

運筆状態規則を用いた 平仮名毛筆文字生成について†

真野 淳治*¹ 江野脇 宏*²
中村 剛士*² 何 立風*³ 伊藤 英則*²

本論文では、ユーザ個人の感性(個性)を反映した平仮名毛筆文字を生成する方法を提案する。本提案方法では、ユーザ独自の情報として、ユーザが動かす電子ペンから得られる座標、筆圧、筆速を用い、これらを入力情報として毛筆書体を生成する。まず、描画入力により得られたサンプリング点列を毛筆書体へ変換するための前処理として、与えられたサンプリング点列から適当な数の制御点を選択し、なめらかな曲線形状になるよう B スプライン曲線により補間処理する。このとき、制御点の選択方法として、ファジィスプライン曲線を用い、サンプリング点列の形状を反映しつつ、かつ少ない制御点でなめらかに補間できるような制御点選択方法を提案する。また、実際の書道における運筆状態に着目し、電子ペンにより描画入力した文字を分割し、各運筆状態に割り当てる方法を提案する。これらの方法を毛筆文字出力システムとしてワークステーション上に実装し、システムにより試作した平仮名文字を示すことにより本方法の有効性を示した。

キーワード：平仮名毛筆文字生成、ペン入力、ファジィスプライン曲線、運筆状態規則

1. はじめに

近年のパーソナルコンピュータの普及によりその利用形態も多様化してきており、特に手紙や挨拶状、パンフレット等の作成に利用されることが多くなってきている。同時にこれらの作成に用いられる文字として毛筆文字に対する要求が強くなっている。そのため、様々な毛筆フォントが作成され利用できるようになっている。

しかし、現在これらの毛筆フォントは単なるグラフィックパターンで構成されているものが多く、ユーザの好みに合わせて文字の大小は変形できるが、文字の形自体は変形することができない。

また、毛筆文字は書道という芸術の一表現であり、単なる文字情報としてだけでなく画像情報としての側面も併せ持っている。しかし、現在ある毛筆フォントのほとんどは一文字単位で作成されており、文章全体の構成でユーザの感性を表現することは困難である。

画像情報に関連した感性情報処理の研究は少なくない

いが、毛筆文字等のフォント設計における感性情報処理の研究はほとんど見られない。毛筆フォント設計において感性情報処理に近い処理を行なっているものとしては張らの研究[1,2,3,4]がある。これらの研究では個人の嗜好や個性に応じて各種パラメータを自由に設定可能であり、それに応じて毛筆文字の形状を変化させる方法を確立している。感性とは各個人単位で見た場合、個人の嗜好または個性であることから、フォント設計において感性情報を処理または表現した研究の一例であるということが出来る。しかし、そのパラメータの設定方法は専門的な知識を必要とし、かつ複雑な設定方法であるため、ユーザの感性を文字形状に反映させることは容易ではない。

我々はこれまで、比較的容易かつ直感的な方法により、各ユーザ個人の感性(個性)を反映した毛筆文字の生成方法を提案してきた[5,6,7,8,9]。これらの研究では感性情報として、位置情報や筆圧、筆速などをマウスや電子ペンから入力し、これらの情報から毛筆文字の形状や掠れや滲みの表現を実現している。しかし、これらの研究では対象となる文字は楷書漢字のみに限定されていた。

そこで本稿では、ユーザの入力に応じて平仮名のよみを持つ文字やストローク間が繋がった連筆文字も生成可能な方法を提案する。張らは、文献[4]において平仮名生成の方法を提案しているが、連筆の箇所に対してユーザがパラメータの設定を行なわなければならない。上に述べたように、ユーザの感性を文字形状に反映するためには専門的知識を必要とする。それに対し、本方法では、ユーザは複雑なパラメータを設定

† Generation of Hiragana Brush-Style Characters Using Pen-Moving Rules

Junji MANO, Hiroshi ENOWAKI, Tsuyoshi NAKAMURA, Lifeng HE and Hidenori ITOH

*¹ 日本電信電話株式会社
NTT

*² 名古屋工業大学 知能情報システム学科
Department of Intelligence and Computer Science,
Nagoya Institute of Technology

*³ 愛知県立大学 情報科学部
Faculty of Information Science and Technology, Aichi
Prefectural University

する必要はなく、電子ペンにより文字を入力するだけでよい。電子ペンによる入力、座標データ等を数値で直接入力したり、マウスで入力するよりも実際の毛筆の動きを比較的再現しやすく、本方法では、起筆や収筆の筆運びの細かな相違を文字形状に反映することができる。

また、電子ペンにより描画入力した文字をなめらかな毛筆文字に変換するための前処理としてファジィスプライン補間[10,11]を用いた手書き入力文字の補間法を提案する。

2. システムの概要

本システムは入力部、文字補間部、文字分割部、文字変換部、擦れ・滲み生成部、出力部の6つの部分から構成される(図1参照)。また、入力装置としてタブレットと筆圧が測定可能な電子ペンを用いた。また、電子ペンはタブレットから一定の高さ(読み取り可能高さ)内であれば座標や筆圧が測定可能である。

従来のシステム[5,6,7,8,9]では1ストローク毎に入力、認識、変換、出力の処理を繰り返して毛筆漢字を生成していた。しかし、この方法では、ユーザが連筆文字を描画しようとしても、ストローク間の情報を使用しないため、ストローク間の繋がりを反映しにくい。そこで本システムでは、“一筆”毎に毛筆書体への変換を行なう。ただし、一筆とは文字を書き始めた後、電子ペンがタブレットから離れ、一定時間内に次の座標及び筆圧の情報が送られてこない時点までを指す。これにより、一旦、電子ペンがタブレットの読み取り可能高さ外に出ても続けて文字を書くことにより、ユーザは容易に連筆文字を描画入力することができる。

この後、文字補間部における手書き文字の補間方法と文字分割部、文字変換部における毛筆文字への変換方法についてそれぞれ詳しく述べる。

3. 電子ペンによる手書き文字の補間法

本節では、ユーザの描画入力により得られたサンプリング点列をその形状を可能な限り損なわず、かつなめらかな曲線として補間する方法を提案する。

曲線補間を行なう方法は大きく次の2つに分けることができる。

- (1) 補間曲線は与えられたすべての制御点を必ず通る。
- (2) 補間曲線は与えられた制御点を必ずしも通らない。

電子ペンによる手書き文字を補間する場合を考えると、実際に手書きされた文字とユーザの意図した文字の間にはある種の「ずれ」がある[10]。また、座標拾得時のタイミングの遅延による「誤差」が生じる可能

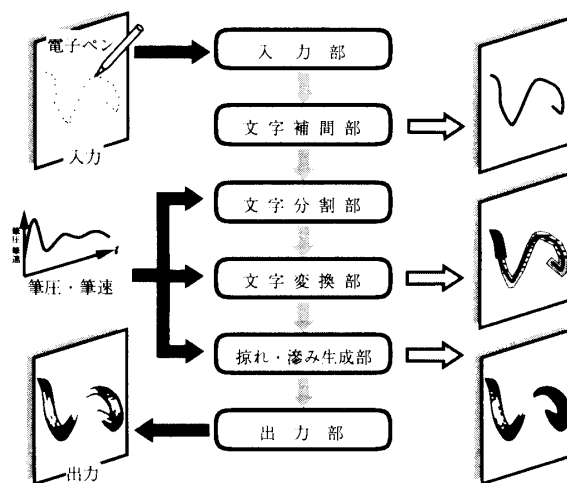


図1：システム構成

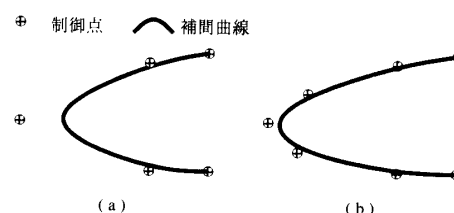


図2：曲線的な入力における制御点と補間曲線の関係

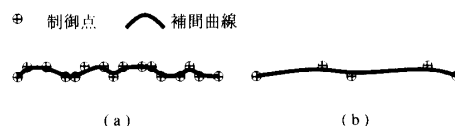


図3：直線的な入力における制御点と補間曲線の関係

性もある。そのためサンプリング点の位置情報には曖昧さが含まれており、サンプリング点上を通る(1)の方法より(2)の方法の方が適しているといえる。

(2)の方法として良く用いられるものにBスプライン補間法があるが、手書き文字を補間する場合以下のような問題点が生じる。まずなめらかな曲線によって補間しようとする場合、与えられる制御点は少ない方がよい。しかし、図2(a)のように少ない制御点で曲線的な入力を補間すると補間曲線は制御点から離れてしまう。これを防ぐためには図2(b)のように制御点を多くし、制御点間の距離を短くする必要がある。

電子ペンによる入力を行なう場合、図2(a)のような問題を防ぐためには、筆記速度が速い場合においても制御点間の距離がある程度短くなるよう、座標データのサンプリング時間間隔を短くする必要がある。しかしその場合、筆記速度が遅くなると制御点間の距離は

極端に狭くなり、図3(a)に示すように直線的な入力に対してユーザが意図した補間結果が得られなくなり、図3(b)のように制御点の選択が必要となる。

そこで本稿では、ファジィスプライン補間法[10,11]を用いてサンプリングされた入力座標点に曖昧さを持たせ、補間された曲線がその曖昧さの範囲を越えないように制御点の選択を行なう。

3.1 ファジィスプライン補間法(FSI)

FSIは、スプライン補間法(SI)をファジィ理論に基づいて拡張したものであり、曖昧な位置情報を有する点列を補間する曖昧な曲線モデルを構成するものである[10]. FSIでは、一般のスプライン補間で用いられる制御点列 $d_i (i=1, 2, \dots, n)$ の代わりにファジィ制御点列 $\tilde{d}_i (i=1, 2, \dots, n)$ が用いられる。ファジィ制御点 \tilde{d}_i は円錐型ファジィ点で与えられ、円錐頂点座標 d_i (SIにおける制御点)と円錐底円半径 r_i を用いて以下のように表される。

$$\tilde{d}_i = \langle \tilde{d}_i, r_i \rangle \quad (1)$$

図4にファジィスプライン補間の一例[10]を示す。(a)のファジィ制御点列から(b)のファジィスプライン曲線を求めるためには、円錐頂点座標のx軸要素、y軸要素、及び円錐底円要素のそれぞれに対して独立に通常のスプライン曲線を求めれば良い。

3.2 FSIを用いた手書き文字の補間

本システムでは、ユーザの入力によって得られたサンプリング点列に曖昧さを与え、これらすべてをファジィ制御点列としてファジィスプライン曲線を求める。次にサンプリング点列からいくつかを制御点として選択し、通常のスプライン曲線により一本の補間曲線

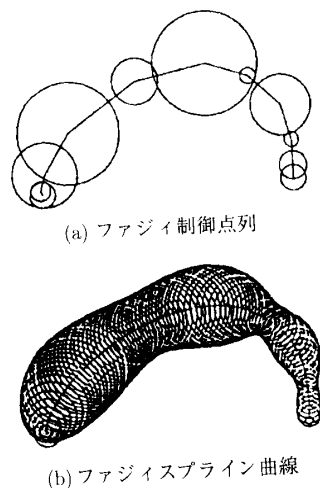


図4：ファジィスプライン補間の例[10]

を求める。このとき求められたBスプライン曲線がファジィスプライン曲線に内包されるように制御点を選択する。図4に示すように、ファジィスプライン曲線は、円錐底面の集合として表現される。ここでは、補間曲線全体が、この円錐底面集合内に位置することを内包と定義する。

3.2.1 ファジィ制御点

文献[10]では、サンプリング点の曖昧さとして円錐型メンバシップ関数を与えているが、本システムでは処理を簡単にするため、これを円で与える。このときファジィ制御点 \tilde{p}_i は、サンプリング点の座標ベクトル p_i と p_i を中心としたファジィ円の半径 r_i を用いて式1と同様に表される。

$$\tilde{p}_i = \langle p_i, r_i \rangle \quad (2)$$

とファジィ円の半径 r_i はその点における加速度 a_i を用いて以下のように決定する。

$$r_i = j \cdot f(|a_i|) + k \quad (j, k \text{ は定数}) \quad (3)$$

なお、サンプリング点と補間された曲線との間が開きすぎないようにするため、本システムでは関数 $f(x)$ を図5で定義し、曖昧さの度合すなわちファジィ円の半径 r_i に上限を設定した。

3.2.2 制御点の選択

本システムでは補間曲線として3次Bスプライン曲線(BSC)を使用しており、この場合における制御点の選択アルゴリズムを図6に、そのアルゴリズムで使用する変数と記号を表1にそれぞれ示す。

本アルゴリズムではまず、サンプリング点列の始点 p_0 と終点 p_N を結んだ直線がファジィスプライン曲線(FSC)に内包されるかどうかを調べ、内包されている場合はその直線を補間曲線とする(10-30行)。

制御点の選択は基本的に120-240行で行なわれるが、制御点 c_j の追加によるBスプライン曲線の確定部分(図7参照)は3次Bスプライン曲線の場合、 $j \geq 3$ でないと出現しない。そこで、 c_0, c_1 として p_0 を選択し(40, 50行)、 c_2 は、可能な限り c_1 から離れた点を選択するものとし、直線 $c_0 p_i$ がファジィスプライン曲線に内包され、かつ最長となる点 p_i を選択した(70-100行)。

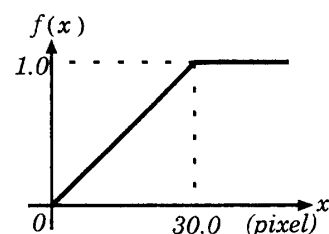


図5：上限関数 $f(x)$

```

10  if (直線  $p_0p_N \in FS(p_0, p_N)$ ){
20    補間曲線として直線  $p_0p_N$  を返す;
30  }
40   $c_0 = p_0$ ;
50   $c_1 = p_0$ ;
60   $i = 1$ ;
70  while (直線  $c_0p_i \in FS(c_0, p_i)$ ){
80     $i = i + 1$ ;
90  }
100  $c_2 = p_{i-1}$ ;
110  $j = 3$ ;
120 while (not 制御点の選択終了){
130   $c_j = p_i$ ;
140  if ( $s_j \in FS(c_{j-3}, c_j)$ ){
150     $i = i + 1$ ;
160  } else {
170     $c_j = p_{i-1}$ ;
180     $j = j + 1$ ;
190    if ( $s_{tail} \in FS(c_{j-3}, p_N)$ ){
200       $c_j = p_N$ ;
210      制御点の選択終了;
220    }
230  }
240 }
250 補間曲線として  $\sum_{k=0}^{j-1} s_k + s_{tail}$  を返す;

```

図6：制御点選択アルゴリズム

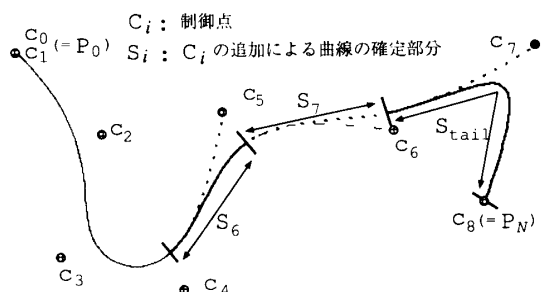


図7：制御点の追加による B スプライン曲線の確定

3.2.3 結果と評価

図8,9に曲線と直線を意図して入力した場合の結果をそれぞれ示す。それぞれの図において(a)は入力により得られたサンプリング点列に曖昧さを与えたファジィ制御点列, (b)はそれにより求められたファジィスプライン曲線, (c)は図6のアルゴリズムによって選択された制御点とその補間曲線である。また, 制御点の選択を行わずにサンプリング点すべてを制御点として用いて補間した結果を(d)に示す。また, その他の入力におけるファジィスプライン曲線や選択された制御点とその補間結果を図10-12に示す。なお, 図10,11は, 図8,9よりも筆記速度を速くしたものである。これらの場合におけるサンプリング点数(S)や選択された

表1：制御点選択アルゴリズムに使用する変数と記号

変数名	意味
p_i	サンプリング点 ($i=0, 1, \dots, N$)
c_j	選択された制御点 ($j=0, 1, \dots, M$)
$FS(p_i, p_j)$	点 p_i, p_j 間の FSC
s_j	c_j の追加による BSC の確定部分
s_{tail}	現時点までの BSC 確定部分の終点から p_N までの BSC 部分
記号	意味
$s \in FS$	s が FS に内包される



図8：出力例1：筆記速度の遅い曲線

表2：サンプリング点数 S と制御点数 C

出力例	S (総移動距離)	C	C / S (%)
図8	68 (692 pixel)	14	20.59
図9	43 (481 pixel)	4	9.30
図10	54 (1351 pixel)	17	31.48
図11	40 (884 pixel)	7	17.5
図12	49 (652 pixel)	7	14.29

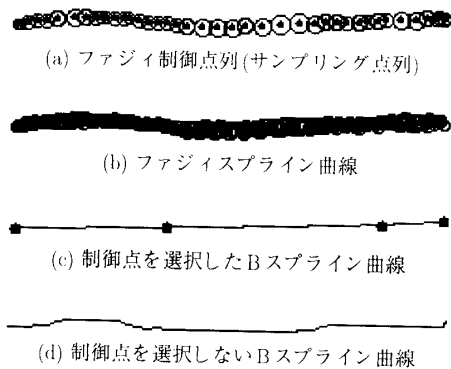


図9：出力例2：筆記速度の遅い直線

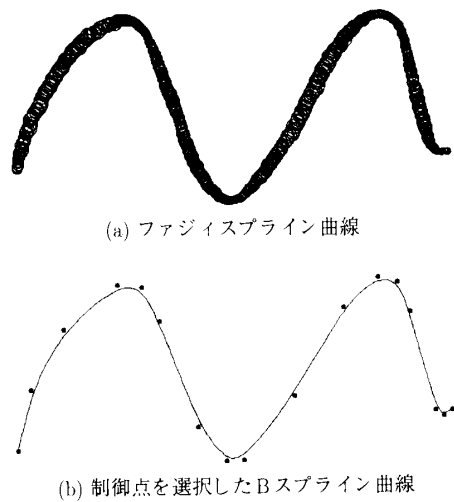


図10：出力例3：筆記速度の速い曲線

制御点数(C)等を表2に示す。なお、本アルゴリズムでは、制御点 c_0, c_1 として同一なサンプリング点 p_0 を選択しており、表中の制御点数はこれらを1点として数えている。

直線の入力時(図9, 11)より曲線の入力時(図8, 10)の方が制御点を選択される割合が高く、また入力筆記速度が遅い場合(図8, 9)は速い場合(図10, 11)より制御点を選択される割合が低くなっており、図2, 図3で示した問題を回避している。図12のように折筆を持つ入力に対しては、その折筆付近のサンプリング点を制御点として多く選択している。

4. 運筆状態遷移規則による連筆文字への変換法

ここでは、補間された入力文字をその筆速や筆圧を用いて毛筆文字に変換する方法を提案する。従来のシ

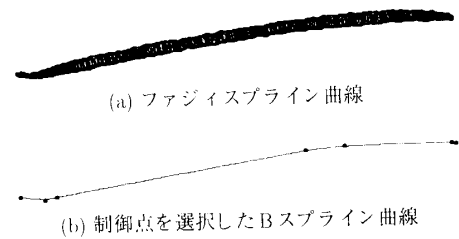


図11：出力例4：筆記速度の速い直線

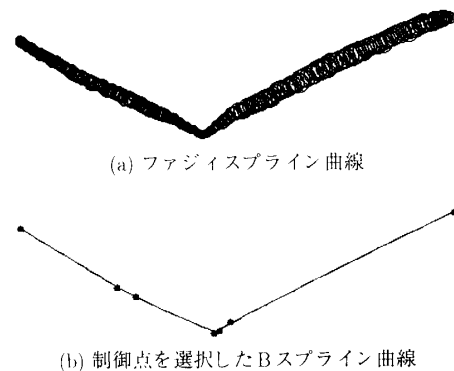


図12：出力例5：折筆

ステム[5,6,7,8,9]では、入力文字をストローク毎に認識し、認識されたストロークの種類に応じてデータベースから基本となる毛筆ストロークを選択し、位置や筆圧などの入力情報に応じてそれを変形させることで毛筆文字を作成していた。そのため、ストローク間が繋がった連筆文字については対応出来なかった。またデータベースのストロークは楷書漢字にしか対応しておらず、平仮名のような丸みを持った文字やストロークが変形した文字の変換には不向きであった。

本システムではこれらの問題を解決するため、電子ペンの動かし方(運筆)に着目し、運筆状態毎に文字を分割し、毛筆文字への変換を行なう。以下に文字の分割方法と毛筆文字への変換方法についてそれぞれ説明する。

4.1 運筆状態遷移規則による文字の分割

毛筆文字はその運筆の状態に応じて起筆部、送筆部、折筆部、収筆部、連筆部の5種類の状態に分割することができる。連筆部とはストローク間の繋がり、すなわちストロークの終端部分(収筆部)と次のストロークの先端部分(起筆部)が繋がった部分を指す。

入力文字をこの5種類の状態に分割するため、運筆状

態の移り変わりに着目し、図13に示す運筆状態遷移規則を作成・利用した。文字の分割は一筆単位で行ない、座標、筆速、筆圧の入力情報を用いて初期状態である起筆部からの状態の遷移地点と経路を決定することにより実現される。なお、本システムでは、収筆部の状態をさらに「とめ」、「はらい」、「はね」の3状態に分割している。

4.2 毛筆文字への変換

毛筆書体への変換は以下のように毛筆書体の輪郭線を求めることで行なわれる。

1. 運筆状態遷移規則により分割された区間ごとに輪郭点を計算。
2. 一筆分の輪郭点を計算した時点で毛筆書体の輪郭線を生成。

なお、本システムでは輪郭点から輪郭線を生成する際の曲線補間として通常の3次Bスプライン補間を用いた。次に各状態の輪郭点の計算方法を述べる。

4.2.1 起筆部・折筆部・収筆部

起筆部では、図14(a)に示すように、起筆部内の補間曲線上に一定間隔で、3個の骨格点を設定する。次に各骨格点から数個ずつ輪郭点を生成する。この時、輪郭点は骨格点からの相対位置で求められ、システムは相対位置を決定するための相対ベクトルを返す関数をデータベースに持つ。本システムではこの関数の引数として骨格点 c_i の座標データと筆圧データ、さらに隣接する骨格点 c_{i-1}, c_{i+1} の座標データを与えることにより、ユーザの入力を毛筆書体に反映させている。

折筆部、収筆部も起筆部と同様に補間曲線上に2個又は3個の骨格点を配置し、データベースにある関数により輪郭点を求める。

4.2.2 送筆部

送筆部では、図14(b)に示すように補間曲線上に一定間隔で骨格点を取り、その骨格点から進行方向に垂直な方向に輪郭点を配置する。この時の骨格点から輪郭点までの距離は骨格点上の筆圧によって決定される。

4.2.3 連筆部

本システムでは、一旦ペンを離れた後もタブレットの読み取り可能高さ内にあり、座標及び筆圧データが送られてくる間、または一旦読み取り可能高さ外に出ても一定時間内に再び読み取り可能高さ内に戻る場合は、ユーザが引き続き描画中であるものとして判断する。描画中に再びペスが着いた場合、これを連筆部として扱い毛筆文字に反映させる。

図13で示した運筆状態遷移規則から分かるように連筆部は必ず収筆部から起筆部に移行する間に現れる。

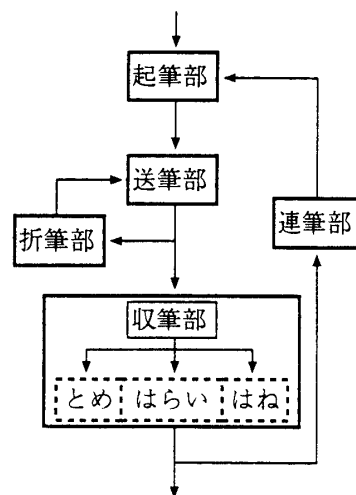


図13：運筆状態遷移規則

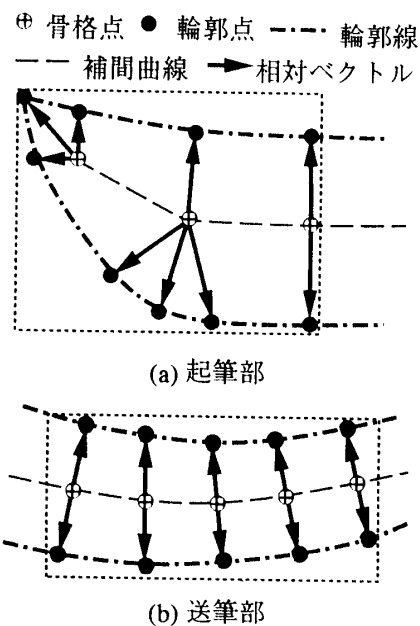
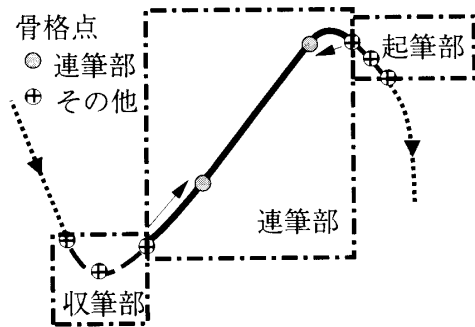


図14：骨格点と輪郭点

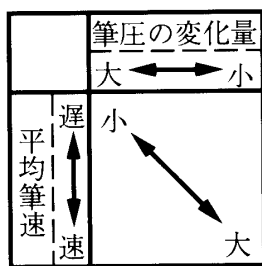
連筆部では、収筆部側と起筆部側にそれぞれ骨格点を1点のみ取る(図15(a)参照)。また、骨格点の位置は、ペスが離れた(又は着いた)付近の平均筆速と筆圧の変化量の絶対値により連筆度合(図15(b)参照)を求め、その度合が大きいほど収筆部(又は起筆部)から離れた地点に配置する。なお、連筆度合は0.0~1.0の範囲を取るものとする。この骨格点の位置に応じて収筆形状(又は起筆形状)が変化するように収筆部(又は起筆部)と同様に輪郭点を配置する。

図16は、収筆部側及び起筆部側の連筆度合を筆圧の変化量、平均筆速に応じて変化させず、一定値を設定

することにより、連筆度合の変化による出力文字への影響を調べたものである。これにより連筆度合が大きくなるにつれて、収筆部と起筆部の繋がりが大きくなっていくのが分かる。なお、図16(d)は図15(b)により連筆度合を決定した結果である。



(a) 骨格点



(b) 連筆度合

図15：連筆部の骨格点と連筆度合

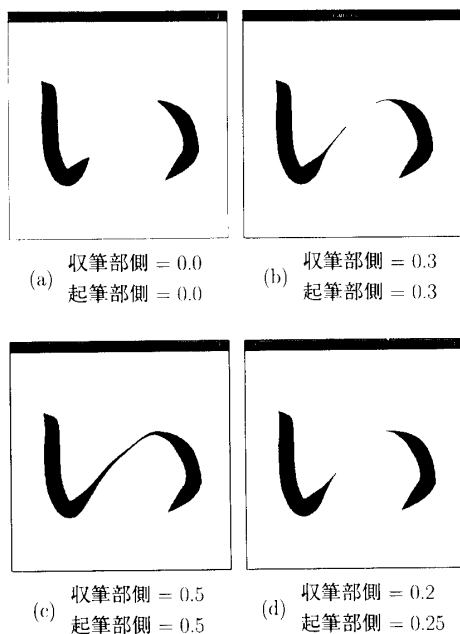


図16：出力文字に対する連筆度合の値とその影響

5. 毛筆文字出力例

本システムにおける毛筆文字出力例を図17に示す。なお、この出力例は掠れ・滲み生成部において文献9で提案しているくりこみ方式を用いて掠れや滲みを付加したものである。

6. おわりに

本稿では、電子ペンによる入力から得られた筆速、筆圧等のユーザ個人の感性情報(個性)を反映した毛筆文字を生成する方法を提案した。この方法では、まず、電子ペンを用いた描画入力により得られたサンプリング点列をなめらかな曲線になるよう補間する。次に毛筆文字の特徴を毛筆の動きの状態により分類した連筆状態遷移規則を用い、補間された入力曲線をその状態毎に分割し、毛筆文字に変換する。また、これらの方法を実装した毛筆文字出力システムを作成してその出力例を示すことにより提案方法の有効性を示した。

本方法を用いることにより、平仮名のような丸みを持った文字やストローク間が繋がった連筆文字の作成

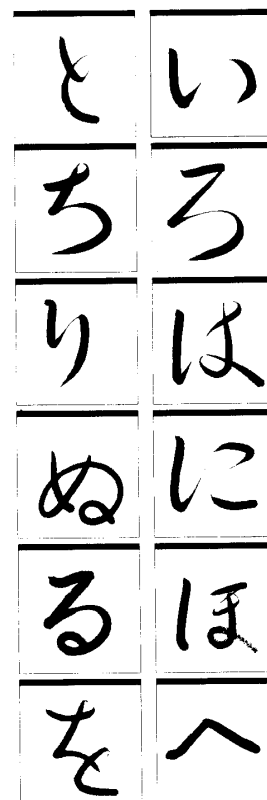


図17：毛筆文字出力例

にほぼ対応できるようになった。また、電子ペンの動きにより入力曲線を分割するので、基本的にはどのような入力曲線に対しても毛筆で書いたように変換できる。

本稿で述べた方法では、分割された入力文字を毛筆文字に変換するときに、毛筆文字の輪郭を形成するための点を入力曲線からの相対位置で決定する。この相対位置を決定するための情報についてはデータベースとして事前に与えなければならないが、現在はこれを手作業で作成している。しかし、毛筆の書体は筆の種類や書道技法などにより様々に変化し、手作業でデータを作成するには非常にコストがかかる。そこで実際の毛筆文字や既存の毛筆フォントからこれらの特徴を捕らえてデータベースを作成するシステムの実装が今後の課題とされる。

参考文献

- [1]張 憲栄, 真田 英彦, 手塚 慶一 : 漢字楷書毛筆字体の計算機による生成, 信学論, J67-D, No.5, pp.599-606 (1984).
- [2]張 憲栄, 真田 英彦, 手塚 慶一 : 階層分解合成法による隷書体漢字の生成, 信学論, J68-D, No.8, pp.1489-1496 (1985).
- [3]張 憲栄, 真田 英彦, 手塚 慶一 : 計算機による様々な書体生成に適合する筆触パターンの提案, 信学論, J69-D, No.6, pp.885-892 (1986).
- [4]張 憲栄, 李 華妹, 真田 英彦, 手塚 慶一 : 自由連筆の可能な毛筆書き平仮名の生成, 信学論, J76-D-II, No.9, pp.1868-1877 (1993).
- [5]Nakamura, T., Itoh, H., Seki, T. and Law, T. : A Writing System for Brush Characters Using Neural Recognition and Fuzzy Interpretation, International Joint Conference on Neural Networks, Vol.3, pp.2901-2904, (1993).
- [6]Nakamura, T., Kuroda, T., Itoh, H. and Seki, H. : Fuzzy-Based Writing System for Acquiring Good Writing Skill of Brush Characters Based on the Analysis of Writing Speed, The 3rd Pacific Rim International Conference on Artificial Intelligence, Vol.2, pp.822-827 (1994).
- [7]中村 剛士, 黒田 崇, 世木 博久, 伊藤 英則 : 筆記速度のファジィ評価方法を導入した毛筆文字生成システムについて, 日本ファジィ学会誌, Vol.7, No.2, pp.371-379 (1995).
- [8]中村 剛士, 松下 政親, 世木 博久, 伊藤 英則 : フラクトルを用いた毛筆文字のかすれ表現について, 日本ファジィ学会誌, Vol.8, No.3, pp.558-566 (1996).
- [9]真野 淳治, 中村 剛士, 世木 博久, 伊藤 英則 : 毛筆書体におけるくりこみ群を用いたかすれにじみ表現, 情報処理学会論文誌, Vol.8, No.4, pp.806-814 (1997).
- [10]佐賀 聡人, 牧野 宏美, 佐々木 淳一 : 手書き曲線モデルの一構成法 : ファジィスプライン補間法, 信学論, J77-D-II, No.8, pp.1610-1619 (1994).
- [11]佐賀 聡人, 牧野 宏美, 佐々木 淳一 : ファジィスプライン曲線同定法, 信学論, J77-D-II, No.8, pp.1620-1629 (1994).

(1998年3月25日 受付)

問い合わせ先

〒466-8555

名古屋市昭和区御器所町

名古屋工業大学 知能情報システム学科

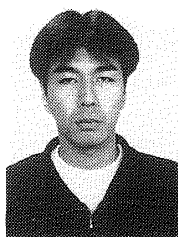
中村 剛士

TEL : 052-735-5475

FAX : 052-735-5477

E-mail : tnaka@ics.nitech.ac.jp

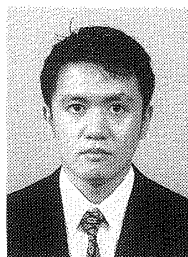
著者紹介



真野 淳治 (まの じゅんじ)

日本電信電話(株)

1996年 名古屋工業大学知能情報システム学科卒業。1998年 同大学院博士前期課程修了。同年、日本電信電話(株)入社。在学中、毛筆文字生成システムの研究開発に従事。情報処理学会会員。



中村 剛士 (なかむら つよし)

名古屋工業大学 知能情報システム学科

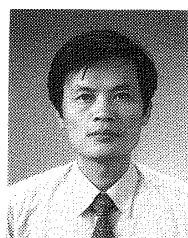
1993年 名古屋工業大学工学部電気情報工学科卒業。1998年 同大学院博士後期課程修了。同年、名古屋工業大学知能情報システム学科助手、現在に至る。感性情報処理、ファジィ推論等に興味を持つ。博士(工学)。情報処理学会、日本ファジィ学会各会員。



江野 脇 宏 (えのわき ひろし)

名古屋工業大学 知能情報システム学科

1997年 名古屋工業大学知能情報システム学科卒業。現在、同大学院博士前期課程在学中。ファジィ理論、ニューラルネットワークに興味を持つ。



何 立風 (かりふう)

愛知県立大学 情報科学部地域情報学科

1982年 中国西北軽工業学院自動制御学科卒業。同年 同大学助手、1987年 同講師。1997年 名古屋工業大学工学研究科博士後期課程電気情報工学専攻修了。工学博士。1998年 愛知県立大学情報科学部地域情報科学科講師、現在に至る。人工知能、定理証明、マルチエージェント分散計算、画像処理、ファジィ推論に関する研究に従事。情報処理学会会員。



伊藤 英則 (いとう ひでのり)

名古屋工業大学 知能情報システム学科

1974年 名古屋大学大学院工学研究科博士課程電気電子専攻満了, 工学博士号取得, 1974年 日本電信電話公社横須賀研究所勤務, 1985年(財)新世代コンピュータ技術開発機構出向, 1989年 名古屋工業大学教授, 現在知能情報システム学科所属, この間, 数理言語理論, 計算機ネットワーク通信, OS, 知識ベースシステムなどの研究開発に従事, 電子情報通信学会, 情報処理学会, 人工知能学会, 形の科学学会, 日本ファジィ学会各会員.

Generation of Hiragana Brush-Style Characters Using Pen-Moving Rules

by

Junji MANO, Hiroshi ENOWAKI, Tsuyoshi NAKAMURA, Lifeng HE and Hidenori ITOH

Abstract :

In this paper, we propose a method to generate Japanese hiragana style characters that reflect user's personality or individuality. Input device is a pen. Our method uses input coordinates (sampling points), pen pressure and pen speed as input information. These information transform the hiragana style variously. In our method, B-spline curves interpolate the selected sampling points, and make smooth input curve. Fuzzy spline curves are applied for the selecting, and enable the shape of sampling-points-queue to be preserved approximately. Moreover, we propose the pen-moving rules. The smooth input curve is divided into several parts by applying the pen-moving rules. The each divided part is transformed into hiragana style part. Our proposed method was implemented on the calligraphic system. Hiragana style characters were generated by using the system and they showed that the proposed method is effective.

Keywords : Generation of Hiragana Brush-Style Characters, Drawing Input by Electronic Pen, Fuzzy Spline Curve, Pen-Moving Rules

Contact Address : **Tsuyoshi NAKAMURA**

*Department of Intelligence and Computer Science,
Nagoya Institute of Technology
Gokiso-cho, Showa-ku, Nayoya, 466-8555 Japan
TEL : 052-735-5475
FAX : 052-735-5477
E-mail : tnaka@ics.nitech.ac.jp*