

ワークステーションにおける手書き認識技術

Handwritten Characters and Graphics Recognition Technology for Workstation

OAの進展に伴い、マンマシン性の良いシステムがますます重要となっている。ワークステーションの高度マンマシンインタフェースとして、最近、従来の紙と鉛筆の感覚で入力できる手書き入力が注目され始めている。

そこで、今回、キーボード不要の次世代マンマシンインタフェースとして、入力と表示機能を一体化構造にした平面ディスプレイと、この平面ディスプレイ上に文字や図形を手書きするとこれを実時間で認識・清書表示する実時間手書き文字・図形認識技術を開発した。

本論文では、入力一体化平面ディスプレイをもった手書き認識システムと、画数、筆順などの入力条件を大幅に緩和した手書き文字・図形認識方式について述べる。

葛貫壮四郎* Sôshirô Kuzunuki
正嶋 博* Hiroshi Shôjima
福永 泰* Yasushi Fukunaga
浅田昭広** Akihiro Asada
三瓶 徹** Tôru Sanpei

1 緒 言

OA (Office Automation) の進展に伴い、マンマシン性の良いシステムの開発がますます重要となっている。ところが、現状はキーボードによる漢字入力のように、人間が機械の制約に合わせているのが実情である。

従来、計算機の入力技術として、図1のハードウェア技術に示すように、文字列入力ではキーボードが、グラフィック入力ではマウス¹⁾やタブレットが主に使用されている。これらのハードウェア機器では、同図のソフトウェア技術に示すように、人間のイメージをいったんマシンが分かる言葉へ翻訳しなければならなかった。

そこで今回、文字列、グラフィックを同時に入力できて、かつ処理内容のイメージをマシンに直接入力できるハードウェア技術(入力一体化平面ディスプレイ)、ソフトウェア技術(実時間手書き文字・図形認識)を開発した。

本論文では、これら技術を用いた手書き認識システムの構

成と入力・編集方式の概要及び手書き文字、図形認識アルゴリズムについて述べる。

2 入力一体化平面ディスプレイをもった手書き認識システム

手書きのストロークを自由に入力するには、現在のように表示画面と入力部が離れている構造では、正確なストローク入力が不可能である。このため、表示画面上からストロークを直接入力できるように、入力一体化平面ディスプレイとそれを用いた手書き認識システムを開発した。図2に入力一体化平面ディスプレイを、図3に手書き認識システムの論理構成を示す。

入力一体化平面ディスプレイは、液晶表示装置と電磁タブレットを組み合わせることで実現した。

入力一体化平面ディスプレイ上に電磁ペンで手書き(ストローク)すると、まず筆跡が表示され、そのストロークは認識

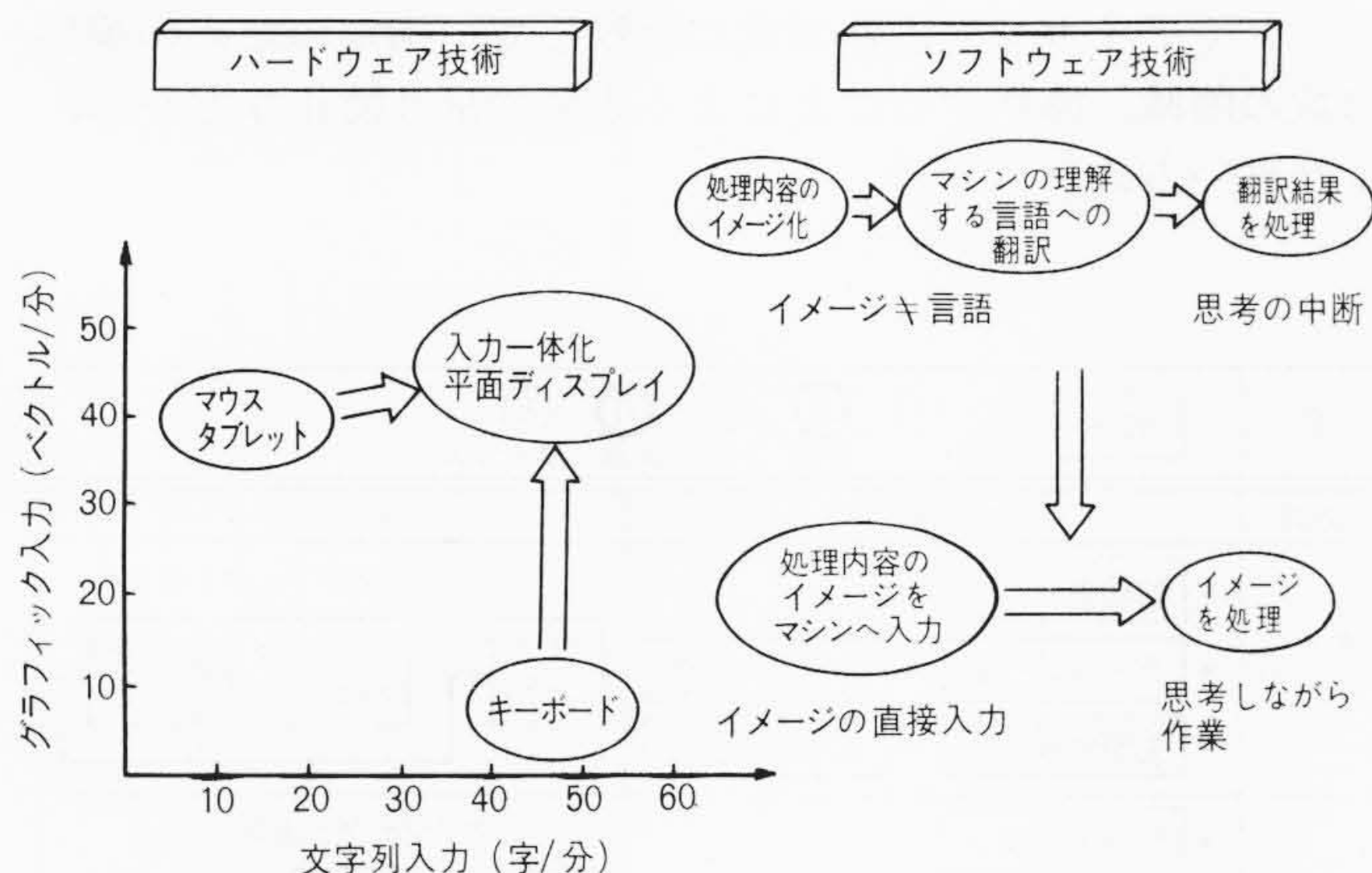


図1 入力方式の革新 現在は、キーボード入力のようにマシンの理解する言語に翻訳しているのが実情である。しかし、文字列やグラフィックを同時に入力できる入力一体化平面ディスプレイと実時間手書き認識技術により、処理内容のイメージをマシンに直接入力できるようになった。

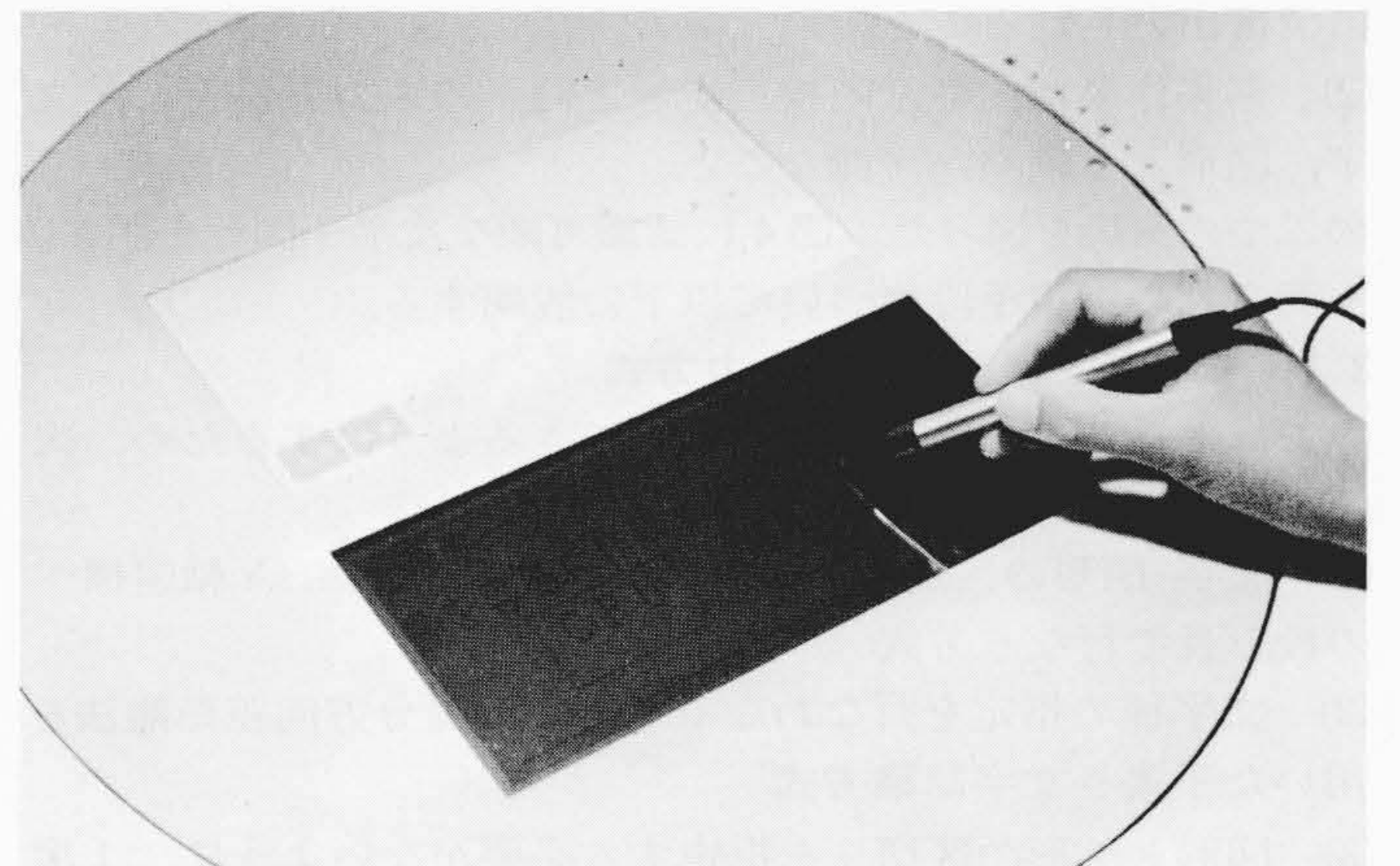


図2 入力一体化平面ディスプレイの外観 電磁式タブレットと大形液晶(640×200ドット)を組み合わせたものである。この平面ディスプレイ上に専用ペン(スタイラスペン)で文字や図形を入力すると、清書されて表示される。

* 日立製作所日立研究所 ** 日立製作所家電研究所

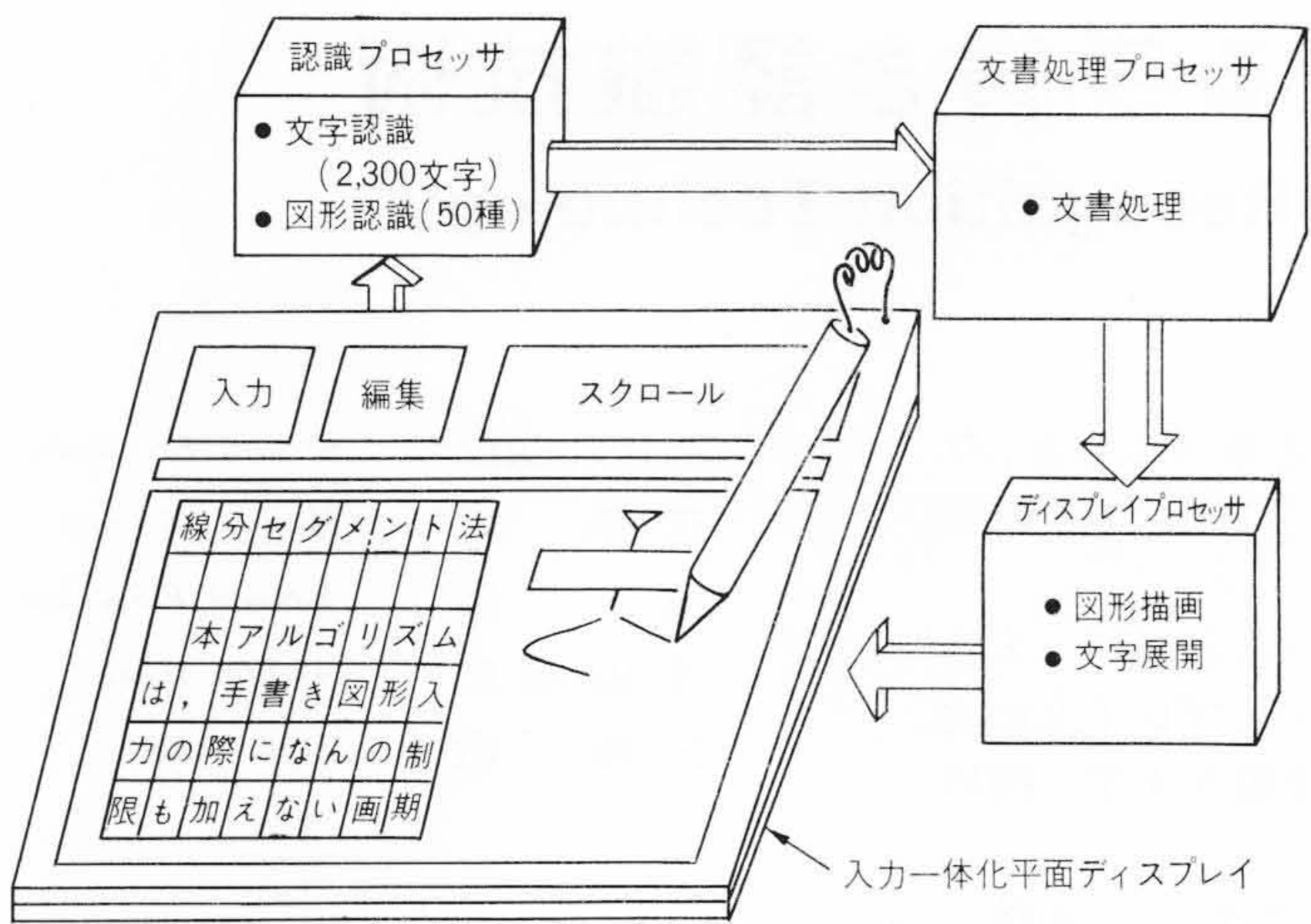


図3 手書き認識システムの構成 入力一体化平面ディスプレイ上に手書きストロークが入力されると、いったんその筆跡が表示されたあと、そのストロークは認識プロセッサで文字あるいは図形のコードとして出力される。文書処理プロセッサでは、送られたコードをデータやコマンドとして使用し、その処理結果をディスプレイプロセッサで平面ディスプレイ上に表示する。

プロセッサで文字あるいは図形として認識される。認識された文字や図形は、文書処理プロセッサ、ディスプレイプロセッサを介して入力一体化平面ディスプレイ上に清書表示される。更に、清書された文字・図形を校正記号を用いて手書きで編集することができる。以上のように、図形入り文書をすべて手書きで作成・編集することができるため、従来の紙と鉛筆の作業形態に一步近づき、マンマシン性を大幅に改善できた。

3 手書き認識を用いた文書入力・編集ソフトウェア

2章で説明したシステムを用いて、図形入り文書入力・編集ソフトウェアを実現した。特に、頭に描いたイメージをそのまま入力できるようにするため、従来の紙と鉛筆による作業と同一スタイルになるように意を用いた。すなわち、

(1) 文字入力や図形入力モードのように、なんのモードも意識せずに、文字や図形を書きたいところに書くと、その場所に清書出力されること。

(2) 編集作業は、従来の校正記号と同一の入力方法とし、文字、図形とも編集が可能なこと。

の二つの実現を図った。図4に認識可能な文字・図形を示す。上記2点の実現手段について以下に説明する。

3.1 モードを意識しない入力方式

モード指定を意識しないで入力できるようにするため、次のような方式を採用した。

(1) 文字・図形の区別を、入力ストロークのX軸、Y軸座標への投影長で行なう方式

(2) 文字種の指定を行わなくてもよい線分方向差距離法を用いた手書き文字認識方式

(3) 図形と図形の区切りを指定する必要があるように、1次元ダイナミックプログラミング法による自動図形切出し方式と、文字と文字の区切りを文字枠で管理する方式

以上の方式により、モードは図3に示すように入力と編集の二つがあるだけである。

3.2 校正記号による編集方式

手書き図形認識を用いて、校正記号による編集ができるようにした。シンボルコマンドによる編集は、CAD(Computer

種類	内容
文字 (2,300字)	漢字, 仮名, 英・数字, 句読点
図形 (30種)	直線図形: 三角形, 正方形, 六角形, 五边形, 十字形 曲線図形: 半円, 楕円, 円, 卵形, 扇形
編集記号 (20種)	文字編集: (削除) A-B-C, (文字列挿入) A B C, (入れ替え) [A] [B] 図形編集: (拡大) □ → □, (90度回転) ㄣ → ㄥ, (移動) ○ → ○

図4 認識可能な文字・図形の例 文字は平仮名, 片仮名, 常用漢字など2,300字, 図形はフローチャート図形などの2次元図形30種, 更に文字・図形の校正のための編集記号20種が認識できる。

Aided Design)システムなどで実用化されているが、この方式は、次のような点に特徴がある。

(1) シンボルをJIS校正記号とし、一般に人が原稿の校正を行なっている形と同じものとした。

(2) 同一の編集を行なう場合でも、数種類の校正記号を定義することで、どの記号で入力しても動作するようにした(例: 削除記号として、—、X、/を定義した)。

(3) 一つの編集作業を行なう場合、オブジェクトの指定コマンドの指定が必要であるが、オブジェクトの指定を校正記号のサイズを変えることによって行なえるようにした。

(4) 編集の一連のストローク入力の書き順をフリーにし、また3.1のモードレスと連動して、どのように入力しても編集できるようにした。

このようにして開発した編集方式を削除を例にとって、他の方式と比較したものが図5である。これは一例であるが、1本のストローク入力で削除というコマンドと、“BCD”(文字列)というオブジェクトを一度に指定できる本方式は、キーボードやアイコンによる方式に比較して、指定のための操作回数の削減、操作することによる思考の乱れ防止などから、より良い方式と考える。

例	削除	(A) 先頭	(B) 末尾	(C) 末尾	(D) カーソル
方式	具体例				考え方
キーボード	削除キー	カーソル移動	カーソル移動	先頭位置指定	オブジェクト指定 コマンド → カーソル移動 → 位置指定
アイコン	削除アイコン	先頭文字指定	末尾文字指定		オブジェクト指定 コマンド → 文字指定
手書き編集記号	(A) (B) (C) (D) (E)				コマンド + オブジェクト指定

図5 各種編集方式の比較 文字列(B)(C)(D)を削除する場合の各種編集方式の操作方法を示す。手書き編集記号では、コマンドとオブジェクト指定を一つのストロークで表現しているため、他の方式に比べ操作性が良い。

4 手書き文字認識方式

4.1 認識方式概要

文字認識方式は、文字特有の性質、構造に着目する構造解析法²⁾と文字を2次元のパターンとしてとらえるパターンマッチング法³⁾とに大別される。ここでは、それぞれ性質の異なる字種(漢字、平仮名、片仮名及び英・数字)を認識対象とすることから、(1)認識処理が字種に依存しない、(2)辞書の作成及び追加が容易、という点で、後者の認識方式とした。図6にその認識処理手順を示す。

4.2 特徴抽出

特徴抽出は、前処理の後で行なう。パターンマッチング法では、文字をどのような特徴パラメータで記述するかが重要である。そこで、「文字は、数種のストロークをある規則(位置、大きさ、接続関係などを規定)のもとに組み合わせたもの」という観点から、文字の記述をストローク単位とし、特徴パラメータは、ストロークの筆記位置、形状、大きさを如実に示すものとした。具体的には、図7に示すように、(1)ストロークの始点、終点座標値 P_S , P_E (位置情報)、(2)ストロークを N 等分した N 個のセグメントの近似方向ベクトル系列 $\{\Delta P_n\}$ (形状、大きさ情報)である。

この特徴パラメータを用いて、入力文字と辞書文字の対応

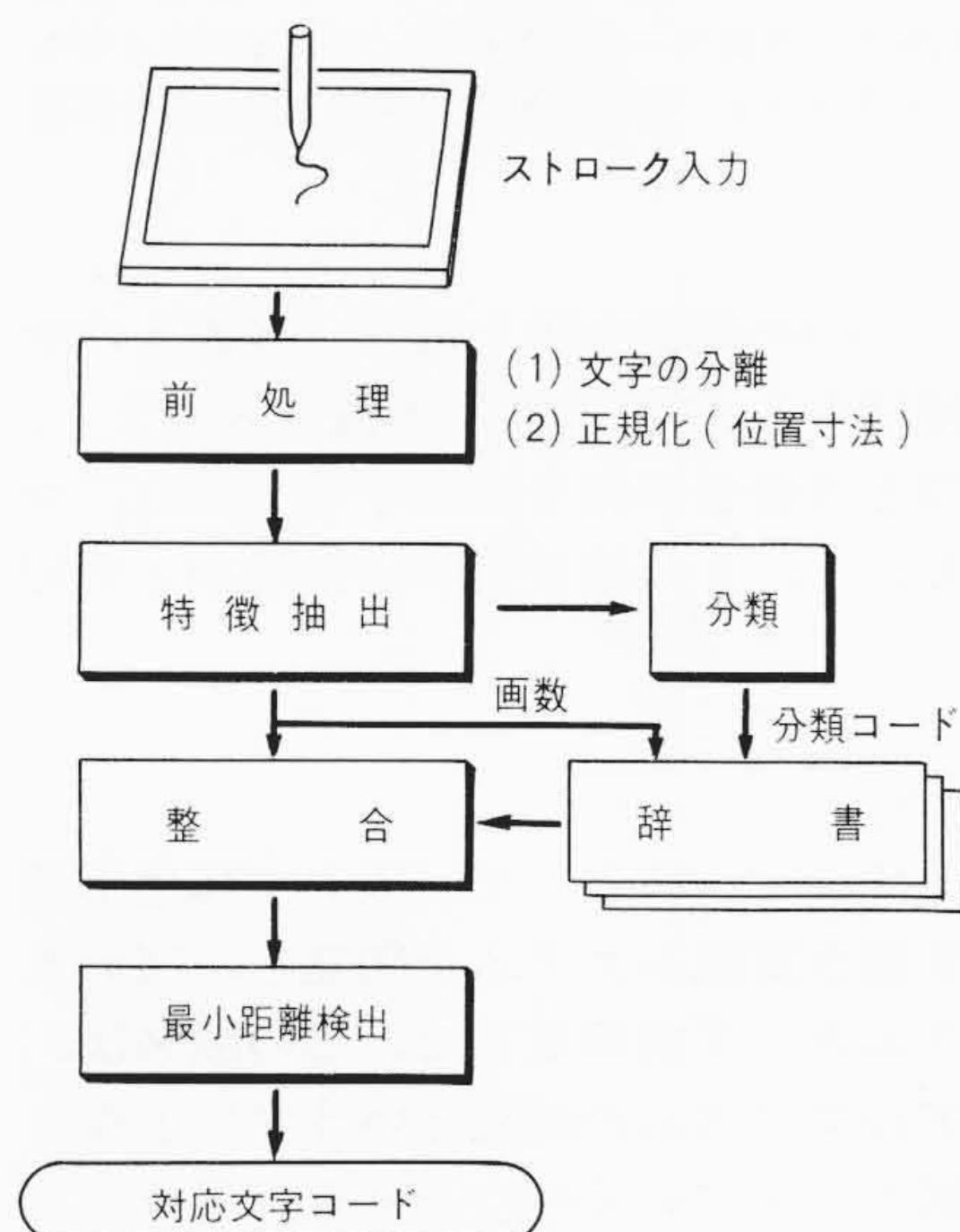


図6 認識処理手順
入力一体化平面ディスプレイ上に筆記した文字の筆点座標系列に対して、文字の大きさ、位置の正規化などの前処理を行なう。その後、文字を構成するストロークの特徴を抽出し、あらかじめ各文字ごとに同様な処理で特徴抽出した辞書との類似度を求め、最大類似度を与えるものを認識結果とする。

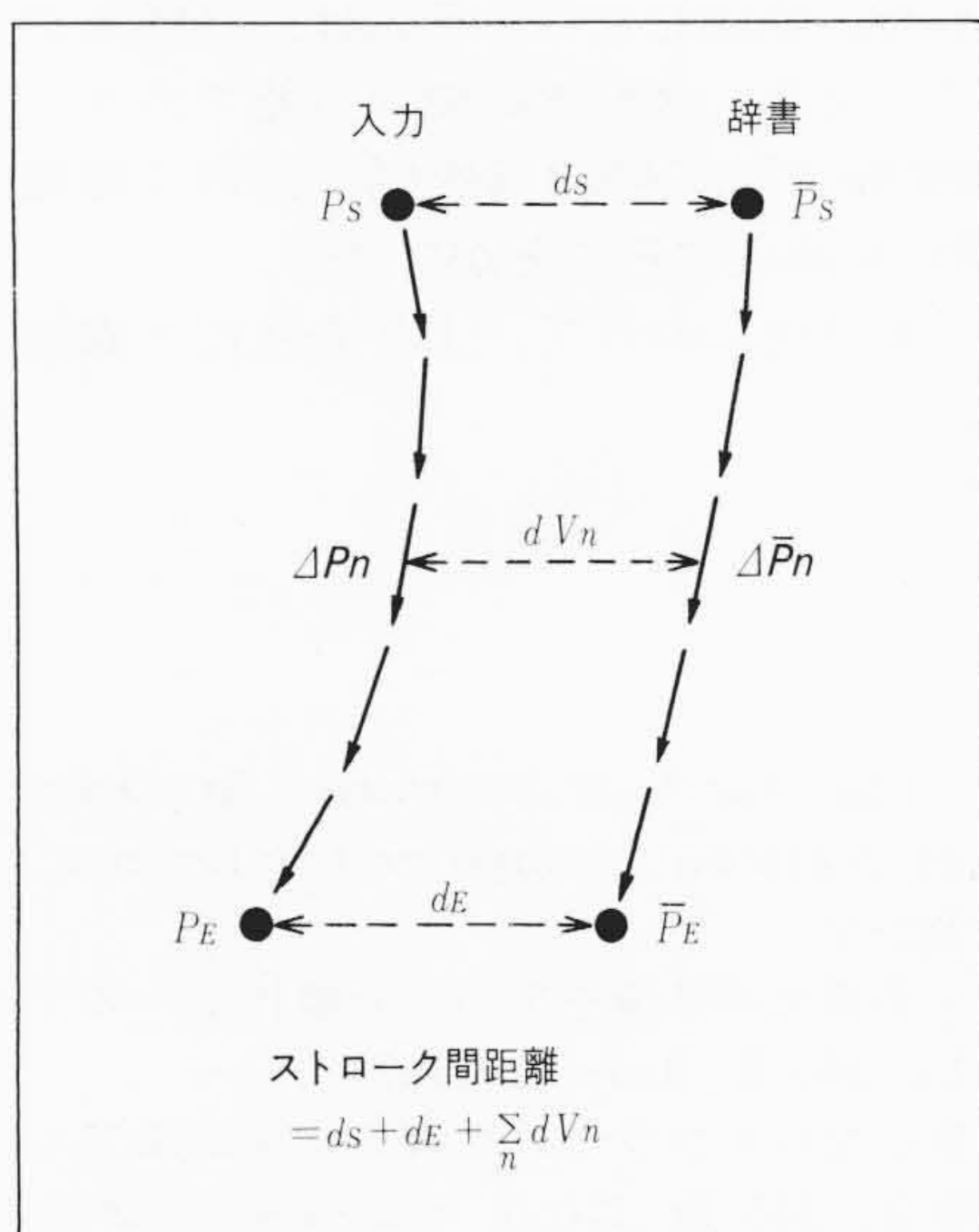


図7 特徴パラメータ
入力、辞書ストロークの特徴パラメータをそれぞれ P_S , P_E , $\{\Delta P_n\}$, $\bar{P}_S$, $\bar{P}_E$, $\{\Delta \bar{P}_n\}$ とすると、ストローク間距離 d は、始点間距離 $d_S = |P_S - \bar{P}_S|$ 、終点間距離 $d_E = |P_E - \bar{P}_E|$ 、方向差ベクトルの大きさ $d_{Vn} = |\Delta P_n - \Delta \bar{P}_n|$ の和を用い、 $d = d_S + d_E + \sum_n d_{Vn}$ で表わされる。 n の値は2、ただし6画以下は5とする。

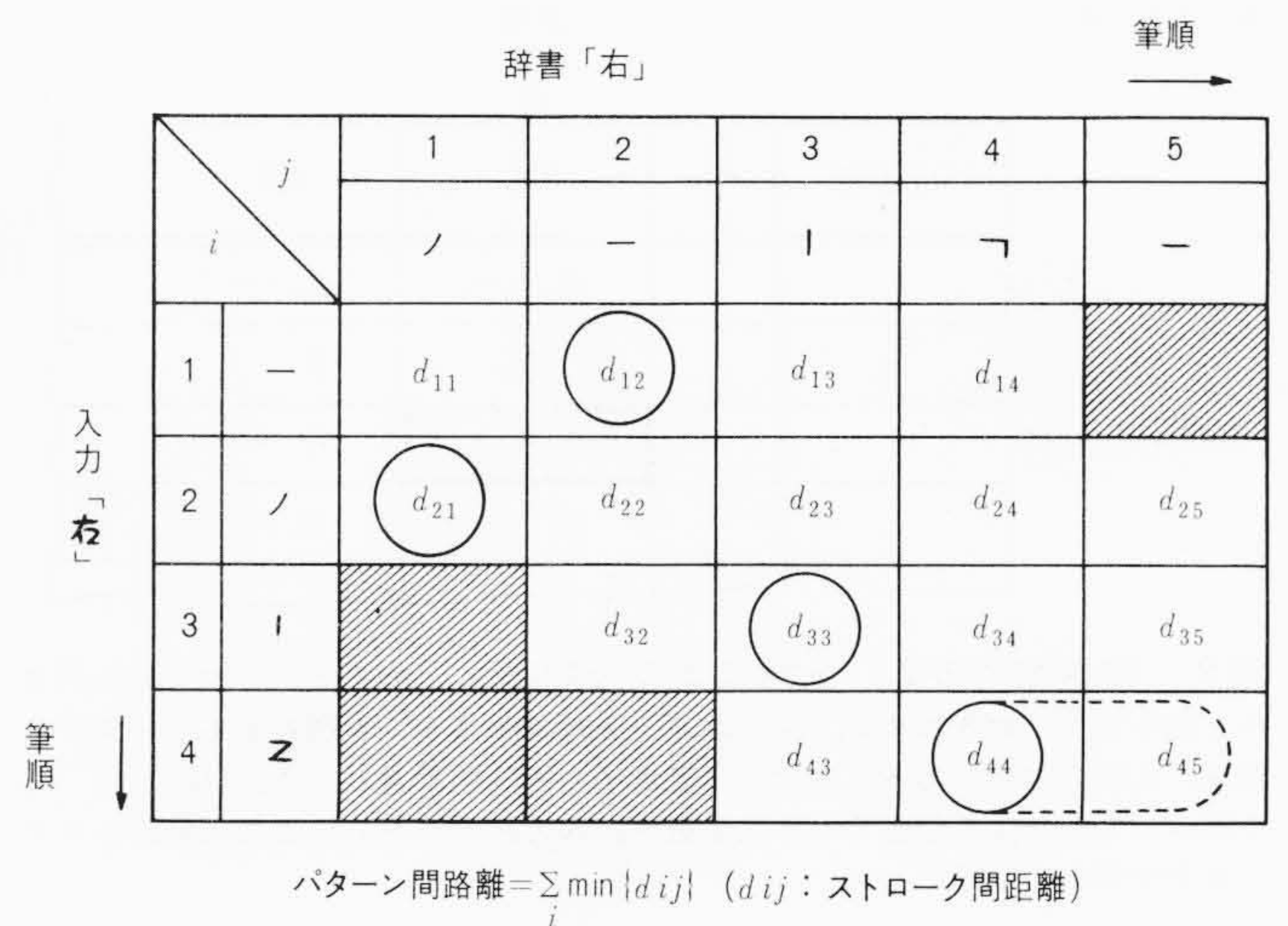


図8 ストローク対応図 ○印を付けた入力と辞書のストロークが対応し、破線は未対応ストロークを他のストロークとつなぎ、一つのストロークとしたことを示す。一般的に変動する筆順や画数の範囲を考慮して、図中の斜線部を整合演算から除外し、認識率を低下させないで処理量の低減を可能とした。

するストローク間の距離を、両者の特徴パラメータ間距離の総和として求める方式を、線分方向差距離法と称している。

4.3 整合

入力文字と辞書文字とのパターン間距離は、対応するストローク間距離の総和として求める。しかし、文字の筆順や画数は筆者によって様々であるため、入力文字と辞書文字のストロークを正しく対応させなければならない。その方法を図8に示す。線形計画法的手法により、両者のストロークを1対1に対応させ、対応したストローク間距離の総和が最小となるストロークの組合せを選択する(同図中○印)。入力と辞書文字の画数が異なる場合、未対応のストロークが生じるので、これをパターン間距離が最小となるように他のストロークとつなぎ、一つのストロークにし(図中○印)、両者の画数を合わせている。これにより、筆順、画数の入力筆記制限の緩和を可能にした。

4.4 入力文字の分類による認識処理の高速化

入力文字をあらかじめ分類して、整合すべき辞書文字数を絞り、整合処理量を低減する「書き始めの2, 3画による分類法」を開発し、認識処理の高速化を図った。これは、図9に示すように、3画まで筆記された時点で、分類用の2画、3画パターンとの類似度(筆点列の相関係数)を計算し、最大類似度を与えるカテゴリーと、入力文字の画数で分類する方法である。これにより、整合すべき辞書文字数を $\frac{1}{5}$ 以下にし、筆記速度に追従する認識応答速度を可能にした。

5 手書き図形認識方法

5.1 認識方式概要

図形認識方式は、文字と比較すると(1)標準的な画数がない。(2)標準的な筆順がない。(3)回転して入力するなどの図形特有の課題を解決しなければならない。そこで、これら三つの課題を解決する手法として入力ストロークを「直線」と「曲線」の2種の線分要素に変換し、認識する「線分セグメント法手書き図形認識方式」を開発した。

更に、図形間の区切りを意識せずに入力できるようにするため、図形入力時の特徴を考慮した「一次元DP(ダイナミックプログラミング)法活用図形切出し方式」を開発した。

例：入力「板」

		画数	
		8画	9画
分類コード	分類辞書		
	イ	価, 供, …	信, ……
	++	苦, 英, …	茶, ……
	木	板, 林, …	柱, ……
	扌	招, 扌, ……	指, ……

図9 書き始めの2, 3画による分類法 「板」という文字が入力された例を示す。書き始めの3画により、分類辞書「木」を選択する。一文字入力後、8画の文字であることから、整合すべき分類辞書として、「板, 林, …」を選ぶ。これにより整合すべき文字数を、入力画数に等しい画数に登録されている全文字の3以下とした。

	処理回数	切出し図形の線分要素番号
手書き入力1	1	1 2 3 4
	2	1 2 3
	3	2 3 4
	4	2 3
	5	1 2
	6	3 4
手書き入力2	10	1 2 3 4 5 6 7 8
	11	1 2 3 4 5 6 7
	12	2 3 4 5 6 7 8
	13	2 3 4 5 6 7

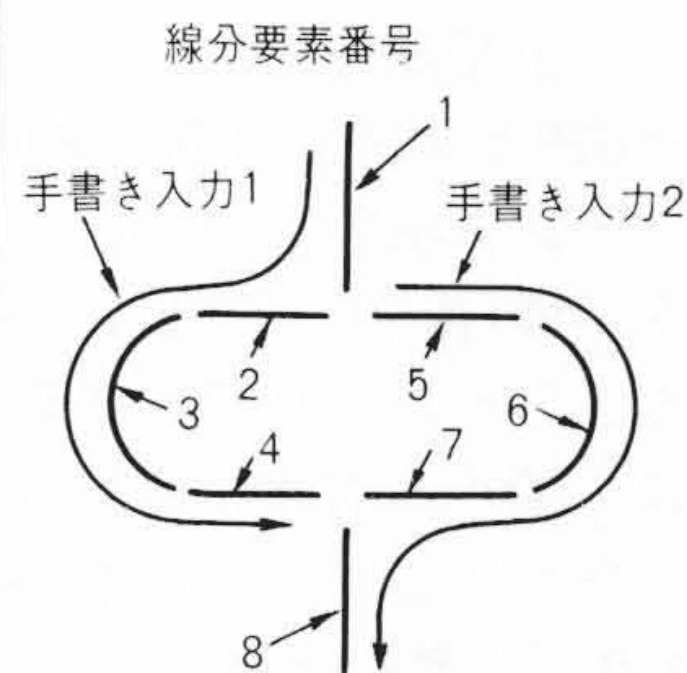


図11 図形の切出し処理 手書き入力1(線分要素番号1~4)では、図形は切り出せなかった。手書き入力2(5~8)を加えた(処理回数10)後、4回目(処理回数13回目)で端子記号に相当する線分要素2~7の切出しに成功している。

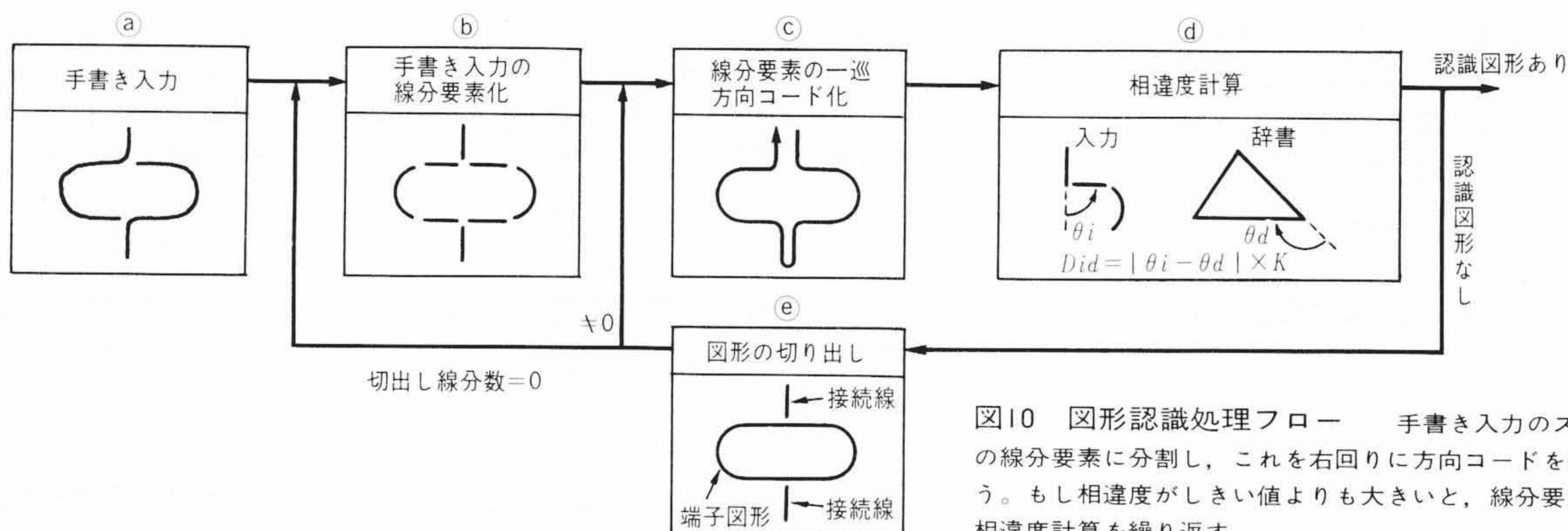


図10 図形認識処理フロー 手書き入力のストロークをまず直線と曲線の線分要素に分割し、これを右回りに方向コードを整理後、相違度計算を行なう。もし相違度がしきい値よりも大きいと、線分要素の再構成を行なって再度相違度計算を繰り返す。

5.2 線分セグメント法手書き図形認識方式

図10に本方式の処理手順を示す。タブレットでサンプリングした手書き入力の座標データは、1ストロークごとに処理される。1回の入力で図形が認識されない場合は、あらかじめ設定した時間内に次の手書き入力があれば、そのストロークデータを加えて再度処理する。以下詳細に本方式について説明する。

(1) 手書き入力の線分要素化

画数フリーとするため、図10aの手書き入力を同図bに示す線分要素(直線及び曲線)に変換する。そのためにまず、手書き入力を折線近似し、次に曲率半径により直線と曲線を抽出する。

(2) 線分要素の一巡方向列発生

図10cに示すように、筆順フリーとするため、一方向に線分要素を巡る線分要素列を作成する。

(3) 相違度計算

図10dに示すように、入力図形と辞書図形の間で下式により相違度 D を求める。

$$D = \sum_{i=1}^n \left| \left(\begin{array}{c} \text{辞書図形の} \\ \text{線分要素間} \\ \text{角度差} \end{array} \right) - \left(\begin{array}{c} \text{入力図形の} \\ \text{線分要素間} \\ \text{の角度差} \end{array} \right) \right| \times K$$

ここに i : 線分要素数

K : 特徴角重みパラメータ

この式に示すように、線分要素間の角度差を比較することで、図形が回転していても的確に認識することが可能である。

(4) 図形の切出し

図形の区切りを意識することなく入力できるようにするため、1次元DP手法を用いて、線分要素の組合せを変更して図形を切り出す方式を採用した。図11に処理の一例を示す。入力された線分要素例から、最後の要素、最初の要素という順

に切り離しては、上記(1)~(3)の処理を繰り返し、辞書とのマッチングがとれるまで続行する。この例では手書き入力2を、入力後13回目に1番目と8番目の線分要素を切り離し、マッチングに成功している。1, 8の線分要素は接続線と判定される。

6 結 言

新しいマンマシンインタフェースとして、入力一体化平面ディスプレイを用いた手書き認識システムを開発し、図形入り文書の入力編集システムとして実用化した。この開発は、手書き文字、図形認識アルゴリズムの開発というソフトウェア技術、入力一体化平面ディスプレイのハードウェア技術、更には、それらを組み合わせたシステム化技術がうまく融合することによって実現した。完成したシステムは、ほぼ人が書くスピードに合わせて、文字、図形の認識が可能であり、所期の目標(文字：認識率99.5%、速度0.5秒/字、図形：認識率95%、速度0.5秒/図形)を達成することができた。

今後はこうした技術を製品化に向けて、より高度化・成熟化していく予定である。

参考文献

- 1) D.C.Smith, et al.: The Star User Interface: An Overview Proceedings of National Computer Conference, Vol.51, 515~528(1982-6)
- 2) 寺井, 外: オンライン手書き文字認識を用いた原稿校正システム, 情報処理, Vol.15, No.6, 419~427(1974-6)
- 3) 小高, 外: 筆順に依存しないオンライン手書き文字認識アルゴリズム, 電子通信学会, J65-D, No.6, 679~686(1982-6)