

# フラクタル画像符号化のスケーリングパラメータ計算量削減 Fast Calculation of Scaling Parameters for Fractal Image Coding

原田 雅樹<sup>†</sup> 木本 伊彦<sup>††</sup> 藤井 俊彰<sup>††</sup> 谷本 正幸<sup>††</sup>

Masaki HARADA<sup>†</sup>, Tadahiko KIMOTO<sup>††</sup>, Toshiaki FUJII<sup>††</sup>, and Masayuki TANIMOTO<sup>††</sup>

<sup>†</sup> (株)SBS 情報システム, 静岡市

<sup>††</sup> 名古屋大学大学院工学研究科電子情報学専攻, 名古屋市

<sup>†</sup> SBS Information System, Shizuoka, 422-8033, Japan

<sup>††</sup> Graduate School of Engineering, Nagoya University, Naoya, 464-8603, Japan

E-mail: m.harada@sys.sbs-np.co.jp, kimoto@nuee.nagoya-u.c.jp, fujii@nuee.nagoya-u.c.jp, and  
tanimoto@nuee.nagoya-u.ac.jp

**あらまし** フラクタル画像符号化の高速化を実現するために、以下の二つの手法を提案する。第一の手法では、スケーリングパラメータの計算回数を削減させるために、一つのレンジブロック内の最大振幅と一つのドメインブロック内の最大振幅との比（以下ここではこれを最大振幅比と呼ぶ）を用いる。符号として選択されるドメインブロックにおける最大振幅比はスケーリングパラメータ付近の値をとるため、ある閾値を超える大きな最大振幅比を持つ場合にはスケーリングパラメータの計算を行わない。第二の手法では、スケーリングパラメータそのものの計算量を削減させるために、一つのレンジブロック内の輝度偏差と一つのドメインブロック内の輝度偏差との比（以下ここではこれを偏差比と呼ぶ）を用いる。偏差とスケーリングパラメータとの間には相関があるため、計算量の少ない偏差比に定数をかけてスケーリングパラメータを近似的に求める。

**Abstract** We propose two schemes to reduce the calculation time of Fractal Image Coding. The first one reduces calculation time of parameters, affine transform and rmse by using the maximum amplitude ratio which is a ratio between the maximum amplitude ratio (MAR) of range block and that of domain block. By using the MAR, domain block which does not seem to choose is excluded before calculating parameters. The second one reduces calculation time of scaling parameter by using the ratio between deviation (DR) of range block and that of domain block. The DR is used instead of the scaling parameter.

**キーワード** 画像符号化, フラクタル, 高速化, 最大振幅比, 偏差比

**Key words** image coding, fractal, maximum amplicude ratio, deviation ratio

## 1. はじめに

近年, WWW(World Wide Web)の普及や携帯電話等の発達に伴い, 動画像・静止画像・音声・テキストなどの各種データを統合的に取り扱うマルチメディアシステムの研究開発が各方面で活発に行われている [1]. このような各種データの中でも特に画像データは, ユーザに与える視覚効果という点で最も重要な情報源の一つである. しかし画像データはテキストや音声に比べて情報量が膨大であるため, 効率良く蓄積, 伝送するための圧縮技術の導入が不可欠である. そのため従来より, 種々の圧縮方式が提案・研究されてきた [2].

画像符号化技術は, 静止画像用の JPEG (Joint Photographic Experts Group), テレビ会議用の H.261 あるいは H.263, 動画像蓄積用の MPEG1 (Moving Picture Experts Group1), 動画像の放送メディア用の MPEG2, オブジェクト指向の MPEG4 [3], および静止画像の新しい規格である JPEG2000 と標準化が進んでいる. これら画像符号化の基本技術は, 変換, 量子化, エントロピー符号化および動き補償に分類することができる [4]. 変換では, 画像信号を分布に著しい偏りを持ったデータに変えることでエントロピーを減少させる. 量子化では, 変換後の連続的に変化するデータを離散量で表す. エントロピー符号化では, 量子化されたデータを最も短い平均符号長で符号化し, 実際の符号を得る. 動き補償は, 動画像のフレーム間において画像中の動きを検出し, その動き分を補償してからフレーム間の予測を行う.

画像符号化の基本技術のうち, 画像の性質に強く依存している変換については DPCM (Differential Pulse Code Modulation), 離散コサイン変換 (DCT: Discrete Cosine Transform) が現在の主流である. DPCM, DCT はいずれも画素間の相関が高いことを利用し, 信号を分布に著しい偏りを持ったデータに変換することでエントロピーを減少させる. このような符号化は波形符号化と呼ばれている. これまでの符号化技術は, 情報理論およびフィルタリング理論に基づく波形符号化が中心であった. しかし, 波形符号化ではさらなる圧縮率の向上は期待できず, 新しい圧縮方式の登場が望まれている. これに対して, 反復関数系として画像の自己相似性に着目したフラクタル画像符号化等全く新しい符号化方式の研究が行われている. フラクタル画像符号化は, 超低レートで JPEG と比べて視覚的に良好な画像を復元でき

る圧縮方式であると言われている. また, 自己相似性を利用しているため, 復号時に画像を拡大・縮小すると解像度に応じた特徴が現れるという興味深い性質を持つ. このため, 圧縮符号化だけでなく, 解像度変換や画像の修復等への応用が期待されている.

Jacquin の方式では, 画像を小さな矩形ブロック (以下これをレンジブロックと呼ぶ) に分割し, レンジブロックよりも大きく, かつこれを最もよく近似するブロック (以下これをドメインブロックと呼ぶ) を画像中から探索する [5]. そしてそのドメインブロックの位置や変換のパラメータを符号とする. 探索処理では, 各ドメインブロックについてスケーリングパラメータを求め, 得られたパラメータ毎に近似誤差を計算する. また, ドメインブロックにはいくつかの幾何変換が施され, 一つの変換毎にスケーリングパラメータと近似誤差の計算を行う. 従って, 計算量が非常に多くなり符号化時間が膨大になるという欠点があり, 実用化への障壁となっている. この問題に対して, 高速化を実現させるための手法がいくつか提案されている [7]~[14]. これらは主に, ブロックの特徴値を用いて一つのドメインブロックに対していくつかある幾何変換のうち一つだけを選択する. これにより, 一つのドメインブロックに対して行われるスケーリングパラメータと近似誤差の計算回数は一回となり, 高速化を実現している. また, 復号に関する高速化手法も提案されている [15], [16]. しかし, 一つの幾何変換だけを選択することでスケーリングパラメータの計算回数は削減されるが, 一つのスケーリングパラメータ計算に要する計算量そのものは変わらない. スケーリングパラメータの計算量は符号化処理全体の中で大きな部分を占めるため, さらなる高速化のためにはスケーリングパラメータの計算量を削減する必要がある.

そこで本稿では, スケーリングパラメータの計算量を削減することを目的として, 以下の二つの高速化手法を提案する. 両手法ともにスケーリングパラメータの計算を行う前に, それよりも少ない計算量で求められる特徴値を計算する. この値を, 第一の手法ではスケーリングパラメータの計算を行うかどうかの判断のために用い, また, 第二の手法ではスケーリングパラメータの近似値として用いて, 計算量を削減する.

第一の手法では, 特徴値として一つのレンジブロック内の最大振幅と一つのドメインブロック内の最大振幅との比 (以下ここではこれを最大振幅比と呼ぶ) を用いる. 最大振幅比の値は, 理想的な近似が得ら

れたときに最小となり、近似誤差が大きくなるにつれて大きくなる。従って、あらかじめ閾値を定めて、これより大きな最大振幅比を持つ場合にはスケーリングパラメータの計算を省く。

第二の手法では、特徴値として一つのレンジブロック内の輝度偏差と一つのドメインブロック内の輝度偏差との比（以下ここではこれを偏差比と呼ぶ）を用いる。偏差とスケーリングパラメータとの間には相関があるため、偏差比に定数をかけてスケーリングパラメータを近似する。

これら二つの手法により、画質の劣化を抑えつつ符号化時間を削減できることを示す。まず第2章で従来法の符号化時間の分析を行う。次に第3章で提案方式について説明し、第4章で提案法の実験結果を示す。

## 2. 符号化時間の分析

はじめに、フラクタル画像符号化の概要を示す。なお本稿では、提案法の効果を明らかにするために固定ブロックサイズの平均値分離方式を用いた。後述する提案法を四分木分割の符号化方式へ適用した時の結果は、参考値として付録に提示する。

まず最初に画像  $F$  を重複しない矩形ブロック  $R_i (1 \leq i \leq N)$  に分割する。このブロックをレンジブロックと呼ぶ。各々のレンジブロック  $R_i$  に対して図1のように、画像中からレンジブロックより大きなブロック（これをドメインブロックと呼ぶ） $D_i$  を選択し、縮小変換  $w_i$  を施す。この操作を繰り返し、レンジブロック  $R_i$  を最も近似するような  $D_i, w_i$  を見つける。すなわち、 $\bigcup_{i=1}^N d(F \cap R_i, w_i(F \cap D_i))$  が最も小さくなるように、 $i = 1 \dots N$  に対して  $w_i$  を見つける。

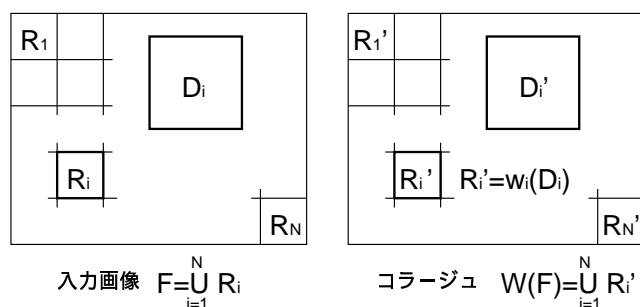


図1 入力画像とコラージュ  
Fig. 1 Input image and collage

縮小変換  $w_i$  には平面上での縮小変換  $S$  の他に、回転、鏡像変換  $I$  (縮小変換も含めて幾何変換と呼ぶ)、輝度方向のアフィン変換  $L$  が含まれる。

- 平面上での縮小変換  $S$

通常、ドメインブロックの一辺の長さはレンジブロックの一辺の長さの2倍に設定される。そのため、縮小変換によってレンジブロックと同じ大きさにするためには縮小率を1/2にする。つまり、下式のよう縦横1/2にダウンサンプリングする。

$$S(D_{i_d+i, j_d+j}) = \{D_{I(i), J(i)} + D_{I(i)+1, J(i)} + D_{I(i), J(i)+1} + D_{I(i)+1, J(i)+1}\} / 4 \quad (1)$$

ただし、 $i_d, j_d$  はドメインブロックの左上の位置、 $I(i) = i_d + 2i$ ,  $J(i) = j_d + 2j$ ,  $D_{x,y}$  はドメインブロック内の位置  $(x, y)$  における輝度値である。なお、以下  $d_{i,j} = S(D_{i_d+i, j_d+j})$  と記述する。

- 回転、鏡像変換  $I$

縮小変換されたドメインブロックに対して、0度回転、90度回転、180度回転、270度回転、0度回転して鏡像変換、90度回転して鏡像変換、180度回転して鏡像変換、270度回転して鏡像変換、以上の8つの変換を行う。すなわち、幾何変換後のドメインブロック内の輝度値を  $d'_{i,j}$  とすると、

$$d'_{i,j} = I(d_{i,j}) \quad (2)$$

である。

- 輝度方向のアフィン変換  $L$

幾何変換をドメインブロックに対して行った後、その変換ドメインブロック内の輝度値  $d'_{i,j}$  に対して以下の変換を行う。

$$L(d'_{i,j}) = \alpha d'_{i,j} + \beta \quad (3)$$

また、 $\alpha$  は以下のように求める。

$$\alpha = \frac{M^2 \sum_{j=0}^{M-1} \sum_{i=0}^{M-1} (r_{i,j} - \mu_r)(d_{i,j} - \mu_d)}{M^2 \sum_{j=0}^{M-1} \sum_{i=0}^{M-1} (d_{i,j} - \mu_d)^2} \quad (4)$$

ただし、 $r_{i,j}$  はレンジブロックの内の位置  $(i, j)$  の輝度値であり、 $d_{i,j}$  は空間縮小後のドメインブロックの各画素の輝度値である。また、 $\mu_r, \mu_d$  はそれぞれレンジブロックとドメインブロックの輝度平均値である。 $M$  はレンジブロックの一辺の長さである。

以上の操作を行い、レンジブロックと変換されたドメインブロックとの誤差が最小となるようにドメインブロックを選択する。なお、コラージュ定理ではハウスドルフ距離を誤差を示す指標としているが、計算処理上、一般的には誤差  $e$  として平均二乗誤差を用いて、誤差が小さくなるようにドメインブロッ

クを選択を行う。ただし本稿では、計算量削減のために下式のように誤差の絶対値の和を計算する。

$$e(\mathbf{r}, \alpha' \mathbf{d}') = \sum_{j=0}^{M-1} \sum_{i=0}^{M-1} |(\mathbf{r}_{i,j} - \mu_r) - \alpha'(\mathbf{d}_{i,j} - \mu_d)| \quad (5)$$

ただし、 $\alpha'$  は  $\alpha$  の量子化値である。

本稿ではレンジブロックの大きさを  $4 \times 4$  画素 ( $M = 4$ ) とし、左上のブロックから順番にパラメータを探索する。探索範囲はレンジブロックを中心とする 256 個のドメインブロックとする。全てのブロックの符号化が終了した後、エントロピー符号化を施す。

以上の符号化処理は次のように分類される。

- 初期化
- パラメータの探索
- エントロピー符号化と符号出力

ただし、初期化には符号化に必要なパラメータの解析および画像の読み込みが含まれる。このうち、パラメータの探索が全体の 99% の処理時間を占め、その他の処理に必要な時間は無視できる程度である。さらに、パラメータ探索の処理内容は以下のように分類される。

- (1) ドメインブロックの切り出しおよび空間縮小。
- (2) ドメインブロックに幾何変換を施す。
- (3) 変換パラメータ  $\alpha$  を求める。
- (4) レンジブロックとドメインブロックとの間の近似誤差を求める。

それぞれの処理時間を表 1 に示す。なお、本論文では CPU として Celeron 466MHz を搭載した Linux マシンを利用した。また、プログラムは C 言語で作成した。なお、コンパイラには gcc を使い、最適化オプションは全て無効とした。パラメータ探索処理のうち、特にスケールパラメータ  $\alpha$  の計算に多くの時間が費やされている。また、幾何変換、近似誤差の計算にも多くの時間が費やされている。従って、スケールパラメータを高速に計算することができれば、符号化にかかる計算時間も削減する。さらに、スケールパラメータの計算を行う前に計算を打ち切ることができれば、その後の近似誤差の計算も行わないため、さらに計算量の削減が見込まれる。

### 3. 提案法

符号化の高速化を実現するために、レンジブロックとドメインブロックとの最大振幅比を用いてスケー

表 1 ブロック符号化関数の処理時間 (Lena, 256, 8bits grayscale)

Table 1 Calculation time to encode blocks (Lena, 256, 8bits grayscale)

演算	割合 [%]
(1) ドメインブロックの切り出し, 縮小	11
(2) 幾何変換	27
(3) $\alpha$ の計算	36
(4) 近似誤差の計算	26

リングパラメータを計算する前に判定を行い、計算を打ち切る手法を提案する。この手法により、スケールリングパラメータ、幾何変換、および近似誤差の計算回数が削減される。また、偏差比を用いてスケールリングパラメータを近似する手法を提案する。この手法により、スケールリングパラメータの計算時間そのものが削減される。

#### 3.1 最大振幅比を用いた計算打ち切り - MAR 法

レンジブロックを  $\mathbf{r}$ 、空間縮小後のドメインブロックを  $\mathbf{d}$  とする。いま、図 2 のようにレンジブロック中およびドメインブロック中の輝度の最大値、最小値をそれぞれ  $r_{max}, r_{min}, d_{max}, d_{min}$  とすると、最大振幅比 (Maximum Amplitude Ratio, 以下ここではこれを  $MAR$  と記し、これを用いる手法を MAR 法と呼ぶ) は以下のように定義される。

$$MAR = \frac{r_{max} - r_{min}}{d_{max} - d_{min}} \quad (6)$$

$MAR = \alpha$  となるのは  $\mathbf{r}$  の交流成分と  $\mathbf{d}$  の交流成分が比例関係にあるときである。このとき、変換パラメータ  $\alpha$  との関係は

$$\mathbf{r}_{i,j} - \mu_r = \alpha(\mathbf{d}_{i,j} - \mu_d) \quad (7)$$

と表される。

図 3 はパラメータ探索の結果  $\alpha = 0.32$  を符号として持つレンジブロックに対する  $MAR$  の分布を示している。ただし、横軸の  $MAR$  を 0.1 ごとに区切り、度数を求めている。また、図中 “All Blocks” は画像中の全てのドメインブロックについて計算した最大振幅比の分布を示しており、“Blocks Selected” はそのうち、実際に選択されたドメインブロックについての最大振幅比の分布を示している。この図から、ある  $\alpha$  に対する  $MAR$  の分布は広がりをもつが、非常に大きな  $MAR$  をもつブロックは最適なブロックとして選択されていないことがわかる。この傾向は他の  $\alpha$  についても同様である。従って、選択される可能性のない部分については  $\alpha$  を計算する必要が

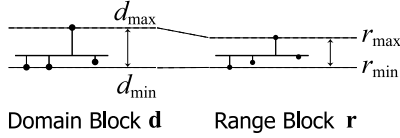


図 2 最大振幅比

Fig. 2 Maximum Amplitude Ratio

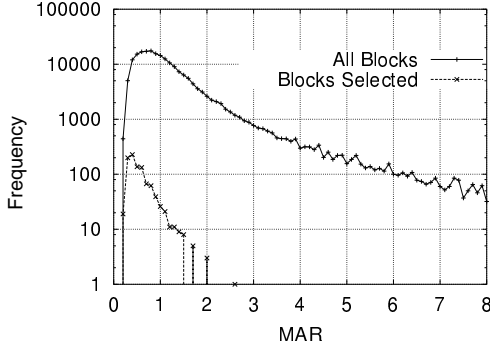

 図 3  $\alpha = 0.32$  となる場合の  $MAR$  の度数分布 (Lena,  $256 \times 256$ )

 Fig. 3 Histogram of  $MAR$  where  $\alpha = 0.32$  (Lena,  $256 \times 256$ )

ない。そこで、あらかじめ閾値を定め、 $MAR$  の値がこれより大きい場合には計算を打ち切る。これにより、 $\alpha$  の計算、幾何変換、および近似誤差の計算が省かれ、計算量が削減される。なお、式 (6) の分母が 0 となる場合、式 (4) の分母も 0 となり、 $\alpha$  は計算できない。従って、この場合にも計算を打ち切る。

### 3.2 スケーリングパラメータの近似値計算—DR 法

式 (7) において  $r$  の交流成分は  $\alpha d$  のみから生成されることを考慮し、それぞれのブロックの分散に着目する。符号化時のレンジブロック及びドメインブロックの分散をそれぞれ  $V_r, V_d$  とすると、もし近似誤差が 0 となるような近似が得られた場合には以下の関係が得られる。

$$\begin{aligned} (r_{i,j} - \mu_r)^2 &= \{\alpha(d_{i,j} - \mu_d)\}^2 \\ \sum_{i,j} (r_{i,j} - \mu_r)^2 &= \sum_{i,j} \{\alpha(d_{i,j} - \mu_d)\}^2 \\ V_r &= \alpha^2 V_d \\ \sqrt{V_r} &= \alpha \sqrt{V_d} \quad (\alpha \geq 0) \end{aligned} \quad (8)$$

$\sqrt{V_r}, \sqrt{V_d}$  はそれぞれレンジブロック、ドメインブロックの偏差であるので、それぞれのブロックの偏差の比 (Deviation Ratio, 以下  $DR$  と記し、これを用いる手法を  $DR$  法と呼ぶ) を

$$DR = \sqrt{V_r/V_d} \quad (9)$$

とおき、偏差比と呼ぶ。このとき、偏差比  $DR$  とス

ケーリングパラメータ  $\alpha$  との関係は図 4 のようになり、 $\alpha$  は偏差比  $DR$  を上限とし、それ以下に分布していることが確認できる。これは、 $\alpha > 0$  とすると、

$$\begin{aligned} DR^2 - \alpha^2 &= \frac{V_r}{V_d} - \left[ \frac{\sum_{i,j} (r_{i,j} - \mu_r)(d_{i,j} - \mu_d)}{\sum_{i,j} (d_{i,j} - \mu_d)^2} \right]^2 \\ &= \frac{\sum_{i,j} (r_{i,j} - \mu_r)^2}{\sum_{i,j} (d_{i,j} - \mu_d)^2} \\ &\quad - \left[ \frac{\sum_{i,j} (r_{i,j} - \mu_r)(d_{i,j} - \mu_d)}{\sum_{i,j} (d_{i,j} - \mu_d)^2} \right]^2 \\ &= \frac{\sum_{i,j} (r_{i,j} - \mu_r)^2 \sum_{i,j} (d_{i,j} - \mu_d)^2}{\left[ \sum_{i,j} (d_{i,j} - \mu_d)^2 \right]^2} \\ &\quad - \frac{\left[ \sum_{i,j} (r_{i,j} - \mu_r)(d_{i,j} - \mu_d) \right]^2}{\left[ \sum_{i,j} (d_{i,j} - \mu_d)^2 \right]^2} \\ &\geq 0 \end{aligned}$$

となることから分かる。ただし、等号はブロック  $r$  と  $d$  とが等しいときに成立する。また、 $\alpha$  は比較的  $DR$  に近いところに分布しており、偏差比とスケーリングパラメータとの間には相関があると考えられる。ただし、スケーリングパラメータには多少分布に広がりがある。従って、次式のように偏差比を定数倍して近似値とする。この定数  $b$  を  $DR$  係数と呼ぶ。

$$\alpha = b \times DR \quad (10)$$

$DR$  の値は幾何変換によって変わらないので、全ての幾何変換に対して 1 回だけ  $\alpha$  を計算すれば良い。従って、式 (4) を用いて計算するよりも  $DR$  を計算する方が計算量が少なくなる。図 5 に  $MAR$  および  $DR$  を用いた場合の符号化の流れ図を示す。

## 4. シミュレーション実験

提案法の有効性を確認するため、計算機シミュレーションを行った。MAR 法における閾値と  $DR$  法における係数を決定した後、MAR 法、 $DR$  法および両者を組合せた方式について検討を行った。なお、全ての場合においてスケーリングパラメータのビット数は 3、平均値のビット数は 7 とした。

### 4.1 MAR 閾値の決定

テスト画像 “Lena,” “Aerial,” “Couple,” および “Cameraman” を用いて、MAR 法における計算打ち切りの閾値を変化させたときの原画像と復号画像との SN 比を図 6 に示す。図より、 $MAR$  が 1.2 以下で急激に SN 比が下がっているが、 $MAR$  が 1.8 以上で

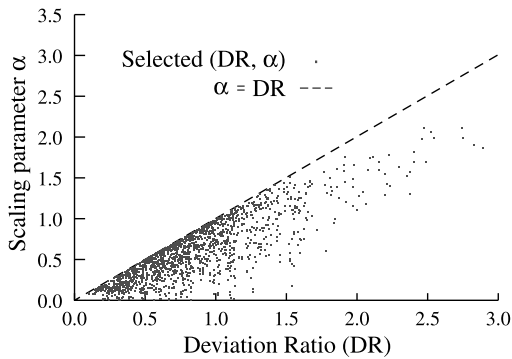


図 4 偏差比とスケーリングパラメータとの関係 (Lena, 256 × 256)

Fig. 4 Relation between the ratio of DR and scaling parameters (Lena, 256 × 256)

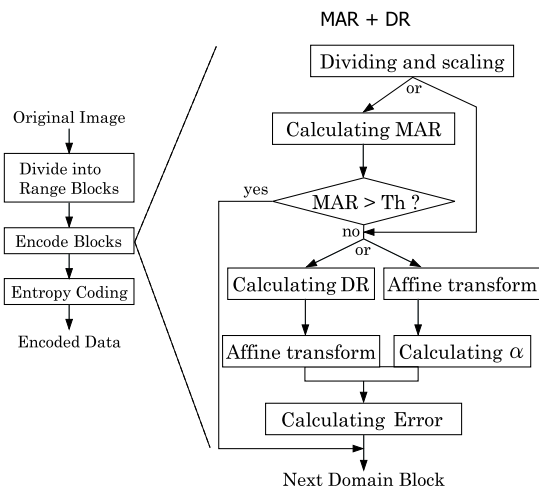


図 5 提案法のフロー図

Fig. 5 Flow of proposed scheme

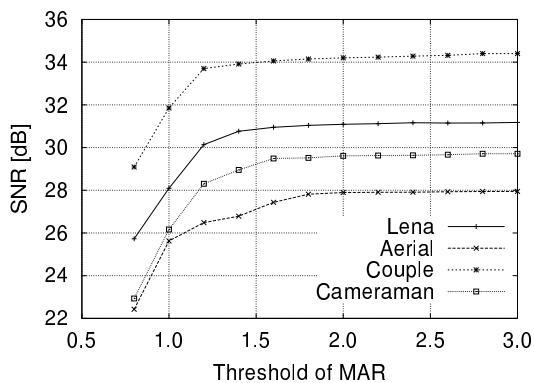


図 6 計算を打ち切る MAR の値と SN 比との関係

Fig. 6 Relation between the threshold of MAR and SNR

はほぼ同様の SN 比が得られていることがわかる。

同様に、閾値を変化させたときの符号化時間を図 7 に示す。ただし、縦軸は MAR 法を適用しない従来法の符号化時間を 1 とした場合の相対値である。図

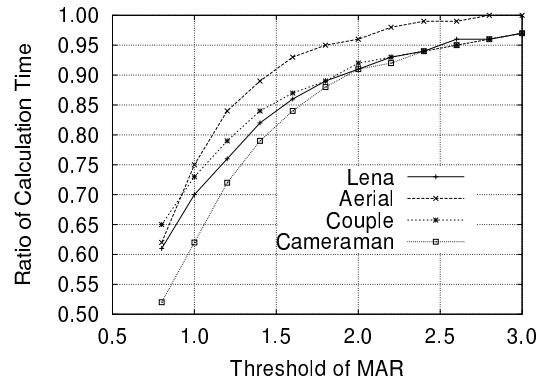


図 7 計算を打ち切る MAR の値と符号化時間との関係

Fig. 7 Relation between the threshold of MAR and Calculation Time

表 2 SN 比が最大となる時の DR 係数  $b$

Table 2 The DR constant where SN ratio has maximum value

画像 (256 × 256)	SN 比最大となる $b$
Lena	0.88
Aerial	0.92
Cameraman	0.94
Couple	0.92

より閾値が低下するにつれて符号化時間も減少していることがわかる。これらの図より、MAR 法では閾値を変化させることで符号化時間を削減できるが、SN 比と符号化時間は相反関係にあることがわかる。本稿では、SN 比の低下を極力抑えることを重視し、MAR の閾値を 1.6 として以下の実験を行った。

#### 4.2 DR 係数の決定

DR 係数  $b$  を変化させたときの原画像と復号画像との SN 比を図 8 に示す。Lena (256×256, 8bits grayscale) の場合には  $b = 0.88$  で SN 比が最大となるが、0.88~0.92 の範囲で同じような SN 比となっている。Aerial (256×256, 8bits grayscale) でも同様の傾向を示し、 $b = 0.92$  で SN 比最大となった。さまざまな画像に対して係数  $b$  と SN 比との関係を求めた結果、ほぼ同じような特性を示した。表 2 に SN 比が最大となるときの  $b$  の値を示す。これより平均値を求め、DR 係数を  $b = 0.92$  として固定した。

#### 4.3 符号化時間

前述の実験により得られた閾値および DR 係数を用いて各方式の符号化時間および復号画像の SN 比を求めた結果を表 3 に示す。MAR 法を用いた場合、従来法に比べてそれぞれ 86%、93% の処理時間になっている。偏差比を用いた場合、62%、66% の処理時間になっている。また、両者を組合せた場合、62%、67% の処理時間になっている。エントロピーについて

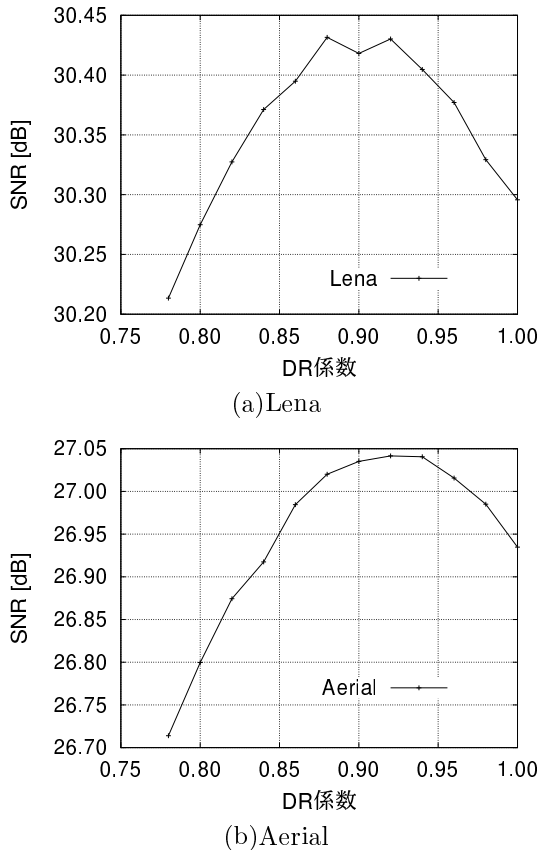


図 8 DR 係数  $b$  と SN 比との関係 (256×256, 8bits grayscale)  
 Fig. 8 Relation between the deviation ratio and DR constant (256 × 256, 8bits grayscale)

では、MAR 法、DR 法、両者を組合せた場合のいずれにおいても若干減少する傾向にある。一方、SN 比に関しては、MAR 法を用いた場合には 0.3dB から 0.5dB の減少しか見られないが、DR 法および両者を組み合わせた手法では 0.8dB から 1dB 程度減少している。これは、偏差比によって近似したスケールパラメータと、本来の計算式によって求められる値との差が大きいことを示している。また、MAR 法では閾値を変化させることにより符号化時間を制御できるが、DR 法と同じ処理時間を実現させようとした場合、SN 比が極端に落ちてしまう。従って、MAR 法では DR 法程には符号化時間を削減することはできず、むしろ劣化の少ない範囲で利用するのが適当である。

#### 4.4 ブロック処理時間の分析

図 9 に処理時間の内訳を示す。ただし、従来法を 100%として処理時間の割合で示してある。この図から、MAR 法を用いた場合、最大振幅比の計算のために約 5%計算時間が増加しているが、スケールパラメータの計算、幾何変換、近似誤差の計算に

表 3 符号化時間 (256×256, 8bits grayscale)  
 Table 3 Amount of calculating time (256 × 256, 8bits grayscale)  
 (a)Lena

手法	Entropy [bits/pel]	SNR [dB]	符号化時間比
0. なし	1.169	31.2	1.00
1. 最大振幅比	1.126	30.9	0.86
2. 偏差比	1.063	30.4	0.62
3. 1+2	1.061	30.4	0.62

(b)Aerial

手法	Entropy [bits/pel]	SNR [dB]	符号化時間比
0. なし	1.189	27.9	1.00
1. 最大振幅比	1.161	27.4	0.93
2. 偏差比	1.095	27.0	0.66
3. 1+2	1.093	26.8	0.67

※符号化時間は 10 回の試行の平均値。同一パラメータ

においてそれぞれ約 6%計算時間が削減され、全体として 14%計算時間が削減されていることが分かる。これは、計算打ち切りにより、以後の処理が行われなかったためである。

DR 法を用いた場合、偏差比そのものの計算のために約 5%要しているが、スケールパラメータの計算(量子化、定数倍のみ)にかかる計算時間が約 3%に削減されている。また、幾何変換、近似誤差の計算においてそれぞれ約 5%計算時間が削減され、全体として約 40%計算時間が削減されていることが分かる。DR 法では、スケールパラメータの計算時間削減の効果が特に大きく現れている。

両者を組み合わせた場合、偏差比、スケールパラメータの計算のための処理時間が増加分に対して以後の計算時間の削減量がそれほど多くないため、全体として約 40%の計算時間削減となり、DR 法とほとんど変わらない。従って、両者を組み合わせて利用するのは計算量削減には効果がない。

#### 4.5 復号画像

図 10～図 14 に原画像および各方式による復号画像の一部分を拡大したものを示す。256×256 の画像うち、左上を (0,0) とした座標 (100,100) から 64×64 の大きさだけ切り出してある。DR 法ではコントラストがやや失われているもののこれらの画像の画質に大きな差は認められない。しかし、SN 比を比べたとき、MAR 法が他の手法より多少優位となる。以上の結果から、劣化をなるべく抑えて高速化を実現させるには MAR 法がより適していると言える。ま

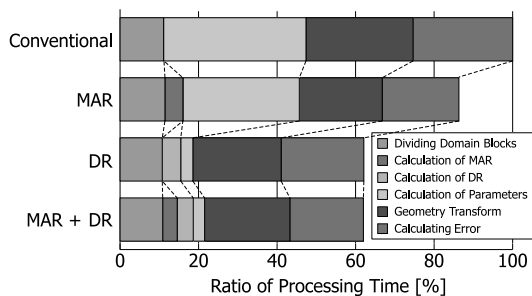


図9 ブロック符号化関数の処理時間の比較 (Lena)

Fig. 9 Details of calculation time to encode blocks (Lena)



図10 原画像の一部 (Lena, 64×64, grayscale)

Fig. 10 Part of Original Image (Lena, 64×64, grayscale)

た、大幅に高速化を実現するには DR 法が有効であると言える。

## 5. む す び

本稿では、レンジブロックとドメインブロックとの最大振幅比を用いて、スケーリングパラメータの計算打ち切りを行う手法および、スケーリングパラメータをレンジブロックとドメインブロックとの偏差の比で近似することとで符号化時間の高速化を実現するフラクタル画像符号化方式を提案した。計算機シミュレーションの結果、画質の劣化を抑えつつ MAR 法で約 10%, DR 法で約 35% 計算量が削減された。特に、SN 比の劣化をより抑えたい場合には MAR 法が、また高速性を重視した場合には DR 法



図11 従来方式の復号画像 (Lena, 1.169bits/pel, 31.2dB)

Fig. 11 Reconstructed Image of Conventional Scheme (Lena, 1.169bits/pel, 31.2dB)



図12 MAR方式の復号画像 (Lena, 1.126bits/pel, 30.9dB)

Fig. 12 Reconstructed Image of MAR Scheme (Lena, 1.126bits/pel, 30.9dB)

が有効である。

さらに高速化を実現させるためには、全体の約 60% を占める幾何変換および近似誤差の計算において、計算量を削減する必要がある。

## 文 献

- [1] 青木秀, 青木輝, 安田, “MPEG-7 を用いた画像検索における動き情報の応用,” 信学技報, IE2001-113, pp.29-36, Nov. 2001.
- [2] 相澤, “画像情報圧縮の手ほどき (第 12 回 新しい画像符号化方式),” テレビジョン学会誌, Vol.44 No.5, pp.625-632, 1990.
- [3] 堅田, 矢ヶ先, 渡辺, 甲藤, “MPEG4 ビジュアル規格,” 映像情報メディア学会誌, Vol.53 No.4, pp.485-491, 1999.
- [4] 原島, 酒井, 吉田, “映像情報符号化” オーム社, 2001.
- [5] A. E. Jacquin, “Image coding based on a frac-





図 13 DR方式の復号画像 (Lena, 1.063bits/pel, 30.4dB)

Fig. 13 Reconstructed Image of DR Scheme (Lena, 1.063bits/pel, 30.4dB)



図 14 MAR+DR方式の復号画像 (Lena, 1.061bits/pel, 30.4dB)

Fig. 14 Reconstructed Image of MAR+DR Scheme (Lena, 1.061bits/pel, 30.4dB)

tal theory of iterated contractive image transformation," IEEE Trans. Image processing, Vol.1, pp.18-30, Jan. 1992.

- [6] M. F. Barnsley, "FRACTALS EVERYWHERE," Academic Press Professional, 1988.
- [7] S. Curinga, "A proposal about fractal coding," Picture Codint Symposium '94, pp.429-432, 1994.
- [8] 富樫, 池原, 野寺, "フラクタル符号化の高速化," 情報処理学会論文誌, Vol.38 No.1, pp.64-72, Jan. 1997.
- [9] 黒田, デン, 今村, ホンヤン, "フラクタル画像符号化における高速ブロックマッチング法," 情報処理学会論文誌, Vol.38 No.8, pp.1543-1553, Aug. 1997.
- [10] 加藤, 田中, "ブロック特性に基づく高速フラクタル画像圧縮法," 第 21 回情報理論とその応用シンポジウム, pp.209-212, Dec. 1998.
- [11] 齊野, 田中, 宮下, 佐藤, "ブロックの特徴付けによるフラクタル符号化の高速化," 信学総大, D-11-37, p.37, 1998.
- [12] J. Mukherjee, P. Kumar and S. K. Ghosh, "A

Graph-Theoretic Approach for Studying the Convergence of Fractal Encoding Algorithm," IEEE Tran. Image Processing, Vol. 9, pp.366-377, Mar. 2000.

- [13] T. K. Truong, J. H. Jeng, I. S. Reed, P. C. Lee and A. Q. Li, "A Fast Encoding Algorithm for Fractal Image Compression Using the DCT Inner Product," IEEE Trans. Image Processing, Vol. 9, pp.529-535, Apr. 2000.
- [14] M. Polvere and M. Nappi, "Speed-Up In Fractal Image Coding: Comparison of Methods," IEEE Trans. Image Processing, vol. 9, pp.1002-1009, Jun. 2000.
- [15] H. T. Chang and C. J. Kuo, "Iteration-Free Fractal Image Coding Based on Efficient Domain Pool Design," IEEE Tran. Image Processing, Vol. 9, pp.329-339, Mar. 2000.
- [16] Y. H. Moon, H. S. Kim and J. H. Km, "A Fast Fractal Decoding Algorithm Based on the Selection of an Initial Image," IEEE Trans. Image Processing, vol. 9, pp.941-945, May. 2000.
- [17] M. Harada, T. Fujii, T. Kimoto, and M. Tanimoto, "Fractal Image Coding Based on Replaced Domain Pools," in Proc. SPIE Visual Communication and Image Processing, Vol.3309-103, pp.1042-1049, Jan. 1998.
- [18] 淵上, 矢野, 小松, 斎藤, "IFS 画像符号化におけるブロックのクラス分類手法の一検討," PCSJ95, pp.179-180, Oct. 1995.
- [19] B.Ramamurthi and A.Gersho, "Classified vector quantization of images," IEEE Transactions on communications, pp.1105-1115, Vol.com-34, No.11, Nov. 1986.
- [20] ケイワーク, "JPEG 概念から C++による実装まで," ソフトバンクパブリッシング, Sep. 1998.

## 付 録

### 1. JPEG との比較

#### 1.1 ブロックの分類

実際に符号化をする際、輝度値の平坦なレンジブロックについてはドメインブロックの探索を行わない [5]. このレンジブロックを SHADE ブロックと呼ぶ. そして輝度値の平坦でないレンジブロックを EDGE ブロックと呼び, EDGE ブロックのみドメインブロックの探索を行う. SHADE ブロックと EDGE ブロックの分類法には, ブロック内の輝度値の分散値をもとに分類する方法 [18] や, エッジ検出法 [19] を用いて分類する方法 [5] がある. 通常はブロック内の輝度値の分散値をもとに SHADE, EDGE の分類を行う.

SHADE ブロックの場合, レンジブロック内の輝度値の平均値を復号側へ伝送し, EDGE ブロックの場合, ドメインブロックを変換する際に用いられた変換パラメータを復号側へ伝送する.

#### 1.2 四分木分割法

四分木分割法は, 矩形ブロックを用いるフラクタル画像符号化の中で, 比較的良好な符号化特性を示

す。四分木分割法では、 $16 \times 16$ 、 $8 \times 8$ 、 $4 \times 4$  という 3 種類の大きさのレンジブロックを用いる [17]。符号化を行う前に、まず  $16 \times 16$  程度の大きなレンジブロックについてブロック内の輝度分散を計算し、ブロックの分類を行う。ブロックの状態には、ブロック内の輝度平均値を符号とする SHADE、フラクタルの変換パラメータを符号とする EDGE、4 つの小さなブロックに分割する DIVIDE がある。判定のための閾値を  $V_{SE}$ 、 $V_{EV}$  とし、あらかじめ与えられるものとする。ブロック内の輝度分散が  $V_{SE}$  よりも小さい場合、SHADE となる。  $V_{SE}$  以上  $V_{EV}$  未満の場合、EDGE となる。  $V_{EV}$  以上となる場合、DIVIDE と判定される。

DIVIDE となった場合には、そのレンジブロックを 4 つの小さなレンジブロックに分割する。分割してできるレンジブロックそれぞれに対して、再度ブロック内の輝度分散を計算し、ブロックの分類を行う。最小のレンジブロックサイズになるまでこの操作を繰り返す。ただし、最小のレンジブロックで DIVIDE と判定された場合には EDGE とする。これにより、パラメータの探索を行う前に画像の分割が終了する。

### 1.3 符号化実験

四分木分割法およびブロックの分類を適用したフラクタル画像符号化に、本稿の提案法を適用した際の符号化特性を図 A・1 に示す。比較のために、JPEG のベースライン方式 [20] による符号化特性も併せて示す。図より、SN 比で比較すると、エントロピーが  $0.7 \text{ bits/pel}$  以上の領域および  $0.4 \text{ bits/pel}$  以下の領域では、JPEG の方が SN 比が高いことが分かる。一方、 $0.4 \text{ bits/pel}$  から  $0.7 \text{ bits/pel}$  までの領域では、ほぼ同程度の SN 比である。DR 法および MAR 法と DR 法を組み合わせた手法では、JPEG を超えることはないが、MAR 法は  $0.5 \text{ bits/pel}$  付近で JPEG を上回る性能を示している。また、表 A・1 に符号化時間を示す。MAR 法を用いた場合、従来法に比べて 77% の処理時間になっている。偏差比を用いた場合、55% の処理時間になっている。また、両者を組合せた場合、51% の処理時間になっている。このように、四分木分割法およびブロックの分類を適用した場合には MAR 法と DR 法を組み合わせた方式も有効である。

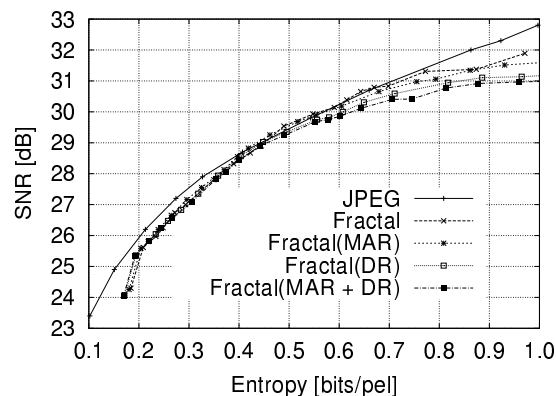


図 A・1 四分木分割法と JPEG の符号化特性 (Lena,  $256 \times 256$ , 8bits grayscale)

Fig. A-1 Coding Performance (Lena,  $256 \times 256$ , 8bits grayscale)

表 A・1 符号化時間 (Lena,  $256 \times 256$ , 8bits grayscale)

Table A-1 Amount of calculating time (Lena,  $256 \times 256$ , 8bits grayscale)

手法	Entropy [bits/pel]	SNR [dB]	符号化時間比
0. なし	0.444	28.897	4.846
1. 最大振幅比	0.418	28.828	0.77
2. 偏差比	0.404	28.567	0.55
3. 1+2	0.399	28.438	0.51

※符号化時間は 10 回の試行の平均値。同一パラメータ。