

(M-7)

# フラクタル幾何学を用いたオフライン手書き文字認識\*

篠沢 佳久†

慶應義塾大学 インフォメーションテクノロジーセンター‡

大駒 誠一§

慶應義塾大学 理工学部 管理工学科¶

## 1 はじめに

本研究においては文字の自己相似性という特徴をオフライン手書き文字認識に利用することを試みる。文字を構成する線を局所的に見ると、他の線と類似した部分もある。これは文字が自己相似性という特徴を持っている可能性があるということであり、これを文字固有の特徴としてとらえることも可能である。本研究では文字の自己相似性という特徴を画像圧縮の分野で利用されているフラクタル変換を利用することによって抽出し、これを認識に利用することを提案する。

## 2 局所的な自己相似性の抽出

フラクタル幾何学ではすべての画像はそれ自身の一部份の縮小写像の集合(フラクタル写像)で近似できる。二値画像のフラクタル写像の求め方は以下の通りである。図1に示すようにまず画素数 $x \times y$ の画像 $I$ を一辺の大きさが $r$ の正方形 $R_i (i = 0, 1, \dots, M)$ に $M$ 個に分割する( $M = x/r \times y/r$ )。この正方形 $R_i$ をRangeブロックと呼ぶ。次にこのRangeブロック一つ一つに対して $P$ 倍の拡大および $\theta$ 度の回転(アフィン変換)を施した図形 $R'_i$ を考える。そしてこの図形 $R'_i$ と最も類似しているブロック $D_i$ を画像 $I$ の中から探し出す。このブロック $D_i$ をDomainブロックと呼ぶ。すなわち各Rangeブロック $R_i$ と対応するDomainブロック $D_i$ への変換 $F_i$ (拡大率 $P$ , 回転角 $\theta$ , Domainブロックの左上の座標 $d_i$ )をそれぞれ求める。この変換 $F_i$ を保存しておくことにより, Domainブロック $D_i$ からRangeブロック $R_i$ の復元はこの逆変換(フラクタル変換)によって可能である。

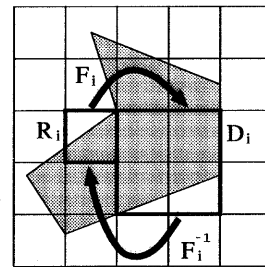


図1: フラクタル変換

このように任意の画像 $I_j$ が与えられた時, 上記で述べた変換の集合 $F_j^t = (F_{j1}, F_{j2}, \dots, F_j)$ をあらかじめ求めておく(フラクタル写像)。するとこのフラクタル写像の逆変換, すなわちフラクタル変換によって画像 $I_j$ の各Domainブロック $D_{ji}$ から対応するRangeブロック $R_{ji}$ を復元して合成すると, 元の画像 $I_j$ を得ることができる。一方で画像 $I_k (k \neq j)$ に対して画像 $I_j$ のフラクタル写像 $F_j$ を利用し, フラクタル変換を施しても元の画像 $I_k$ を得ることはできない。この性質を認識に利用する。

## 3 自己相似性を利用した認識

文字の局所的な自己相似性を利用した文字パターン認識方法は以下の通りである。

1. 学習文字パターンとしてあらかじめ認識対象となる二値の文字画像 $I_j (j = 1, 2, \dots, N)$ からそれぞれフラクタル写像 $F_j$ を求めておき, データベースに保存しておく。
2. 未知の文字パターン $U$ を認識する場合は, 求めておいたフラクタル写像 $F_j$ によって文字パターン $U$ にフラクタル変換を施し, 新しい画像 $U'_j$ を生成する。この場合認識対象数の $N$ 回行ない,  $N$ 個の画像 $U'_j$ を新たに生成する(図2)。

\*The Offline Handwritten Character Recognition by Fractal Geometry.

†Yoshihisa SHINOZAWA

‡Information Technology Center, Keio University

§Seichi OKOMA

¶Faculty of Science and Technology, Keio University

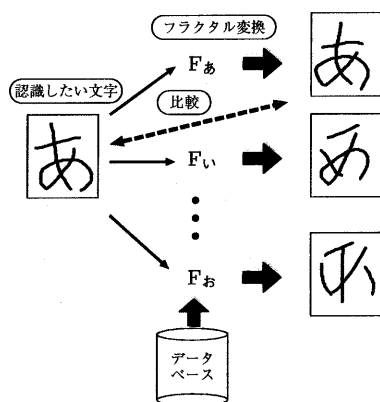


図 2: 自己相似性を利用した認識

- 次にフラクタル変換前の画像  $U$  と変換後の  $N$  個の画像  $U'_j$  との相違を一つ一つ比較し、その差が最も小さくなる変換を施したフラクタル写像  $F_k$  を求め、文字種  $k$  を認識結果とする。

#### 4 認識実験

認識実験のため文字パターンについては通産省工業技術院電子技術総合研究所提供の ETL9B の最初の 100 パターンを利用した。学習文字として最初の 40 セットからフラクタル写像を抽出し、未学習文字として次の 40 セットを認識対象とした。認識方法は以下の通りである。まず最初に  $64 \times 63$  の文字画像に対し、非線形正規化処理を施し  $64 \times 64$  の大きさにし、196 次元の輪郭線特徴を抽出する。そしてこの特徴を利用してユークリッド距離による大分類を行ない、文字候補を第 10 候補まで絞り込む。次にフラクタル変換によって抽出した自己相似性を利用して認識を行ない、最終的な候補を一つに絞り込む。しかし 3 節で述べた認識手法をそのまま適用しても文字を認識することは困難である。そこでいくつかの改良を施す。

- 前処理として非線形処理化により文字画素の密度を一定にし、文字線を太くした上でフラクタル写像はこの前処理により  $72 \times 72$  の大きさとした文字画像から抽出する。
- 抽出するフラクタル写像において Domain ブロックは Range ブロックの大きさの 2 倍と固定し、変換は回転角を考慮せず恒等写像のみとする。
- 任意の Range ブロックにおいてそれと対応する Domain ブロックを求める際、その変換によって大分類で絞った他の文字候補の Range ブロックと同一の Domain ブロックが一致しない変換のみを保存し、認識に利用する。

- 上記の条件を満たすフラクタル写像全てを保存しておく。そして認識時においては複数個の中からランダムに一つ選び、変換に使用する。
- 認識時において一字種につき 40 個抽出されるフラクタル写像をそれぞれ別々に使用し、別々の文字画像を生成し比較する。
- 大きさの異なる Range ブロック ( $3 \times 3$ ,  $4 \times 4$ ,  $6 \times 6$ ,  $1 \times 8$  など) からのフラクタル写像をいくつか求めておく。認識時においてはこれらのフラクタル写像を別々に利用して復元を行ない、最後に一つの文字画像に合成する。

上記の条件に基づいて Range ブロックの大きさやその組合せを変えた結果を表 1 に示す。表 1 に提示した認識率は未学習文字についてであるが、学習文字については復元が可能なので、大分類で誤分類をしない限り誤ることはない。表 1 より最良の結果である  $1 \times 8$  と  $8 \times 1$  の組合せで認識対象を ETL9B の最初の 1000 文字として、候補数を変化させながら同様の実験を行った結果を表 2 に示す。

表 1: Range ブロックの大きさを変えた時の認識率

Range ブロックの 大きさとその組合せ	特徴量	認識率 (%)
重み付けユークリッド距離	196	92.33
$3 \times 3$	576	89.58
$4 \times 4$	324	88.93
$6 \times 6$	144	87.50
$1 \times 8$	648	91.98
$4 \times 4, 6 \times 6$	468	90.08
$3 \times 3, 4 \times 4$	900	92.03
$1 \times 8, 8 \times 1$	1296	92.90

表 2: ETL9B による認識実験

候補数	認識率 (%)
重み付けユークリッド距離	92.40
2	92.96
3	92.88
4	92.76
5	92.72

#### 5 まとめ

以上の実験結果より文字の局所的な自己相似性をフラクタル変換によって抽出し、これを特徴として利用したオフライン手書き文字の認識は可能であることを示した。現在のところデータベース、計算量の巨大化から、大規模な文字認識に適用することは困難である。今後はこうした問題点を解決すると共に、より認識率の向上が可能なフラクタル写像の抽出などの改良を試みていく予定である。