

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第3671575号
(P3671575)

(45) 発行日 平成17年7月13日(2005.7.13)

(24) 登録日 平成17年4月28日(2005.4.28)

(51) Int. Cl.⁷D05B 19/08
D05C 5/06

F I

D05B 19/08
D05C 5/06

請求項の数 5 (全 13 頁)

(21) 出願番号	特願平9-36672	(73) 特許権者	000005267
(22) 出願日	平成9年2月20日(1997.2.20)		ブラザー工業株式会社
(65) 公開番号	特開平10-230089		愛知県名古屋市瑞穂区苗代町15番1号
(43) 公開日	平成10年9月2日(1998.9.2)	(72) 発明者	田口 彰一
審査請求日	平成15年12月22日(2003.12.22)		名古屋市瑞穂区苗代町15番1号ブラザー工業株式会社内
		審査官	蓮井 雅之
		(56) 参考文献	特開平08-052291 (JP, A) 特開平07-236784 (JP, A)
		(58) 調査した分野(Int. Cl. ⁷ , DB名)	D05B 19/00-21/00 D05C 5/00-6/06

(54) 【発明の名称】 刺繍データ処理装置

(57) 【特許請求の範囲】

【請求項1】

刺繍図柄を表す画像データに基づいて、該図柄の刺繍を形成するためのデータを処理する装置であって、

前記画像データを細線化する細線化手段と、

前記細線化手段により変換された細線画像に関して細線追跡する細線追跡手段と、

前記細線追跡において枝があった場合、その枝を削除する枝削除手段と

を具備することを特徴とする刺繍データ処理装置。

【請求項2】

前記細線化手段により細線化された細線画像より、前記刺繍図柄を構成する輪郭線を抽出する輪郭線抽出手段と、

前記抽出された輪郭線が成す図形の輪郭を構成する縫目若しくは抽出された閉領域が成す図形の内部を縫いつくす縫目の少なくとも一方のための縫製データ作成手段を備えることを特徴とする請求項1に記載の刺繍データ処理装置。

【請求項3】

前記細線追跡手段によって既に細線追跡を行った画素を再び追跡した場合、閉ループを検出したとし、それを内部輪郭線として領域処理する閉ループ処理手段を備え、前記枝削除手段は前記閉ループ処理手段よりも先に適用されることを特徴とする請求項1に記載の刺繍データ処理装置。

【請求項4】

前記細線追跡手段による細線追跡において、その追跡方向が前回の追跡方向と180度異なるか否を判断する方向判断手段を有し、前記枝削除手段は前記方向判断手段によって、前回の追跡方向と今回の追跡方向が180度異なることが判断された場合、前回の追跡に関わる枝を削除することを特徴とする請求項2に記載の刺繍データ処理装置。

【請求項5】

前記枝削除手段及び前記閉ループ処理手段は、図形の一部ずつを繰り返し処理することを特徴とする請求項3に記載の刺繍データ処理装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、被縫製体に対して所定の図柄の刺繍を形成するミシンのための、刺繍の形成に必要な刺繍縫製データを作成するための刺繍データ処理装置に係り、特に輪郭画を原画として用いて図柄の刺繍データを処理する装置に関する。

【0002】

【従来の技術】

従来より、需要者の嗜好の多様化、高級化、刺繍ミシンの性能の向上などの諸事情を背景にして、家庭用の刺繍マシンであっても、予め記憶されている刺繍データに基づく図柄の刺繍形成だけでなく、使用者の所望する図柄の刺繍を形成することを可能とする比較的安価で操作の容易な刺繍データ処理装置が提供されている。これらの多くは、原紙上にペンで書いた絵をハンディースキャナー等を用いて読み取り、刺繍データを作成するものである。

【0003】

従来、この種の刺繍データ処理装置では、複数の色を用いた刺繍データを作成するためには、各色に対応する複数の原紙を用い色ごとに原紙のスキャンを行うものや、特開平7-236784号公報に示すように初めに輪郭線を描いた原紙をスキャンし、後に指定する部分を塗りつぶし再び原紙のスキャンを行うもの等があった。

【0004】

【発明が解決しようとする課題】

先に記した従来の刺繍データ処理装置のうち前者のものでは、輪郭線や各色ごとにそれぞれ原画を作成し、スキャナー等を用い入力を行う。そのために、図17に示すような「葉」の模様のデータを作成する時は、タタミ縫いの原画では枝の部分を塗りつぶしておけば(図18)、輪郭線のサテン幅が非常に狭い時や、輪郭線が走り縫いであっても図17の部分Dにおいてタタミ縫いと輪郭線との間に隙間を生じる事は無かった(図19)。しかし、各色ごとに異なった原画を作成しなくてはならない為に、各原画にずれがあった場合には、出来上がるデータ間にずれが発生する。また、原画にずれが無い場合においても、各スキャンごとの読み取り誤差などによってもずれは発生する。そのため、ずれの少ない刺繍データを作成するには作業に多くの時間がかかり、面倒で意外と熟練を要するものであった。

【0005】

また、後者の刺繍データ処理装置においては、初めにスキャンされた輪郭線画像をもとに、内部タタミ縫い領域のデータを作成するため、輪郭線がタタミ縫いの時には、ずれが発生しにくい。しかしながら、輪郭線をサテン縫いにすると、ずれが発生しやすかった。

【0006】

この理由としては、次のような事情が背景として存在する。図柄の原画をイメージスキャナ等で読み取って画像データとし、それから刺繍データに自動変換を行う手法の1つとして、図柄を表す画素の集まりの輪郭を抽出し、その結果得られるいわゆるアウトラインで図柄形状を扱うものがある。輪郭線として描かれた図柄を、イメージスキャナで読み取って変換される画像データでは、輪郭線とはいってもある程度の広がりを持った面状の領域とならざるを得ないため、そこからアウトラインを取り出したときには原画の(ある程度太さをもつ)輪郭線の両側に経路を有するようなアウトライン(円のような閉領域の場合

10

20

30

40

50

、離間している少なくとも2本の線)が取り出される。輪郭線がタタミ縫いの場合はそのアウトラインで挟まれた内部を縫うようなデータを作成すれば、内部タタミ縫いと輪郭線との間の縫いずれの問題は発生しにくいのである。しかし、輪郭線を走り縫い、サテン縫い等にしようとする時には、このようなデータでは、自動処理によって刺繍データを作成できない。

【0007】

なぜかという、輪郭線のようなものに対しては、その両側に径路があるような形状規定線ではなく、その中心にただひとつの径路を有する径路規定線で表されるのが、何かと都合が良いためである。そして、このような要請に対しては、画像データ処理技法として知られている細線化処理が適しており、それを適用することによって得られる細線を径路規定線とすれば、前述の走り縫い、サテン縫いのデータ変換も自在に可能となるのである。

10

【0008】

しかし、内部タタミ縫い領域を細線化前のデータより作成すると、細線化を行う際に、細線化前後のアウトラインデータにずれが生じるために、輪郭線が走り縫いの場合やサテン幅が非常に細い場合には、輪郭線と内部タタミ縫い部分との間に隙間が生じてしまうことになっていた。

【0009】

それに対しては、内部タタミ縫いのデータを輪郭線のデータと同じく、入力された輪郭画像を細線化した細線画像より作成するとすれば問題は発生しない。しかし、この場合においては図17で示した部分Dに関するデータが、同じ画素を2回辿るために非常に近くなったり、交差をおこしてしまう。すると、縫目が詰って仕上がりが美しくなかったり、運針データが作成できない恐れがあった。

20

【0010】

本発明は上記事情に鑑みてなされたものであり、その目的は、入力画像に対する内部タタミ縫いの刺繍データを作る際に、部分Dのような枝があった場合その枝を削除し、できるだけ単純なデータを作成することであり、専門的な知識や熟練がなくても容易に短時間で美しい刺繍縫いデータの作成を可能とする刺繍データ処理装置を提供するにある。

【0011】

【課題を解決するための手段】

このような課題を克服するために本発明の請求項1に記載の刺繍データ処理装置は、刺繍図柄を表す原画の画像データを細線画像に変換する細線化手段と、前記細線化手段により変換された細線画像を細線追跡する細線追跡手段と、前記細線追跡において枝があった場合、その枝を削除する枝削除手段とを備えている。従って、枝のある原画を入力した場合、枝を削除し刺繍データが自動的に作成される。

30

【0012】

また、本発明の請求項2に記載の刺繍データ処理装置では、細線化手段により細線化された細線画像より、前記刺繍図柄を構成する輪郭線を抽出する輪郭線抽出手段と、抽出された輪郭線が成す図形の輪郭を構成する縫目若しくは抽出された閉領域が成す図形の内部を縫いつくす縫目の少なくとも一方のための縫製データ作成手段を備えるため、多様な刺繍データの作成が可能となる。

40

【0013】

また、本発明の請求項3に記載の刺繍データ処理装置では、細線追跡によって既に細線追跡を行った画素を再び追跡した場合、閉ループを検出したとし、それを内部輪郭線をして領域処理する閉ループ処理手段を備え、枝削除手段は閉ループ処理手段よりも先に適用されることを特徴とするために、枝を発見した時に閉ループを発見したという誤認識をしないでデータの作成が可能となる。

【0014】

また、本発明の請求項4に記載の刺繍データ処理装置では、細線追跡において、その追跡方向が前回の追跡方向と180度異なるか否を判断する方向判断手段を有し、枝削除手段は方向判断手段によって、前回の追跡方向と今回の追跡方向が180度異なることが判断

50

された場合、前回の追跡に関わる枝を削除することを特徴とするため、枝を発見及び削除するデータ処理が容易となる。

【0015】

さらに、本発明の請求項5に記載の刺繍データ処理装置では、枝削除手段及び閉ループ処理手段は、図形の一部ずつを繰り返し処理することを特徴とするために、正しく枝及び閉ループの処理を行いデータの作成が可能になる。

【0016】

それから、輪郭線データや内部縫いデータの少なくとも一方のために更に針落ち点を確定する処理を施されるデータを、同一の細線化されたデータに基づき作成するので、輪郭線の縫い幅を非常に細くした場合においても、輪郭線と内部縫い間の隙間が発生しにくく、刺繍内部において下地の出現が少なくなり、専門的な知識や熟練がなくても簡単に高品質の刺繍データの作成が可能になる。また、輪郭線の入力より、その輪郭線縫いのみや内部タタミ縫いのみ、或いはその両方のデータを作成することも可能になり、多様な刺繍データの作成作業が可能になる。さらに、輪郭線及び内部縫いのデータを同一の細線画像より作成するために輪郭線と内部縫い間の隙間が発生しにくく、刺繍内部において下地の出現が少なくなり、専門的な知識や熟練がなくても簡単に高品質の刺繍データの作成が可能になる。

10

【0017】

【発明の実施の形態】

以下、本発明を家庭用刺繍ミシンのための刺繍データ処理装置に適用した一実施の形態について、図を参照して説明する。なお、本実施の形態では、図4に示すような「ねこ」の刺繍図柄Aにおける刺繍データを作成する場合を具体例として挙げながら、同処理装置で行われる刺繍データ処理の内容を説明する。

20

【0018】

まず、図示はしないが、家庭用刺繍マシンについて簡単に触れておく。刺繍マシンは、マシンベッド上に配置され刺繍が施される加工布を保持する刺繍枠を、水平移動機構により装置固有のX、Y座標系で示される所定位置に移動させつつ、縫い針及び釜機構による縫製動作を行うことにより、その加工布に所定の図柄の刺繍を施すようになっている（特開平5-49766号公報参照）。

【0019】

この場合、前記水平移動機構や針棒などは、マイクロコンピュータなどから構成される制御装置により制御されるようになっており、従って、一針毎の加工布のX、Y方向の移動量、即ち針落ち位置を指示する刺繍データ（ステッチデータ）が与えられることにより、制御装置は、刺繍動作を自動的に実行することが可能となるのである。また、本実施の形態では、刺繍マシンにはフラッシュメモリ（カードメモリ）により、外部から刺繍データが与えられるように構成されている。本実施の形態に係わる刺繍データ処理装置は、このような刺繍データを自動的に作成する機能を有するものである。

30

【0020】

次に、本実施の形態に係わる刺繍データ処理装置の全体構成について、図1および図2を参照して述べる。図1は刺繍データ処理装置の外観を示し、また図2はその電氣的構成を表している。ここで、処理装置本体1はマイクロコンピュータを主体として成り、CPU2、ROM3、RAM4、フラッシュメモリ装置（FMD）5、入出力インターフェイス6（I/O）がバスを介して相互に接続して構成されている。

40

【0021】

処理装置本体1の上面部には、読み取った図柄や刺繍領域等を画面7aに表示するための液晶ディスプレイ（LCD）7が設けられている。この液晶ディスプレイ7は、表示制御装置（LDC）8により制御されるようになっており、この表示制御装置8には表示記憶装置（VRAM）9が接続され、モノクロのビットマップグラフィックス表示が可能ないように構成されている。また、前記フラッシュメモリ装置5には、記憶媒体としてのフラッシュメモリ10が着脱可能に装着されるようになっており、そして、処理装置本体1に

50

は、操作者が縫製様式など各種の選択や設定の指示を行うための操作キー 11、および、図柄原画を読み込むためのイメージスキャナ 12 が、前記入出力インターフェイス 6 を介して CPU 2 に接続されている。

【0022】

このイメージスキャナ 12 は、モノクロの図柄原画を二値のビットマップ画像データとして読み取り可能ないわゆるハンドスキャナから成り、操作者がその上部を手で持って下面の読み取り部を図柄原画用紙上に宛い、ボタンを押しながら原画に沿ってなぞるように一定方向に移動させることにより、図柄の画像の読み取りが行われるようになっている。読み取られた図柄画像データはラスタ形式のビットマップとして、画素毎に白であれば「0」、黒であれば「1」の値を持つ 1 ビットデータで表現され、図 3 に示すような RAM 4 の図柄画像データ記憶領域 4a に記憶されるようになっている。

10

【0023】

さて処理装置本体 1 は、そのソフトウェア構成により、例えば図 9 に示すような「ねこ」の刺繍図柄 A の原画に基づいて、刺繍データの作成処理を自動的に行うように構成されている。このソフトウェアは、ROM 3 に CPU 2 を制御するプログラムコードとして格納されている。以下ではこの場合の動作を、図 4 から図 7 のフローチャートを参照しながら詳述する。この刺繍データを作成するにあたって、まず操作者は、図 9 に示す図柄輪郭画を、例えば白色の原紙に黒のペンで描くなどして用意する。

【0024】

操作者は、イメージスキャナ 12 により図柄 A の原画を読み取らせる。このイメージスキャナ 12 により読み取られた画像は図 10 のような 2 値のビットマップ画像データとしてメモリ上の図柄画像データ記憶領域 4a に記憶される。次に、この画像データは細線化処理（細線化手段）により図 11 のように細線化され、RAM 4 の細線画像記憶領域 4b に記憶される（S1000）。細線化の方法はすでに多くが知られているが、この場合、線幅が 1 画素になればどのような方法を用いてもよい。

20

【0025】

図 5 及び図 6 に示すように、細線化処理が行われた細線画像記憶領域 4b に記憶された図柄 A の細線画像データはベクトル化処理によって、各々が適当な長さも方向をもつ線分データの連鎖、つまりショートベクトルの集合に変換され、RAM 4 のベクトル記憶領域 4c に記憶される。ベクトル化処理の方法としては、画像中の細線図形を構成する画素のうち任意の画素を開始点として、細線図形を構成する画素の連鎖を順に辿りながら適当な間隔で細線図形を構成する画素の座標値をサンプルして形状構成点の集合を得る方法などがある。そしてこのベクトルデータをもとに輪郭線の刺繍データを作成する（S2000）。

30

【0026】

次に、操作者は先ほど入力した原画に対して、タタミ縫いを行う領域を指定すると、CPU 2 はその領域を抽出する（S3000：Yes, S4000）。この指定の方法は多種考えられるが、一例を挙げると、先に入力した原画のタタミ縫いを希望する領域を塗りつぶし、再度スキャナにより読み込む方法などが考えられる。そして、この指定された閉領域に対するベクトルデータの抽出方法は、以下に示す手順（図 6 参照）を内部輪郭線（図 5：S5100）及び外部輪郭線に適用し（図 5：S5300）、選られた径路を変換することによって実現される。

40

【0027】

手順 1：指定された領域の輪郭線に対応する細線ビットマップ上の最上点を開始点 P とする（図 6：S1 及び図 20）。

【0028】

手順 2：開始点 P に隣接する構成点に向かう径路のうち、内部輪郭線では最も左に向かう経路を、外部輪郭線では最も右に向かう径路を選択し、そちらへ進む方向に径路を辿る（S1）。

【0029】

50

手順3：進行方向に関して最も右方に向かう径路を選択し進む、という規則に従って径路を辿る（S2）。

【0030】

手順4：辿った先の画素が開始点Pの場合は径路を記憶し、終了する（S3：Yes, S10）。また、今回辿った径路と前回辿った径路の方向が180度異なる時（S4：Yes）は、枝を発見したとして前回の径路を消去し（S5）、手順3に戻る。尚、本発明で言う「枝」とは、閉領域の内部に向かって伸張する線分である。

【0031】

手順5：今回辿った経路と前回辿った経路の方向が180度異ならないとき（S4：No）、今回辿った径路を記憶する（S6）。今回辿った径路の先の画素が既に辿られているか調べ（S7）、辿られていなければ（S7：No）今回の画素を記憶し（S12）、手順3に戻る。尚、このS7では、開始点P以外の点について辿ったか否かが調べられる。

10

【0032】

手順6：今回の径路の先の画素が既に辿られていた場合は（S7：Yes）、前回その画素に入ってきている径路と、今回これから出て行く径路の方向を調べ（S8）180度異ならなければ（S8：No）手順3に戻る。

【0033】

手順7：今回これから出て行く径路の方向が180度異なっている時（S8：Yes）、閉ループを発見したとして、その閉ループを今処理している領域の内部輪郭線として処理し（S9）、手順3に戻る。

20

【0034】

手順3の規則を詳しく説明すると以下のようになる（図7参照）。

【0035】

手順1'：前回の径路が上、下、右、左か、あるいは右上、左上、左下、右下かを調べる（S100）。ただし、文中の上、下、左、右、右上、左上、左下、右下は図13に示すxy座標に対しての方向である。また、「右方」とか「・・・方」とあるのは、進行方向に対する方向である。

【0036】

手順2'：前回の径路が上、下、右、左の場合（一方の座標軸に直交する方向の場合）は（S100：No）前回の径路に対して右前方、前方、左前方、後方の順番で（S105～S125：図8（a））、前回の径路が右上、右下、左上、左下の場合（座標軸に直交しない方向の場合）は（S100：Yes）前回の径路に対して右方、右前方、前方、左前方、左方、後方の順番で画素が存在するかを調べる（S135～S160）。

30

【0037】

手順3'：最初に調べて存在した画素およびその画素への径路を、次の画素、次の径路として、RAM4の径路記憶領域4d、画素記憶領域4eに記憶する（S130）。

【0038】

次に、図8に例として、上と右上に径路を辿ってきた場合の画素を調べる順番を示す。

【0039】

図柄Aを細線化したビットマップデータについて、以上の手順実施によりなされる処理を説明するものが図11である。「ねこの顔」の領域（図12（a）斑点部）のデータ作成を例として説明する。このビットマップデータは、先の細線画像記憶領域4bに記憶されている。

40

【0040】

この「ねこの顔」の領域は内部輪郭線1及び2、外部輪郭線3を持つ（図12（b））。まず内部輪郭線1及び2の径路が、上述した手順1から手順7に従い辿られ（図12（c））、ベクトルデータに変換される。内部輪郭線1及び2では、枝及び閉ループが存在しないために手順4、6、7は行われない。

【0041】

続いて外部輪郭線3の径路を、手順1から手順7に従い辿る。図12（d）の矢印部分は

50

枝及び閉ループが存在しないため、手順4、6は行われず、手順5により辿った径路がRAM4の径路記憶領域4dに、辿った画素がRAM4の画素記憶領域4eに記憶されて行く。

【0042】

図12(d)の部分Bを拡大した図13に示すように、上方から辿ってきて、枝分かれている。画素1まで画素を辿ってくると、次の画素は今までの径路の進行方向に対して左方、前方、右方に存在するので、次の径路は右方へと進む(矢印a:S2)。画素2まで辿った時の径路は、図13に示されている部分では、左下、下、左下、左上(矢印a)、左上(矢印b)となっている。次のとるべき径路は右下(矢印c)となっており、この矢印cは前回の径路(矢印b)とは方向が180度異なっている(S4:Yes)。そのため、この場合は手順4により前回の径路(矢印b)は径路記憶領域4dから消去され、今回の径路(矢印c)も記憶されない(S5)。

10

【0043】

そうすると、今記憶されている径路は、左下、下、左下、左上(矢印a)となり、今辿られた画素は画素3である状態になる。今回の径路は右下(矢印d)であり、やはり記憶されている前回の径路(矢印a)と方向が180度異なるために、前回の径路は消去され、今回の径路は記憶されない。この様にして、図13に示されたように画素を全て辿り終わると、記憶されている径路は、左下、下、左下、左下、左下、左下、左となり、矢印a~dの枝の部分は、一旦記憶されているが結局径路から消去されている。

【0044】

次に、図12(d)の部分Cを拡大した図14に示すように、右側より手順1から6に従い径路を辿ってくる。矢印eに従い画素4を辿った時は、この画素を辿るのはまだ1回目なので手順5は行われない(S7:No)。続いて、環状部を反時計まわりに辿って、矢印gに従い再び画素4を辿ると、先ほど画素4を辿った時に、手順5により辿った画素がRAM4の画素記憶領域4eに記録されているために、この画素を辿るのは2回目と分かり(S7:Yes)、閉ループがある事が発見される。この画素から、次に辿る経路(矢印h)と前回この画素に入ってきている径路(矢印e)の方向を調べ、180度異なっていた場合のみ閉ループを切り離す処理を行う(S9)。これは、閉ループが接している時(図15)には切り離しを行わないためである。閉ループが接している場合には、同様の2つの径路(矢印iと矢印j)の方向は180度異ならない。

20

30

【0045】

図14の場合では、矢印fから矢印gまでの径路を閉ループとして切り離し、現在処理している領域の内部輪郭線として処理をする。そのため、記憶されている径路は左、左上、上、上、となる。そして、今回の径路は下であるために、手順4に従い、記憶されている最後の径路(矢印e)は消去され、今回の径路(矢印h)も記憶されない。

【0046】

つまり、閉ループが存在した場合には、閉ループ部分を切り離し内部輪郭線として処理するために、残った部分では先ほど説明した枝の削除と同じ処理が行われるのである。そのため、図14に示された画素を全て辿った後記憶されている径路は、左、左、左上、左下、左、左、となる。

40

【0047】

細線追跡の開始点は、図20に示すフローチャートのように、図11の画素について図13のような直交する座標系を想定し、図12の「ねこ」の絵の右上の座標を1画素ずつ座標値に演算する。その手順は、有効な画素(黒塗りの画素)について、X方向で最大座標を検出する度にその最大座標(Xmax)を更新すると共に引き続きその画素のY座標についても同様に最大値(Ymax)を越えるかを調べ、越える場合その座標を更新する(S410乃至S442)。そして、先の処理を繰り返してすべての画素について座標を調べた後に、最終的に抽出した(Xmax、Ymax)に相当する画素を開始点とする(S440:YES, S444)。

【0048】

50

以上の説明に従い、「ねこの顔」の内部輪郭線及び外部輪郭線を上記処理により作成したものが図12(e)である。内部輪郭線は処理前に比べて、1つ増え(新内部輪郭線)3つになっており、4本のひげ及び鼻の下部の線が消去されたデータができていたことが分かる。このため、輪郭線のサテン幅を非常に細くした場合や、輪郭線を走り縫いで行うような場合においても、輪郭線縫製部(図16(b),(d))とタタミ縫いの領域(図16(a))との間に隙間が生じる事が無くなり、その両方によって形成される模様(図16(c),(e))を美しく刺繍する事が可能になるのである。尚、輪郭線縫製部を走り縫い及びサテン縫いの両方で構成してもよい。

【0049】

以上の説明から明らかなように、枝削除処理によれば、4連結、8連結の連結性をもつ細線ビットマップに関して枝を削除するために、内部縫いデータの形状が単純になりデータ量が減少し、輪郭線の幅を細くできるため、高品質の刺繍データの作成が可能になる。また、枝削除は閉領域及び輪郭線のデータ作成時に適用可能のため、データが単純になり、高品質の刺繍データの作成が可能になる。また、ノイズ除去にも使用できる。更に、既に細線追跡を行ったか否かに基づき処理し、枝削除は閉ループ処理より先に行うので、枝と閉ループの判断をすることができるために正しくデータを作成することが可能になる。そして、追跡の前回と今回とで180度方向が異なることに基づいて処理しているため、細線追跡中に枝を発見した時、その枝をデータ上より削除することが可能になる。図名の一部ずつ繰り返し処理しているため、各ステップごとに枝削除、閉ループ発見を行うために、効率的にデータを作成することができる。

【0050】

それから、輪郭線データや内部縫いデータの少なくとも一方のために更に針落ち点を確定する処理を施されるデータを、同一の細線化されたデータに基づき作成するので、輪郭線の縫い幅を非常に細くした場合においても、輪郭線と内部縫い間の隙間が発生しにくく、刺繍内部において下地の出現が少なくなり、専門的な知識や熟練がなくても簡単に高品質の刺繍データの作成が可能になる。また、輪郭線の入力より、その輪郭線縫いのみや内部タタミ縫いのみ、或いはその両方のデータを作成することも可能になり、多様な刺繍データの作成作業が可能になる。さらに、輪郭線及び内部縫いのデータを同一の細線画像より作成するために輪郭線と内部縫い間の隙間が発生しにくく、刺繍内部において下地の出現が少なくなり、専門的な知識や熟練がなくても簡単に高品質の刺繍データの作成が可能になる。

【0051】

また、再び追跡をしたか否かに基づき閉ループを検出するので、閉領域内に閉領域が存在し、それらが枝によってつながっている時は内側の閉領域を内部輪郭線として処理することが可能になり、高品質の刺繍データの作成が可能になる。更に、閉ループ検出を行う際に追跡方向に基づくので、一層細かな判断を行え、高品質なデータの作成が可能になる。それから、細線追跡された画素のデータに基づき既に追跡されたか否かが判断されるので、細線追跡中に閉ループが存在した場合、容易に発見することが可能になる。さらに、追跡方向に対する180度に基づき閉ループの検出を行うので、閉領域内に閉領域が存在し、それらが枝によりつながっている時に、閉領域どうしの距離等の条件により、内側の閉領域を内部輪郭線として処理するか否かの判断をすることができる。

【0052】

上述した実施の形態においては、縫製データのうち縫い目を形成するための針落ち点データを作成したが、刺繍図柄を表す表示データを、同様にドットデータより作成しても良い。また、上述した実施の形態では、輪郭部分とその内部部分との両方の縫目を形成できるように構成されている装置であるが、その片方しか作成できない装置であっても良い。例えば、輪郭内部に相当する部分をアップリケにする場合、その輪郭内部を縫い尽くすデータは不要であるため、輪郭部分だけのデータが必要である。その逆に、輪郭部分を無くして意識的に布地を露出させて、輪郭線とする場合も考えられ、その場合、輪郭内部だけのデータが必要である。そのいずれの場合であっても、原画に忠実であることが望まれるこ

10

20

30

40

50

とは言うまでもなく、本発明を実施することによって良好なデータを取得することができる。

【0053】

【発明の効果】

以上の説明から明らかなように、請求項1に記載の刺繍データ処理装置によれば、枝削除手段は4連結、8連結の連結性をもつ細線ビットマップに関して枝を削除するために、内部縫いデータの形状が単純になりデータ量が減少し、輪郭線の幅を細くできるため、高品質の刺繍データの作成が可能になる。

【0054】

また、請求項2に記載の刺繍データ処理装置によれば、枝削除は閉領域及び輪郭線のデータ作成時に適用可能のため、データが単純になり、高品質の刺繍データの作成が可能になる。また、ノイズ除去にも使用できる。

【0055】

また、請求項3に記載の刺繍データ処理装置によれば、枝と閉ループの判断をすることができるために正しくデータを作成することが可能になる。

【0056】

また、請求項4に記載の刺繍データ処理装置によれば、細線追跡中に枝を発見した時、その枝をデータ上より削除することが可能になる。

【0057】

さらに、請求項5に記載の刺繍データ処理装置によれば、各ステップごとに枝削除、閉ループ発見を行うために、効率的にデータを作成することができる。

【図面の簡単な説明】

【図1】刺繍データ処理装置の外観を示す斜視図である。

【図2】刺繍データ処理装置の電気的構成を示すブロック図である。

【図3】刺繍データ処理装置のRAMを示す図である。

【図4】本発明の実施の形態の刺繍データ処理装置による刺繍データ処理の全体の手順の一例を示すフローチャートである。

【図5】本発明の実施の形態の刺繍データ処理装置による指定領域のデータ処理手順の一例を示すフローチャートである。

【図6】本発明の実施の形態の刺繍データ処理装置による輪郭線のデータ処理手順の一例を示すフローチャートである。

【図7】本発明の実施の形態の刺繍データ処理装置による径路を調べるデータ処理手順の一例を示すフローチャートである。

【図8】前回径路に対する周囲8近傍の検索の順番を示す図である。

【図9】刺繍図柄原画の一例を示す図である。

【図10】刺繍図柄原画を読み取ったビットマップ画像の部分を示す図である。

【図11】刺繍図柄の細線化ビットマップ画像の部分を示す図である。

【図12】閉領域の内部輪郭線及び外部輪郭線の抽出過程を説明する図である。

【図13】枝があった場合の径路の辿りかたとx-y座標を示す図である。

【図14】閉ループがあった場合の径路の辿りかたを説明する図である。

【図15】閉ループが接していた場合の径路の辿りかたを説明する図である。

【図16】作成される刺繍縫目の一例を示す図である。

【図17】枝の存在する原画を示す図である。

【図18】領域を指定する原画を示す図である。

【図19】作成される刺繍縫目の一例を示す図である。

【図20】本発明の実施の形態の刺繍データ処理装置による追跡の開始点を抽出するためのフローチャートである。

【符号の説明】

2 CPU

3 ROM

10

20

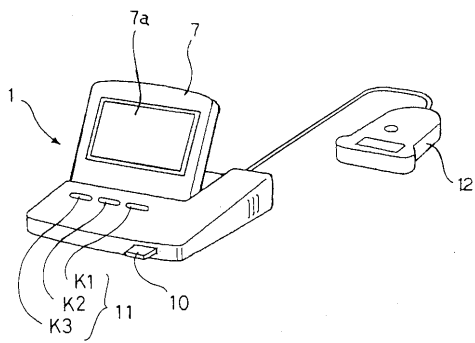
30

40

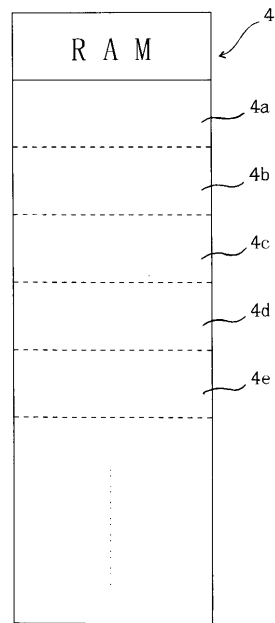
50

- 4 R A M
- 5 フラッシュメモリ装置
- 6 入出力インターフェイス
- 7 液晶ディスプレイ (L C D)
- 10 フラッシュメモリ 10
- 12 イメージスキャナ 12

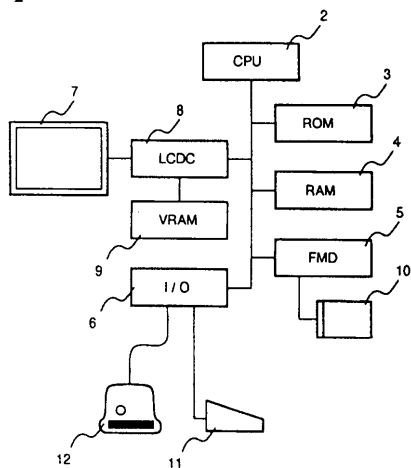
【 図 1 】



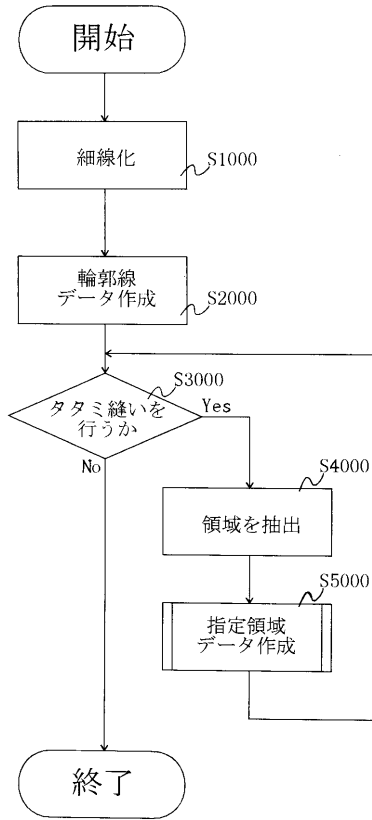
【 図 3 】



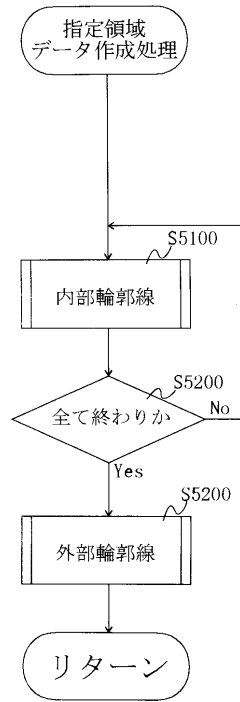
【 図 2 】



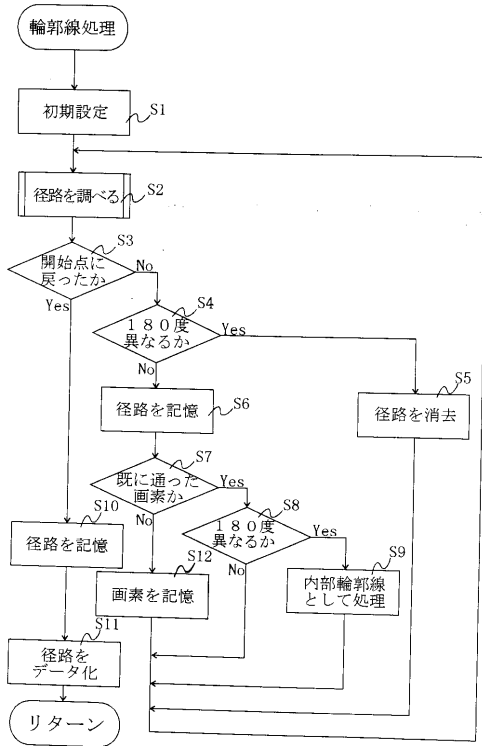
【 図 4 】



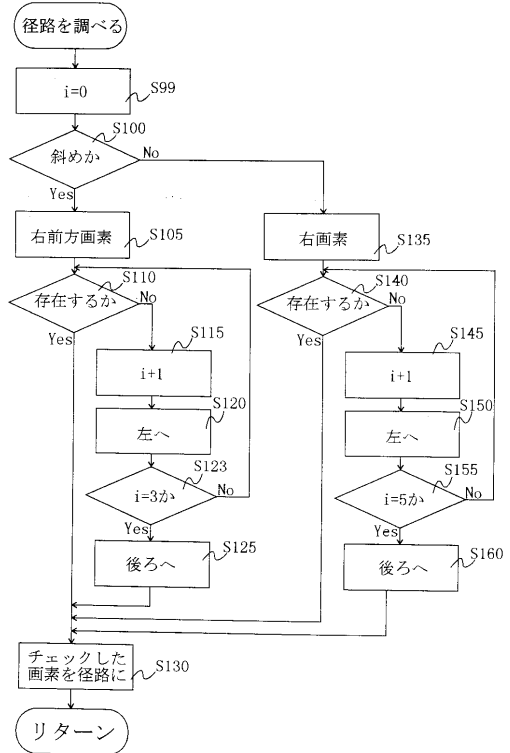
【 図 5 】



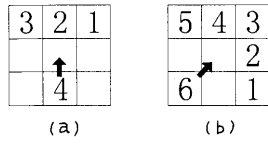
【 図 6 】



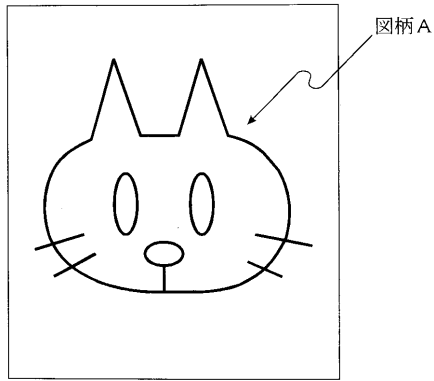
【 図 7 】



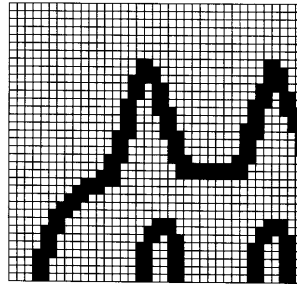
【 图 8 】



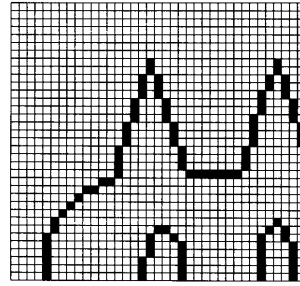
【 图 9 】



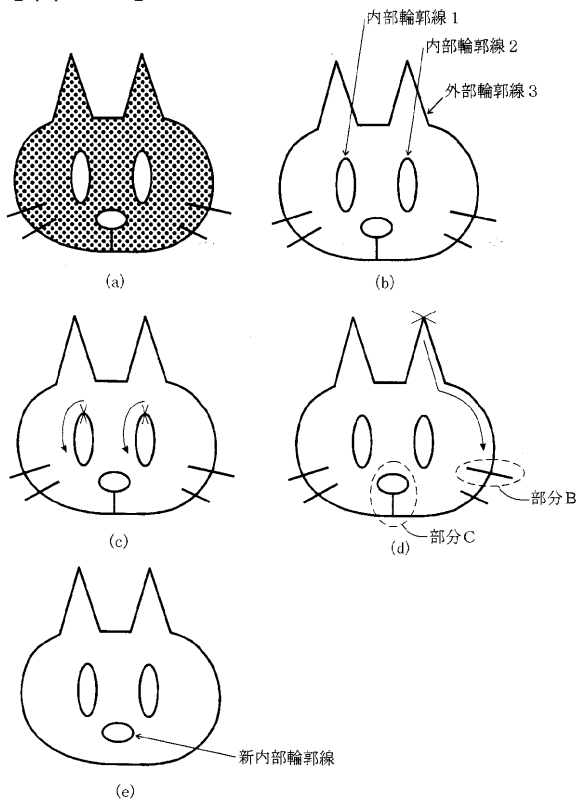
【 图 10 】



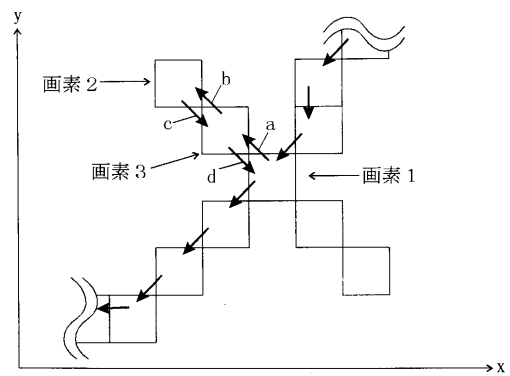
【 图 11 】



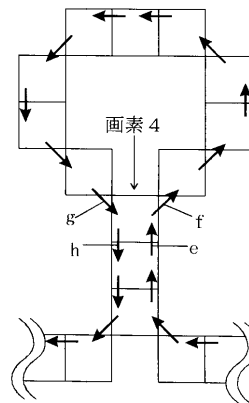
【 图 12 】



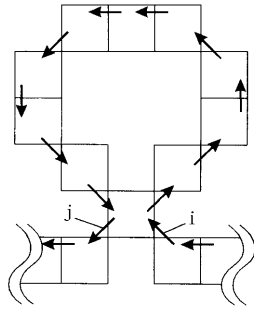
【 图 13 】



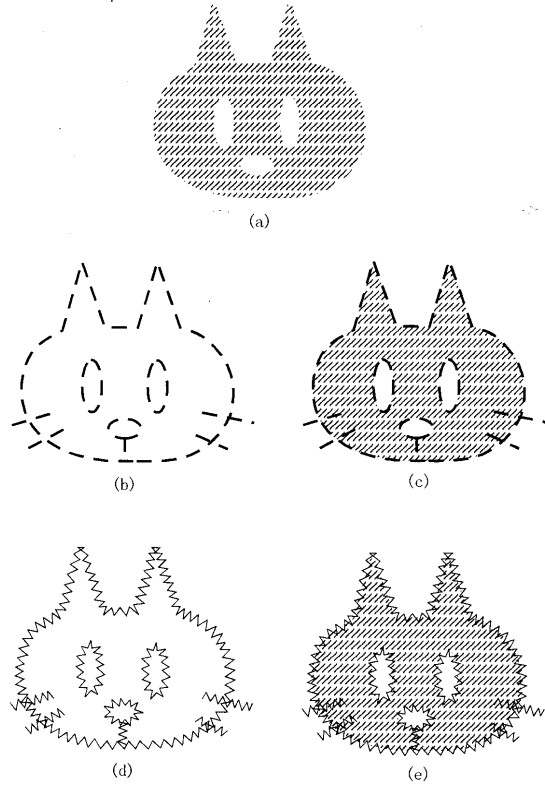
【 图 14 】



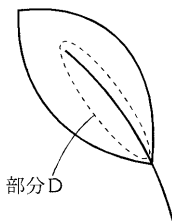
【図15】



【図16】



【図17】



【図18】



【図19】



【図20】

