

手書き文字の筆順復元

1 P-4

加藤美治*

保原信*

電気通信大学 大学院 情報システム学研究所

1 はじめに タブレットなどの入力デバイスを用いるオンライン手書き文字認識と比べると、スキャナーなどから入力された二次元の静止画像の情報だけを用いなければならないオフライン手書き文字認識には、より困難さが伴う。しかし、もし画像データから筆順を復元することができれば、オフライン文字認識問題を時間情報を利用するオンライン文字認識問題に帰着させることが可能になる。そこで、静止画像から筆順を復元するために、濃淡画像上のストロークの特徴を解析する方法 [1]、ストロークの方向、長さや幅などの特徴を考慮した連続性に基づく方法 [2]、大局的な滑らかさを評価する方法 [3]、巡回セールスマン問題を解くことによって最小の曲率となるハミルトン道を求める方法 [4] などが試みられている。本稿では交点の次数と最短経路の探索に基づく方法を報告する。

2 筆順の復元問題 筆順を復元するにあたっては、(1) 一筆書きで書かれていること、(2) 始点と終点が存在し、互いに異なり、かつ指定されること、(3) 3本以上の線分が交差する交点が存在しないこと、(4) 2本以上の線分が接触するだけで交差していない交点が存在しないことを前提とする。書かれた文字をグラフと見なすと、頂点の集合を $V = \{v_1, \dots, v_k\}$ 、辺の集合を $E = \{e_1, \dots, e_l\}$ としたとき、連結な無向グラフ $G = (V, E)$ が始点 v_1 から出発して各辺を各々一度だけ通って終点 v_k まで一筆書き可能であるためには、 $degree(v_1) = 1$ 、 $1 < i < k$ に対して $degree(v_i) \bmod 2 \equiv 0$ 、および $degree(v_k) = 1$ であることが必要である (Euler の一筆書きの定理)。そこで、ここでは各々の交点の次数に着目することにする。また、「h」という文字などでは

ある辺が二度書きされることが必要なので、このような辺を「ダブル・トレースされた線分」あるいは「D-line (Double-traced line)」と呼んで二度書きを認めることにする。

3 筆順の復元方法 筆順の復元は入力画像データを細線化して得られる1ピクセル幅の中心線を始点から終点まで追跡することによっておこなう。そのためにはまず、細線化された画像を3x3ピクセルのマスクによって走査し、次数が0と2以外のピクセルをノード(節点)として検出して、隣接行列Aを作成する。その後、A上で始点から追跡を開始して終点に達するまで、各ノードの次数に応じて表1に示されている処理を行う。

表 1: Trace algorithm

| DEG | EXAMPLE | NODE TYPE | ACTION |
|-----|---------|--------------|-------------------------------|
| 0 | | Isolated | Ignore |
| 1 | | Terminal | ① Terminate ② Double-trace |
| 2 | | Line | Ignore |
| 3 | | Branch | S-Line |
| 4 | | Intersection | Select middle path |

本来、次数3のノードは存在しないはずであるが、入力文字のデジタル化や細線化によって図1の $\overline{B_1 B_2}$ のような線分が発生することがあるので、このような線分を「S-line (Spurious-line)」と呼ぶことにする。S-lineは次数3のノード間に存在し、本来は連続している線分を分断して筆順の追跡を難しくする。そこで、S-lineに対処するために、隣接する次数3のノードを図1に示すクラスタリング処理によってひとつの次数4のノードに統合し、隣接行列Aを更新する。

*Stroke Restoration from Handwriting Characters
Yoshiharu Kato and Makoto Yasuhara
Graduate School of Information Systems, The University of
Electro-Communications
1-5-1 Chofugaoka, Chofu, Tokyo 182-0021 Japan

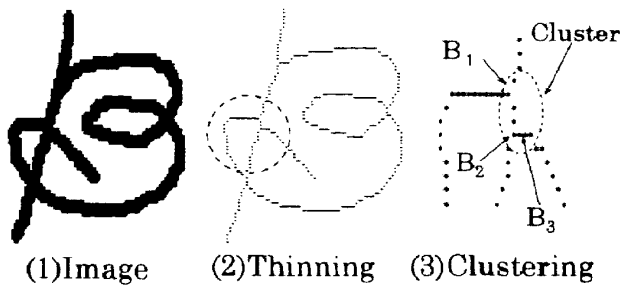


図 1: Clustering and S-Line handling

なお、D-line は、表 2 のように分類できるが、SD-line は長い S-line として扱うことができ、PD-line は前提条件 (4) を満足していないので、本稿においては最も出現頻度の高い、ループが縮退してできる LD-line を処理対象とする。

表 2: Double-trace types

| TO FROM | DEGREE 1 | DEGREE 3 |
|------------|-------------|---------------------|
| DEGREE 1 | | LD-Line |
| DEGREE 3 | LD-Line | SD-Line PD-Line |

次数 1 のノードから最短経路で到達できる次数 3 のノードを求めることによって LD-line を検出し、この最短経路を二重化し該当するノードの次数を変更する。この処理を繰り返して始点と終点以外の全ノードの次数が偶数になれば、隣接行列 A に従って筆順を追跡することによって与えられた文字を一筆書きすることができる。図 2 に例を示す。ノード N_7 から最短経路で到達できる次数 3 のノード N_2 への最短経路 $N_7N_6N_4N_2$ が LD-line として検出される。この LD-line を二重化すると、 N_1 と N_5 以外のすべてのノードの次数が偶数になる。

4 実験結果および考察 筆順復元のアルゴリズムを Java アプレットとして実装して実験を行った結果を図 3 に示す。入力された画像、あるいはマウスやタブレットによって書かれた文字から筆順が復元され、D-line は色によって区別され、筆順は鉛筆のアニメーションによって示される。実験の結果、D-line を

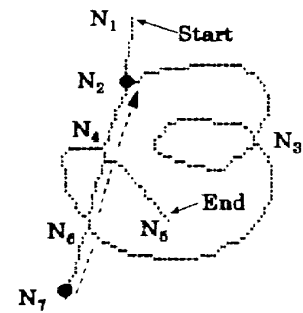


図 2: Double-trace detection

含む複雑な文字からも正しく筆順を復元できることが示された。本手法は、線分が交差する角度や曲線の滑らかさ等の算出が不要であるために正確で安定な復元が可能であり、交点数が増えると試すべき組み合わせの数も増加するといった問題がないので複雑な文字にも対応可能である。

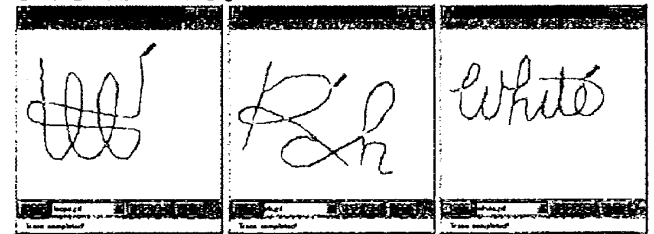


図 3: Examples

5 まとめ 静止画像上の手書き文字の筆順を復元する手法について報告した。ダブル・トレースを含む一筆書きの文字を、交点の次数を考慮して追跡することにより、複雑な文字であっても組み合わせ問題を解くことなく筆順を復元できることを示した。

参考文献

- [1] D.S. Doermann and A. Rosenfeld, "Recovery of Temporal Information from Static Images of Handwriting", In *Proc. IEEE Comput. Soc. Conf. on Computer Vision and Pattern Recognition*, pp.162-168, 1992.
- [2] G. Boccignone, A. Chianese, L.P. Cordella and A. Marcelli, "Recovering Dynamic Information From Static Handwriting", *Pattern Recognition*, Vol. 26, No. 3, pp.409-418, 1993.
- [3] T. Huang and M. Yasuhara, "Recovery of Information on the Drawing Order of Single-Stroke Cursive Handwritten Characters from Their 2D Images", *IPSJ Trans.*, Vol. 36, No. 9, pp.2132-2143, Sep. 1995.
- [4] S. Jäger, "Recovering Writing Traces in Off-Line Handwriting Recognition: Using a Global Optimization Technique", In *Proc. ICPR '96*, pp.150-154, 1996.