossless Image Compression”, IEEE Trans. on Image Processing, Vol.10, No.3, pp.383–392, Mar. 2001.
- [31] 春日 秀夫, 山本 博章, 岡本 正行 : “高速 K-means 法を用いたカラー画像の色量子化”, 電子情報通信学会論文誌 D-II, No.7, pp.1120–1128, Jul. 1999.
- [32] R.G.Keys: “Cubic convolution interpolation for digital processing”, IEEE Trans. on Acoustics, Speech and Signal Processing, Vol.ASSP-29, No.6, pp.1153–1160, Jun. 1981.
- [33] W.H.Chen, C.H.Smith, S.C.Fralick: “A fast computational algorithm for the discrete cosine transform”, IEEE Trans. on Communication, Vol.25, No.9, pp.1004–1009, Sep. 1977.
- [34] C. Loeffler , A. Ligtenberg , and G. S. Moschytz: “Practical fast 1-D DCT algorithms with 11 multiplications” , IEEE Int. Conf. Acoustics , Speech and Signal Processing , Vol.2 , pp.988–991 , May 1989.
- [35] Y. Yang, N. P. Galatsanos and A. K. Katsagglos: “Projection–based spatially adaptive reconstruction of block–transform compressed image”, IEEE Trans. on Image Processing, Vol.4, pp. 896–908, Jul. 1995.
- [36] Wei Dai and Trac D. Tran: “Regularity–constrained pre– and post–filtering for block DCT–based systems”, IEEE Trans. on Signal Processing, Vol.51, No.10, pp.2568–2581, Oct. 2003.

- [37] Trac D. Tran, Jie Liang and Chengjie Tu: “Lapped transform via time-domain pre- and post-filtering”, IEEE Trans. on Signal Processing, Vol.51, No.6, pp.1557–1571, Jun. 2003.
- [38] Draft ITU-T Recommendation H.263 Version 2, “Video coding for low bit rate communication”, 1998.
- [39] 貴家 仁志: “マルチレート信号処理” 昭晃堂, 1995.
- [40] Tomio Goto, Etsuro Hayahara: “A Study on Image Coding Technique Based on DCT - Wavelet Transform Coding”, International Symposium on Information Theory and Its Applications(ISITA2002), pp.639 - 642, Oct. 2002.

発表論文一覧

査読論文

1. 後藤 富朗, 新海 善弘, 桜井 優, 北村 正: “隣接ブロックを用いた DCT 領域での高解像度化手法の高速化に関する一検討”, 映像情報メディア学会誌, Vol.61, No.9, Sep.2007 に掲載予定.
2. 後藤 富朗, 山崎 達也, 桜井 優, 北村 正: “高圧縮静止画像の高画質化に関する研究”, 電子情報通信学会論文誌 (D), Vol.J90-D, No.7, pp.1679–1681, Jul. 2007.
3. 後藤 富朗, 奥村 剛史, 早原 悦朗: “カラー成分間の関係を利用したカラー画像符号化手法の一検討”, 電気学会論文誌 (C), Vol.124, No.4, pp.947–952, Apr. 2004.
4. 後藤 富朗, 川北 淳, 早原 悦朗: “ウェーブレット – DCT による画像符号化手法に関する一検討”, 電気学会論文誌 (C), Vol.121, No.12, pp.1860–1865, Dec. 2001.
5. 後藤 富朗, 早原 悦朗: “マルチ走査によるウェーブレット変換符号化に関する一検討”, 電子情報通信学会論文誌 (A), Vol.J84-A, No.11, pp.1325–1332, Nov. 2001.

国際学会論文

1. **Tomio Goto**, Yoshihiro Shinkai, Masaru Sakurai, Tadashi Kitamura: “A Study on the Expansion of a Resolution Conversion Method into Rational Scale using Neighboring Blocks’ DCT Coefficients”, *International Symposium on Information Theory and its Applications (ISITA2006)*, pp.836 – 841, Dec. 2006.

第 6 章 発表論文一覧

2. **Tomio Goto**, Naoki Goh, Tadashi Kitamura: “A Method of Reducing Blocky Noise Using Inverse Wavelet Transform For Moving Pictures”, *International Symposium on DSP and Communication Systems (DSPCS2005)*, pp.318 – 322, Dec. 2005.
3. **Tomio Goto**, Toshiaki Hanamura, Takashi Negami, Tadashi Kitamura: “A Study on Resolution Conversion Method using DCT coefficients”, *International Symposium on Information Theory and Its Applications (ISITA2004)*, pp.260 – 265, Oct. 2004.
4. **Tomio Goto**, Manabu Ishikawa, Tadashi Kitamura: “A Method of Reducing Blocky Noise using Inverse Wavelet Transform”, *IASTED International Conference on Signal and Image Processing (SIP2004)*, pp.167 – 172, Aug. 2004.
5. **Tomio Goto**, Takeshi Okumura, Tadashi Kitamura, Etsuro Hayahara: “A Color Image Coding Technique using Correlations between Color Components”, *IASTED International Conference on Visual, Imaging and Image Processing (VIIP2003)*, pp.239 – 243, Sep. 2003.

国内学会発表等

1. 市川 拓摩, 平野 智, 後藤 富朗, 北村 正, 内匠 逸: “環境電磁波観測回路に用いる タイミング同期回路の開発”, 電子情報通信学会 総合大会, A-4-24, pp.135, Mar. 2007.
2. 山崎 達也, 郷 直輝, 後藤 富朗, 桜井 優, 北村 正: “逆ウェーブレット変換を用いた 動画像高画質化に関する一検討”, 情報理論とその応用シンポジウム (SITA2006), 24.1, pp.637 – 640, Nov. 2006.
3. 新海 善弘, 後藤 富朗, 桜井 優, 北村 正: “隣接ブロックを用いた DCT 領域での解 像度変換の高速化”, 情報理論とその応用シンポジウム (SITA2006), 42.3, pp.633

– 636, Nov. 2006.

4. 鬼頭 孝宏, 後藤 富朗, 北村 正, 内匠 逸: “ケプストラム分析による大域・局所モデルを用いた異常環境電磁波の検出に関する一検討”, 情報理論とその応用シンポジウム (SITA2006), 24.1, pp.363 – 366, Nov. 2006.
5. 山崎 達也, 郷 直輝, 後藤 富朗, 桜井 優, 北村 正: “静止画像の高画質化に関する研究”, 画像符号化シンポジウム (PCSJ2006), p-5.07, pp.99 – 100, Nov. 2006.
6. 新海 善弘, 後藤 富朗, 桜井 優, 北村 正: “DCT 領域での解像度変換の高画質化に関する検討”, 画像符号化シンポジウム (PCSJ2006), p-2.13, pp.59 – 60, Nov. 2006.
7. 豊倉 行崇, 南角 吉彦, 後藤 富朗, 徳田 恵一, 北村 正: “基本動作 HMM に基づく手話単語認識手法”, 電子情報通信学会 ソサイエティ大会, A-4-5, pp.72, Sep. 2006.
8. 鬼頭 孝宏, 後藤 富朗, 平野 智, 北村 正, 内匠 逸: “ケプストラム分析を用いた異常環境電磁波の検出”, 電子情報通信学会 ソサイエティ大会, A-4-16, pp.83, Sep. 2006.
9. 後藤 富朗, 山崎 達也, 桜井 優, 北村 正: “ウェーブレット変換を用いた動画像のブロックひずみ低減手法”, 第 5 回科学技術フォーラム (FIT2006), J-010, pp.209 – 212, Sep. 2006.
10. 新海 善弘, 花村 敏明, 後藤 富朗, 北村 正: “隣接ブロックを用いた DCT 係数上での解像度変換の有理数倍率への拡張に関する一検討”, 電子情報通信学会 総合大会, A-4-6, pp.87, Mar. 2006.
11. 郷 直輝, 石川 学, 後藤 富朗, 北村 正: “JPEG・MPEG におけるブロックひずみ低減手法の一検討”, 情報理論とその応用シンポジウム (SITA2005), pp. 845 – 848, Nov. 2005.
12. 山本 貢嗣, 野村 健, 南角 吉彦, 後藤 富朗, 北村 正: “HMM に基づく日本手話認識のための特徴の統合に関する検討”, 電子情報通信学会 ソサイエティ大会, A-4-28,

第 6 章 発表論文一覧

Sep. 2005.

13. 郷 直輝, 石川 学, 後藤 富朗, 北村 正: “動画像におけるブロックひずみ低減手法の一検討”, 電子情報通信学会 ソサイエティ大会, A-4-45, Sep. 2005.
14. 新海 善弘, 後藤 富朗, 北村 正: “隣接ブロックを用いた動画像の解像度変換に関する一検討”, 電気関係学会 東海支部連合大会, O-345, Sep. 2005.
15. 土井 勇人, 浅野 雄介, 後藤 富朗, 平野 智, 北村 正, 内匠 逸: “環境電磁波観測回路に用いるクロック同期回路の開発”, 電気関係学会 東海支部連合大会, O-430, Sep. 2005.
16. 後藤 富朗, 杉本 和彦, 花村 敏明, 北村 正: “DCT 係数上での動画像解像度変換の高画質化に関する研究”, 第 4 回情報科学技術フォーラム (FIT2005), J-063, Sep. 2005.
17. 杉本 和彦, 花村 敏明, 後藤 富朗, 北村 正: “動画像の DCT 係数上での解像度変換”, 電子情報通信学会 総合大会, A-4-6, Mar. 2005.
18. 郷 直輝, 石川 学, 後藤 富朗, 北村 正: “逆ウェーブレット変換を用いた DCT 係数復号化手法”, 電子情報通信学会 総合大会, A-4-7, Mar. 2005.
19. 野村 健, 後藤 富朗, 徳田 恵一, 北村 正: “アクティブ画像探索法を用いた HMM ジェスチャー認識の特徴量抽出の検討”, 電子情報通信学会 総合大会, D-12-29, Mar. 2005.
20. 山本 貢嗣, 野村 健, 南角 吉彦, 後藤 富朗, 北村 正: “日本手話認識に用いる特徴量の検討”, 電子情報通信学会 総合大会, D-12-121, Mar. 2005.
21. 郷 直輝, 石川 学, 後藤 富朗, 北村 正: “逆ウェーブレット変換によるブロックひずみ低減手法”, 情報理論とその応用シンポジウム (SITA2004), pp.233 – 236, Dec. 2004.
22. 花村 敏明, 根上 崇, 後藤 富朗, 北村 正: “隣接ブロックを用いた D C T 係数上での解像度変換”, 情報理論とその応用シンポジウム (SITA2004), pp.229 – 232, Dec.

2004.

23. 郷 直輝, 後藤 富朗, 北村 正, 石川 学: “Wavelet 係数への変換を用いたブロックひずみ低減手法”, 電子情報通信学会 ソサイエティ大会, A-4-8, Sep. 2004.
24. 花村 敏明, 後藤 富朗, 北村 正: “3 次畳み込み補間法を用いた DCT 係数上での解像度変換”, 電子情報通信学会 ソサイエティ大会, A-4-13, Sep. 2004.
25. 野村 健, 柳 哲, 後藤 富朗, 徳田 恵一, 北村 正: “HMM ジェスチャー認識における MDL を用いた最適状態数の検討”, 電子情報通信学会 ソサイエティ大会, A-4-34, Sep. 2004.
26. 石川 学, 後藤 富朗, 北村 正: “逆ウェーブレット変換を用いたブロックひずみの軽減法”, 電子情報通信学会 総合大会, A-4-14, pp.105, Mar. 2004.
27. 花村 敏明, 根上 崇, 後藤 富朗, 北村 正: “隣接ブロックを用いた DCT 係数上での解像度変換”, 電子情報通信学会 総合大会, A-4-16, pp.107, Mar. 2004.
28. 野村 健, 柳 哲, 後藤 富朗, 徳田 恵一, 北村 正: “HMM ジェスチャー認識における MDL を用いた最適状態数の検討”, 電子情報通信学会 総合大会, D-12-117, pp.283, Mar. 2004.
29. 花村 敏明, 根上 崇, 後藤 富朗, 北村 正: “DCT 係数上での解像度変換法に関する研究”, 情報理論とその応用シンポジウム (SITA2003), pp.229 – 232, Dec. 2003.
30. 高田 佳明, 奥村 剛史, 後藤 富朗, 北村 正: “色変換によるカラー画像の符号化効率改善に関する一検討”, 情報理論とその応用シンポジウム (SITA2003), pp.233 – 236, Dec. 2003.
31. 石川 学, 後藤 富朗, 北村 正: “ウェーブレット・直交変換符号化の一検討”, 電子情報通信学会 ソサイエティ大会, A-4-20, pp.81, Sep. 2003.
32. Hoang Khang Nguyen Vu, 後藤 富朗, 北村 正: “周波数成分を考慮した Wavelet-DCT 符号化”, 電子情報通信学会 ソサイエティ大会, A-4-19, pp.80, Sep. 2003.
33. 石川 学, 後藤 富朗, 早原 悦朗: “Wavelet – DCT を用いた動画画像の再現性に関する

第 6 章 発表論文一覧

- る一検討”, 電気関係学会 東海支部連合大会, pp.304, Sep. 2002.
34. Hoang Khang Nguyen Vu, 後藤 富朗, 早原 悦朗: “整数型ウェーブレット DCT 符号化に関する検討”, 電子情報通信学会 ソサイエティ大会, A-4-24, pp.86, Sep. 2002.
35. 森川 純次, 後藤 富朗, 早原 悦朗: “視覚的特性を考慮したブロックひずみの一般化に関する研究”, 電子情報通信学会 ソサイエティ大会, A-4-25, pp.87, Sep. 2002.
36. 奥村 剛史, 後藤 富朗, 早原 悦朗: “カラー画像の成分間の関係を考慮した符号化方法”, 電子情報通信学会 ソサイエティ大会, A-4-26, pp.88, Sep. 2002.
37. 後藤 富朗, 平野 智, 早原 悦朗: “マルチ走査によるウェーブレット変換符号化に関する一検討”, 情報理論とその応用シンポジウム (SITA2001), pp.187 – 190, Oct. 2001.
38. 早原 悦朗, 平野 智, 後藤 富朗, 奥村 剛史: “カラー画像の成分間の相関を考慮したウェーブレット変換符号化”, 電子情報通信学会 情報・システムソサイエティ, D-11-2, pp.87, Sep. 2001.
39. 川北 淳, 後藤 富朗, 平野 智, 早原 悦朗: “DCT を組み込んだウェーブレット変換の一構成”, 電子情報通信学会 基礎・境界ソサイエティ, A-4-46, Sep. 2001.
40. 後藤 富朗, 平野 智, 早原 悦朗: “斜め走査によるウェーブレット変換符号化に関する一検討”, 情報理論とその応用シンポジウム (SITA2000), pp.635 – 638, Oct. 2000.
41. 赤尾 嘉紀, 後藤 富朗, 平野 智, 早原 悦朗: “ジグザグ走査を用いたウェーブレット変換符号化手法”, 電子情報通信学会 基礎・境界ソサイエティ, A-4-31, pp.122, Sep. 2000.
42. 川北 淳, 後藤 富朗, 平野 智, 早原 悦朗: “DCT を組み込んだウェーブレット変換”, 電気関係学会 東海支部連合大会, 444, pp.222, Sep. 2000.