

# オンライン手書き文字認識の前処理に対する 信号処理的手法の応用

相澤 正、中川正樹、高橋延匡  
(東京農工大学 工学部 数理情報工学科)

## 1. はじめに

最近の日本語ワードプロセッサの低価格化は激烈を極め、一部では4万円を下回る商品も売り出されている。このため、日本語ワードプロセッサは情報処理機器としてより、あたかも家庭電化製品のごとく一般家庭にまで浸透するに至っている。無論、オフィスにおける日本語ワードプロセッサの導入は改めて言うまでもないことである。日本においてはこれまで言語情報処理の点で、英語文化圏に大きく水を開けられていただけにユーザとしては歓迎すべき傾向である。

しかし一度その入力法に眼を転じると、未だ仮名漢字変換方式が主流を成しており、入力者に対する負担は改善されてはいないのが現状である。仮名入力は比較的短時間の訓練によって習得可能であり、その点で現実的かつ素人向きの方式と言えるが、一方、同音異義語選択の必要性、分ち書きの必要性、図表入力との親和性、等の問題があり、この方式に淘汰されてしまうとは考えがたい。マンマシン・インターフェイスの点でも入力者に対する多大な負担、強制を然殺している感がある。今後は人間が計算機に合わせるのではなく、計算機が人間に歩みよるといふマンマシン・インターフェイスの改善を志向すべきである。

このような観点にたち、本研究室においてはより現実的であるオンライン手書き文字認識方式を前提とした日本語入力システムの研究を行ってきた。この研究は、JOLIS (Japanese On-Line Input System) 計画と名づけられ、1981年度には実験システム JOLIS-1 の開発[1]を、また、1984年度にはその評価[2,3]を終了した。JOLIS-1 は平仮名、片仮名、教育漢字および句読点等の記号を構造解析的手法により、字種に依らない統一的な認識を行う。JOLIS-1 は、タブレットより入力される筆点位置座標点列に、前処理(平滑化処理、間引き処理)を施した後、8方向の直線セグメント系列に変換する。次に直線セグメント列として表現されたストロークは、方向変化とその前後の方向セグメント列として情報圧縮される。その後、ストローク辞書によって表現される29種類の基本ストロークの何れかに識別される。JOLIS-1 は、このように文字を基本ストロークの列で表現し、その列を標準字体の文字辞書と相違度マッチングをとることにより入力文字を認識するものである。この JOLIS-1 の評価の結果、その問題点の一つとして前処理が指摘されている。これは前処理によって入力パターンが変形されることに起因する問題である。JOLIS-1 においてこの問題は、認識率に数%の影響を及ぼすに過ぎないが、反面、この変形は前処理後のいかなる処理によっても救済不可能な傾向が確認されている。現在この変形の存在を前提とした文字辞書を生成しなければならないという弱点を JOLIS-1 は含んでいる。したがって、JOLIS の最も重要な研究課題の一つであるパタンプリミティブの設定、識別の研究[4]に先立ち、前処理を抜本的に考察し改善を行う必要がある。

また現在、計算機に対する図表の入力はキーボードより行っているが、これが著しく不便であることは否めない。このような背景を考えた時、最近急激に低価格、小型化されてきたタブレットは入力デバイスとして期待される。また、その適用範囲も広がることであろう。このような現状を考え、前処理方式を考察しその特性を明らかにすることは意義があると考えられる。

以下の節ではまず、JOLIS-1 も含めて現在一般的に用いられている前処理方式の概要とその問題点を述べる。次に本研究の目的、方針を述べた後、実際に行った実験内容について述べる。

## 2. 前処理方式の現状

前処理の目的は安定的なストローク形状の抽出にある。この安定的特徴の抽出を妨げる要因として、①被験者の手振れによる雑音成分、②タブレットの量子化誤差、ハードウェア的な雑音成分、等が挙げられる。平滑化処理はこれらの要因の除去を目的としたものである。

平滑化処理として、これまで用いられてきた方式には、

$$\left. \begin{aligned} X_1 &= \Delta S_1 \cdot X_{1-1} + \Delta S_2 \cdot x_1 \\ Y_1 &= \Delta S_1 \cdot Y_{1-1} + \Delta S_2 \cdot y_1 \end{aligned} \right\} \dots\dots\dots (1)$$

ただし、 $\Delta S_1 + \Delta S_2 = 1$

$$\left. \begin{aligned} X_1 &= (x_{1-1} + x_1) / 2 \\ Y_1 &= (y_{1-1} + y_1) / 2 \end{aligned} \right\} \dots\dots\dots (2)$$

$$\left. \begin{aligned} X_1 &= (x_{1-1} + x_1 + x_{1+1} + x_{1+2}) / 4 \\ Y_1 &= (y_{1-1} + y_1 + y_{1+1} + y_{1+2}) / 4 \end{aligned} \right\} \dots\dots\dots (3)$$

いずれも、 $(x_1, y_1)$  : 平滑化処理前座標点、 $(X_1, Y_1)$  : 平滑化処理後座標点

などが知られている[5,6,7]。このうち JOLIS-1 では、式(1)を採用し  $\Delta S_1=0.75$ 、 $\Delta S_2=0.25$  という値を実験的に最適と確認した上で使用してきた。しかし、実際には式(1)自身が示すように  $i$  が進むにつれて蓄積効果が大きくなり、筆跡の進行方向に対して遅れが生じるという欠点が指摘された。すなわち、“ね”の“ $\eta$ ”のような長いストロークの場合、ストロークの末尾になるほど平滑化処理前と後の座標点が遠く離れたものとなり、その後の認識処理に悪影響を残す。実際、これが原因となって認識できなかった文字は誤認識、リジェクト文字の約5%を占めており、しかも悪質なことにこれらの文字はその後の処理において文字辞書の学習や相速度効果では救済できない。この傾向は低画数文字に特に顕著である。また、JOLIS-1 ではこのような“遅れ”まで考慮した文字辞書を作成せざるを得ない状況であった。

一方、①『スタイラスペンの停滞による冗長な座標点の除去』の目的で、座標点列に対し間引き処理を施す。また、JOLIS-1 にあっては特に、②『ストロークの始末端にみられるストローク修飾の除去』の目的も合わせもっている。この間引き処理は、一定の空間分解能で座標点をマスクするものであり、正方形マスク、十字形マスク、円形マスクなどが知られている。このうち JOLIS-1 では正方形マスクを採用しており一辺を0.8mm、始末端においては1.6mmと実験的に最適値を求め使用してきた。しかし、JOLIS 開発環境の整備(JOLIS-1E(JOLIS-1のM C68000版)の実現)にともない使用するタブレットの性能が向上し、文字記入枠を1cm角と小さくすることにより間引き処理におけるマスクの大きさを考えなおす必要がある。加えて、JOLIS-1 では始末端に対し特別に大きなマスクを用いているが、これが原因となって本来あるべきセグメントを除去してしまうという問題が存在する。この要因によって認識できなかった文字は誤認識、リジェクト文字の約2%を占めている。

また、JOLIS-1 では平滑化処理、間引き処理の順に処理を行っているが、この順で処理を加えると平滑化処理の効果が筆速に依存することが確認されている。すなわち、速く書くと強く平滑化されるが、ゆっくり書くとあまり平滑化されないという現象がおこる。JOLIS-1 設計時において実験的にこの順に定めたものであるが、この平滑効果の不均等については予測されていなかった。現時点で再考する必要があるであろう。

以上述べたように、現在の前処理には数々の問題点が存在する。JOLIS-1 におけるこれらの問題は全認識対象文字の認識率に及ぼす影響はたかだか数%に過ぎないが、特に平仮名などの低画数文字においてこの問題は致命的な要因となっている。なぜなら、前処理における阻害要因はその後のどんな処理によっても救済できないという傾向をもっているからである。また、これまで JOLIS も含めオンライン手書き文字認識システムで用いられてきた前処理方式に対しては、オンライン手書き入力における特性の考察、定量的比較検討といったことが行われていない。このような現状を考え、前処理方式を抜本的に考え直し、その処理の特性等の側面を明らかにすることには意義があると考えられる。

### 3. 前処理方式のフィージビリティテスト

以上の問題点を受け、前処理の設計方針を次の通りとした。

#### (1) 入力筆点座標系列をタブレットから発生する信号とみなし、前処理は信号処理の観点から設計、考察する。

前処理という性質上、リアルタイム処理に適した方式を採用する必要があり、計算量の制約は無視できない。また、好ましい特性のみを抽出するという点では前処理イコール筆点座標へのフィルタとして扱うことが得策と考えられる。

#### (2) 一般性を持たせた前処理方式を志向する。

最近ではマイコンなどに比較的簡単に、しかも安価にタブレットが接続可能である。このような状況に裏づけられて、今後このデバイスは広く普及していくと思われる。また文字入力に限らず図形、表など種々な情報をタブレットを通じて入力することが可能なようになるであろう。なぜなら、現在図表等もキーボードより入力しているが、これが著しく不便であることは否めないからである。このような考えに基づき文字認識だけに狭く限定された処理方式ではなく、一般性を考慮したものとする。

#### (3) 量的尺度を用いた比較検討を行う。

JOLIS にとって最適な前処理とは、以後の認識過程を直して最高の認識率を与えうる方式である。しかし、これを各前処理方式の優劣の尺度としたのでは前述の一般性を持たせるという議論に矛盾する。したがって、最終的にはストローク識別率の最も高い前処理方式を JOLIS として採用するとしても、その前段階としてこれとは別の尺度で、考案される前処理方式案の中から候補を絞りこみ、その中で JOLIS としてはどの方式を採用するか考慮する。そこで、認識率とは別の尺度として、①筆跡進行方向の角度成分変化(方向安定性)、②サンプリング座標点と前処理後座標点の距離(追従度)、を設定する。この2点が十分良好であれば JOLIS にとっても最適な方式だという保証はないが、一般性を保ちつつ JOLIS にとって好ましいものと期待できる。これによって、これまで目視に頼ってきた方式評価に量的な尺度を導入することになる。

### 3.1 平滑化処理の方式案

これまでは単に座標値という量に対し平滑化処理を施してきた。しかし、二次元の量である筆跡を  $x$ 、 $y$  成分に分解して独立に扱って適当かという疑問も残る。また、平滑化処理をサンプリング信号に対するフィルタリングという見方をすれば、その特性というものをもっと明らかにし最適な特性を持つフィルタを設定する必要がある。

このような考え方から、まず2次元の筆跡からどのような量を抽出しフィルタに入力するかという問題を考慮する。ここで次の3種類の量を候補として提起する(図1参照)。

- (1) x、y座標値  $(x_i, y_i)$
- (2) 角度成分  $(\theta_i)$
- (3) 角度成分および移動距離成分  $(\theta_i, r_i)$

角度成分  $\theta_i$  は  $-\infty$  から  $+\infty$  までの値をとる、X軸に対する偏角とする。移動距離成分  $r_i$  は、ストロークの始点から  $(x_i, y_i)$  までの累積距離とする(線形フィルタを用いる限りにおいて積分量であっても差分量であっても問題は同値である)。このような3種類の量を平滑化フィルタへの入力量として考察する。

次に上述の量に対し適当なフィルタを設計するというを考慮する。ここではフィルタ型として①巡回型フィルタ(回掃型フィルタ)、②移動平均型フィルタ、の2通りの候補について考慮した。

前者の巡回型フィルタは次の式で表わされる。 $g_i$  をフィルタに対する入力量、 $f_i$  を出力量とする。

$$f_i = \alpha \cdot f_{i-1} + (1 - \alpha) \cdot g_i \quad \text{ただし、} 0 \leq \alpha \leq 1 \quad (\alpha: \text{巡回比率})$$

この巡回型フィルタは計算が単純で簡単であるが、その反面、 $i$  が大きくなるほど(入力が後になるほど)蓄積効果が大きくなるという特徴を持つ。したがって、時間的な対称性というものは保証されない。

後者の移動平均型フィルタは次の式で表わされる。 $g_i$  をフィルタに対する入力量、 $f_i$  を出力量とする。

$$f_i = w(-N+1) \cdot g_{i-N+1} + w(-N+2) \cdot g_{i-N+2} + \dots + w(0) \cdot g_i + \dots + w(N-2) \cdot g_{i+N-2} + w(N-1) \cdot g_{i+N-1} \quad (N: \text{平均化項数})$$

この式から分かるように、このフィルタは入力量の前後  $2N-1$  点の加重平均をとるものである。前者と比べて  $f_i$  が単純に“ところ天”式に決定できないかわりに時間的な対称性は保証される。このフィルタの場合、前後  $N$  点の加重比率  $w(n)$  が一つの問題となるが一般的に最も特性が良いとされているハミング・ウィンドウ[8](図2参照)を用いる。これは次の式で表わされる。

$$W(n) = \begin{cases} 0.54 + 0.46 \cos(2\pi n / (2N-1)) & -(N-1) \leq n \leq N-1 \\ 0 & \text{otherwise} \end{cases}$$

ハミング・ウィンドウはサイドローブが一律に  $-40\text{dB}$  以下になり、なおかつ比較的周波数分解能がよいという性質を持っている。

以上の3通りの入力量と2通りのフィルタの組合せのうち先の評価基準を最高にするものを平滑化処理として採用する。

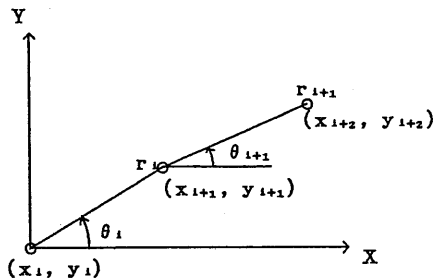


図1 フィルタへの入力量

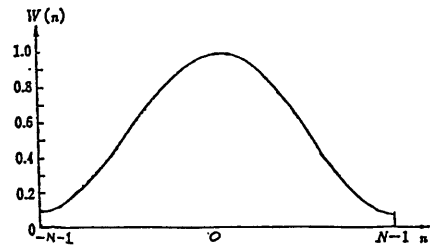


図2 ハミング・ウィンドウ

### 3.2 間引き処理の設計

間引き処理の設計は以下の通りとした。

#### (1) 円形マスクを考察する。

JOLIS-1における間引き処理は正方形マスクを使用してきたが、後継システムでは円形マスクも考察する。マスクの形については、正方形、十字形、菱形、円形などが提案されてきたが、これまでは計算量の観点から円形マスクは採用していなかった。しかし、ハードウェア環境の整備にともない乗算の2回程度は問題にならないほどの余裕があり計算量の制限により円形マスクを候補から除外する必要はほぼなくなった。また、前処理をフィルタリングの見地から考察することを考え合せれば、間引き処理ではこの議論にあわせ特別な方向選択性の効果をもたないマスクで実験する方が好ましい。その上で実用上の制約が生じた場合、他の形状のマスクで近似するものとする。

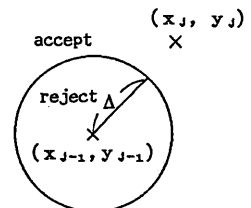


図3 円形マスク

## (2) 始末端への特別なマスクを廃止する。

ストローク修飾への対処はあくまでストローク識別で行い、副作用を持つような前処理は行わない。始末端マスクによっても除去できないストローク修飾は当然あり、また逆に始末端マスクにより有意なセグメントが欠落するという副作用が JOLIS-1 の評価を通じて確認されている。したがって、困難であってもストローク識別処理等でストローク修飾に抜本的に対処していく必要がある。

この設計より、間引き処理は次の式によって定義される (図3参照)。

$$(\overline{X}_j, \overline{Y}_j) = (x_i, y_i) \quad \text{if } (x_i - \overline{X}_{j-1})^2 + (y_i - \overline{Y}_{j-1})^2 \geq \Delta^2$$

(  $x_i, y_i$  ): 間引き処理前座標点, (  $\overline{X}_j, \overline{Y}_j$  ): 間引き処理後座標点

パラメータ  $\Delta$  については、抽出したいストロークの微妙な動きを十分表現できるように実験を通じて最適な値を見出し採用する。

以上のような設計のもとに、フィージビリティテストを通じて最良な方式を決定するものとした。

## 3.3 平滑化処理、間引き処理に対するテスト

以上の設計に基づき、以下のフィージビリティテストを行った。

### 3.3.1 テスト環境およびテストデータ

テストデータは、JOLIS-1E(MC68000システム)のタブレット(サンプリングレート150point/sec、分解能0.1mm)によって収集し、汎用計算機ACOSシステム600Sに転送した後テストを行った。これは後に述べるとおり、評価尺度の一つとしてパワースペクトル密度を求めるためである。

また、使用したデータは、①筆速の速いパターン、遅いパターン、②鋭い尖点の存在するパターン、③ループの存在するパターン、等の様々な観点から性質の異なるデータを英数字、平仮名、片仮名、漢字の字種にわたり14通りを用いた。図5(データA)、図6(データB)に実験データの一例を示す。両データとも、1cm角の大きさの記入枠に筆記されたものであるが、筆速の点で大きく異なる。データAは比較的速く書かれたものであり、入力点数70(筆速2.37mm/sec)である。一方、データBは非常にゆっくりと書かれたものであり、入力点数632(筆速0.4mm/sec)である。



図4 データAの入力パターン

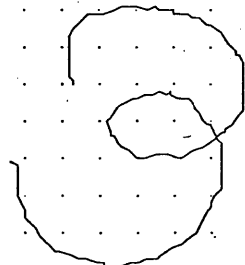


図5 データBの入力パターン

### 3.3.2 評価尺度

これまで述べた方式案は、平滑化フィルタへの入力量(3通り)、フィルタ型(2通り)および平滑化処理と間引き処理の処理順序の組合せの中から最適なものを選択するというものである。したがって、前処理方式の候補はその組合せだけ存在し計12通りとなる。

本実験ではまず、各データを各方式案により処理し、データによらず方向安定性、追従性の2点を一様に満足する方式を求めた。まず方向安定性を計る尺度として、①角度成分のパワースペクトル密度、②角度成分変化の2乗和、③ JOLIS セグメント化処理におけるN分割方向量子化安定性、の3つを用いた。①は角度成分を  $\theta(t)$  なる時間関数として表現した場合のパワースペクトル密度である。②は図7に示すように、直前の角度成分との変位の2乗和である。この尺度は、直線なら0、また、極端な角度変化が存在すると鋭敏に値に反応し、緩やかな角度変化に対しては比較的小きな値をとるという性質を持つ。③は JOLIS 特有のセグメント化処理における安定性を確認するものである。この尺度はセグメント化処理において、N分割方向量子化を行った場合、その結果が安定するか否かを調べるものであり、Nとしては8、16、32を用いた。以上3つの量的尺度を用いて方向変化の安定性を表わすものとした。

次に追従性の尺度としては、①サンプリング座標点とそれに対応する前処理後座標点とのユークリッド距離、を用いた。これは図8に示すように、各サンプリング座標点と、それが前処理によって移動した前処理後座標点とのユークリッド距離を用いるものである。ただし、間引き処理によってサンプリング座標点除去される場合

が存在するため前処理後の座標点数によって平均化を行い用いた。

これらの量的尺度によって前処理後座標点の方向安定性、追従性を評価し、各方式案の優劣を比較するものとした。

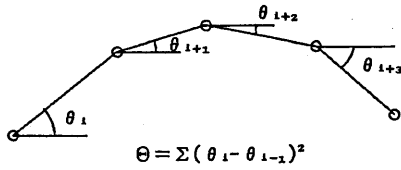


図6 角度変化の2乗和

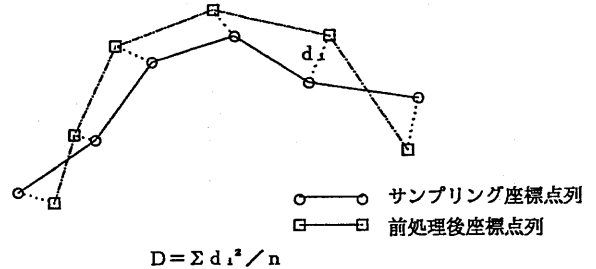


図7 追従性の尺度

### 3.3.3 テスト結果

平滑化フィルタに対する入力量（[1]座標値、[2]角度成分、[3]角度成分+筆速成分）によって大きく異なる結果を得た。したがって、この点を中心に以下に順を追って、各方式案による実験結果を示す。

#### [1] 座標値に対し平滑化処理を行う場合

座標値に対して平滑化処理(巡回型、移動平均型)を行う場合の結果は、次に要約される。

(1) 平滑化処理と間引き処理の処理順序について

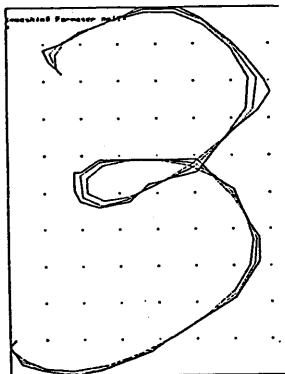
- ① 平滑化処理を先に行った場合、最適な平滑化パラメータは対象パタンの筆速に大きく依存する。筆速の幅広い変化に統一して適用可能な平滑化処理を設定することは不可能であった。
- ② 逆に、間引き処理を先に行う場合、間引きマスクの大きさによって方向安定性が大きく変化する傾向を得た。間引きマスクを比較的大きく(半径0.3mm)設定した場合、一様に追従性が悪化する傾向があるが、逆にマスクを小さく(半径0.1mm)設定した場合、筆速の速いパターンについて追従性の不足を確認した。

(2) 平滑化フィルタの型について

- ① 巡回型フィルタを用いた場合、パラメータのいかんをとわず追従度の悪化を確認した。これは特に、筆速の速いパターンに顕著である。
- ② 移動平均型フィルタを用いた場合、平均化項数5において良好な結果を得た。しかしなお筆速の速いパターンについて改良の余地をもつ。

以上の結果の一例として、データA、データBに対する間引き処理後、移動平均型平滑化処理を行った結果について図9、図10にそれぞれ示す。筆速の緩やかなデータBについては良好な追従性を示しているのに対し、筆速の速いデータAでは追従性に改良の余地が残されている。

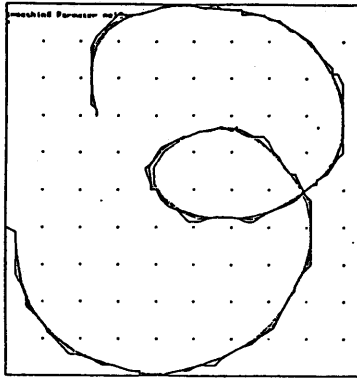
また、角度成分パワースペクトル密度において高周波数領域が十分減衰しているにも係らず、JOLIS セグメント化処理において不安定な結果を得るケースが40例中17例に確認された。周波数分解を行った結果、有意な信号成分とノイズ成分との閾値が得られれば、単純な周波数フィルタを前処理とすることが可能である。しかしこの結果から、高周波数領域が減衰するだけでは十分な条件とは成りえず、単純なローパスフィルタでは有効な前処理とは成りえないと判断できる。



平滑化項数=3(点線)および5(一点鎖線)  
間引きマスク半径=0.3mm

追従度D=1.42および4.77  
角度変化2乗和 $\Theta$ =10.04および8.02  
JOLISセグメント安定性:  
3/2分割安定および3/2分割安定

図8 データAに対する前処理結果  
(間引き処理+移動平均型平滑化処理)



平滑化項数 = 3 (点線) および 5 (一点鎖線)  
間引きマスク半径 = 0.3mm

追従度  $D = 0.21$  および  $0.81$   
角度変化 2 乗和  $\Theta = 7.47$  および  $6.17$   
JOLISセグメント安定性:  
3 2 分割安定 および 3 2 分割安定

図9 データBに対する前処理結果  
(間引き処理+移動平均型平滑化処理)

[2] 角度成分に対し平滑化処理を行う場合

まず、抽出する角度成分量として図11に示す次の2つのタイプが存在する。前者のタイプ1は、平滑化処理がなされた座標点  $\bar{P}_i(\bar{x}_i, \bar{y}_i)$  と入力座標点  $P_i(x_i, y_i)$  を結ぶベクトルがX軸となす角度変位を抽出するものである。一方、後者タイプ2は、入力座標点  $P_i(x_i, y_i)$ 、 $P_{i+1}(x_{i+1}, y_{i+1})$  を結ぶベクトルがX軸となす角度変位を抽出するものである。これは、入力座標点を一度極座標系に変換し、その角度成分を抽出することに等しい。この場合、極座標系の距離成分については保存するものとして扱う。

以上、単に角度成分と言っても2つのタイプの量が定義できる。本実験ではこの両タイプについて考察した。実験の結果は次に要約される。

(1) タイプ1の角度変化について

図12aは比較的平滑化効果が強いケースを示している。この場合、微小な角度変化に対して比較的鈍重であるために、直線的な部分では十分な追従を示すが、転回部分において座標点位置関係上大きな遅れが生じ、それが蓄積されて徐々に追従性が悪化する。偏角における遅れは後の処理において回復可能であるが、移動距離成分で遅れが生じてしまった。逆に図12bは平滑化効果が弱いケースを示している。この場合では、角度変化に対して過敏であるためにノイズによる角度変化にも追従してしまう。この図12a、bの中間の状態を探し得ることができれば、求められるフィルタ特性が得られると考えたが結果的にそのような状態は存在しないことを確認した。

(2) タイプ2の角度変化について

この角度変化は、入力パタンの角度変化のみが独立して扱われ、一度、平滑化処理後座標点と平滑化処理前座標点との偏角が増大しても、再び縮小させるための力は何処にも生じない。したがってタイプ2の角度変化を扱った場合には追従性が一概に不足する傾向を示した。

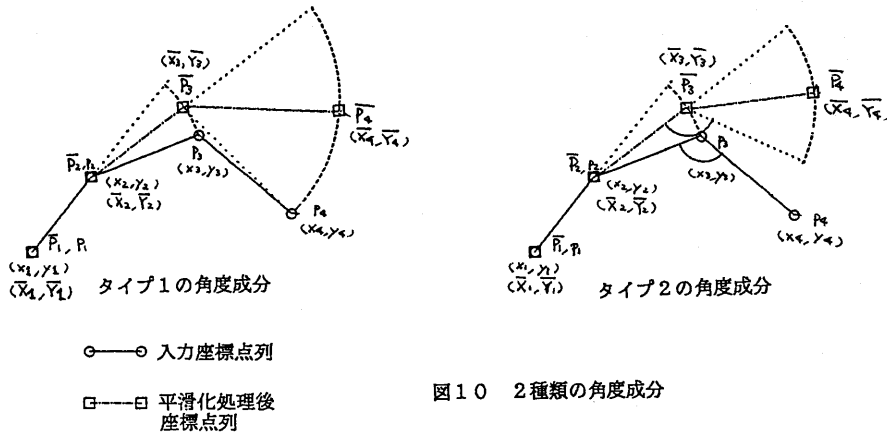
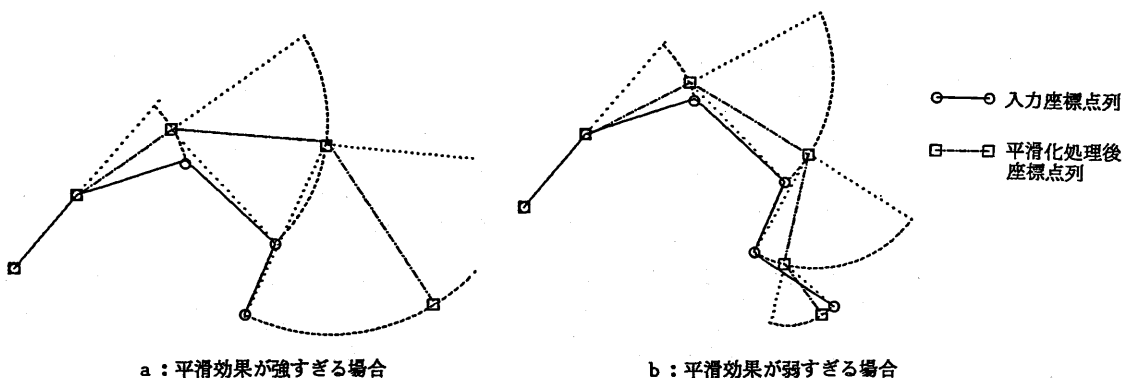


図10 2種類の角度成分



a : 平滑効果が強すぎる場合  
このケースでは、平滑効果が弱すぎるためささいな角度変化に対しても追従してしまう。

b : 平滑効果が弱すぎる場合  
このケースでは、平滑効果が強すぎるため転回部分において追従性が悪化し、それが蓄積されてしまう。

図1.1 角度変化に平滑化を行う場合の問題点

### [3] 角度成分および移動距離成分に対する平滑化処理を行った場合

この方式における議論は、前述の角度成分タイプ1の処理において“遅れ”が蓄積し追従性が徐々に悪化していく現象に対し、移動距離成分も操作することで改善が得られるか否かに絞られる。しかし結論として、移動距離成分に対しフィルタを施しても“遅れ”を解消するほどの改善は見られず、本方式も前処理として不適当であるとの答えを得た。

#### 3.3.4 テスト結果のまとめ

以上の結果をまとめると、次のように言える。

- (1) まず、角度成分、および角度成分と移動距離成分に対する平滑化処理は、適当な成果を得られなかった。したがって、座標値に対する平滑化処理が適当と考えられる。
- (2) 座標値に対する平滑化処理として、巡回型フィルタを用いた場合、追従性の悪化は避けられない。
- (3) サンプリング座標点に対して間引き処理を行った後、平滑化処理として移動平均型フィルタを施した場合に、最も良好な結果を得た。これは、筆速の速いパタンと遅いパタンとの不均一なサンプリング点間隔を、間引き処理において均一化し、この結果に適する平滑化パラメータを設定することによって筆速に比較的依存しない前処理方式を実現できたことによる。しかし筆速の速いパタンの場合、サンプリング点間隔が間引きマスクサイズより大きい場合この均一化が十分実現できない。よって、この方式においても筆速の速いパタンの追従性の面で改良の余地が残された。

#### 3.4 補間処理、可変サンプリング処理の導入

サンプリング座標点に対し間引き処理を施した後、移動平均型フィルタを用いた平滑化処理を行うことで比較的良好な結果が得られた。これは、筆速の不均一なパタンに対して、まず最初に間引き処理を行い座標点間隔をほぼ均一化したために、筆速に比較的依存しない平滑化フィルタが設定できたことによる。しかし、この方式の問題は、

- (1) 間引きマスクによっても座標点間隔を均一化できないほど、筆速の速いパタンの存在
- (2) 尖点における角の鈍化

にある。(1)は間引きマスクの大きさに比べてサンプリング座標点間隔が広すぎる筆速の速いパタンが存在するという事である。このようなパタンでは、平滑度の不足という現象が生じる。これに対し、間引きマスクを大きく設定すれば座標点間隔を均等化できるが、このように一律に座標点を除去した場合、他パタンにも副作用が生じ、好ましい結果は得られない。(2)は入力座標点に対し、一律に間引き処理を行うことによる副作用であり、“L”や“μ”などの尖点においても一律に間引き処理を行うために、尖点の先端付近の座標点の間引きが、角が鈍化するものである。これらの作用により特徴点を失うことが考えられ、可能な限りこれらの要因に対処する必要がある。そこで、この問題を解決すべくさらにフィージビリティテストを続行した。この結果を以下に述べる。

##### 3.4.1 補間処理の導入

まず、間引きマスクによって座標点間隔を均一化できないほど筆速の速いパタンに対処することを目的として、補間処理の導入を図った。この要因は、筆速が速すぎるためにタブレットのサンプリング動作が追従できないことに起因するものであり、したがってタブレットの性能不足を前処理において補う必要がある。このために、補間処理を導入する。すなわち、タブレットの性能が不足するために抽出できなかった座標点を補うものである。

補間処理方式としては、種々と知られているがオンライン処理に適した方式という制約、および補間処理に要

求される精度を考慮し一次補間を採用した。

補間処理は、次の式によって定義される。

$$\text{if } (\bar{X}_J - X_1)^2 + (\bar{Y}_J - Y_1)^2 \geq R^2 \quad (\bar{X}_J, \bar{Y}_J) = (\bar{X}_{J-1} + r \cos \phi, \bar{Y}_{J-1} + r \sin \phi)$$

$$\text{else } (\bar{X}_J, \bar{Y}_J) = (X_1, Y_1)$$

$$\phi = \tan^{-1} \frac{Y_1 - \bar{Y}_{J-1}}{X_1 - \bar{X}_{J-1}} \quad \begin{array}{l} (X_1, Y_1) : \text{補間処理前座標点} \\ (\bar{X}_J, \bar{Y}_J) : \text{補間処理後座標点} \end{array}$$

すなわち、直前の補間処理後座標点と新たなサンプリング座標点の距離がR以上離れた場合、間隔rで補間するというものである。本報告では、R=0.4mm、r=0.2mmを用いた。間引き処理と補間処理の導入により、時系列データ  $(x(n\Delta t), y(n\Delta t))$  または  $(r(n\Delta t), \theta(n\Delta t))$  から、ほぼストロークの単位長  $\Delta S$  ごとの系列データ  $(x(n\Delta S), y(n\Delta S))$  または  $(r(n\Delta S), \theta(n\Delta S))$  ( $n=1, 2, \dots$ ) に変換することができる。したがって以後の平滑化処理が筆速に影響されないという効果を持つ。

### 3.4.2 可変サンプリング処理の導入

以上の処理で問題として残るのは、単に間引き処理を施したのであっては尖点付近の点も一様に除去してしまう、それに平滑化処理を行うことによって角が鈍るという結果を得ることである。これらの問題意識より冗長な点を除去する処理に対して、次のような要求を見出すことができる。

- ① 一様にゆっくりとしたペンの動きによって入力された冗長な座標点は十分に間引く必要がある。
- ② ただし、尖点付近でペン移動速度の低下により入力された座標点列は残しておく必要がある。

これに対し上記間引き処理に換えて、可変サンプリング手法に基づく処理を次に提起する。

入力座標点列を一定の大きさの領域  $a_1 \sim a_n$  に分割する。それぞれの領域に  $b_1 \sim b_n$  個の座標点が存在した場合、各領域  $a_i$  より抽出する点数の上限を次の  $N_i$  によって規定する。

$$N_i = \frac{b_i}{(b_{i-1} + b_i + b_{i+1})/3} \quad (i=1 \sim n)$$

これはサンプリングされたパタンを一定の大きさの領域  $a_1 \sim a_n$  に区切り、それぞれの領域に  $b_1 \sim b_n$  個の点が存在した場合、各領域  $a_i$  内より実際に抽出する点数の上限を  $N_i$  ( $i=1 \sim n$ ) 個に定めるといものである。ただし、この  $N_i$  を  $a_i$  および  $a_i$  の前後1個の領域  $a_{i-1} \sim a_{i+1}$  中に存在する点数  $b_{i-1} \sim b_{i+1}$  によって決めようとするものである (図12参照)。

実際には、区間0.3mmの領域分割を行い、 $a_i$  およびその前後の領域に属する点数の平均を用いた。これによれば、筆速が比較的一定している場合、領域  $a_{i-1}$ 、 $a_i$ 、 $a_{i+1}$  に含まれている点数はほぼ等しくなり、 $a_i$  の領域より1個の点が抽出されることになる。すなわち筆速が一定している場合には、これまでの間引き処理とまったく同じ効果を持つことになる。一方、筆速が変化する場合、例えば尖点が存在するような場合、領域  $a_k$  において尖点が存在し筆速が低下したと仮定する。当然領域  $a_k$  中に存在する点数は  $b_{k-1}$ 、 $b_{k+1}$  に比べて多いため、この処理によって領域  $a_k$  より抽出される点数は1より大きくなる。

この処理は、上記領域ごとに平均して1点を採集することから、冗長に入力された座標系列を除去する(間引く)ことになり擬似的にタブレットの性能を低下させることになる。しかし、尖点付近において周囲より筆速が低下する場合、サンプリング性能を高めてその筆点情報を強調し、以後の平滑化処理で尖点が鈍らないようにする効果がある。そこで、以後この処理を可変サンプリング処理と呼ぶことにする。この処理は、補間処理がタブレットの性能不足を補うための処理であったことを考えると非常に好対称をなす。この可変サンプリング処理、補間処理の導入によって、理論的にはタブレットの性能に依存しない前処理方式を構築することができると言えよう。また、最終的に冗長点を除去するために間引き処理を行う。入力パタンの形状をどの程度保存する必要があるか、その必要性に応じて間引きマスクの大きさを決定すれば良い。本報告では実際に0.3mmの円形マスクを用いた。

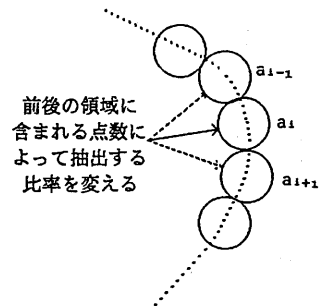


図12 可変サンプリング処理

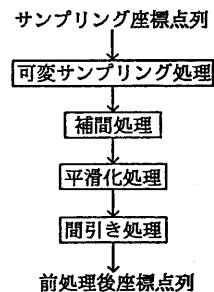
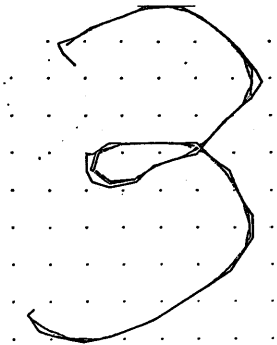


図13 前処理の流れ



以上の議論により結果的に前処理の流れは図13となった。この処理を前述のデータA、Bに施した例を図14、図15に示す。この結果が前述の各前処理方式に比べて追従性、セグメント安定性の点で優秀であることが確かめられよう。

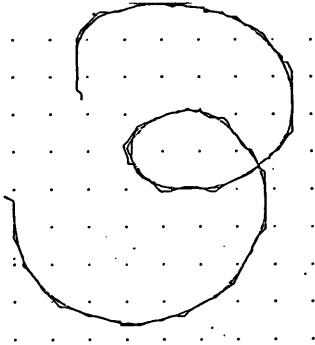


平滑化項数=5(点線)および7(一点鎖線)  
 間引きマスク半径=0.3mm  
 補間間隔 $r=0.2$ mm  $R=0.4$ mm

追従度 $D=1.31$ および $1.65$   
 角度変化2乗和 $\Theta=7.85$ および $7.38$   
 JOLISセグメント安定性:  
 3 2分割安定および3 2分割安定

図14 データAに対する前処理結果

(可変サンプリング処理+補間処理+間引き処理+移動平均型平滑化処理)



平滑化項数=5(点線)および7(一点鎖線)  
 間引きマスク半径=0.3mm  
 補間間隔 $r=0.2$ mm  $R=0.4$ mm

追従度 $D=0.62$ および $0.75$   
 角度変化2乗和 $\Theta=6.50$ および $5.84$   
 JOLISセグメント安定性:  
 3 2分割安定および3 2分割安定

図15 データBに対する前処理結果

(可変サンプリング処理+補間処理+間引き処理+移動平均型平滑化処理)

#### 4. おわりに

本研究における結論は以下の3点である。

- (1) タブレットより得た時系列座標点に対し、信号処理(フィルタリング)の観点からアプローチした結果、リアルタイム処理を可能とする前処理方式を得られた。
- (2) これまで経験的に用いられてきた前処理方式も含め、各種方式を追従性、方向安定性の点から量的比較を行うことができた。
- (3) 相反する性質である追従性、方向安定性の両者を満足する前処理を、可変サンプリング処理、補間処理を導入することによって実現した。

これらの実験は、文字パターンについて行ったが、特にその性質を利用した点は皆無であり、他の図形入力等における前処理にも適用可能であると確信する。

#### 謝辞

本研究は、東京農工大学工学部数理情報工学科、阿刀田央一助教授を始めとする本学科の諸先生方に数々の御助言を頂いた成果である。また、JOLIS 計画に参加した五十嵐道弘氏(現・日本電気㈱)、木村慎一氏(現・キャノ

ン備)、青木克郎氏(現・日本電気備)、真鍋俊彦氏(現・東芝)、池田裕治氏(現・キャノン備)の各位に感謝する。

#### 参考文献

- [1] 中川正樹, 他: “オンライン手書き文字認識システムJOLIS-1の設計と試作”, 情報処理学会日本文入力方式研究会資料3-1 (1982)
- [2] 中川正樹, 他: “オンライン手書き文字認識システムJOLIS-1の定量的評価”, 情報処理学会日本文入力方式研究会資料16-1 (1984)
- [3] 相澤正, 他: “オンライン手書き文字認識システムJOLIS-1の定量的評価 (文字辞書の拡張とその効果)”, 情報処理学会第30回全国大会講演論文集4N-6, pp. 1279 - 1280 (1985)
- [4] 相澤正, 他: “オンライン手書き文字認識におけるパタンプリミティブの情報理論的決定とその辞書生成”, 情報処理学会日本語文書処理研究会資料5-3 (1986)
- [5] Grouner, G. F. : “Realtime recognition of handprinted text”, AFIPS Proc. FJCC, Vol. 29, pp. 591 (1966)
- [6] 佐伯元司, 他: “2次元データ入力のためのオンライン手書き文字認識”, 電子通信学会研究会資料AI 23-3 (1981)
- [7] 藤原塩和, 他: “接線ベクトル列を用いたオンライン手書き文字の認識”, 情報処理, Vol. 17, No. 3, pp. 191-199 (1976)
- [8] 齋藤収三, 他: 音声処理の基礎, オーム社, 1981