

漢字ドットパタンの作成処理

森 克己

中野 博隆

日本電信電話公社

武蔵野電気通信研究所

1. ま え が き

1.1 目的

コード情報としての計算機の出力情報は最終的に人間に容易に理解できる文字情報へと変換されることが必要である。

この変換過程で文字パターン発生器が必要となる。文字パターン発生器では、そのシステムに含まれる出力装置の特性に応じて、最終的に表示あるいは記録される文字が読み易くなる文字パターンを発生してやる必要がある。ドット文字パタンの発生を考える時には、大きさ、字体ともに適切にデザインされた文字パターンを文字パターン発生器に準備しておくことが必要となる。

一方、文字情報を扱おう計算機システムの普及に伴ない、漢字プリンタ、ファクシミリ受信機あるいは液晶パネル、プラズマパネル、キャラクタディスプレイなど、種々の出力装置が開発され、使用されるに至ってきた。

この結果、種々の漢字ドットパターンが必要とされるようになった。

従来、漢字ドットパタンの作成は専門のデザイナーの手作業によって行なわれており、漢字字種の多さ、パタンの複雑さのために、多大の労力と時間を要し、種々の漢字パターンを手軽に準備することは困難であった。

より効率的な漢字パタンの作成法として、計算機と人間との会話形の漢字パターン作成システムを考え、本報告では、パターン作成のための計算機による漢字ドットパタンの処理について述べる。

1.2 漢字ドットパターン作成システムの概要

図1に漢字パターン作成処理の流れの概略を示している。

本システムでは、カード上の印刷文字を母形として用いており、処理は次の3段階に大別することができる。

(A) 印刷文字の読みとりと2値化

FSSを用いて印刷文字を 96×96 のドットマトリクスパターンとして読取る。

(B) 計算機による漢字ドットパタンの処理

FSSで読みとられた 96×96 の漢字ドットパターンを 32×32 の標準次数の漢字ドットパターンに変換した後、

- ① 品質改善処理
- ② 任意次数への次数変換処理
- ③ 字体変換処理

を行なう。

(C) 会話修正処理

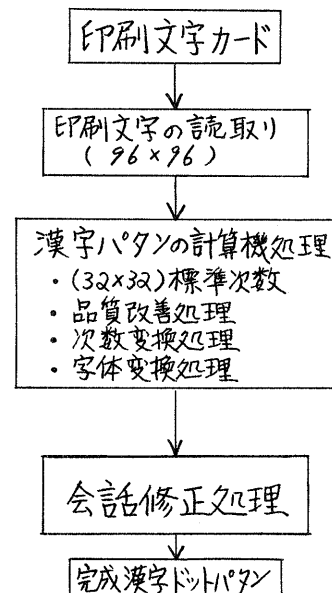


図1. 漢字パターン作成処理の流れ

計算機処理が行われた漢字ドットパターンをディスプレイ上に表示し、ライトペンを用いて、オペレータが漢字パターンの最終的な修正を行なう。

2. 基本的考察

2.1 印刷文字

我が国で用いられる字体は、ゴシック体と明朝体に大別されるが、両者の形状の差は、おおむね次の2点にあると考えられる。

①ストロークの形状：明朝体では、ストローク幅がそれぞれの中で変化し一定でなく、線端には、いわゆる‘うろこ’などの飾りを有している。他方、ゴシック体では、ストローク幅はほぼ一定で、飾りもない。

②水平・垂直ストローク幅の比：水平ストロークと垂直ストロークの幅の比が、ゴシック体では1に近い値をもつものに対して、明朝体では $\frac{2}{3}$ と1に比べて小さい値をとるものが多い。

2.2 漢字ドットパターン

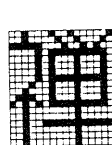
漢字ドットパターンの例を図2に示している。
漢字ドットパターン d を次のように定義する。

$$d = \{a(i, j)\} \quad \begin{matrix} i = 1 \sim m \\ j = 1 \sim n \end{matrix} \quad (2-1)$$

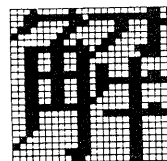
こゝで

$$a(i, j) = \begin{cases} 0 & ; \text{白ドット} \\ 1 & ; \text{黒ドット} \end{cases}$$

m, n をそれぞれ縦サイズ、横サイズとよぶ。



(18×16)



(24×24)

図2. 漢字ドットパターンの例

縦、横サイズがそれぞれ m, n の漢字ドットパターンを $m \times n$ 漢字ドットパターンと表わす。

漢字ドットパターンのサイズとしては $18 \times 16 \sim 32 \times 32$ が主に使用される。

2.3 漢字パターンの性質

漢字パターンの主要な特徴としては、水平・垂直方向のストロークが多いことが良く知られている。

本報告が対象としている 32×32 程度以下の漢字ドットパターンに限定すると、パターンの表現能力に大幅な制限をうけ、微妙なニュアンスの表現は不可能となり、新たに次のような特徴を追加することができる。

(1) 水平・垂直に近い勾配のストロークは全て完全な水平・垂直直線としてしか表現できない。また、線幅の微妙な差は表現できず、 $1 \sim 3$ ドット幅の範囲で、ドット単位で量子化される。

(2) 斜線・曲線の滑らかな表現は困難である。斜線・曲線の表現形式は、勾配や曲率、それに線幅によって異なり、一意的に表現形式が定まっていない。

(3) (1), (2)をうけて、ドット配置の乱れは水平・垂直直線部で特に目立ち、斜線・曲線部でのそれは、あまり目立たない。

以上述べたような点から、漢字ドットパターン作成のための処理としては、水平

垂直直線部に対する整形処理が、主要な検討対象となる。

2.4 基本的な処理方法

前節で、水平・垂直直線が、斜線や曲線に比べて、漢字ドットパタンの文字品質に対してより重要な役割を持っていることを述べた。

本報告では、漢字ドットパターンに対する基本的処理方法として、水平・垂直直線と斜線・曲線とを分離して処理する方法を用いた。

図3に処理の流れの概要を示している。

(1) ランの分離

32×32の原パターンから、水平・垂直方向に連続する黒ドットを「ラン」として分離する。

(2) 水平・垂直直線の抽出

分離したランを用いて、水平・垂直直線を抽出する。

以下、水平・垂直直線のみから成るパターンを水平・垂直直線パターン、残されたパターンを斜線・曲線パターンとよぶ。

(3) 水平・垂直直線処理と斜線・曲線処理

水平・垂直直線と斜線・曲線とを別々に分離して処理を行なう。

水平・垂直直線処理としては、線幅規格化処理、次数変換処理における座標変換処理、字体変換処理における飾り付け処理が含まれる。

一方、斜線・曲線処理としては、不要点除去処理、座標変換処理が含まれる。

(4) 再合成

必要な処理を終えた、水平・垂直直線パターンと斜線・曲線パターンを再合成し、最終的な漢字ドットパターンを作成する。

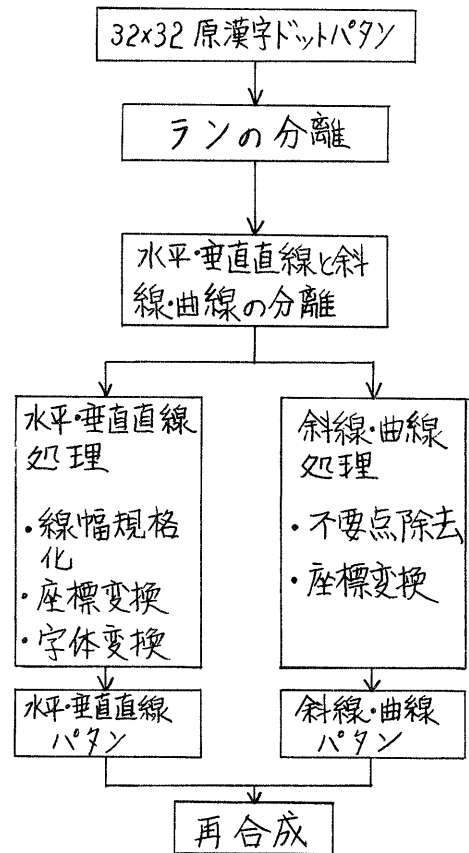


図3. 漢字ドットパタンの計算機処理

3. 漢字パターン整形処理¹⁾

本処理では、フライングスポットスキャナから読みとった96×96漢字ドットパターンを標準次数の32×32漢字ドットパターンへ変換した後、ランの分離・統合処理を行ない線幅の規格化、不要点除去を行なう。

3.1 標準次数への変換

フライングスポットスキャナから読みとった96×96漢字ドットパターンを3×3のサブマトリクスへ分割し、各サブマトリクス内の黒ドットの個数に閾値

処理を行ない、 32×32 の標準次数漢字ドットパターンへと変換する。変換結果の標準次数漢字ドットパターンを、以下、パターン処理における原漢字ドットパターンとする。

原漢字ドットパターンの1例を図4に示している。

図4にも見られるように、原漢字ドットパターンは、水平・垂直直線幅が統一されておらず、また、直線に多数のノッチ(ギザギザ)がある。人手によるこれらの修正は大きな作業量となるので計算機による修正処理が重要となる。

前割剛

図4. 原漢字ドットパターンの例

3.2 水平・垂直直線と斜線の分離

ランとランレングスを次に定義する。

ラン：水平・垂直方向に連続する黒ドットの集まり。

ランレングス：ランを構成する黒ドットの個数

図5に水平・垂直直線と斜線の分離処理のフローを示している。以下、図に従って説明する。

(1) ランの分離

原漢字ドットパターンから、閾値 R_{th} 以上のランレングスをもつランを分離する。分離して得られるパターンを「ランパターン」とよび、残ったパターンを「オブリークパターン」とよぶ。 $R_{th}=7$ に対して得られるランパターンとオブリークパターンを図6に示す。

適切な閾値 R_{th} を設定することにより、水平・垂直直線を構成するランの大部分を分離することが可能である。

この R_{th} の最適値は漢字ドットパターンで異なるため、漢字パターン作成システムとしてみた時には、オペレータが指定するパラメータとなっている。

(2) 水平・垂直直線の抽出(ランの統合)

図6からも分かるように、1本の水平・垂直直線から1~4本のランが分離される。したがって、これらの位置、長さが異なる複数本のランから1本の水平・垂直直線を抽出することが必要となる。

そこで、次のような「ラン束」を定義する。

縦ラン束：次の条件を満たす1本のランの集まり。

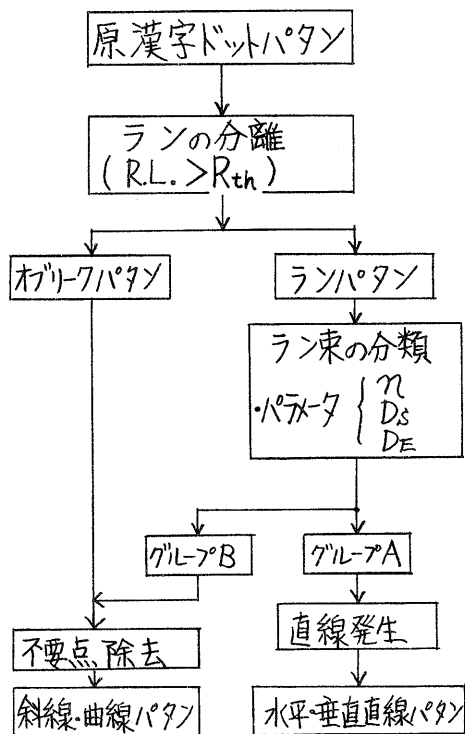


図5. 水平・垂直直線と斜線・曲線の分離

前割剛剂力助勢

前割剛剂力助勢

前割剛剂力助勢

図6. ランパターンとオブリークパターンの分離

$$\begin{aligned} & \{(ISV_1, JV_1), (IEV_1, JV_1)\} \\ & \{(ISV_2, JV_2), (IEV_2, JV_2)\} \\ & \vdots \\ & \{(ISV_n, JV_n), (IEV_n, JV_n)\} \end{aligned} \quad (3-1)$$

$$\begin{aligned} \text{但し } & JV_2 = JV_1 + 1 \\ & JV_3 = JV_1 + 2 \\ & \vdots \\ & JV_n = JV_1 + n - 1 \end{aligned} \quad (3-2)$$

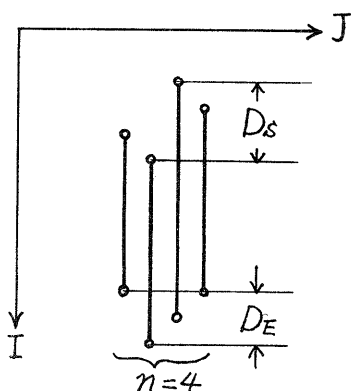


図7.座標系と縦ラン束

かつ

$$ISV_k \leq ISV_{k+1} \leq IEV_k \quad \text{or} \quad ISV_{k+1} \leq ISV_k \leq IEV_{k+1} \quad k=1, 2, \dots, n \quad \text{である。}$$

横ラン束：縦ラン束と同様に定める。

ここで $\{(IS, JS), (IE, JE)\}$ は始点 (IS, JS) 、終点 (IE, JE) のランまたは直線を意味する。

図7に座標系と縦ラン束を図示している。

以下、縦ラン束を中心に説明する。

ここでラン束の特性パラメータとして n, D_s, D_E を以下のように定める。

n ：ラン束の幅

$$\begin{aligned} D_s &= \max\{ISV_1, ISV_2, \dots, ISV_n\} - \min\{ISV_1, ISV_2, \dots, ISV_n\} \\ D_E &= \max\{IEV_1, IEV_2, \dots, IEV_n\} - \min\{IEV_1, IEV_2, \dots, IEV_n\} \end{aligned} \quad (3-3)$$

図7にこれらのパラメータを図示する。

これらのパラメータ n, D_s, D_E を用いて、ランパターン内のラン束を2つのグループA, Bに分類する。このグループへの分類の適否が、最終の漢字ドットパターンの品質を左右する。このグループへの分類法については後に詳述する。

分類の結果、グループAに属するラン束は「水平・垂直直線から分離されたランから成るもの」と最終的に判定されたものであり、グループBに属するラン束は「斜線部を構成するラン」と最終的に判定されたものである。この結果、グループBに属するラン束は再びオブジェクトパターンの中へもどされる。

一方、グループAのラン束を用いて、垂直直線を次のように抽出する。

$$\text{抽出された直線: } \{(\min(ISV_k), JV_n), (\max(IEV_k), JV_n)\} \quad k=1 \sim n.$$

次にラン束のグループ分けの方法について述べる。

ここで問題となるのは、「カ」などに見られるように、殆んど垂直直線でありながら、その一部が曲線であったり、垂直に近い斜線を含む線分の処理である。これらの線分を直線として処理すると漢字ドットパターンは不自然なものになってしまう。この例を図8に示す。

そこで、次の3通りの方法によって、ラン束のグループ分けのシミュレーションを行なった。

方法I：32 × 32 程度以下の漢字ドットパターンでは水平・垂直直線幅は3ド

ット以下であるとして, D_S , D_E に無関係に $n \leq 3$ のラン束はグループAに, それ以外はグループBに分類する。

方法[IV]: 水平・垂直直線から分離されたラン束の D_S , D_E は斜線・曲線から分離されたラン束の D_S , D_E より十分小さいとして, n に無関係に $D_S \leq 5$, $D_E \leq 5$ ならば, そのラン束はグループAに, そうでなければグループBとする。

方法[V]: 斜線の勾配あるいは曲線の曲率の概念を判定基準に導入して, 次の3条件

$$\begin{cases} \bullet n=1 \\ \bullet n=2 \text{ and } D_S \leq 3 \text{ and } D_E \leq 3 \\ \bullet n=3 \text{ and } D_S \leq 5 \text{ and } D_E \leq 5 \end{cases} \quad (3-4)$$

のいずれかを満たすラン束はグループA, そうでないものはグループBとする。

グループ分けの結果は, 直接, 斜線などを含むオブリークパターンの処理にも関係しているので, グループ分けのシミュレーション結果は3.4節で考察する。

3.3 不要点除去処理

図6に示すように, 原漢字ドットパターンから分離されたオブリークパターンには, 斜線・曲線以外に多数の不要点が含まれている。

これらの不要点の大多数は原漢字パターンの水平・垂直直線のノッチに対応するものであり, 水平・垂直直線と並んで現われるため, 次の処理で斜線・曲線と容易に区別し除去することができ。

オブリークパターン上で

$a(i, j) = 1$ に対して, 次の2条件

$$\begin{cases} \bullet a(i+1, j) = \phi \text{ and } a(i-1, j) = \phi \\ \bullet a(i, j+1) = \phi \text{ and } a(i, j-1) = \phi \end{cases}$$

のいずれかが満足されれば

$a(i, j) = \phi$ とする。

(3-5)式の条件を図9に図示する。また, 図10に不要点除去処理の結果を示している。

d	d	d
0	$a(i, j)$	0
d	d	d

d	0	d
d	$a(i, j)$	d
d	0	d

d: don't care

図9. 不要点除去条件。

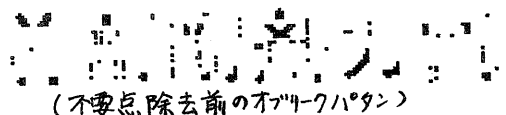
3.4 再合成

3.2節で抽出した水平・垂直直線 $\{(ISV, JV), (IEV, JV)\}$ を用いて, n 本の直線

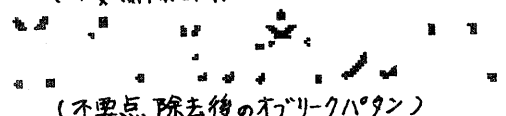
$$\begin{aligned} & \{(ISV, JV), (IEV, JV)\} \\ & \{(ISV, JV-1), (IEV, JV-1)\} \\ & \vdots \end{aligned}$$

前割剛剤力割

(原パターン)



(不要点除去前のオブリークパターン)



(不要点除去後のオブリークパターン)

図10. 不要点除去処理の結果。

{(ISV, JV-n+1), (IEV, JV-n+1)}

を発生することにより、線幅が規格化された水平・垂直直線パターンが得られる。
この水平・垂直直線パターンと3.3節で得られた斜線・曲線パターンを合成することにより、整形処理を施した漢字ドットパターンが得られる。

図11に、3.2節で述べた通りのグループ分けの方法について、その結果を示している。

同図から分かるように「前」,
「割」,「副」のように殆んど水平・
垂直直線のみから成る漢字パタ
ンに対しては方法[I]で最も良
く線幅の規格化処理が行なわれ
ている。方法[II],[III]につい
ては、水平・垂直直線の識別条件
が方法[I]に比べて強いため、
線幅の規格化、ノッチ除去効果
が充分得られていない。

逆に「力」,「加」に含まれる水
平・垂直に近い傾きの斜線・曲
線については、方法[II],[III]が
これらの線分のニュアンスを良
く保存している。

また、「劣」,「努」などに含まれる水平・垂直方向に対して大きな角度をなす斜
線・曲線に対しては、いずれの方法を用いても、大差なく良好に保たれることが
分かる。

水平・垂直直線が漢字パターン内に占める割合が大きいこと、斜線・曲線の乱れ
が文字品質に及ぼす影響が比較的小さいことを考慮して、水平・垂直直線に対す
る整形効果が最も大きい、方法[I]をこゝでは用いることにした。

3.5 整形処理の評価

以上の整形処理の結果は次のようになる。

- ① 斜線を含める漢字ドットパターンに対しては人手による修正が不要となる。
- ② 水平・垂直直線と、これらと大きな角度をなす斜線・曲線のみから成る漢
字ドットパターンに対しては、人手による修正は斜線部のみ行なえば良い。
- ③ 水平・垂直に近い傾きの斜線・曲線を含む漢字パターンに対しては、それら
の線分を損なうことがある。

32×32程度の漢字ドットパターンでは、上記①,②の範ちゅうに属する漢字
が大きな割合を占め、人手による修正作業量を減らす目的としての整形処理は極
めて大きな効果をもつものと思われる。

4. 次数変換処理^{2),3)}

4.1 次数変換の問題点

こゝでは与えられた標準次数の漢字ドットパターンから所要の次数をもつ漢字ド

前 割 副 力 加 劣 努

方法[I]

前 割 副 力 加 劣 努

方法[II]

前 割 副 力 加 劣 努

方法[III]

図11. 整形処理結果.

ットパターンを作成する次数変換処理について述べる。

最も単純な次数変換法としては、原漢字ドットパターンの各ドットの座標に対して、比例的な座標変換処理を行なう方法である。しかし、この方法では水平・垂直直線の幅に不揃いが生じる欠点がある。

この欠点を避けるためには、水平・垂直直線をドットの集合としてでなく、線分として扱うことが必要となる。ドットパターンから線分情報を全て抽出することは困難な問題であるが、幸いなことに、漢字パターンの次数変換で特に問題となる水平・垂直直線が、3章で述べた整形処理において、既に抽出されているので、本漢字ドットパターン作成システム内で容易に次数変換処理が実行できる。

4.2 次数変換処理

3.2節の(2)で抽出された水平・垂直直線 $\{(i_1, j_1), (i_2, j_2)\}$ を次式に従って次数変換後の漢字ドットパターン内の水平・垂直直線 $\{(I_1, J_1), (I_2, J_2)\}$ へ変換する。

$$\text{変換式; } I_k = \left[\frac{M-1}{m-1} \times (i_k - 1) + 1.5 \right], J_k = \left[\frac{N-1}{n-1} \times (j_k - 1) + 1.5 \right] \quad k=1, 2 \quad (4-1)$$

こゝで $(m, n), (M, N)$ はそれぞれ原漢字ドットパターンと変換後の漢字ドットパターンの次数である。 $[]$ はガウス記号である。

一方、斜線・曲線については、各黒ドット $a(i, j)$ を式(4-1)に従って、変換後の漢字ドットパターン内の黒ドット $a^t(I, J)$ に変換する。

次数変換結果の例を図12に示している。

前前前前前前前前前前前前前前前前

前前前前前前前前前前前前前前前前

(32X32) (31X31)

(20X20)

剣剣剣剣剣剣剣剣剣剣剣剣剣剣剣剣

剣剣剣剣剣剣剣剣剣剣剣剣剣剣剣剣

図12. 次数変換処理結果の例

5. 字体変換

2.1節で述べたように、明朝体とゴシック体の主要な相違点は、ストローク端における、いわゆる‘ウロコ’と呼ばれる飾りの有無と線幅の違いである。

図13に主要なウロコの種類を示す。

こゝでは、ゴシック体漢字パターンにこれらの飾りを付加することによる、ゴシック体から明朝体への字体変換処理について述べる。

5.1 ウロコの付加

ウロコの種類と位置は、水平・垂直直線の接続状態に応じて規則性をもっている。以下、水平・垂直直線の接続状態の識別処理について述べる。

1個の漢字ドットパターン内で抽出された水平・垂直直線がそれぞれ

垂直直線: $\{(ISV_m, JV_m), (IEV_m, JV_m)\}$ $m=1, 2, \dots, L_m$
 水平直線: $\{(IH_n, JSH_n), (IH_n, JEH_n)\}$ $n=1, 2, \dots, L_n$

であるとする。

(A) 「あたま」と「左上端飾」

垂直直線の始点 (ISV_m, JV_m) が他の水平直線 k と接しない場合、すなわち、次のいずれかの条件が成り立つ時

- $ISV_m \neq IH_k$
- $JV_m < JSH_k$
- $JV_m > JEH_k$

$k=1, 2, \dots, L_n$

のとき「あたま」をつける。また、垂直直線の始点に水平直線の始点 (IH_n, JSH_n) が接するとき、すなわち、

$ISV_m = IH_n$ and $JV_m = JSH_n$

のとき、「左上端飾」をつける。

(B) 「ウロコ」と「カドウロコ」

水平直線の終点 (IH_n, JEH_n) がどの垂直直線とも接しないとき、すなわち、

$JEH_n \neq JV_k$ or $JH_n < JSV_k$ or $IH_n > IEV_k$

$k=1, 2, \dots, L_m$

のとき「ウロコ」をつける。

また、水平直線の終点 (IH_n, JEH_n) が垂直直線の始点 (ISV_m, JV_m) に接するとき、すなわち、

$IH_n = ISV_m$ and $JEH_n = JV_m$

のとき「カドウロコ」をつける。

(C) 「左下端飾」と「右下端飾」

垂直直線の終点 (IEV_m, JV_m) が水平直線の始点 (IH_k, JSH_k) か終点 (IH_k, JEH_k) に接するとき、すなわち

$IEV_m = IH_k$ and $JV_m = JSH_k$

又は $IEV_m = IH_k$ and $JV_m = JEH_k$

のとき「左下端飾」または「右下端飾」をつける。

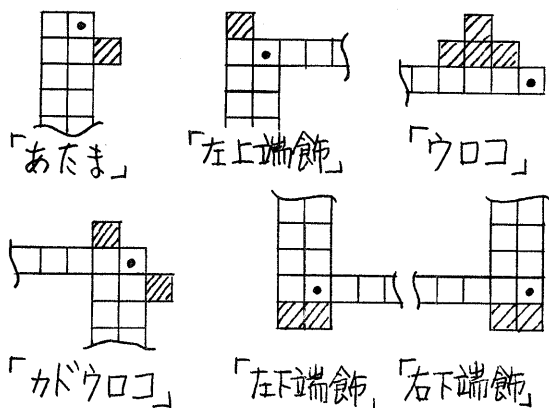


図13. 明朝体線端飾り。

5.2 線幅制御

水平直線を線幅1ドットで、垂直直線を線幅2ドットで発生し、水平直線と垂直直線の線幅を変える。

以上の字体変換処理結果を図14に示している。

前 剖 剣 劑 副

(原ゴシック体漢字パターン)

前 剖 剣 劑 副

(明朝体)

図14. 字体変換例

6. むすび

漢字の字種の多さ、字形の複雑さから人手による漢字ドットパタンの作成には多大の労力と時間を要するため、より効率的な漢字ドットパタンの作成システムが必要とされるようになってきた。

本報告では計算機との会話処理による漢字ドットパターン作成システムの一環として、計算機による漢字ドットパタンの整形処理、次数変換処理、字体変換処理について述べた。

作成対象である(32×32)程度の漢字ドットパターンでは、パタンの表現力には大きな制限をうけ、特に、斜線や曲線がもっと微妙なニュアンスの表現は不可能である。また、漢字パタンの特徴として、斜線・曲線がパターン内に占める割合は、水平・垂直直線に比べて小さいことがあげられる。

以上のような点から、斜線や曲線部のドットの乱れが漢字パターン全体の品質に与える影響は比較的小さく、主要な処理対象は水平・垂直直線となる。

そこで本報告では、漢字ドットパターンを水平・垂直直線と斜線・曲線に分離して処理することを基本的な処理方法とした。

水平・垂直直線と斜線・曲線の分離手段として、水平・垂直方向への黒ドットの連なりである「ラン」を導入することにより、十分有効な水平・垂直直線と斜線・曲線の分離が可能であることを示した。この結果を用いて、線幅の規格化、不要点の除去処理を行なうことにより、人手による修正作業量の大幅な軽減という所期の目的を達成することができた。

また、次数変換処理は整形処理過程で抽出された水平・垂直直線情報を用いることにより、線幅の不揃いという漢字ドットパタンの品質劣化を生じることなく次数変換が行なえる。

最後に、水平・垂直直線の接続状態を識別し、線端に適切な飾りを付加することにより、ゴシック体から明朝体へ字体変換が行なえることを示した。

以上述べたように漢字ドットパターン処理において水平・垂直直線と斜線・曲線の分離処理は極めて有効な手段であることが確認できた。

謝 辞 本検討を通して御指導いただきました釜江室長をはじめ、御討論いただいたグループの名氏に感謝いたします。

参考文献

- 1) 中野 森「漢字パターン作成のための前処理」信学会総合全国総会全国大会11/2
- 2) 森「ドット漢字パターンマトリクス of 次数変換法」信学論 Vol.60-D No.10
- 3) 斎藤 他「インクジェット式漢字記録の検討」電子通信学会研究会資料EC77-27.