

## エッジを保った画像圧縮法——GBTC法との比較

服部 進\*、関 章良\*、井上 徳仁†

### Edge Preserving Image Compression——Comparison with the GBTC Method

Susumu HATTORI, Akiyoshi SEKI and Norihito INOUE

#### ABSTRACT

This paper proposes an image compression method suitable for digital image-based plotters. This plotter is an image processing system designed for aerial photogrammetry for map production. Since data volume is quite large, efficient data compression techniques are important. In aerial photogrammetry different pixel sizes are required in orientation of images and plotting object contours. In this aspect pyramidal image compression, including the GBTC (Generalized Block Truncation Coding) algorithm, looks most appropriate. But the GBTC algorithm does not preserve edges of images, and loses significant information when plotting object contours. The authors propose a method to store positions of edges instead of quantization levels in GBTC. The experiments illustrates the quality comparison of decompressed images with the GBTC and the authors' method. The result shows the latter is superior both in precision of identifying point pairs in stereo images and the compression rate.

Keywords: Image Compression, Edges, LoG filter, Precision, GBTC Algorithm, Digital Image-Based Plotter, Digital Photogrammetry, Measurement

#### 1. はじめに

計算機の発達によって、従来精密機械で作られていた図化機を汎用の画像処理機で作ることができるようになってきた。著者らは現在デジタル画像をベースにした図化機（以下デジタル図化機と呼ぶ）「ごくう」を開発中であり<sup>1),2)</sup>、とくに空中写真測量（地形図作成）用に設計している。デジタル図化機の実用化に際しては、膨大な画像データの格納と検索および表示の問題がある。たとえば実用的には1つの図化プロジェクトでは100枚以上の空中写真を用いることは珍しくないので、大容量の記憶装置が比較的安価に市販されるようになった現在でも、画像データを能率よく圧縮する方法が望まれている。静止画像の圧縮法としては多くの方法が提案されているが<sup>2),3)</sup>、現在JPEG委員会が提唱した方式が一般的になっている<sup>3)</sup>。この方式は離散コサイン変換とハフマ

ン符号化を組み合わせたものであるが、高周波域をカットするため、計測を目的にするデジタル図化機には必ずしも向いていない。

画像の標定点を計測する段階では小さい画素のデータが必要である。たとえば空中写真の銀粒子の大きさから期待できる画像座標の限界測定精度は $3\mu\text{m}$ 程度である。デジタル画像を拡大すれば、 $1/3$ 画素程度の計測精度があると期待できる<sup>4)</sup>ので、この精度を得るためには $8\mu\text{m}$ の画素幅が必要になる。1枚の空中写真画像（ $23\text{cm}\times 23\text{cm}$ ）をデジタル化すると、 $8\mu\text{m}\times 8\mu\text{m}$ の画素では1GBのデータ量になる。しかし図化のときにはこのような高い精度の計測を必要とする場合は限られる。標準的な場合では、写真縮尺の5倍程度の縮尺の地図データを作成することが多い。このときには写真上 $20\mu\text{m}$ 程度の測定精度を想定している。なおこれはかつてのアナ

\*情報処理工学科、†工学研究科情報処理工学専攻

ログの1級関化機の精度と同じである<sup>7)</sup>。

このため標定は小さな画素の画像でおこない、関化を行うときには大きな画素の画像を使うようにすればデータ量を節約できる。しかしときには高い精度の座標測定が必要になるので、必要に応じて画像の解像力を復元できるような(すなわち多重解像度の)データ構造で圧縮しておくのが便利である。多重解像度の画像圧縮法としては、ピラミッド符号化<sup>10)</sup>がもっとも自然である。ピラミッド符号化は粗い画素から細かい画素へ階層的にブロックに分割し、対応する画素値の差分を順次記憶する方法である。しかし単純なピラミッド符号化は画像分割のブロックの大きさを各段階で一定にとるので、画素値の分布特性を考慮しない。このため可逆的な符号化を考えるとデータの量を減らすことができない。ブロックの大きさを可変にするピラミッド圧縮法<sup>9),11)</sup>は、きわめて圧縮率の高い方法にも拘らず画質の劣化が少ない方法であることが知られていて、多くの論文がでている。ただし任意の部分を読み出して計測するデジタル関化機では、ブロックの大きさを可変にすると適用が複雑になる。

この点で2段階のピラミッド構造を維持し、かつ非可逆的な圧縮法でありながら画質の劣化が小さい方法としてGBTC (Generalized Block Truncation Coding) 法が注目できる<sup>9),11),12)</sup>。この方法は原画像を  $m \times m$  画素(通常は  $m=4$ ) のブロックに分割し、そのブロック内の画素値の最大値と最小値との差(最大較差)を調べ、較差の大きさに応じて、画素値の平均値からの残差を3段階の符号化モードに分類して、量子化したレベルを圧縮していく方法である。GBTC法は、少ない符号量で画像全体の階調性を保ったまま符号化できると言われている。しかし著者らの経験ではGBTC法での復元画像ではエッジ部分の劣化が比較的大きく、精密座標計測にはやや問題が生じることが分かった。

デジタル関化機では画像は座標計測の媒体にすぎないので、画質の劣化は座標測定の精度を落とさなければあまり問題にならない。地形図を作成するときには建築物、道路、水路などを画像から描画するので、エッジの位置情報をもっとも重要である。そこでこのGBTC法での画素値の量子化レベルの代わりにエッジの情報を格納する方法が考えられる。

この論文ではGBTC法で画像を圧縮したとき、エッジを保存したとき(以下便宜上エッジ保存法と呼ぶ)の圧縮率、および復元画像の性質を調べる。エッジの情報としてLoG (Laplacian of Gaussian) フィルタの出力のゼロ交差を用いた。エッジ保存法はGBTC法より圧縮率は高く、かつ復元画像と元の画像とで位置計測の精度が変わらないことを実例で示す。

## 2. GBTC法によるデータ圧縮と復元画像の画質

### 2.1 GBTC法による画像データの圧縮

図1<sup>9)</sup>に示すように画像を  $m \times m$  画素のブロックに分

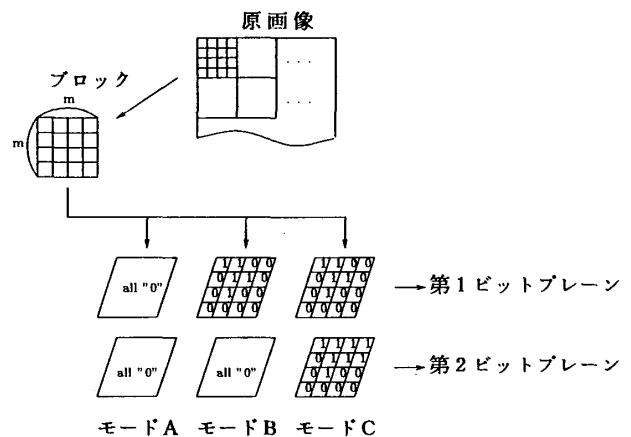


図1 ブロックごとの3つの量子化モード

Fig.1: Thee Quantization Modes in a Block

割し、ブロックごとにより少ない階調に量子化する。 $m$ の値は通常4が使われるが、座標計測には縮小率が大きすぎるように思われたので、 $m=4$ 以外に、 $m=2$ の場合を取り扱った。ブロック内の最大較差 ( $DR = |\text{ブロック内最大値} - \text{ブロック内最小値}| = |L_{\max} - L_{\min}|$ ) を計算し、適当に設定した符号化パラメータ  $T_1, T_2$  ( $T_1 < T_2$ ) と比較して以下の3つの符号化モードに分類する。符号化はモードとモード内での量子化レベルを記憶することでなされる。画素値を  $D$ 、ブロック内の平均値を  $L_a$ 、適当に設定する量子化のしきい値を  $L_d$ 、符号化した量子化レベルを  $Q$  とする。

#### ●符号化の手順(図1参照)

- モードA: 最大較差が非常に小さい場合 ( $D_R \leq T_1$ )、各画素を1レベルに量子化する ( $Q=00$ )。
- モードB: 最大較差が  $T_1 \leq D_R \leq T_2$  のとき、各画素を2レベルに量子化する。 ( $D \leq L_a$  なら  $Q=01$ ,  $D \geq L_a$  なら  $Q=00$ )。
- モードC: 最大較差が非常に大きい場合 ( $T_2 \leq D_R$ )、各画素をブロック内の階調分布の範囲内で4レベルに量子化する ( $D \leq L_a - 1/2L_d$  なら  $Q=11$ ,  $L_a - 1/2L_d \leq x \leq 0$  なら  $Q=10$ ,  $0 \leq D \leq L_a + L_d/2$  なら  $Q=01$ ,  $D \geq L_a + 1/2L_d$  なら  $Q=00$ )。

各モードともブロック内の平均画素値を標準8ビットで保存し、モードを2ビット、さらに各モードについて量子化レベル  $Q$  の第1ビットと第2ビットの各プレーンを保存する。

次に復号は次のようになる。復号した画素値を  $p$  で表すと、量子化したときのしきい値の間をとって次のよう

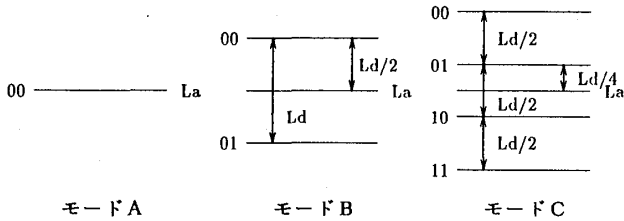
になる。

●復号化の手順 (図2<sup>9),12</sup>参照)

モードA :  $p=La$

モードB :  $Q=01$ のとき  $p=La-1/2Ld$ ,  
 $Q=00$ のとき  $p=La+1/2Ld$

モードC :  $Q=11$ のとき  $p=La-3/4Ld$ ,  
 $Q=10$ のとき  $p=La-1/4Ld$ ,  
 $Q=01$ のとき  $p=La+1/4Ld$ ,  
 $Q=00$ のとき  $p=La+3/4Ld$



La:ブロック内平均値    Ld:レベル差

図2 3つのモードでの復号化

Fig.2 : Decompression of Data for the Three Modes

ここではGBTC法は実験的に行ったので、符号化パラメータ  $T_1$ ,  $T_2$  と量子化のしきい値  $Ld$  を一定にとっている。しかし本来はこれらの値もブロックごとに換え、さらにしきい値を画素値の分布に合わせて非線形に変えるのが画質の保全には適切である。この論文では符号化パラメータ、量子化のしきい値は、画素値の標準偏差  $\sigma$  をあらかじめ求め、実験的に  $T_1=1/4\sigma$ ,  $T_2=1/2\sigma$ ,  $Ld=1/2\sigma$  とした。

3つのモードを表すには2ビット必要であるから、元の画像が8ビットの奥行きであるとする、それぞれのモードに必要なビット数は表1のようである。モードB、

表1 GBTC法での1ブロックあたりの必要ビット

Table.1 : Numbers of Bits Necessary for Coding One Block in the GBTC

|      | m = 2     | m = 4    |
|------|-----------|----------|
| モードA | 10 bit    | 10 bit   |
| モードB | 10+16 bit | 10+4 bit |
| モードC | 10+32 bit | 10+8 bit |

モードCでは偏差を示すビットプレーンがそれぞれ1枚および2枚必要になる。任意の画像部分を読みとるには固定長のデータが望ましいが、GBTC法では可変長になる。一般にビット単位の操作性は悪いので、実際には

適当にバイト単位で処理するようデータをブロック化する。場合によっては使用しないビットが出現して圧縮率が悪くなる。以降ではビット単位で圧縮するものとして話を進める。

もし3つのモードの出現率が同じであれば、ビット単位で圧縮したときのデータの圧縮率は次のようになる。

$$m=4 : (10+26+42)/3 / (16 \times 8) = 0.20$$

$$m=2 : (10+14+18)/3 / (4 \times 8) = 0.44$$

すなわちビット単位の処理が必要になるにも拘らず、 $m=2$  では半分強の圧縮率しか期待できない。

2.2 GBTC法での復元画像の画質と圧縮率

実験で使用した画像データは、図3に示す512×480画

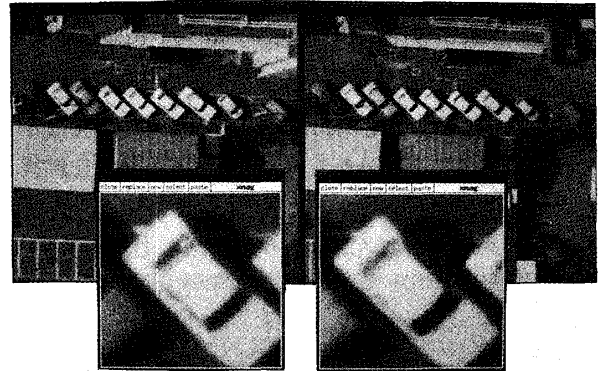


図3 実験に使った原画像

Fig.3 : Original Image Pair

素(奥行き8ビット)のステレオ空中写真画像の一部である。画素幅は25μm×25μm、測定濃度範囲は8ビット(0-3D)である。図4(a)は4×4画素の平均値で圧縮・復号したもの(すなわちもとの画像の4×4画素の平均値を4×4画素に与えて復元した画像)、図4(b)は2×2画素の平均値で圧縮・復号したものである。図

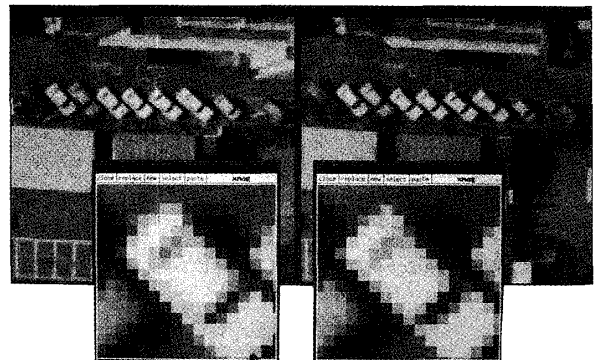


図4(a) m = 4

Fig.4(a) m = 4

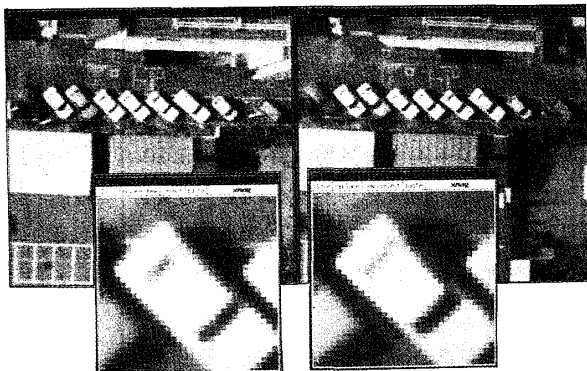


図4 (b) m=2  
Fig.4(b) m=2

図4 m×m画素の平均値をm×m画素に与えて復号した画像

Fig.4 : Decompressed Images, m×m Pixel Values of which are Replaced by Their Average

4 (a) は立体視しても地物の境界を押さえることはかなり困難であることが分かる。図5はGBTC法で圧縮後上記の方法で復号したである。図5 (a) はm=4、図5(b) はm=2の場合である。図5から判断して、図5 (a) は

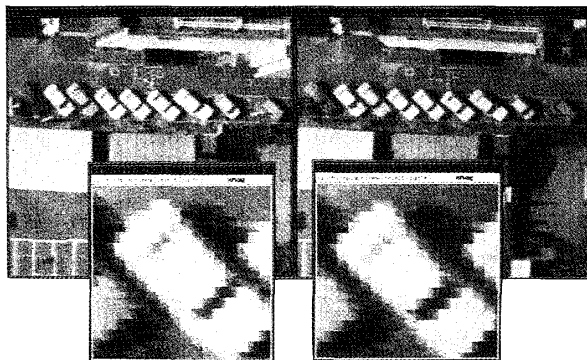


図5 (a) m=4  
Fig.5(a) m=4

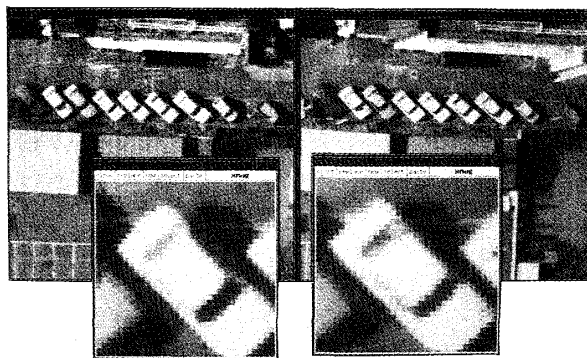


図5 (b) m=2  
Fig.5(b) m=2

図4 GBTC法で圧縮後復号した画像  
Fig.5 : Decompressed Images after Compression with GBTC

図4(a) よりはましであるものの劣化が激しく計測の実用に耐えないように思われる。図5 (b) の画質は比較的良好であるが、やはりエッジ部分は元の画像と比べて不明瞭になっている。これはGBTC法では本質的に画像のエッジを厳密に保持できないからである。

もしビットパターンの出現確率が異なれば Huffman 符号化やランレングス符号化を使って信号を圧縮できる。ビットパターンの発生頻度の分布を知るため、量子化レベルQの2枚のビットプレーンのビット模様の分布パターンを調べた。表2 (a) は左画像についてビットパターンの分布を示したものである。m=4の場合、図6の様に

|   |   |   |   |
|---|---|---|---|
| a | b | a | b |
| c | d | c | d |
| a | b | a | b |
| c | d | c | d |

図6 4×4画素ブロックでのビットパターンの番号付け

Fig.6 : Ordering of Bit Patterns in a 4×4 pixel Block

表2 GBTC法でのビットパターンの出現頻度  
Table.2 : Frequencies of Incidences of Bit Patterns in the GBTC

(a) m=4

| abcd       | abcd       |
|------------|------------|
| 0000 24.8% | 1000 3.6%  |
| 0001 5.8%  | 1001 2.3%  |
| 0010 1.6%  | 1010 3.2%  |
| 0011 3.8%  | 1011 3.5%  |
| 0100 3.4%  | 1100 3.6%  |
| 0101 4.7%  | 1101 1.8%  |
| 0110 2.7%  | 1110 5.1%  |
| 0111 3.8%  | 1111 26.2% |

(b) m=2

| abcd       | abcd       |
|------------|------------|
| 0000 8.2%  | 1000 4.5%  |
| 0001 3.3%  | 1001 0.2%  |
| 0010 6.6%  | 1010 3.5%  |
| 0011 12.5% | 1011 6.8%  |
| 0100 8.2%  | 1100 18.6% |
| 0101 3.9%  | 1101 6.9%  |
| 0110 0.2%  | 1110 5.1%  |
| 0111 3.5%  | 1111 9.9%  |

ブロックを4つの $2 \times 2$ 単位に分けて、パターンの統計を求めた。これによると0000、1111の両パターンが卓越して25%ずつ発生し、他はほぼ同じであった。このことからHuffman符号化を行えば、25%の出現確率の0000を00、同じく1111を01とし、他を5ビットで符号化することができ、1ブロック16画素を表現するためのビットプレーンの平均長は

$$(0.5 \times 2 + 0.5 \times 5) \times 4 = 14.0(\text{bit})$$

になり、圧縮率は

$$m=4: (8+2+14.0)/(8 \times 16) = 0.19$$

となり少しだけ改善できる。また $m=2$ についても同様にビットパターンの分布を調べると表2(b)のようになった。0011、1100のパターンが約15%ずつで、残りのパターンの出現率に有意の差があるとは思えないので。前者に2ビット、後者に5ビットを与え、Huffman符号化を行くと4ビットの平均長は

$$(0.3 \times 2 + 0.7 \times 5) = 4.1(\text{bit})$$

で、圧縮率は

$$m=2: (8+2+4.6)/(8 \times 4) = 0.440$$

となり、この場合は変わらない。

### 2.3 エッジ保存法の画像の圧縮

エッジの抽出法はいくつか提案されているが、ここではLoG (Laplacian of Gaussian) フィルタ<sup>3)</sup>のゼロ交差を使う。LoG フィルタは画像をガウス関数で低域フィルタリングした後、ラプラシアンを施して高域フィルタリングしたもので一種の帯域通過フィルタである。そのゼロ交差は雑音を抑えて、安定したエッジを出力することが知られている。LoG フィルタの応答関数は次式で定義される。

$$L(x, y) = \left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) e^{-\frac{x^2+y^2}{2s^2}} \quad (1)$$

ここで $x, y$ は画像座標、 $s$ はガウス関数のスケールで、 $s$ が大きいと出力されるエッジのスケールも大きくなり、その結果エッジのコーナーが丸くなる。これを避けるた

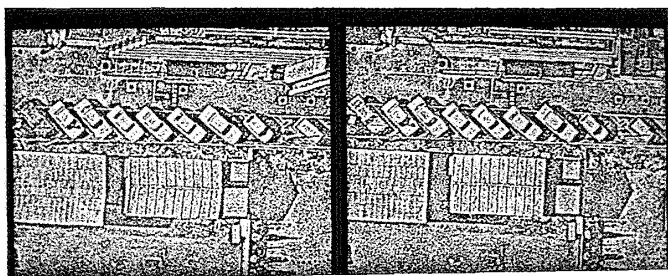


図7 エッジの画像 ( $s=1$ )  
Fig.7: Extracted Edges ( $s=1$ )

めここでは $s=1$ とした。図7は $s=1$ のときに出力された左右画像のエッジの図である。 $s$ が小さいため、微少なエッジが無数に現れる。フィルタリングされた画像の画素値を $D(x, y)$ で表すとき、エッジの出力の手順は次のようである。

#### ●エッジの出力と符号の定義

LoGフィルタを施し、出力の画像の各画素値 $D(x, y)$ で

$D(x, y) > 0$ のとき

$D(x+1, y) < 0$ のとき  $D(x+1, y)$  は-のエッジ

$D(x-1, y) < 0$ のとき  $D(x-1, y)$  は+のエッジ

$D(x, y) < 0$ のとき

$D(x+1, y) > 0$ のとき  $D(x, y)$  は+のエッジ

$D(x-1, y) > 0$ のとき  $D(x, y)$  は-のエッジ

すなわち $x$ または $y$ 方向でゼロ交差の起こったエッジの一侧の画素をエッジとしている。図8(a)、図8(b)は

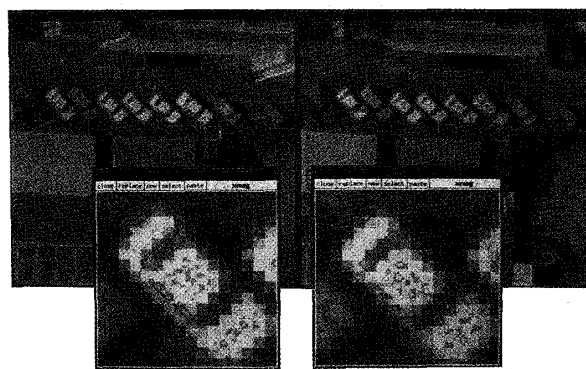


図8(a)  $m=4$   
Fig.8(a):  $m=4$

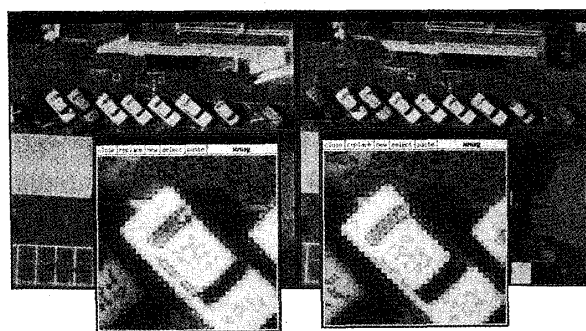


図8(b)  $m=2$   
Fig.8(b):  $m=2$

図8 図4にエッジ画像(図7)を重ねて表示した画像  
Fig.8 Images in Fig.4 Overlapped with the Edge Images (Fig.7)

図4(a)、図4(b)の粗い画像に図7を重ねて表示した画像である。図8ではエッジを強調して表示するため、エッジの位置でもともとの画素値 $I(x, y)$ を $I(x, y) - \sigma$ として表示した。 $\sigma$ は画像全体の標準偏差である。

視覚的に評価すると $m=4$ の画像(図8(a))はエッ

ジ位置と背後のぼけた画像との一致が悪く、対応エッジを押さえるのに時間がかかる。 $m=2$ の画像(図8(b))では背後の画像とエッジの画像は違和感なくよく一致していて、1画素の精度で位置ぎめするのは容易であることが分かる。

エッジ保存法では、 $m=4$ 、 $m=2$ のブロックで平均値と共にエッジの有無を0,1で表したビットプレーンを記憶する。そのときの圧縮率はつぎのようである。

$$m=4 : (8+2+16)/(8 \times 16) = 0.20$$

$$m=2 : (8+2+4)/(8 \times 4) = 0.44$$

したがって、ブロックの大きさが一定であるにも拘らず、GBTC法に比べて圧縮率は改善されている。さらに表2と同様に左画像 $2 \times 2$ 画素ごとのビットパターンの出現確率を調べたところ表3のようになった。表3によれ

表3 エッジ保存法でのビットパターンの出現頻度

Table.3: Frequencies of Incidences of Bit Patterns in the Edge Preserving Coding

|      |        |      |       |
|------|--------|------|-------|
| abcd |        | abcd |       |
| 0000 | 83.4 % | 1000 | 1.2 % |
| 0001 | 1.2 %  | 1001 | 1.4 % |
| 0010 | 1.3 %  | 1010 | 1.2 % |
| 0011 | 2.3 %  | 1011 | 0.5 % |
| 0100 | 1.2 %  | 1100 | 2.1 % |
| 0101 | 1.2 %  | 1101 | 0.4 % |
| 0110 | 1.3 %  | 1110 | 0.4 % |
| 0111 | 0.4 %  | 1111 | 0.5 % |

ば、エッジのない画素が大部分で残りは画像の特性に依存する。この例では縦と横のエッジがたまたま多いので多少そのエッジ(0011, 1100など)の比率が高い。この0000が卓越する性質を利用すれば、大きく信号の圧縮ができる。Huffman符号化を行うと80%の出現確率の0000を0、他を5ビットで符号化すれば、1ブロック16画素を表現するためのビットプレーンの平均長は

$$(0.8 \times 1 + 0.2 \times 5) \times 4 = 7.2(\text{bit})$$

になり、約1バイトでビットプレーンを表現でき、圧縮率

$$m=4 : (8+2+7.2)/(8 \times 16) = 0.13$$

$$m=2 : (8+2+7.2/4)/(8 \times 4) = 0.36$$

のデータ圧縮が実現できる。この圧縮率はGBTC法よりも優れている。ただしこの方式では信号長が可変になるので任意の位置の画像を複号するには手続きが複雑になる。

### 3. 結 論

この論文ではデジタル図化機に適した画像の圧縮法として、エッジを保った圧縮法を提案した。 $2 \times 2$ 画素

ごとの平均値とLoGフィルタで抽出したエッジ情報を格納する形でデータ圧縮を実現した。デジタル図化機ではピラミッド型のデータ構造が便利であるが、著者らの方法はこの条件を満たしている。ピラミッド圧縮の代表であるGBTC法と復元画像の画質および圧縮率を比較した結果、著者らの方法はどちらの点でもGBTC法をやや上回る結果を得た。

### 参考文献

- 1) 八木俊之、服部進、関章良：デジタル画像ベースの図化機「ごくう」の図化機能、日本写真測量学会講演論文集、pp.115-118(1995)。
- 2) 服部進、関章良、行司菅男、八木俊之、et. al. : デジタル画像ベースの図化機「ごくう」の空中写真測量のための機能設計、福山大学工学部紀要、Vol.18, No.1, pp.39-52(1994)。
- 3) 二宮佑一：画像の帯域圧縮と符号化技術、日刊工業新聞、pp.51-90(1994)
- 4) 原島博：画像情報圧縮、オーム社、pp.191-200(1991)。
- 5) Leger Alain, J. L. Mitchell and Y. Yamazaki : Still Picture Compression Algorithms Evaluated for International Standardisation, Conf. Record on IEEE Global Telecommunication, No.2, pp.1028-1032(1988)。
- 6) 行司菅男、服部進、関章良：デジタル画像ベースの図化機のためのパスポイントの半自動計測法、日本写真測量学会講演論文集、pp.99-102(1995)。
- 7) 木本氏寿：写真測量の実際、山海道、pp.148-165(1970)。
- 8) 中川裕志、羽村秀樹：階層型ブロック符号化による濃淡静止画像の帯域圧縮符号化、電子情報通信学会技術研究報告、IE-82-3, Vol.82, No.15, pp.17-24(1982)。
- 9) 越智宏、小林誠、茨木久：濃淡画像の拡張型差分適応ブロック符号化方式、電子情報通信学会論文誌、pp.337-347(1990)。
- 10) 安居院猛、中嶋政之：画像情報処理、森北出版、pp.136-139(1991)。
- 11) 井上徳仁：デジタル画像ベースの図化機に適した画像の圧縮法の研究 - GBTC法の拡張 -、福山大学工学部情報処理工学科卒業論文(1995)。
- 12) 安居院猛、中嶋政之：画像情報処理、森北出版、pp.104-105(1991)。
- 13) Marr, D. and E. Hildreth: Theory of Edge Detection. Proc. of the Royal Society of London, Vol. B-207, pp.187-217(1980)。