

## フラクタル画像符号化方式

黒田 英夫\* デン ポペスク\*\* ホン ヤン\*\*

\* 長崎大学工学部 \*\* シドニー大学工学部

本稿では、フラクタルモデルを用いた高速処理、高圧縮可能な画像符号化方式について述べている。フラクタル画像符号化においては、自分自身の画像の中から最適ドメインブロックを探索するのに、長時間を要し、かつ最適ドメインブロックのアドレス情報も大量となる問題があった。ここでは、最適ドメインブロックの探索時間が短く、かつアドレス情報を大幅に削減できる方法を提案している。本方式は、最適ドメインブロックを積極的に探索する方法として、レンジブロックを中心とするドメインブロックが有力な最適ブロックの候補となる筈であるという考え方を取っている。提案する方式に対して実画像を用いた符号化実験を行ない、良好な実験結果を得ている。

## Image Coding Method Based on Fractal Models

Hideo Kuroda\*, Dan C. Popescu\*\* and Hong Yan\*\*

\*Faculty of Engineering, Nagasaki University

1-14, Bunkyo-machi, Nagasaki, 852 Japan

\*\*Dept. of Electrical Engineering, The University of Sydney  
NSW 2006, Australia

This paper presents a fast block matching technique for image data compression based on fractal models. In fractal coding, domain blocks in an image are searched and the one most similar to a range block is selected as the best matching domain block. We propose a fast search method to improve the encoding time and the data compression rate. In our method a positive aim at a domain block centered on the range block as the best matching domain block is proposed. The method has been tested on real image data with good results.

# 1 まえがき

近年フラクタルモデルを用いた実画像の符号化の研究が行なわれている。[1]-[6] フラクタル符号化においては、復号過程における処理は比較的容易であるが、符号化において符号化処理の単位であるレンジブロックのパターンに最も良く似たドメインブロックを探索するための処理時間が非常に長くなるという問題がある。この点がフラクタル符号化の大きな欠点となっている。このブロック探索にかかる処理時間を改善するためには、探索するドメインブロックの数を削減する必要がある。

フラクタル符号化においては、ブロックマッチングの観点からレンジブロックサイズを余り大きくできず、このため如何に情報圧縮を行なうかということがもう一つの重要なポイントである。上述したような問題点に対して、従来方式はあるブロックを最適ドメインブロックの候補として積極的に用いようとする試みはなされていなかった。Beaumont はレンジブロックの周辺ブロックに注意を払い、それらのブロックを他のブロックより優先的に探索する方法を採用しているが [6]、そこでは、各ドメインブロックの大きさをレンジブロックの9倍の大きさにしている。この大きさは、レンジブロックをそのドメインブロックの中心に持ってきてはいるが、レンジブロックと似ているというためには余りにも大き過ぎ、かつレンジブロックの周辺に最適ドメインブロックがあるという根拠も期待できない。

本稿では、処理時間の短縮と情報圧縮を改善する高速ブロック探索法を採用したフラクタル画像符号化方式を提案する。この方法は、レンジブロックを中心に含むドメインブロックが最適ブロックである可能性が高いとして優先的に使用するものである。

## 2 フラクタル画像符号化の基本原 理

図1に示す画像を考える。実線はレンジブロック、点線は従来選択されていたドメインブロック、破線は本稿で提案するドメインブロックである。レンジブロックが符号化の単位となる。各レンジブロックに対して、自分自身の画像の中から、似通ったパターンをもつドメインブロックを選択し、そのアドレス情報をレンジブロックの平均値、標準偏差とともに復号器側に伝送する。

符号器においては、式1に示す $\delta$ が最小となるド

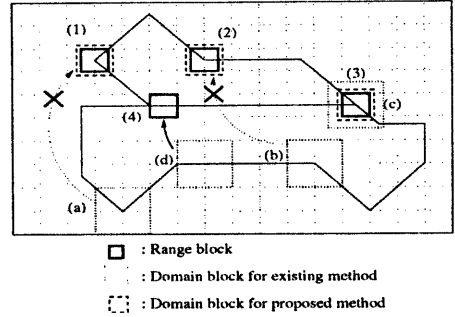


図1: フラクタル画像符号化の基本原理

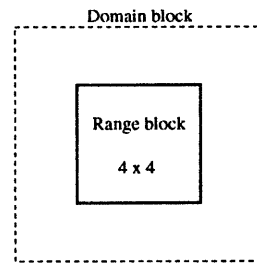


図2: レンジブロックとドメインブロックの関係

メインブロックを最適ブロックとして選択する。探索されるドメインブロックの中には、アフィン変換後のブロックも含めて探索を行なう。

$$\delta = \delta(r, d) = \sqrt{\frac{1}{n^2} \cdot \sum_{0 \leq i, j \leq n-1} (r_{ij} - d_{ij})^2} \quad (1)$$

ここで、 $r$  はレンジブロック内の画素値であり、 $d$  はレンジブロックと同じブロックサイズにリスケールされたドメインブロック内の画素値である。そして、レンジブロックの平均値と標準偏差および選択されたドメインブロックのアドレス情報が伝送される。

復号器側においては、これらのデータを受信する。送られてきたドメインブロックのアドレス情報を用いて受信側のフレームメモリの中からリスケールされたドメインブロックの平均値および標準偏差を作成することができる。ドメインブロックのこれらの値と、受信したレンジブロックの平均値および標準偏差を用い、式2より再生画像の画素値を得る。[1][2]

$$\hat{r} = (\hat{r}_{ij}) = ((d_{ij} - m(d) \cdot dev(r)) / dev(d) + m(r)) \quad (2)$$

ここで、 $f$  はレンジブロックの再生画素値、 $m$  及び  $dev$  は平均値、標準偏差である。そして、この復号処理は良好な再生画像品質が得られるまで繰り返し行なわれる。上述したように、フラクタル画像符号化においては、2つの重要なポイントがある。ブロックマッチングと繰り返し演算システムである。本稿は前者のブロックマッチングに関するものである。

### 3 ブロック構成

#### 3.1 ドメインブロックのサイズと位置関係

従来の選択方法では図1のレンジブロック(1)(2)(4)はレンジブロックと距離の離れたドメインブロックが使用されていた。また、レンジブロック(3)は自分と同じ中心をもつドメインブロックを、右下の部分で1部違いはあるものの、これを最適ブロックとして選択することになる。従って、最適ドメインブロックのアドレス情報を復号器側に伝送する必要がある。ここで提案する方法では、破線で示されたドメインブロックに注目している。レンジブロック(1)～(3)に対しては、個々のレンジブロックを中心に持つドメインブロックが使用される。従って、そのことを表すフラグ1ビットのみ伝送すれば良い。また、レンジブロック(4)に対しては、レンジブロックを中心に持つドメインブロックではレンジブロックとの類似性が小さいため、他のドメインブロックの中から最適ブロックが選択されることになる。この場合はドメインブロックのアドレスを伝送する。

上述したように、従来方式とのもっとも大きな相違点は次の点である。

従来方式：最適ドメインブロックは画面内のどこかある筈である。

提案方式：最適ドメインブロックはここ、レンジブロックと同じ位置にある筈である。

図2はレンジブロックとドメインブロックの関係を示す。レンジブロックの内部画素はドメインブロック内に含まれており、このドメインブロックとレンジブロックが良く似通っていることは明白である。一方、レンジブロックの周辺画素については、レンジブロックの最外部の画素と大きい相関を持っている。これはフレーム内予測符号化方式から類推できる。図3及び図4は最適ブロックの発生頻度を表す。ドメインブロックサイズはそれぞれ6x6及

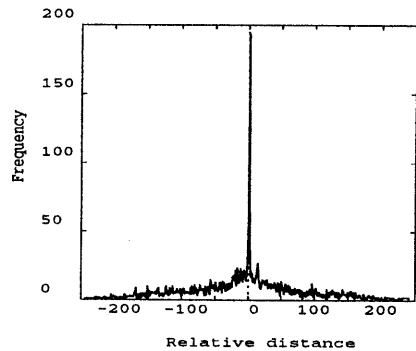


図3: 最適ブロックの発生頻度、レンジブロック 4x4、ドメインブロック 6x6

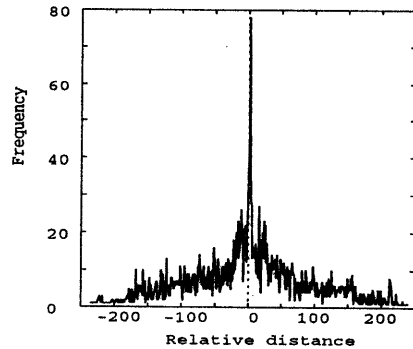


図4: 最適ブロックの発生頻度、レンジブロック 4x4、ドメインブロック 8x8

び8x8、レンジブロックサイズはいずれも4x4である。これらは、全ての位置のドメインブロックを探索の対象として測定したものである。ドメインブロックサイズ8x8の場合は、6x6の場合より中心への集中度が低い、それでもレンジブロックと同じ位置のドメインブロックを最適ブロックの最有力候補として使用し得ることが分かる。このことより、レンジブロックの外側1画素、あるいは2画素までからなるドメインブロックが効果的に使用可能である。

#### 3.2 ブロック探索

ブロック探索は下記のような手順で行なわれる。

- 先ず、レンジブロックを中心位置に持つドメインブロックについて評価を行なう。
- 次に、レンジブロックの周辺8個のドメインブロックについて評価を行なう。
- 最後に、その他のドメインブロックについて評価を行なう。

### 3.3 データ圧縮

レンジブロックを2種類のブロックに分けて符号化を行なう。1つは、フラットブロックであり、もう1つはノンフラットブロックである。フラットブロックに対しては、6~8ビットの平均値のみを伝送する。ノンフラットブロックに対しては、最適ドメインブロックアドレス情報(16ビット)、平均値(6~8ビット)、スケーリングファクタ(6~8ビット)及びアフィン変換の変換コード(3ビット)が伝送される。本稿では、上述したようにレンジブロックを中心にもつドメインブロック、次に周辺の8ドメインブロックを優先的に評価することにより、ドメインブロックのアドレス情報を大幅に圧縮する。

### 3.4 ブロック探索処理時間

フラクタル画像符号化では、ドメインブロックは、先ず画面全体の中から準備されなければならない。そのための処理時間はドメインブロックの存在する場所に依存するのではなく、その数によって異なる。画面全体に存在するドメインブロックの数は同じとすれば、この処理における消費時間は従来方式と同様である。しかし、ブロック探索に関わる処理時間は、探索されるドメインブロック数によって異なる。画面サイズ 256 x 256、ドメインブロックサイズ 8x8 水平及び垂直方向のステップサイズを2とする時、探索するドメインブロック数は通常数百ブロックであるが、最悪ケースでは 15625 (= 125 x 125) となる。もし、レンジブロックの近傍9ブロックのみの探索で終了する場合は探索処理時間を約 1/20 に圧縮することができる。もし、レンジブロックを中心にもつドメインブロック1つの評価だけで終了する場合は特に効率が良く、約 1/100 に圧縮することができる。図3及び図4に示されたように、中心ドメインブロックが極めて頻繁に発生する場合には、画像全体に対する処理時間は大幅な短縮が可能となる。

| Block    |                | 6x6      |            | 8x8      |            |
|----------|----------------|----------|------------|----------|------------|
|          |                | Centered | Uncentered | Centered | Uncentered |
| Flat     |                | 25.66    |            |          |            |
| Non-flat | Center         | 41.67    | 74.34      | 30.64    | 74.34      |
|          | Neighbor eight | 3.03     |            | 5.32     |            |
|          | Others         | 29.64    |            | 38.38    |            |

[%]

表1: ブロック種別の発生確率

## 4 シミュレーション結果

ここで提案したフラクタルモデルを用いた画像符号化方式についてシミュレーション実験を行なった。テスト画像として、画像サイズ(256 x 256 x 8)の“Lena”を使用した。本方式では、レンジブロックを中心にもつドメインブロックを使用しており、それと比較するため、レンジブロックを中心にも含まないドメインブロックの構成方式についてもシミュレーションを行なった。先ず、ブロック種別の発生確率を測定した。表1は、ブロック種別の発生確率を示す。フラットブロックの発生確率はドメインブロックサイズ6x6あるいは8x8、また、レンジブロックとの位置関係に関わりなく同じである。ノンフラットブロックは画素値にバラツキのあるブロックではあるが、それに関わらず中心ブロックが頻繁に発生していることが分かる。このことは、レンジブロックとその外側2画素までの周辺画素からなるドメインブロックがレンジブロックと似通ったパターンを持つという我々の仮定が正しいことを証明するものである。表2は、ドメインブロックのアドレスと画素値に対する符号長を示す。フラットブロックとノンフラットブロックを識別するための符号は振幅値データに含めている。アフィン変換を表すデータ3ビットはドメインブロックの種類に関するものとして、アドレスデータに含めているが、中心ドメインブロックではこれは不要である。

中心ドメインブロック及びその周辺8ドメインブロックについては、それぞれ13ビット、7ビット圧縮できている。しかし、その他のドメインブロックでは1ビット増加している。表3は、表1及び表2を用いて計算した各ブロック種別に対するエン

| Block     |          | Centered       | Uncentered |
|-----------|----------|----------------|------------|
| Address   | Flat     | 0              | 0          |
|           | Non-flat | Center         | 2          |
|           |          | Neighbor eight | 8          |
|           |          | Others         | 16         |
| Amplitude | Flat     | 9              | 9          |
|           | Nonflat  | 17             | 17         |

[bits]

表 2: ドメインブロックのアドレスと画素値に対する符号長

| Block              |          | Domain block 6x6 |                | Domain block 8x8 |                |
|--------------------|----------|------------------|----------------|------------------|----------------|
|                    |          | Centered         | Uncentered     | Centered         | Uncentered     |
| Address            | Flat     | 0                |                |                  |                |
|                    | Non-flat | Center           | 0.83           | 0.61             | 11.15          |
|                    |          | Neighbor eight   | 0.24           | 0.42             |                |
|                    |          | Others           | 4.74           | 6.14             |                |
| Amplitude          | Flat     | 2.31             |                |                  |                |
|                    | Nonflat  | 12.64            |                |                  |                |
| Total [bits/block] |          | 20.76<br>(79.8%) | 26.1<br>(100%) | 22.12<br>(84.7%) | 26.1<br>(100%) |

表 3: ブロック種別に対するエントロピー

トロピーを示す。エントロピーの合計は、ドメインブロックサイズ 6x6 及び 8x8 の場合について、非中心ドメインブロックの場合に比べ、それぞれ約 20.2% 及び 15.3% 改善されている。

表 4 は、実験結果を示す。情報発生量はドメインブロックサイズ 6x6 及び 8x8 の場合、それぞれ、0.33 ビット/画素 及び 0.25 ビット/画素 改善されている。ドメインブロックの平均探索数は約 1/2 に改善されている。ドメインブロック数の探索数を制限することにより SNR が犠牲になっているが、それでもドメインブロックサイズ 6x6 及び 8x8 の場合について、それぞれ 33dB 及び 34.4dB を得ている。図 5 に原画像、図 6 及び 図 7 に、それぞれドメインブロックサイズ 6x6 及び 8x8 の場合の再生画像を示す。レンジブロックサイズは 4x4 である。良好な再生画像が得られている。図 8 は

|                        | Domain block 6x6 |            | Domain block 8x8 |            |
|------------------------|------------------|------------|------------------|------------|
|                        | Centered         | Uncentered | Centered         | Uncentered |
| Data rate [bits/pixel] | 1.30             | 1.63       | 1.38             | 1.63       |
| The number of searches | 417              | 1043       | 537              | 1038       |
| SNR[dB]                | 33.0             | 35.0       | 34.4             | 35.3       |

表 4: 実験結果

復号処理の繰り返し回数に対する再生画像の SNR を示している。ドメインブロックサイズが 8x8 の場合は、6x6 の場合に比べ収束が早く 3 回目で飽和している。6x6 の場合でも 4 回目で飽和している。前述した通り、復号化処理は簡単のため、このことは大きな問題とはならない。

## 5 むすび

高速処理、高圧縮可能な、フラクタルモデルを用いた画像符号化方式について述べた。従来方式が画面のどこかに最適ドメインブロックがある筈であるとするのに対し、本稿ではレンジブロックのところ、即ちレンジブロックを中心にもつドメインブロックに最適ドメインブロックがある筈であるという考えにたつた方式を提案した。まず、中心ドメインブロックを最優先して評価し、次にその周辺 8 ドメインブロックについて評価し、最後にその他のドメインブロックを評価するものである。このことにより探索するドメインブロック数を大幅に削減でき、かつ最適ドメインブロックのアドレス情報も大幅に圧縮できる。シミュレーション実験により、提案する方式が高速処理、高圧縮とともに、良好な画像品質を得られることが明らかとなった。今後その他のドメインブロックのサイズを可変にする方式について検討する予定である。

## 参考文献

- [1] L.M.Beaumont, Image data compression using fractal techniques, BT Technology Vol.9, No.4.



図 5: 原画像



図 6: 再生画像、ドメインブロック 6x6



図 7: 再生画像、ドメインブロック 8x8

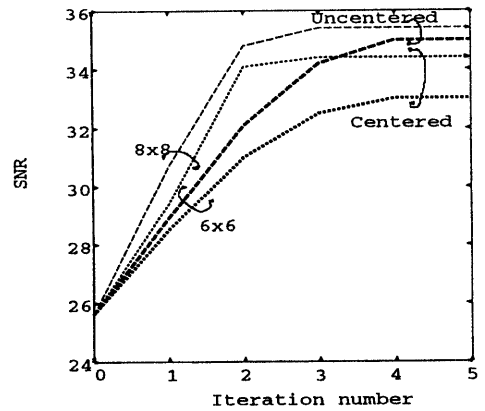


図 8: 復号繰り返し回数に対する SNR

pp.93-109 Oct.1991

- [2] Dan C.Popescu, Hong Yan, MR IMAGE COMPRESSION USING ITERATED FUNCTION SYSTEMS, Magnetic Resonance Imaging, Vol.11, No.5, pp.727-732 1993
- [3] Barnsley, M.F.Fractals Everywhere, New York:Academic Press, 1988
- [4] A.E.Jacquin, Image coding based on a fractal theory of iterated contractive image transformation, IEEE Trans. Image Processing 1:18. 1992
- [5] A.E.Jacquin, A novel fractal-coding technique for dijital images, IEEE Intn. Conf. on Acoustics, Speech and Signal Processing,M8.2, pp.2225-2228
- [6] L.M.Beaumont, Advaces in Block based Fractal Coding of Still Pictures,IEEE Colloguim on Fractal, Dec.1990