

フラクタル符号に基づく類似性抽出についての検討

横山貴紀, 渡辺俊典, 菅原 研, 上野芳朗

{yokotaka, watanabe, sugawara, yoshiro}@sd.is.uec.ac.jp

電気通信大学 大学院情報システム学研究科

概要

フラクタル符号を用いた画像の構造的な類似性に関する特徴抽出手法を提案する。フラクタル画像圧縮により圧縮された画像の符号は、画像中の相似領域の関係を写像として記録している。この写像をベクトルとして扱い、ベクトル集合から代表ベクトルを生成することにより画像間の類似度を求めることができる。この類似度は構図の異なる画像を判別することが可能である。本報告ではこの提案手法の画像分類への可能性を実験によって検証している。

キーワード：フラクタル画像圧縮, フラクタル符号, 画像分類, 相似性, 構造的類似度, 特徴抽出

Feature Extraction of Structural Similarity between Images using PIFS code

Takanori Yokoyama, Toshinori Watanabe, Ken Sugawara, Yoshiro Ueno

{yokotaka, watanabe, sugawara, yoshiro}@sd.is.uec.ac.jp

Graduate School of Information Systems, University of Electro-Communications

Abstract

We propose a new scheme of feature extraction about structural similarity between images using PIFS code. In fractal image encoding, PIFS code has similarity relation maps between domains and ranges in the same image. These maps are treated as vectors, and representative vectors are generated from vector sets. The structural similarity between images is calculable from representative vectors. In this report, we demonstrated this possibility experimentally.

key words : Fractal Image Compression, PIFS code, Image Categorization, Similarity, Structural Similarity, Feature Extraction

1 はじめに

一般に圧縮技術は非常にコストの掛かる演算を必要とする。この圧縮で費やされるコストをデータの圧縮と復号だけに消費してしまうのは、あまりにも勿体ない。そこでこのコストを利用し、認識の問題へ応用するのが本研究の狙いである。

この報告では圧縮手法としてフラクタル画像圧縮の技術を取り上げている。この手法が提供する圧縮技術は、画像の幾何形状に着目していることから、画像の認識について非常に高い潜在能力を有していると考えられる。

この特徴に着目し、圧縮手法の技術の中心である符号生成のメカニズムから、認識に供する特徴を抽出する手法を提案する。

本手法には以下の特徴がある。

- 圧縮符号の非解凍 (非復号) による特徴抽出
- 画像の構造を反映した特徴を抽出

本報告は、まず、研究の背景を述べ、フラクタル画像圧縮の概要を説明する。次に特徴の抽出手法のアルゴリズムと、その詳細についての説明を行い、いくつかの基礎的な実験データを示し、本手法が持つ妥当性と性質を検証する。

2 研究の背景

2.1 研究の目的

研究ではフラクタル画像圧縮をデータベースに応用するための基礎技術の確立を目指している。データベースから画像を取り出す、あるいは選択する際に、非解凍のまま画像を判別できることが望ましい。このような技術が完成すれば、データベースから任意の画像を、各処理系 (人物の識別や、文字認識などの各処理) に渡すことができる。

2.2 関連研究

フラクタル画像圧縮に関する研究は、圧縮効率全般についての改善が主流を占める。特に高い品質を保ったままの圧縮率の改善と、圧縮演算の高速化などが中心である [1]。

一方で主に国内を中心にフラクタル画像圧縮の技術を応用する研究が行われている。これは井田らの領域分割 [2] と輪郭線抽出 [3]、それらの改善などに焦点が絞られている。本研究は、この応用技術に属するものである。

私たちはこれまでに応用技術として、フラクタル符号の半解凍過程における画像変化に着目した画像分類技術を報告した [4]。

この手法は比較対象の画像と、非解凍時に得られる画像間の誤差から、画像間の類似性を抽出することを試みたものであった。この報告により、符号そのものが与える情報が有用であること、比較対象の画像が持つ構造も重要であることが明らかになった。

一方で私たちの研究グループでは、別の圧縮手法を用いた認識技術 [5] の研究も行っている。本報告はこれらの知見と技術を基に、さらに改善を重ねたものである。

3 フラクタル画像圧縮

3.1 概要

本手法はフラクタル画像圧縮を利用したものである。フラクタル画像圧縮は、図 1 のように同一画像中のフラクタル性を利用した圧縮手法である。

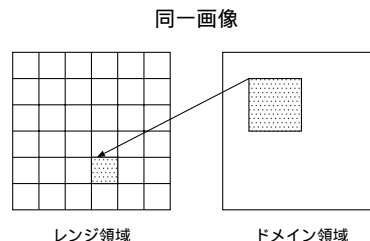


図 1: 同一画像中のレンジとドメイン領域と、その相似関係

圧縮手法のアイデアそのものは Barnsley らのフラクタル画の作成法に基づいている [6]。特に、この時用いられた反復関数系 (IFS: Iterated Function System) が圧縮の基本となっている。詳細については [7] や [8] などを参照されたい。

フラクタル画像圧縮は Fractal Image Compression、あるいは手法そのものを示す用語として PIFS (Partitioned IFS) などと呼ばれる。本報告では、

フラクタル画像圧縮されたデータそのものを、フラクタル符号および PIFS code と呼ぶ。

3.2 レンジサイズによる多重解像度的表現

PIFS で最も使用される領域分割手法は Quadtree Partitioning である [1]。この分割手法により画像の局所的な変動に対し可変サイズのレンジを割り当てることができる。

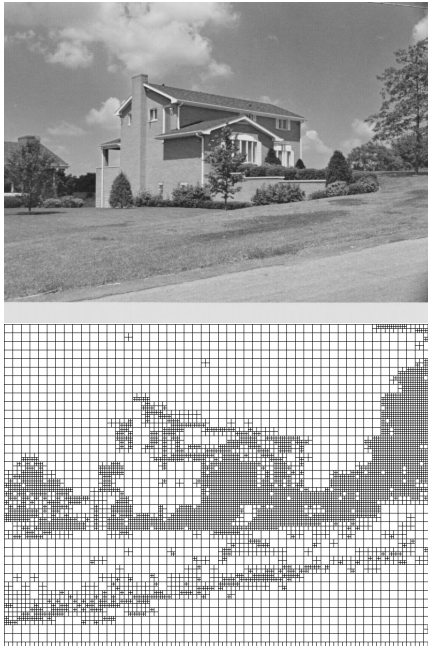


図 2: 多重解像度的な表現 (上) 原画像 house (下) quadtree partition

図 2 のように、大きいレンジサイズには変動の穏やかな部分、小さいレンジには変動の激しい領域が割り当てられている。

多重解像度のようにも見えるが完全には対応しない。よって本報告ではこれらを区別するため、レンジサイズによる階層を「多重解像度的な表現」であると称する¹。

¹解像度によって対応関係を決定する場合、多重解像度と一致するが、一般的な PIFS では対応領域をマッチングにより求めている。よって解像度の高い領域に大きなレンジサイズに割り当てられることも、その逆の現象も生じる

4 提案手法

4.1 概要

本研究の目的は画像圧縮により得られた符号から、解凍 (復号) を必要とすること無く、画像間の構造的な類似性を示す特徴量を決定することにある。

本手法のアルゴリズムは次のようになる。

1. フラクタル符号から写像を取り出す
2. 写像の移動項のみに着目し、ベクトル表現する
3. 写像ベクトルの集合から代表ベクトルを決定する
4. 画像間の代表ベクトルを比較することにより、構造的な類似性を求める

このアルゴリズムの詳細について、次節から順に説明する。

4.2 フラクタル符号とその写像

フラクタル符号にはレンジ領域とドメイン領域の相似関係が記録されている。記録された写像は以下の様に表現される。画像中のある領域について着目し、符号中の i 番目の要素と考える。レンジと呼ばれる領域の位置を rx_i, ry_i と表し、その位置にある画素の輝度を rz_i とする。レンジ領域と相似関係にあるドメインの領域を dx_i, dy_i とし、輝度を dz_i と表現する²。

レンジとドメインは以下の写像 w_i を用い、以下の関係式となる。

$$\begin{aligned} \begin{bmatrix} rx_i \\ ry_i \\ rz_i \end{bmatrix} &= w_i \begin{bmatrix} dx_i \\ dy_i \\ dz_i \end{bmatrix} \\ &= \begin{bmatrix} a_i & b_i & 0 \\ c_i & d_i & 0 \\ 0 & 0 & s_i \end{bmatrix} \begin{bmatrix} dx_i \\ dy_i \\ dz_i \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \\ o_i \end{bmatrix} \end{aligned} \quad (1)$$

特に提案手法で着目したのは、レンジとドメインの位置の対応関係のみである。輝度項については今回無視した。よって次の v_i だけを着目したことになる。

$$\begin{aligned} \begin{bmatrix} rx_i \\ ry_i \end{bmatrix} &= v_i \begin{bmatrix} dx_i \\ dy_i \end{bmatrix} \\ &= \begin{bmatrix} a_i & b_i \\ c_i & d_i \end{bmatrix} \begin{bmatrix} dx_i \\ dy_i \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \end{bmatrix} \end{aligned} \quad (2)$$

² rx_i, ry_i, dx_i, dy_i は画素ではなく、領域中の画素集合を表すが、本手法では後述のように画素集合の左端点として扱う

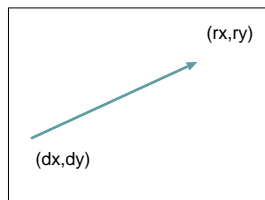
4.3 写像のベクトル表現

着目した式 2 について、特に移動項に注目する。このとき、回転行列も無視し、 x_i と y_i を領域全体ではなく、領域の左端点と考えると、結果として e_i と f_i に着目することになる。

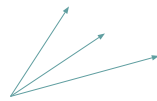
つまり相似関係を持つ対応領域を表す情報である式 3 の 4 次元ベクトルを、写像 v_i から抜きだす。

$$\begin{bmatrix} rx_i \\ ry_i \\ dx_i \\ dy_i \end{bmatrix} = \begin{bmatrix} dx_i + e_i \\ dy_i + f_i \\ dx_i \\ dy_i \end{bmatrix} \quad (3)$$

4 次元ベクトルは、図 3 のように 2 次元平面上に位置を含んだベクトルとして表現することができる。



位置情報を含んだベクトル



通常ベクトルは始点を合わせる

図 3: 写像の 2 次元平面上でのベクトル表現

本研究では、写像から抜き出した対応領域の位置関係によりベクトル表現したものを「写像ベクトル」と呼ぶこととする。

4.4 クラスタリングを用いた代表ベクトルの生成

写像ベクトルを各レンジサイズ毎に求めることができる。この写像ベクトルを 2 次元平面に表示すると各階層で異なる粗密が生じる。このベクトル線の密な部分に着目し、代表的なベクトルを生成する。この代表的な写像ベクトルのことを「代表ベクトル」と呼ぶ。

代表ベクトルは写像ベクトル集合の特徴を表す。写像ベクトルは画像の構造によって形成される傾向があ

り、代表ベクトルもまた、画像の構造の骨格を表すと見ることができる。

本手法では写像ベクトル集合を 4 次元空間上で扱うことにより代表ベクトルを求めることとした。写像ベクトルが密になる部分とは、4 次元空間上に表現した場合、点が密集する部分である。

このことから、4 次元空間上でのクラスタリングにより、ベクトルが密になっている部分によりクラスターを形成することができる。そしてクラスターの重心位置を、図 4 のように、代表ベクトルとすることができる。

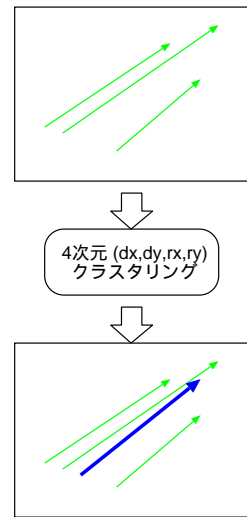


図 4: 4 次元空間でのクラスタリングと代表ベクトル

本手法ではクラスタリングに k -means 法を選択した。これは重心位置を更新し、もっとも好ましい位置で代表ベクトルを決定することができるからである。

図 5 は標準画像 Lenna に対する写像ベクトルと、代表ベクトル (ベクトル数は 16) を実際に求めた結果である。

4.5 代表ベクトルの比較と構造的類似度

前節までの処理によって、画像の構造的な特徴を表す代表ベクトルが求めることができた。次に、この代表ベクトルの違いに着目することにより、画像間の類似性を見ることを試みる。

本手法ではこの代表ベクトルを比較した結果を、構造的類似度を用いて表現する。

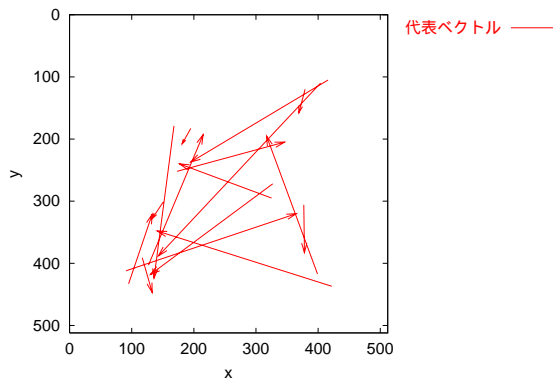
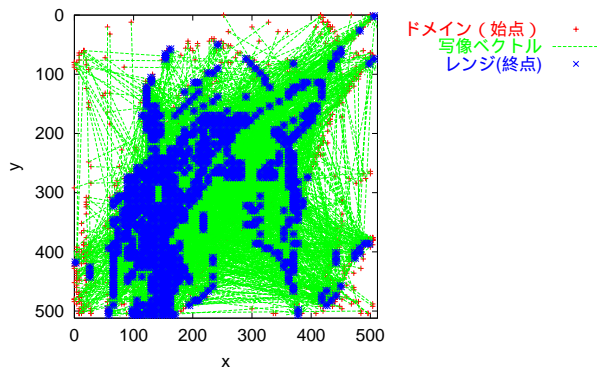


図 5: (上) Lenna の画像ベクトル (下) 代表ベクトル

二つの画像の代表ベクトルの集合を X と Y とする。それぞれの集合がもつベクトルの各要素について、最短距離法 (手法の概要は [9] など) により対応づけを行う。この対応関係に着目し、以下の指標を考案した。

$$s = \frac{|X \text{ と } Y \text{ の要素が 1 対 1 に対応}|}{|X|} \quad (4)$$

この式 4 から得られる値を、本研究では構造を表す類似度と考え「構造的類似度」と呼ぶ。

この類似度は、 X および Y が一致すれば 1.0 になる。代表ベクトルの構成が異なることにより、1 対 1 の対応関係が崩れることで値は 1.0 より小さくなる。図 6 はこの代表ベクトルの違いにより、値が異なる状況を示したものである。

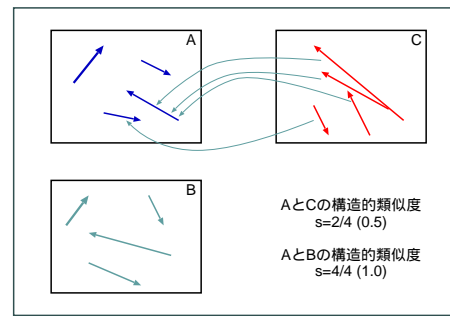


図 6: 異なる代表ベクトルの構造的類似度

5 実験

5.1 実験環境

実験は Linux 2.2.17、PentiumIII 550MHz で Memory 384MB の PC 上で行った。フラクタル画像圧縮は実験用プログラム Mars-1.0³を元に、提案手法を復号側にのみ実装した。

5.2 分類実験

提案手法の画像分類などへの有効性を確認するために、基礎的な実験を行った。CCD カメラにより撮影した、 320×240 pixel のモノクロ 256 階調の画像を、図 7~図 9 のような 3 つのカテゴリーにあらかじめ分類したものを実験対象とした。



図 7: 風景 (land)

このカテゴリーから 1 枚ずつ画像を取り出し、その圧縮符号から代表ベクトルを生成した結果が図 10 で

³<http://inls.ucsd.edu/y/Fractals/Mars-1.0.tar.gz>



図 8: 人物 (man)

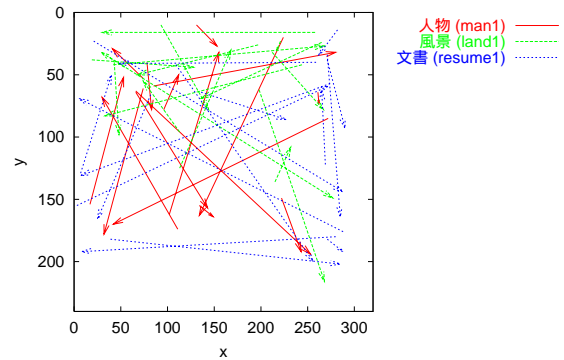
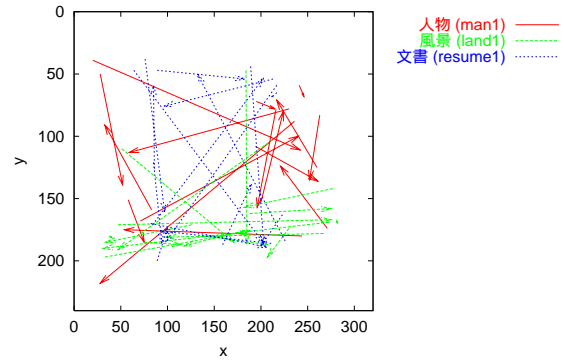


図 10: 各カテゴリーごとの代表ベクトルの違い。(上) レンジサイズ 4 (下) レンジサイズ 16

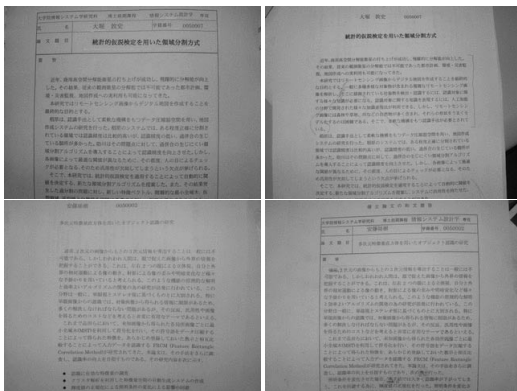


図 9: 文書 (resume)

ある。それぞれのカテゴリーは異なる画像の構図(構造)を持つ。代表ベクトルからも、それらの構造を反映した結果であると見ることができる。

風景から取り出した画像に対して、各画像ごとの類似度を算出したものが表 1 である。列の 4、8、16 はレンジサイズ毎に計算した構造的類似度を表し、それらの平均値も掲載している。この表の類似度の平均が高いものから順に挙げていくと、風景のカテゴリーが上位を占める。これは特定の画像を基準として、構造的類似度を用いた検索が可能であることがわかる。

それぞれのカテゴリーから画像を 1 枚ずつ取り出し、この計 3 枚の画像と他すべての画像との構造的類似度を求める。得られた構造的類似度を軸として、特徴空間を生成したものが図 11 である。この図から、あらかじめ分類を行ったカテゴリーに従ってクラスターを形成していることが確認できる。

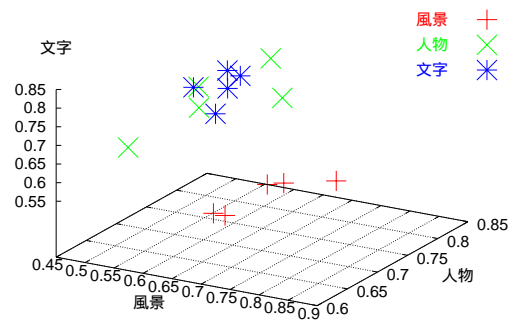


図 11: 風景、人物、文書から選んだ 1 枚と、各画像との構造的類似度を軸とした特徴空間

以上の結果から、構造的類似度による画像検索、および分類が行える可能性を見出すことができる。

表 1: 風景 (land1) に対する各画像の構造的類似度

| | 4 | 8 | 16 | 平均 |
|---------|------|------|------|------|
| land1 | 0.86 | 0.86 | 0.85 | 0.86 |
| land2 | 0.86 | 0.79 | 0.70 | 0.78 |
| land3 | 0.74 | 0.69 | 0.71 | 0.71 |
| land4 | 0.72 | 0.71 | 0.69 | 0.71 |
| land5 | 0.76 | 0.66 | 0.61 | 0.69 |
| man1 | 0.46 | 0.72 | 0.49 | 0.56 |
| man2 | 0.51 | 0.69 | 0.55 | 0.58 |
| man3 | 0.45 | 0.66 | 0.53 | 0.55 |
| man4 | 0.40 | 0.68 | 0.51 | 0.53 |
| man5 | 0.38 | 0.60 | 0.45 | 0.47 |
| resume1 | 0.50 | 0.70 | 0.62 | 0.61 |
| resume2 | 0.49 | 0.68 | 0.62 | 0.60 |
| resume3 | 0.50 | 0.74 | 0.57 | 0.60 |
| resume4 | 0.35 | 0.72 | 0.45 | 0.51 |
| resume5 | 0.45 | 0.66 | 0.59 | 0.57 |

6 まとめと今後の課題

フラクタル符号から直接写像ベクトル集合を取り出し、代表ベクトルを生成した。この代表ベクトルの画像間の対応関係により構造的類似度を求める手法を本報告では提案した。

フラクタル画像圧縮が生成した符号に対し、解凍(復号)を行うことなく、また複雑な処理を要せずに認識問題へ応用することが可能であることが示された。また実験から、本手法が画像の構造に基づく分類や検索などに適用できる可能性を見い出すことができた。

フラクタル画像圧縮が、その圧縮に多大なコストを払うが、そのコストに見合う認識能力を持っている可能性が、本研究を通して確認されたと言える。

本研究は基本的な処理を組み合わせた段階である。今後さまざまな個別の問題を解決していかなくてはならない。

特に画像間の比較に構造的類似度を定義して用いたが、この類似度を各ベクトル間の距離などの指標を取り入れるなどして発展させる必要があると考える。

参考文献

- [1] B.Wohlberg, G. J.: A Review of the Fractal Image Coding Literature, *IEEE Transactions on Image Processing*, Vol. 8, No. 12 (1999).
- [2] Ida, T. and Sambonsugi, Y.: Image segmentation using fractal coding, *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 5, No. 6, pp. 567–570 (1995).
- [3] 井田孝, 三本杉陽子, 渡邊敏明: LIFS を用いた被写体輪郭の高度な抽出, *電子情報通信学会論文誌*, Vol. J82-D-II, No. 8, pp. 1282–1289 (1999).
- [4] 横山貴紀, 渡辺俊典, 菅原研: フラクタル画像圧縮の復元作用に基づく画像分類について, *情処研報*, Vol. 2001, No. 87(CVIM-129-3), pp. 17–23 (2001).
- [5] T. Watanabe, K. S. and Sugihara, H.: A new pattern representation scheme using data compression, *IEEE Trans. PAMI*, Vol. 24, No. 5 (2002 to appear).
- [6] Barnsley, M. F.: *Fractals everywhere*, Academic Press, San Diego (1993, 1988).
- [7] Fisher, Y. ed.: *Fractal image compression: theory and application*, Springer-Verlag New York, Inc. (1995).
- [8] M.F. バーンズレイ, L.P. ハード: マルチメディアフラクタル画像圧縮, トッパン (1995).
- [9] 舟久保登: パターン認識, 共立出版 (1991).