

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4085183号
(P4085183)

(45) 発行日 平成20年5月14日(2008.5.14)

(24) 登録日 平成20年2月29日(2008.2.29)

(51) Int. Cl.		F I			
G09G	5/24	(2006.01)	G09G	5/24	690
G06F	17/21	(2006.01)	G06F	17/21	562Z
G06K	9/62	(2006.01)	G06K	9/62	G
G09G	5/00	(2006.01)	G09G	5/00	510J

請求項の数 2 (全 10 頁)

(21) 出願番号	特願2002-159808 (P2002-159808)	(73) 特許権者	398030229 株式会社 エヌティーアイ 三重県四日市市中村町2293番地1
(22) 出願日	平成14年5月31日(2002.5.31)	(74) 代理人	110000051 特許業務法人共生国際特許事務所
(65) 公開番号	特開2004-4302 (P2004-4302A)	(72) 発明者	中村 貴利 三重県四日市市松寺2丁目19番8号 株式会社中村技術研究所内
(43) 公開日	平成16年1月8日(2004.1.8)	審査官	福永 健司
審査請求日	平成17年5月16日(2005.5.16)		

最終頁に続く

(54) 【発明の名称】 遺伝的アルゴリズムによるフォント生成システム

(57) 【特許請求の範囲】

【請求項1】

手書き入力された文字パターンより、手書き文字フォントを生成して手書き文字を出力する文字フォント生成システムにおいて、

手書き文字を入力して記憶する入力手段と、

前記入力手段により入力される前記手書き文字を文字パターンに変換して出力するパターン変換手段と、

前記パターン変換手段により出力される前記文字パターンを認識して該当する文字コードを出力する文字認識手段と、

前記入力手段により入力される前記手書き文字を該文字の特徴を含むベクトル情報に変換して出力するベクトル変換手段と、

前記ベクトル情報を手書き文字の座標点フォントとして記憶するベクトル情報記憶手段と、

前記ベクトル情報記憶手段の座標点フォントを読み出し補間曲線関数による曲線フォントを遺伝的アルゴリズムGAにより演算生成するGA演算処理手段と、

前記GAにより演算生成された曲線フォントを記憶する手書きフォント記憶手段と、を備え、

前記GA演算処理手段は、

前記座標点フォントを入力して前記補間曲線関数をセグメント分割により直線で近似する補間曲線関数処理部と、前記セグメントの係数を個体として複数の個体を含む集団を形

10

20

成する集団形成部と、前記遺伝的アルゴリズム G A の世代数上限、個体数、交叉点、突然変異、評価関数を少なくとも含むパラメータの設定条件を記述した条件設定テーブルと、前記補間曲線関数処理部におけるセグメントの係数を個体とする近似計算に前記条件設定テーブルに設定された前記パラメータを用いて G A 演算を行う G A 制御部と、を有し、
予め記憶された標準パターン辞書の標準文字パターンに前記遺伝的アルゴリズムを適用して前記手書き文字フォントを生成することを特徴とする遺伝的アルゴリズムによるフォント生成システム。

【請求項 2】

前記 G A 演算処理手段は、G A により曲線フォントを演算生成する際に、複数の異なる文字パターンを生成して記憶する文字パターン生成部と、生成された複数の異なる文字パターンの中から利用者により選択されて記憶される文字パターン選択部と、を有することを特徴とする請求項 1 に記載の遺伝的アルゴリズムによるフォント生成システム。

10

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、主としてペン等の入力装置から入力された手書き文字を文字フォントデータとして出力する遺伝的アルゴリズムによるフォント生成システムに関するものである。

【0002】

【従来の技術】

従来、ペン等を用いて手書き文字を入力できる手書きワードプロセッサなどでは、最初に入力された手書き文字を認識して得られる文字コードによる一律の、キーボード入力と同じような定型文字パターンにして印字出力していたが、最近では各個人特有の肉筆感のある署名などの必要性から、手書き文字を入力して各個人特有の筆跡が生かされた手書き文字も出力可能な手書き文字処理装置が多くなっている。

20

【0003】

そうした従来の手書き文字処理装置としては、例えば、特開平 05 - 143769 号に開示されている「手書き文字処理装置」が知られている。

図 8 はこの従来の手書き文字処理装置の構成図である。ユーザーはペン 52 を持ってタブレット 51 上に、図 9 に示すような「T」という手書き文字を入力すると、手書き文字の筆跡は演算処理部 53 のベクトル変換回路 56 とパターン認識回路 57 へ入力される。

30

【0004】

パターン認識回路 57 はペン 52 の位置データに基づいて入力される文字をパターン展開する。パターン展開された手書き文字パターンにより標準パターン辞書 58 をアクセスして、パターンマッチングを図り該当する標準文字パターンを特定して、対応の文字コードを導出する。この導出した文字コードは「T」を表すもので、文字認識情報記憶部 60 にストアされる。

【0005】

次に、ベクトル変換回路 56 では、ペン 52 の位置データに基づいて図 9 に示すような、複数のサンプリング点 $a_1 \sim a_n$ 、 $b_1 \sim b_n$ を得て、手書き文字を構成する各ベクトルの始点、移動量及び終点を算出する。算出したベクトル情報（座標点フォント）は、図 10 に示すような形式でベクトル情報記憶部 59 にストアされる。ストアされるメモリ内容は、「移動量： $a_1 \sim a_2$ 」が座標点 a_1 から a_2 までの移動量 $[X_1, Y_1]$ として、「移動量： $a_2 \sim a_3$ 」が座標点 a_2 から a_3 までの移動量 $[X_2, Y_2]$ として示しており、「AD2」はメモリ・アドレスである。

40

また「始点： a_1 」～「終点： a_n 」までがベクトル A を表し、「始点： b_1 」～「終点： b_n 」はベクトル B を表している。このようにベクトル変換処理は、手書き入力された文字を点の集まりにして変換記憶している。

【0006】

次にラスト描画については、まず、プリンタ 55 は通常印字の場合は、文字認識情報記憶部 60 にストアされている文字コードを逐次読出し、欧文書体の場合は a r i a l、t i

50

mes など、漢字書体なら明朝、ゴシックなどのフォントを指定して通常のワープロ用の定型文字パターンに展開して印字を行う。

【0007】

一方、手書き文字データをそのまま手書き文字のまま印字出力するには、ベクトル情報記憶部59にストアされている図10に示すようなベクトル情報(座標点フォント)を読み出して、読み出したベクトル情報を曲線に変換するための関数(例えば、B-スプライン曲線関数、ベジェ曲線関数など)を用いて、プログラム処理によりドットの集合で表される曲線に変換する。このようにしてドットの集合で表された曲線のイメージデータはプリンタ55に与えられ逐次印字出力されるので、手書き文字が微妙な手書き文字の弧を顕著に表示しながら印字される。

10

【0008】

このように、図10に示すベクトル情報(座標点フォント)の各点間の弧をベジェなどの曲線関数により補間生成して、図9に示すような手書き文字を復元するが、通常ワープロの定型パターン文字印字の場合は、ASCII、JIS漢字コードなどの文字を指定する文字コードをストアしておいて、arial、明朝体などのフォントを指定し呼び出して印字を行うのに対して、手書き文字復元の場合は、明朝体などに相当するフォントを、ベクトル情報(座標点フォント)と、その座標点間の弧を補間するスプライン曲線関数やベジェ曲線関数を用いた曲線フォントで表す。従って、以降、本明細書の中で手書き文字フォントは、座標点フォント+曲線フォント、を示すものとする。

【0009】

ここで、弧補間用の曲線関数の例としてベジェ曲線関数を上げると、図11に示すように、一般的なベジェ曲線は、始点P0、終点P3、制御点P1、P2で構成され、形状を変えずに2つに分割可能で、分割を繰り返し再帰的に適用することで際限なく細分して、最小単位まで細分した直線の集合で近似的にベジェ曲線を復元するものである。

20

例えば、dmaxを設定された許容値とし、直線P0とP3の中点Qと曲線上の点Pとの距離をdとした場合に、 $d < d_{max}$ 、を分割の判定基準として曲線を分割・細分化する。dmax(フラットネス)が小さい程滑らかな曲線に復元できる。

【0010】

【発明が解決しようとする課題】

しかしながら、このような従来技術では、手書き文字フォントの演算は、ベジェ曲線関数の演算等を含み文字毎に逐次演算を行っているので、膨大な演算量となりシステム全体として処理速度の劣化を招くという問題があった。

30

そこで、本発明は、手書きフォントの生成に遺伝的アルゴリズムを導入することによって、システム全体として処理時間を削減して、システムの処理速度の改善を図ると共に、ユーザーによる修正を反映する遺伝的アルゴリズムによるフォント生成システムを提供することを目的としている。

【0011】

【課題を解決するための手段】

上記目的を達成するため、本発明の遺伝的アルゴリズムによるフォント生成システムは、手書き入力された文字パターンより、手書き文字フォントを生成して手書き文字を出力する文字フォント生成システムにおいて、手書き文字を入力して記憶する入力手段と、前記入力手段により入力される前記手書き文字を文字パターンに変換して出力するパターン変換手段と、前記パターン変換手段により出力される前記文字パターンを認識して該当する文字コードを出力する文字認識手段と、前記入力手段により入力される前記手書き文字を該文字の特徴を含むベクトル情報に変換して出力するベクトル変換手段と、前記ベクトル情報を手書き文字の座標点フォントとして記憶するベクトル情報記憶手段と、前記ベクトル情報記憶手段の座標点フォントを読み出し補間曲線関数による曲線フォントを遺伝的アルゴリズムGAにより演算生成するGA演算処理手段と、前記GAにより演算生成された曲線フォントを記憶する手書きフォント記憶手段と、を備え、

40

前記GA演算処理手段は、前記座標点フォントを入力して前記補間曲線関数をセグメン

50

ト分割により直線で近似する補間曲線関数処理部と、前記セグメントの係数を個体として複数の個体を含む集団を形成する集団形成部と、前記遺伝的アルゴリズム G A の世代数上限、個体数、交叉点、突然変異、評価関数を少なくとも含むパラメータの設定条件を記述した条件設定テーブルと、前記補間曲線関数処理部におけるセグメントの係数を個体とする近似計算に前記条件設定テーブルに設定された前記パラメータを用いて G A 演算を行う G A 制御部とを有し、予め記憶された標準パターン辞書の標準文字パターンに前記遺伝的アルゴリズムを適用して前記手書き文字フォントを生成することを特徴としている。

また、前記 G A 演算処理手段は、G A により曲線フォントを演算生成する際に、複数の異なる文字パターンを生成して記憶する文字パターン生成部と、生成された複数の異なる文字パターンの中から利用者により選択されて記憶される文字パターン選択部と、を有することを特徴としている。

10

【 0 0 1 2 】

【発明の実施の形態】

以下、本発明の実施の形態について図を参照して説明する。

図 1 は、本発明の実施の形態に係る遺伝的アルゴリズムによるフォント生成システムの構成図である。

図 2 は、図 1 に示す G A 演算処理部の構成図である。

図 3 は、図 2 に示す G A 演算処理部の処理のフローチャートである。

図 4 は、図 2 に示す補間曲線の説明図である。

図 5 は、図 4 に示す補間曲線の評価の説明図である。

20

図 6 は、図 2 に示す G A 制御部の処理の説明図である。

図 7 は、図 2 に示す G A 演算処理部で生成する文字パターンの説明図である。

【 0 0 1 3 】

図 1 において、1 は G A フォント生成システムで、2 はその G A (遺伝的アルゴリズム) 演算処理部であり、遺伝的アルゴリズムによる演算によって手書き文字入力のフォント生成演算を行う。3 は生成された曲線フォントを記憶する手書きフォント記憶部、4 は明朝体等の定型文字フォント辞書である。

なお、従来技術の図 8 と同一構成には同一符号を付して重複する説明は省略する。

【 0 0 1 4 】

図 2 において、2 は G A 演算処理部、5 はベクトル情報記憶部 5 9 から手書き文字のベクトル情報を入力して、ベジェ曲線関数等の補間関数を用いた曲線近似処理を行う補間曲線関数処理部、6 はベジェ曲線の分割単位 (セグメント) を 1 個体として複数の個体を含む集団 (1 フォント曲線) を形成し、複数の集団による集団群 (1 ストローク) を形成する集団形成部、7 は集団形成部 6 により構成される複数の集団を並列処理するように配置した集団群を示す。8 は G A 演算処理の世代数上限、個体数、交叉点、突然変異、評価関数、ベジェ曲線関数の閾値 d_{max} 等の、パラメータの設定条件等を記述した条件設定テーブル、9 は条件設定テーブル 8 に設定の各パラメータを用いて並列演算により G A 演算処理を行う G A 制御部、10 はユーザーによる選択が可能ないようにフォント生成過程に設けた文字パターン生成・選択部であり、11 は処理結果 (フォント) の出力部である。

30

【 0 0 1 5 】

つぎに G A 演算処理の具体的な処理について説明する。

G A 演算処理部 2 は、補間曲線関数処理部 5 によりベクトル情報記憶部 5 9 から図 9 に示すようなベクトル情報 (座標点フォント) を読み出し、ベジェ曲線などの曲線関数による補間演算を G A 演算によって行い曲線フォントを生成する。

40

【 0 0 1 6 】

ここで、ベジェ曲線関数とは、図 4 に示すように、始点 P_0 、終点 P_3 、制御点 P_1 、 P_2 で定義されるベジェ曲線を、 $f(t)$ 、 $(0 \leq t \leq 1)$ というパラメトリック曲線で表現した際の、 $R_2(0.5)$ なる点において形状を変えずに 2 つのベジェ曲線 (セグメント)、すなわち (P_0, Q_0, R_1, R_2) と、 (R_2, R_3, Q_2, P_3) に分割できる性質を持つ。ここで、

50

$$Q_{01} = (P_0 + P_1) / 2$$

$$R_1 = (P_0 + 2P_1 + P_2) / 4$$

$$R_2 = \{P_0 + 3(P_1 + P_2) + P_3\} / 4$$

$$R_3 = (P_1 + 2P_2 + P_3) / 4$$

$$Q_{23} = (P_2 + P_3) / 2$$

である。

【0017】

このように、ベジエ曲線は分割を繰り返し再帰的に適用することで際限なく細分することができるので、このベジエ曲線の性質がベジエ曲線の描画に利用されている。つまり、描画しようとするベジエ曲線を再帰分割して、分割後の各セグメントが予め設定されている閾値 d_{max} より平坦であると判断された時点で分割を止め、この時の各セグメントを始点と終点を結ぶ直線で近似して描画することにより、最小単位に分割した直線群の集合という形で、近似的なベジエ曲線を描画できる。

10

この場合に各セグメントを平坦と判断する基準は、図5に示すように、 d をベジエ曲線と直線 P_0 、 P_3 との距離の最大値、 d_1 、 d_2 を各制御点 P_1 、 P_2 と直線 P_0 、 P_3 との距離、フラットネスの閾値を d_{max} と定義すると、

$d_1 \leq d$ 、 $d_2 \leq d$ であって、

$$d_1 < d_{max} \quad (1)$$

$$d_2 < d_{max} \quad (2)$$

の2つの条件が共に満足された場合に平坦と判断して曲線の分割を止め、分割した各セグメントでベジエ曲線を近似する。

20

【0018】

以上のベジエ曲線関数を用いて、補間曲線関数処理部5は、図9のようなベクトル情報を読み込み、始点 a_1 ~ 次の座標点 a_2 間の弧を近似描画するために、ベジエ曲線 a_1 ~ a_2 を再帰分割する。

この場合にGA制御部9は遺伝的アルゴリズムの前処理として、セグメント毎の d_1 、 d_2 値を個体としてコーディングする。

例えば、今、 d_1 、 d_2 値が、2~5、6~9だとすると、以下のように2進コードによってコーディングする。

d_1	2進コード	d_2	2進コード
6	「0110」	2	「0010」
7	「0111」	3	「0011」
8	「1000」	4	「0100」
9	「1001」	5	「0101」

30

この場合の各ビット0, 1はそれぞれ遺伝子に対応する。

【0019】

次に、個体の集合体としての初期集団を、例えば、 a_1 ~ a_2 、 a_2 ~ a_3 、... a_{n-1} ~ a_n のように各座標点間の弧として定義する。従って、図9に示すような、ストロークAは a_n 個の集団によって構成される集団群として表され、並列演算による高速処理が可能な形態に構成される。

40

【0020】

GA演算処理の詳細を、図3に示すGA演算処理のフローチャートを参照して説明する。ここでの処理はGA制御部9が、条件設定テーブル8の設定パラメータを用いて行う並列演算の内容である。

最初に、世代数をカウントアップする(S100)。

遺伝的アルゴリズム(GA)は、

(1)、2進数表現の個体群の初期集団を生成する。

(2)、設定された評価関数により各個体の評価を行う。

(3)、評価に基づいて淘汰、増殖による個体の選択を行う。

(4)、遺伝子を入れ替える交叉を行う。

50

(5)、遺伝子を変化させる突然変異を行う。

の(1)～(5)を1ループとする処理を世代と呼び、世代を繰り返し実施することにより最小値に収束させる処理を行うことから、先ず、現在の世代数をカウントして確認する。例えば、ここでの世代数は具体的には、セグメントの分割回数などに相当する。

【0021】

次に、世代数の上限に達したか?を判断する(S101)。

先に、条件設定テーブル8に世代数上限として世代処理回数を設定してあるので、これを参照して世代数上限に達していれば処理を終了し(S102)、未だ上限に達していなければ、各個体の評価を行う(S103)。

評価処理は個々の個体毎に行われ、条件設定テーブル8に予め設定されている評価関数を用いる。この場合の評価関数は、条件設定テーブル8に設定されている閾値 d_{max} を用いた、

$$d_1 < d_{max}$$

$$d_2 < d_{max}$$

により評価される。

【0022】

続いて、個体評価を基に選択処理を行う(S104)。

先ず、条件設定テーブル8に設定されている「しきい値」を用いて、個体の淘汰と増殖を行う。「しきい値」は特に限定しないが、例えば、セグメント中の d_1 と d_2 の差が「しきい値」より大きいようなアンバランスなセグメントが出現したら削除して淘汰し、削除した分に残った良いセグメントをコピー(補間により増殖)させる等の選択・淘汰を行う。

【0023】

次に、個体同士を組にして交叉を行う(S105)。

交叉は条件設定テーブル8に設定されている交叉条件に従って、例えば、個体(親A)の遺伝子と個体(親B)の遺伝子を入れ替え、第1、第2の個体(子)を生成する。今、親 $A = 1011$ 、親 $B = 0100$ 、として後半の遺伝子同士を入れ替えると、第1の子 = 1000 、第2の子 = 0111 という交叉となる。

この他に、交叉の方法は、それぞれ前半と後半の遺伝子を入れ替える方法や、一点交叉、他点交叉、等各種の方法がある。

【0024】

続いて、いくつかの個体の遺伝子を変化させる突然変異の処理を行う(S106)。

突然変異は条件設定テーブル8に設定された、突然変異の起こる確率と条件に従って、例えば、個体「 0100 」「 0110 」、と言うように個体の遺伝子を変化させる。

これら交叉、突然変異などの処理は、例えば、セグメントを分割して行く過程で d_1 、 d_2 の減少率、バランス等をダイナミックに調整することで演算のやり直しを避けたり、分割回数を早めたりして、より最小値への収束を促進するための処理である。

このように、ベジェ曲線の近似処理に、最適化を図って迅速に収束させることができる遺伝的アルゴリズムを導入したことによって、従来の逐次的な近似演算に比較して大幅に処理時間が短縮できる。

【0025】

こうして、ベジェ曲線が最小直線のセグメントに収束して、弧 $a_1 \sim a_2$ のベジェ曲線描画、つまり集団1の曲線フォントが生成されたら、結果出力部10から出力して、ベクトル情報記憶部59のベクトル情報とリンクして対応する曲線フォントを手書きフォント記憶部3に記憶する。

同様にして、 $a_2 \sim a_3 \dots a_{n-1} \sim a_n$ 、の曲線フォントを手書きフォント記憶部3に記憶する。

ストロークBについても同様な処理を行う。

こうしたGA演算処理全体としては、各集団は並列処理が行われ、各集団内では図6に示すように、各セグメント(個体)が並列処理され、更に、ストロークAとストロークBも

10

20

30

40

50

並列処理により生成されるので高速処理が可能になる。

【 0 0 2 6 】

文字パターン生成・選択部 1 0 は、フォント生成のための前述した G A 演算処理に加えて、ユーザーによる好みなどをフォント生成に反映させるために、ユーザー認知が可能な文字パターンの生成と、生成された文字パターンの選択手段を提供する。

図 7 は、生成された文字パターンの一例を示しており、縦長、横長、右あがり、右下がり、太った、痩せた、等の変形した様々な文字パターンを生成して、ユーザーに生成した文字パターンを選択させるものである。

このようにして選択された文字パターンの各パラメータは再度 G A 処理部で使用され、最終的なフォントにはユーザーの修正内容が反映される。

10

【 0 0 2 7 】

最後にラスト描画については、手書き入力を定型印字する場合は、従来と同様に文字認識情報記憶部 6 0 から文字コードを読み出し、フォント辞書 4 で書体フォントを指定して定型印字する。

一方、手書き文字を手書きのまま印字出力する場合は、ベクトル情報記憶部 5 9 と手書きフォント記憶部 3 のデータを読み出しビットマップ展開して印字する。これによって認証用の署名、サイン、肉筆の宛名書、自筆の手紙などが可能になる。

また、文字の拡大・縮小、太字印字、などは手書き文字のサイズが正規化されているので、手書きフォントの曲線関数の定数を可変設定するだけで、特徴のある手書き文字を表示し、印字出力することができる。

20

その他の文字位置などの制御コードについては従来フォントに準ずる。

【 0 0 2 8 】

ここまでは、手書き文字のフォント生成処理に遺伝的アルゴリズム G A を導入する例について説明したが、これ以外に、例えば、パターン認識部 5 7 における手書き文字パターンと、標準パターン辞書の標準文字パターンとのパターンマッチング(パターン照合)処理なども、大量のデータを処理する画像認識演算が必要となるので本発明の遺伝的アルゴリズムの適用は有効である。

【 0 0 2 9 】

【発明の効果】

以上説明したように、本発明によれば、手書き入力された文字パターンより手書き文字フォントを生成して手書き文字を出力する文字フォント生成システムにおいて、入力された手書き文字を記憶・変換した文字パターンを認識して該当する文字コードを出力する文字認識手段と、入力される手書き文字を該文字の特徴を含むベクトル情報に変換して手書き文字の座標点フォントとして記憶するベクトル情報記憶手段と、ベクトル情報記憶手段の座標点フォントを読み出しベジェ曲線等の補間曲線関数による曲線フォントを遺伝的アルゴリズム G A により演算生成する G A 演算処理手段と、曲線フォントを記憶する手書きフォント記憶手段と、を備えているので、手書き文字フォントの生成演算時間が短縮できてスムーズな手書き文字出力が可能になる。

30

また、G A 演算処理手段は、座標点フォントを入力してベジェ曲線等の補間関数をセグメント分割により直線で近似し、セグメントの係数を個体として複数の個体を含む集団を形成して、遺伝的アルゴリズム G A の世代数上限、個体数、交叉点、突然変異、評価関数などのパラメータの設定条件を基に、セグメントの係数を個体とする近似計算によって G A 演算を行うので、G A による最適化処理によってベジェ曲線などの補間関数の収束が迅速に行われ、手書きフォントの生成時間が短縮される。

40

さらに、G A 演算処理手段は、G A により曲線フォントを演算生成する際に、複数の異なる文字パターンを生成・記憶すると共に、生成された複数の異なる文字パターンの中から利用者による選択・記憶を可能にしているため、より、ユーザーの好みを反映した手書き文字フォントを作成することができる。

【図面の簡単な説明】

【図 1】本発明の実施の形態に係る遺伝的アルゴリズムによるフォント生成システムの構

50

成図である。

- 【図2】図1に示すGA演算処理部の構成図である。
- 【図3】図2に示すGA演算処理部の処理のフローチャートである。
- 【図4】図2に示す補間曲線の説明図である。
- 【図5】図4に示す補間曲線の評価の説明図である。
- 【図6】図2に示すGA制御部の処理の説明図である。
- 【図7】図2に示すGA演算処理部で生成する文字パターンの説明図である。
- 【図8】従来の手書き文字処理装置の構成図である。
- 【図9】図8に示す手書き文字処理装置の座标点フォントの説明図である。
- 【図10】図9に示す座标点フォントのメモリ内容を示す図である。
- 【図11】図8に示す手書き文字処理装置の補間関数の説明図である。

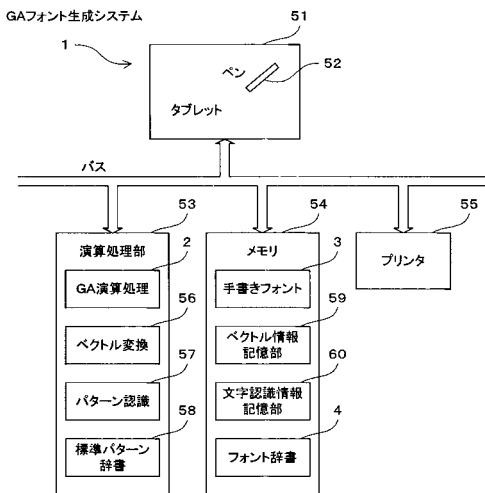
10

【符号の説明】

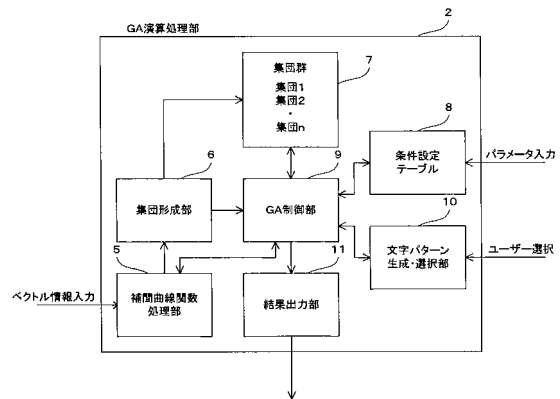
- 1 GAフォント生成システム
- 2 GA演算処理部
- 3 手書きフォント記憶部
- 4 フォント辞書
- 5 補間曲線関数処理部
- 6 集団形成部
- 7 集団群
- 8 条件設定テーブル
- 9 GA制御部
- 10 文字パターン生成・選択部
- 11 結果出力部

20

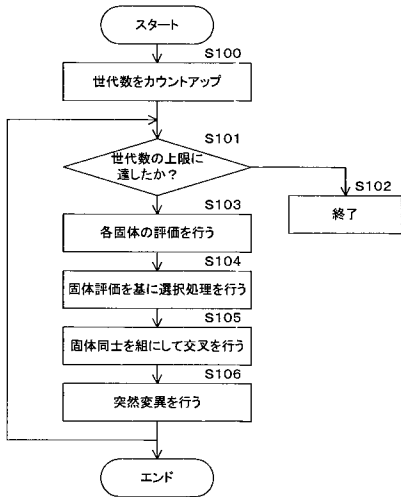
【図1】



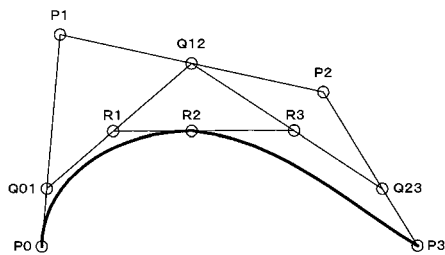
【図2】



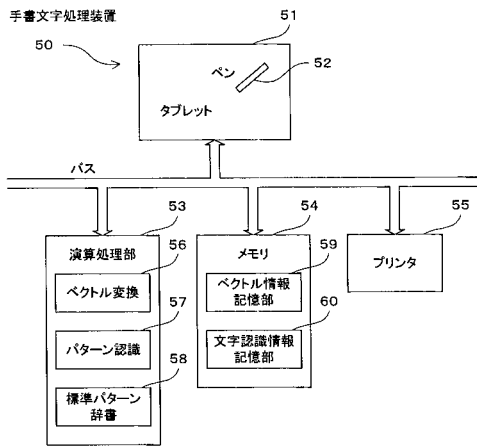
【図3】



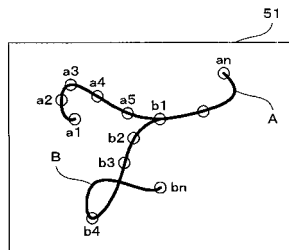
【図4】



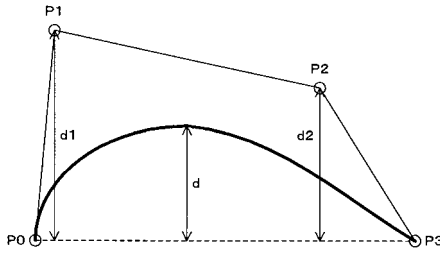
【図8】



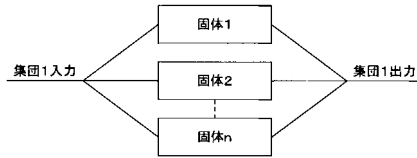
【図9】



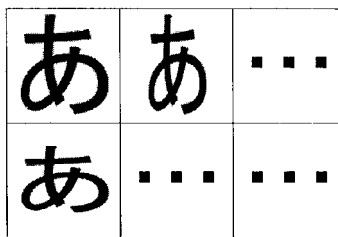
【図5】



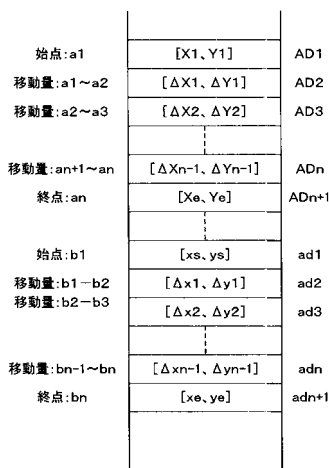
【図6】



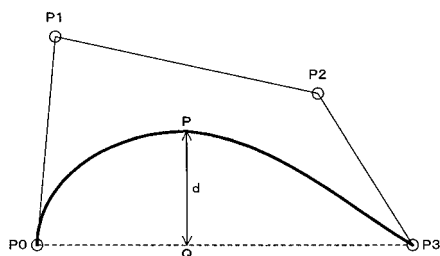
【図7】



【図10】



【図11】



フロントページの続き

- (56)参考文献 特開平05-143769(JP,A)
特開平10-021385(JP,A)
特開2001-022725(JP,A)
特開平10-333663(JP,A)
特開平08-194781(JP,A)
特開平08-016570(JP,A)

(58)調査した分野(Int.Cl., DB名)

G09G5/00-5/42
G06F17/20-17/26
G06K9/00