

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第3634574号
(P3634574)

(45) 発行日 平成17年3月30日(2005.3.30)

(24) 登録日 平成17年1月7日(2005.1.7)

(51) Int.Cl.⁷

F I

G 0 6 K 9/68

G 0 6 K 9/68

E

G 0 6 K 9/70

G 0 6 K 9/70

G 1 0 L 15/06

G 1 0 L 3/00

5 2 1 C

請求項の数 9 (全 14 頁)

(21) 出願番号 特願平9-187019
 (22) 出願日 平成9年7月11日(1997.7.11)
 (65) 公開番号 特開平11-31226
 (43) 公開日 平成11年2月2日(1999.2.2)
 審査請求日 平成15年12月11日(2003.12.11)

(73) 特許権者 000001007
 キヤノン株式会社
 東京都大田区下丸子3丁目30番2号
 (74) 代理人 100076428
 弁理士 大塚 康德
 (74) 代理人 100093908
 弁理士 松本 研一
 (74) 代理人 100101306
 弁理士 丸山 幸雄
 (72) 発明者 吉井 裕人
 東京都大田区下丸子3丁目30番2号 キ
 ヤノン株式会社内

審査官 新井 則和

最終頁に続く

(54) 【発明の名称】 情報処理方法及び装置

(57) 【特許請求の範囲】

【請求項1】

記憶媒体に記憶されている制御プログラムを実行することによりパターンを分類するための分類木を生成するように情報処理装置を制御するための情報処理方法であって、
学習パターンから各次元が1つの座標値で表される複数次元の特徴量を取り出し、各次元の座標値が属する座標軸を所定の方法で再帰的に断片化し、当該次元の座標値が該断片化された座標軸のいずれの断片に属するか判断することにより、各次元についての階層構造を生成する第1生成工程と、

前記第1生成工程で生成された各次元の階層構造に基づいて、分類木の上位ノードから順に下位ノードへ生成していく際、生成途中の分類木のノードにおける各次元の分類効率に
基づいて、いずれの次元で分類を行なうべきかを選択し、当該選択された次元を用いて前記分類木のノードを生成していく第2生成工程とを備えることを特徴とする情報処理方法

【請求項2】

各次元の座標軸を再帰的にn分割(nは複数次元)することにより前記座標軸を再帰的に断片化していくことを特徴とする請求項1に記載の情報処理方法。

【請求項3】

各座標軸に応じて、前記再帰的に断片化する際の所定の方法が異なることを特徴とする請求項1に記載の情報処理方法。

【請求項4】

10

20

前記第 2 生成工程では、最も分類効率が上がる次元の階層構造を用いて前記分類木のノードを生成していくことを特徴とする請求項 1 に記載の情報処理方法。

【請求項 5】

前記第 2 生成工程で生成された前記分類木のデータを格納する格納工程を更に備えることを特徴とする請求項 1 に記載の情報処理方法。

【請求項 6】

前記格納工程によって格納された前記分類木のデータに基づいて入力パターンを認識する認識工程を更に備えることを特徴とする請求項 5 に記載の情報処理方法。

【請求項 7】

前記認識工程は、

前記入力パターンの各次元の特徴量を取得する第 1 取得工程と、

前記第 1 取得工程で取得した該入力パターンの各次元の特徴量を用いて、前記第 2 生成工程で生成された前記分類木をたどることにより、前記入力パターンに対応するカテゴリを認識する認識工程とを備えることを特徴とする請求項 6 に記載の情報処理方法。

【請求項 8】

パターンを分類するための分類木を生成するための情報処理装置であって、

学習パターンから座標値で表される複数次元の特徴量を取り出し、各次元の座標値が属する座標軸を所定の方法で再帰的に断片化し、当該次元の座標値が該断片化された座標軸のいずれの断片に属するか判断することにより、各次元についての階層構造を生成する第 1 生成手段と、

前記第 1 生成手段で生成された各次元の階層構造に基づいて、分類木の上位ノードから順に下位ノードへ生成していく際、生成途中の分類木のノードにおける各次元の分類効率に基づいて、いずれの次元で分類を行なうべきかを選択し、当該選択された次元を用いて前記分類木のノードを生成していく第 2 生成手段とを備えることを特徴とする情報処理装置。

【請求項 9】

パターンを分類するための分類木を生成するための情報処理方法をコンピュータに実行させるための制御プログラムを格納した記憶媒体であって、該情報処理方法が、

学習パターンから各次元が 1 つの座標値で表される複数次元の特徴量を取り出し、各次元の座標値が属する座標軸を所定の方法で再帰的に断片化し、当該次元の座標値が該断片化された座標軸のいずれの断片に属するか判断することにより、各次元についての階層構造を生成する第 1 生成工程と、

前記第 1 生成工程で生成された各次元の階層構造に基づいて、分類木の上位ノードから順に下位ノードへ生成していく際、生成途中の分類木のノードにおける各次元の分類効率に基づいて、いずれの次元で分類を行なうべきかを選択し、当該選択された次元を用いて前記分類木のノードを生成していく第 2 生成工程とを備えることを特徴とするコンピュータ読取可能な記憶媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は画像、文字、音声等のパターンを認識する際に使用する分類木を作成する情報処理方法及び装置に関するものである。

【0002】

【従来の技術】

従来、文献“Classification and Regression Trees” by Breiman, Friedman, Olshen, and Stoneにあるように、いわゆる分類木を用いてパターンを分類する手法が存在した。この方法は認識速度が比較的速く、適用できるパターンの種類を問わないという利点を有する。このため、画像、文字、音声等のパターン認識において広く採用されている。

【0003】

【発明が解決しようとする課題】

しかしながら、この方法では、パターンを分類するための分類木を作成するのに時間がかかるという欠点があった。特にパターンを構成する特徴量の次元が多くなると、分類木の作成に膨大な時間がかかってしまう。

【0004】

例えば、分類木の各ノードではある特徴量の次元に着目し、入力パターンのその次元（特徴量）がある値より大きい小さいかについての判断を行う。その際使用される“ある値”のことを通常“閾値”と呼ぶ。つまり、分類木を作成するにあたり、“ n 次元の中でどの次元に着目するか？”ということと、“その次元のどこに閾値を設定するか？”という

10

【0005】

本発明は上記の問題に鑑みてなされたものであり、複数の次元で表される特徴量を有するパターンを分類するための分類木を高速に作成することを可能とする情報処理方法及び装置を提供することを目的とする。

【0006】

また、本発明の他の目的は、少量の学習パターンでも高認識率を実現できる情報処理方法及び装置を提供することにある。

20

【0007】**【課題を解決するための手段】**

上記の目的を達成するための本発明の情報処理方法は、

記憶媒体に記憶されている制御プログラムを実行することによりパターンを分類するための分類木を生成するように情報処理装置を制御するための情報処理方法であって、

学習パターンから各次元が1つの座標値で表される複数次元の特徴量を取り出し、各次元の座標値が属する座標軸を所定の方法で再帰的に断片化し、当該次元の座標値が該断片化された座標軸のいずれの断片に属するか判断することにより、各次元についての階層構造を生成する第1生成工程と、

前記第1生成工程で生成された各次元の階層構造に基づいて、分類木の上位ノードから順に下位ノードへ生成していく際、生成途中の分類木のノードにおける各次元の分類効率に基づいて、いずれの次元で分類を行なうべきかを選択し、当該選択された次元を用いて前記分類木のノードを生成していく第2生成工程とを備える。

30

【0009】

更に、本発明の他の態様による情報処理装置は以下の構成を備える。すなわち、

パターンを分類するための分類木を生成するための情報処理装置であって、

学習パターンから座標値で表される複数次元の特徴量を取り出し、各次元の座標値が属する座標軸を所定の方法で再帰的に断片化し、当該次元の座標値が該断片化された座標軸のいずれの断片に属するか判断することにより、各次元についての階層構造を生成する第1生成手段と、

40

前記第1生成手段で生成された各次元の階層構造に基づいて、分類木の上位ノードから順に下位ノードへ生成していく際、生成途中の分類木のノードにおける各次元の分類効率に基づいて、いずれの次元で分類を行なうべきかを選択し、当該選択された次元を用いて前記分類木のノードを生成していく第2生成手段とを備える。

【0011】**【発明の実施の形態】**

以下、添付の図面を参照して本発明の好適な実施形態を説明する。

【0012】

<第1の実施形態>

図1は以下に説明する各実施形態すべてに係わるパターン認識方法が適用される情報処理

50

装置の概略の構成を示すブロック図である。図 2 に示されるように、本実施形態による情報処理装置（パターン認識装置）は、パターン入力装置 201、表示装置 202、中央処理装置（CPU）203、メモリ 204 から構成される。

【0013】

パターン入力装置 201 は、例えばオンライン文字認識を行う装置ならば、デジタイザとペンを有し、デジタイザの上にペンによって入力された文字や図形の座標データを CPU 203 に渡す。また、音声認識を行う装置ならば、パターン入力装置 201 はマイクロフォンを備えることになる。すなわち、このパターン入力装置 201 は、これから認識しようとするパターンが入力できるものであればよく、スキャナ等でもよい。表示装置 202 はパターン入力装置 201 に入力されたパターンデータや CPU 203 がパターン認識した結果を表示する。CPU 203 は入力されたパターンの認識処理を行なうなど、すべての装置の制御を行なう。メモリ 204 は CPU 203 が使用する認識プログラムやパターン認識のための辞書を記録する ROM や、入力されたパターンデータ、認識プログラムの使用する変数等を一時的に記憶する RAM 等を備える。

【0014】

図 2 は、第 1 の実施形態による分類木生成の為の制御構成を説明する図である。同図において、101 は元の学習用パターンである学習用パターン、102 は学習用パターン 101 を分類するための特徴量の各次元を予め階層的に断片化する階層断片化前処理部、103 は階層断片化前処理部 102 によって階層的に処理された後の階層化された学習用パターン、104 は階層化された学習用パターン 103 を元に分類木を作成する分類木作成部、105 は分類木作成部 104 が分類木を作成する過程で使用する展開変数判別部、106 は分類木作成部 104 によって生成された分類木である。本実施形態におけるインプットは学習用パターンであり、アウトプットは分類木 106 となる。

【0015】

図 3 は階層断片化前処理部 102 による特徴量の断片化を説明する図である。一般にパターンは n 次元のベクトルを使って表すことができる。上述の “Classification and Regression Trees” にも記載されているが、通常この n が大きい場合、分類木の各ノードではある特徴量の次元に着目し、入力パターンのその次元（特徴量）がある値より大きい小さいかについての決定を行う。その際使用される “ある値” のことを通常 “閾値” と呼ぶ。つまり、分類木を作成するにあたり、“ n 次元の中でどの次元に着目するか？”ということと、“その次元のどこに閾値を設定するか？”ということを決めないと、各ノードでの Decision を作成できないわけである。具体的には、通常は n 次元のすべての次元に対して、仮にその次元に着目したとした時の分類効率が最高になる閾値を求め（求めた閾値の総数は n となる）、その中で最も分類効率の高くなる次元と閾値を導く。このため、特徴量の次元が増加すればするほど計算量が増大し、分類木の作成に膨大な時間を費やすようになってしまう。

【0016】

本実施形態では、図 3 の（a）にあるように、例えばあらかじめ n 次元の各次元を再帰的に 2 等分しておく。こうすることによって、上記の「各次元に対して閾値を求める」という処理を省くことができるようになる。

【0017】

なお、以上の論議はいわゆる 2 分木（Binary Tree）の構造の分類木を作成する場合の話で、一般的な構造の分類木に対しては、本発明の分割は 2 分割である必要はなく、例えば図 3（b）の様に 3 分割であっても構わない。また、この分割は等分である必要もなく、例えば図 3（c）の様に不均一分割であっても構わない。

【0018】

以上のように、階層断片化前処理部 102 は、パターンを分類するための特徴量の各次元を、所定の方法で分割し、分類木の各ノードにおいて使用される閾値を前もって決定しておく。

【0019】

10

20

30

40

50

以下、上記の構成による分類木の生成手順を説明する。本実施形態では、オンライン手書き文字認識技術に本発明を適用した例を用いて説明することにする。なお、本発明が一般のパターン認識技術に対しても、なんら変更なく、適用可能であることは明らかである。

【0020】

まず、図4を用いてパターンを表現するための特徴量を説明する。図4では「あ」の文字が入力文字となっている。401の枠は、特徴量を求めるときの基準となる枠であり、例えば実際の認識文字板の入力枠でもよいし、入力された文字の外接矩形でもよい。「あ」を構成する3つのストロークは点列から構成されており、まずその点列からストロークを記述するのに妥当な点（いわゆる特徴点）を求める。例えば、図4では「あ」の第1画（402）の始点402aと終点402bを特徴点としている。例えばこのように特徴点を取ると、「あ」という文字のように3つのストロークで構成される文字は、 $3 \times 2 = 6$ 次元の特徴ベクトルで記述されることになる。また、各特徴点をX座標のみを用いれば3ストロークで構成される文字の次元は6次元であり、X座標とY座標を用いれば12次元となる。

10

【0021】

なお、始点、終点の他に、例えば各ストロークの1/4点、中点、3/4点、または重心等の特徴点の座標を使って文字を記述してもよいし、文字全体の最上点、最下点、最右点、最左点等の特徴点の座標を使ってもよい。

【0022】

以上述べた特徴量の求め方は、極めて常識的、標準的なものである。これに対して、例えば図4に示されるような、あるストローク（403の場合、第1画）の特徴点（403の場合、始点）からあるストローク（403の場合、第2画）の特徴点（403の場合、終点）への差分ベクトル403から得られる特徴量を用いてもよい。

20

【0023】

以上説明したように、一旦パターンを記述する特徴量の次元（＝変数の数）が決定されれば、その後は、それら変数のそれぞれを図3の（a）、（b）、（c）にある様に再帰的に断片化する。そして、学習用パターンを、断片化された次元によって階層化し、階層化された学習用パターン103を生成する。これが階層断片化前処理部102による処理である。

【0024】

図5は本実施形態による分類木作成の手順を説明する図である。図5ではX軸を再帰的に3回、2分割し、X軸を8個に断片化している。そして、「あ」の第1画のストロークの始点は左から2番目の断片に属していることになる。もちろんY軸に関しても、同様に断片化をする。但し、X軸とY軸が全く同じ形式で断片化されなくてはならないわけではない。また、図5には第1ストロークの始点のX座標という特徴量の次元を分割する様子を示しているが、第1ストロークの終点、第2、第3ストロークの始点、終点のX座標の各次元についても同様に次元の分割を行う。ただし、各次元の分割数は同一である必要は無く、また、すべてが等分割である必要も無い。

30

【0025】

図5の通りX軸を断片化した後、分類木を作成するわけであるが、例えば特徴量が第1画のストロークの始点のX座標しかない場合（特徴量の次元画1次元＝1変数）、図5の「あ」パターンからは図5下の分類木が作成される。

40

【0026】

特徴量の次元が n （ $n > 1$ ）である場合、分類木の各ノードで、どの次元に注目すれば最も分類効率が上がるかを評価しながら分類木を作成していく。この分類効率の評価を行うのが展開変数判別部105である。なわち、展開変数判別部105は、階層化された学習用パターン103を用いて、予め定めた閾値に基づいて分類効率を計算する。

【0027】

なお、分類効率の評価としては、例えば文献“Classification and Regression Trees”の“Gini Criterion”を用いる方法

50

、情報理論で一般的に使用されている“Mutual Information”を用いる方法等、周知の手法を適用できる。

【0028】

また、その他の分類木を作成するための一般的な手法は全て公知のものが使用できる。例えば、文献“Classification and Regression Trees”にはリーフノードを作成する際に、何の尺度を用いれば良いかが記述されている。例えば、エントロピーをその尺度として採用するならば、本実施形態の分類木作成処理のフローチャートは図6のようになる。なお、エントロピーは $P_i \cdot \log P_i$ で定義される。

【0029】

図6において、ステップS11で処理対象のノードに着目する。まず最初に分類木のルートノードから出発することになる。ルートノードはもちろん複数のカテゴリーの学習パターンが存在するので、ステップS12からステップS13進む。次に、ステップS13において、そのノードにおけるエントロピーを算出し、得られたエントロピーがある閾値より小さい場合は、ステップS16へ進み、そのノードをリーフノードとして、最も存在確率の高いカテゴリー番号をそのノードに対応させる。例えば、そのノードに「な」と「は」が存在し、「な」が圧倒的に多い場合、当該ノードは「な」のリーフノードとする。もちろんルートノードはリーフノードにはなり得ないので、ステップS13からステップS14へ進むことになる。

【0030】

ステップS14では、展開変数判別部105による分類効率の判定を行い、その結果に基づいて、当該ノードに用いるべき特徴変数（特徴量の次元）に関するDecisionを行う。その結果、2分岐の場合は2つのノードが新たに生成される。なお、一般化して述べれば、n分岐の場合はn個のノードが生成されることになる。そして、そのそれぞれのノードに対してもとの「あるノードに着目」に戻る。

【0031】

図7は第1ストロークのX座標とY座標を変数（特徴量の次元）とする、2変数の場合の分類木の生成を説明する図である。図7の（a）に示されるように、本例では、X座標は7つの閾値（X1～X7）で8つの階層に断片化され（図5と同様）、Y座標は3つの閾値で4つの階層に断片化されている。図7の（b）は、各次元の閾値の階層構造を示す図である。例えば、X4より左にあると分類された場合、次に比較される閾値はX2となる。

【0032】

図7の（c）は図6のフローチャートに従って生成される分類木の一例を示す図である。ノード701はルートノードであり、ステップS14によってX座標の次元が選択されたことにより、ステップS15において閾値であるX4が設定される。そして、この結果、2つのノード702、703が生成される。次に、ノード702が着目ノードとなり、ノード702が複数のカテゴリーに属する学習パターンを有する場合、ステップS14においてノード702で展開変数判別部105が採用すべき次元を決定する。ここでは、Y座標が選択されたので、ステップS15において閾値としてY2が設定される。以下、同様に、ノード704ではY座標が選択されて閾値Y3が、ノード705ではX座標が選択されて閾値X2が設定されている。なお、各閾値の設定は、図7の（b）に示した階層構造に則ってなされる。すなわち、X4より右側へ分岐した部分（ノード701において右側に分岐した部分）において、次にX座標が選択された場合の閾値はX2ということになる（従って、ノード705の閾値はX2となる）。

【0033】

さて、着目しているノードが単一のカテゴリーの学習パターンしか含まない場合は、当該ノードをリーフノードに決定する（ステップS17、S16）。また、ステップS17でNOに分岐する場合は、学習パターンが存在しないことを意味しており、その場合はステップS18において当該ノードを分類木から削除する。

10

20

30

40

50

【 0 0 3 4 】

以上の手順に従って分類木を作成していった、すべてのノードに対して処理を終了した時点で、分類木の作成終了とする（ステップ S 1 9、S 2 0）。

【 0 0 3 5 】

図 7 は本実施形態の情報処理装置の構成例を示すブロック図である。8 1 は C P U であり、メモリ 8 3 に格納された制御プログラムに従って各種制御を実現する。8 2 はハードディスクであり、入力データとして全学習パターンが保持されるとともに、出力データとして分類木の作成結果が保持される。8 3 はメモリであり、プログラム部 8 4 とデータ部 8 5 を有する。プログラム部 8 4 には、図 6 で説明したフローチャートの手順を実現する制御プログラム等がサブルーチンとして存在し、C P U 8 1 は個々のサブルーチンを読み出して実行する。データ部 8 5 には、個々の学習パターンを暫定的に保持する学習パターンバッファ 8 5 a と、階層断片化された学習パターン領域 8 5 b と、作成途中の分類木を保持する分類木バッファ 8 5 c が存在する。

10

【 0 0 3 6 】

学習パターンバッファ 8 5 a には、ハードディスク 8 2 から全学習パターンを逐次格納する。そして、階層断片化前処理部 1 0 2 によって階層化された学習用パターン 1 0 3 を得て、これを階層化された学習パターン領域 8 5 b に格納する。その後、分類木作成部 1 0 4 は図 6 による分類木作成の手順を実行して、分類木を作成する。

【 0 0 3 7 】

以上説明したように、上記第 1 の実施形態によれば、各ノードにおいて採用すべき次元が決定されると、階層断片化前処理部 1 0 2 において予め決定された閾値が自動的に設定されることになるので、計算量が減少し、分類木の作成時間が短縮できる。

20

【 0 0 3 8 】

なお、上記実施形態では 3 ストロークの文字を分類するための分類木の生成を示したが、他のストローク数、例えば 4 ストロークの文字や 2 ストロークの文字を分類するための分類木の生成も同様に行なわれる。

【 0 0 3 9 】

< 第 2 の実施形態 >

図 9 は、第 2 の実施形態によるパターン認識処理の機能構成を説明するブロック図である。8 0 1 は入力パターンであり、認識対象（分類対象）となる入力パターンである。8 0 2 は入力パターン 8 0 1 を階層的に断片化する階層断片化前処理部である。また、8 0 3 は断片化前処理された後の階層化された入力パターン、8 0 4 は分類木である。8 0 5 はカテゴリー判別部であり、階層化された入力パターン 8 0 3 と分類木 8 0 4 を元にカテゴリーの判別確率を求める。第 2 の実施形態におけるインプットは“入力パターン 8 0 1”で、アウトプットは“認識候補 8 0 6”となる。また、分類木 8 0 4 としては、第 1 の実施形態の方法で作成し得る分類木を採用するものとする。

30

【 0 0 4 0 】

入力パターン 8 0 1 は、第 1 の実施形態の学習用パターン 1 0 1 に対応するもので、実質的には何ら変りはない。階層断片化前処理部 8 0 2 は第 1 の実施形態の階層断片化前処理部 8 0 2 と同様のものである。階層化された入力パターン 8 0 3 は、第 1 の実施形態の場合は学習パターンの数だけ存在していたが、本第 2 の実施形態においては、入力パターン 8 0 1 から導きだされた 1 つのみとなる。

40

【 0 0 4 1 】

カテゴリー判別部 8 0 5 は、階層化された入力パターン 8 0 3 に基づき、分類木 8 0 4 をたどって葉（リーフノード）に到達した時点で、その葉に存在する最も確からしいカテゴリーを認識結果として出力する。また、葉に到達しなかった場合は、最後に通過したノードに含まれるカテゴリー確率（カテゴリー存在確率）を結果として出力する。

【 0 0 4 2 】

次に、第 2 の実施形態の情報処理装置の構成例を図 1 0 に示す。1 0 1 0 は C P U であり、メモリ 1 0 4 0 に格納された制御プログラムに基づいて各種の制御を実現する。1 0 2

50

0 は入力装置であり、入力パターン 8 0 1 を入力するのに用いられる。入力装置 1 0 2 0 は認識処理の対象となるパターンを入力するのに適したものであり、例えば手書き文字認識であれば座標入力装置が、音声認識であればマイクロフォンが用いられる。1 0 3 0 はハードディスクであり、分類木を格納する。メモリ 1 0 4 0 はプログラム部 1 0 5 0 とデータ部 1 0 6 0 とに別れる。プログラム部 1 0 5 0 には、図 9 のブロック図の個々の部品がサブルーチンとして存在し、CPU 1 0 1 0 は個々のサブルーチンを読み出して実行する。データ部 1 0 6 0 には、入力装置 1 0 2 0 からユーザが入力した入力パターン 1 0 6 a (入力パターン 8 0 1 を格納する) と、入力パターンを階層断片化された入力パターン 1 0 6 b (階層化された入力パターン 8 0 3 を格納する) と、認識候補 1 0 6 c が存在する。カテゴリ判別部 8 0 5 は階層化された入力パターン 8 0 3 と分類木 8 0 4 に従って認識候補 8 0 6 を決定し、データ部 1 0 6 0 の認識候補 1 0 6 c に格納する。

10

【0043】

< 第 3 の実施形態 >

図 1 1 は、第 3 の実施形態による分類木作成機能を説明するブロック図である。1 0 0 1 は学習用パターン、1 0 0 2 は学習用パターン 1 0 0 1 をサブパターン毎に切り出すサブパターン取り出し部、1 0 0 3 は学習用サブパターンを階層的に断片化する階層断片化前処理部、1 0 0 4 は階層断片化前処理部 1 0 0 3 によって断片化前処理された後の階層化された学習用サブパターン、1 0 0 5 は階層化された学習用サブパターン 1 0 0 4 を元に分類木を作成する分類木作成部、1 0 0 6 は分類木作成部 1 0 0 5 が分類木を作成する過程で使用する展開変数判別部、1 0 0 7 は分類木作成部 1 0 0 5 による処理の結果として得られた分類木である。第 3 の実施形態におけるインプットは、“学習用パターン 1 0 0 1” で、アウトプットは、“分類木 1 0 0 7” となる。なお、本例において、サブパターンの概念は漢字の偏と旁からきている。例えば、「漢」は 1 3 画で、x、y で 2 6 次元となるが、「最初の 3 画 (偏)」と「最後の 1 0 画 (旁)」に分けて、それぞれの次元で分類木を作ることができる。

20

【0044】

第 1 の実施形態で述べたオンライン手書き文字認識を例にとって説明する。第 1 の実施形態で述べたとおり「あ」という 3 画のストロークで書かれる文字において、始点終点の座標値を特徴点として用いると、パターンは結局 $3 \times 2 = 6$ 次元の特徴ベクトルで記述されることになる。例えば、この特徴ベクトルを構成する 6 次元から 2 次元ずつ抽出すると、 $6C2 = 15$ 個の 2 次元ベクトルがサブパターンとして抽出できることになる。第 3 の実施形態ではこの 15 個のベクトル毎に第 1 の実施形態と同様な手順で分類木を作成するのである。この方法は、学習パターンの情報を冗長に保持し、それに関するパターン認識を行うことに相当する。よって、この方法を用いると、結果として少量の学習パターンでも高認識率が実現できることになる。但し、結果的に生成される全分類木の合計サイズは、冗長性が高まれば高まる程大きくなる。

30

【0045】

例えば極端な例として、6 次元の特徴ベクトルから 1 次元ずつ抽出すると、最も単純な分類木作成が行え、かつ分類木のサイズは小さくなるが、冗長性が全くなり、分類能力は低下する。

40

【0046】

6 次元ベクトルの場合、最大 $(6C6 + 6C5 + 6C4 + 6C3 + 6C2 + 6C1) = 63$ 個の次元の異なるベクトルがえられ、これらそれぞれについて分類木を作成すると、最も冗長性は高くなり認識率も高くなる。

【0047】

なお、第 3 の実施形態の具体的な装置構成例は、第 1 の実施形態の図 7 に示したものと基本的に同じである。

【0048】

以上のように、分類木の作成を第 1 の実施形態で説明したのと同様の手法を行うことにより、分類木をより短時間で作成できるので、上述のような分類木を冗長に作成するように

50

しても十分実用に耐えるスループットが得られる。そして、このように冗長な分類木を作成して、後述の第4の実施形態の如く認識処理を行うことにより、認識率を高めることができる。

【0049】

<第4の実施形態>

図12は、第3の実施形態による分類木作成機能を説明するブロック図である。1101は入力パターン、1102は入力パターン1101をサブパターン毎に切り出すサブパターン取り出し部、1103はサブパターンを階層的に断片化する階層断片化前処理部、1104は断片化前処理された後の階層化されたサブパターン、1105は分類木、1106は階層化されたサブパターン1104と分類木1105を元にカテゴリーの判別確率を求めるカテゴリー判別部、1107はカテゴリー判別部1106による処理の結果得られた各カテゴリーの判別確率を統合する判別確率統合部である。第4の実施形態におけるインプットは“入力パターン1101”で、アウトプットは“認識候補1108”となる。なお、上記の分類木は、第3の実施形態の方法で作成しうる分類木であることが望ましい。

10

【0050】

入力パターン1101は、第3の実施形態の学習用パターン1001に対応するもので、実質的には何ら変りはない。サブパターン取り出し部1102、階層断片化前処理部1103の2つの機能は第3の実施形態のそれぞれに対応する構成（サブパターン取り出し部1002、階層断片化前処理部1003）と同様である。ただし、階層化された入力サブパターン1104は、第3の実施形態の場合は学習パターンの数だけ存在したが、本第4の実施形態においては、入力されたパターンから導き出された1つのみとなる。

20

【0051】

判別確率統合部1107は、カテゴリー判別部1106のサブパターン毎の結果について相加平均、または相乗平均等の平均処理を行う。

【0052】

なお、第4の実施形態の具体的な装置構成例は第2の実施形態の図9に示したものと基本的に同様となる。

【0053】

なお、本発明は、複数の機器（例えばホストコンピュータ、インタフェイス機器、リーダ、プリンタなど）から構成されるシステムに適用しても、一つの機器からなる装置（例えば、複写機、ファクシミリ装置など）に適用してもよい。

30

【0054】

また、本発明の目的は、前述した実施形態の機能を実現するソフトウェアのプログラムコードを記録した記憶媒体を、システムあるいは装置に供給し、そのシステムあるいは装置のコンピュータ（またはCPUやMPU）が記憶媒体に格納されたプログラムコードを読み出し実行することによっても、達成されることは言うまでもない。

【0055】

この場合、記憶媒体から読出されたプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。

40

【0056】

プログラムコードを供給するための記憶媒体としては、例えば、フロッピディスク、ハードディスク、光ディスク、光磁気ディスク、CD-ROM、CD-R、磁気テープ、不揮発性のメモリカード、ROMなどを用いることができる。

【0057】

また、コンピュータが読出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているOS（オペレーティングシステム）などが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言

50

うまでもない。

【 0 0 5 8 】

さらに、記憶媒体から読出されたプログラムコードが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書込まれた後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【 0 0 5 9 】

【 発明の効果 】

以上説明したように、本発明によれば、複数の次元で表される特徴量を有するパターンを分類するための分類木を高速に作成できるという効果がある。 10

【 0 0 6 0 】

また、本発明によれば、少量の学習パターンでも高認識率が実現できるという効果がある。

【 0 0 6 1 】

【 図面の簡単な説明 】

【 図 1 】 本実施形態による情報処理装置の概略の構成を示すブロック図である。

【 図 2 】 第 1 の実施形態による分類木生成の為に制御構成を説明する図である。

【 図 3 】 階層断片化前処理部 1 0 2 による特徴量の断片化を説明する図である。

【 図 4 】 実施形態の情報処理装置におけるパターンの記述を説明する図である。 20

【 図 5 】 本実施形態による分類木作成の手順を説明する図である。

【 図 6 】 第 1 の実施形態における分類木作成過程を示すフローチャートである。

【 図 7 】 第 1 ストロークの X 座標と Y 座標を変数（特徴量の次元）とする、2 変数の場合の分類木の生成を説明する図である。

【 図 8 】 第 1 の実施形態の情報処理装置の構成例を示すブロック図である。

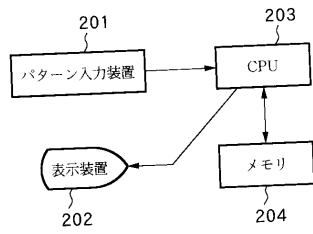
【 図 9 】 第 2 の実施形態によるパターン認識処理の機能構成を説明するブロック図である。

【 図 1 0 】 第 2 の実施形態による情報処理装置の構成例を示すブロック図である。

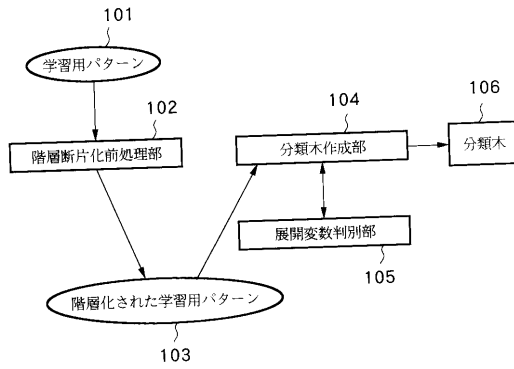
【 図 1 1 】 第 3 の実施形態による分類木作成機能を説明するブロック図である。

【 図 1 2 】 第 4 の実施形態による分類木作成機能を説明するブロック図である。 30

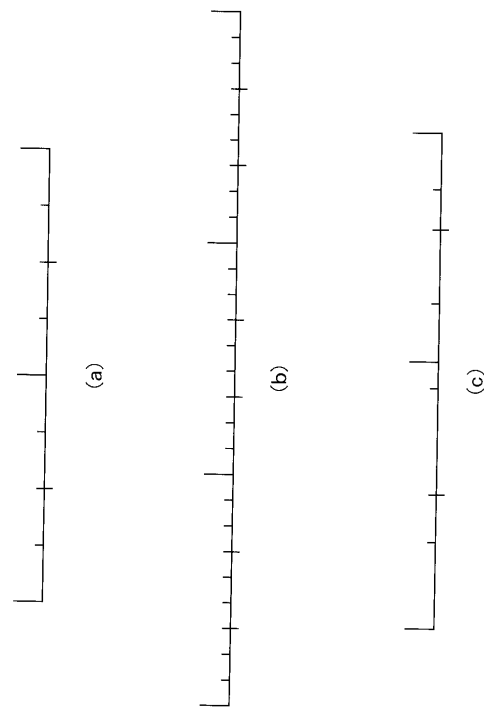
【図 1】



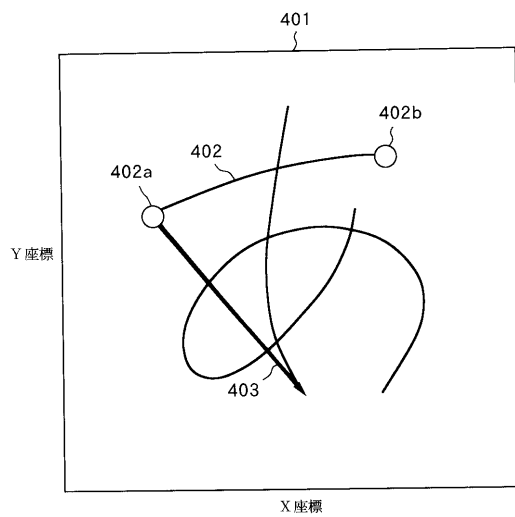
【図 2】



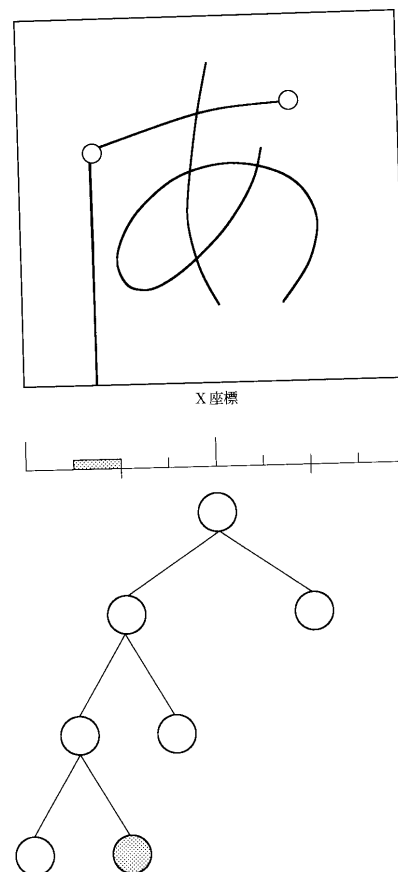
【図 3】



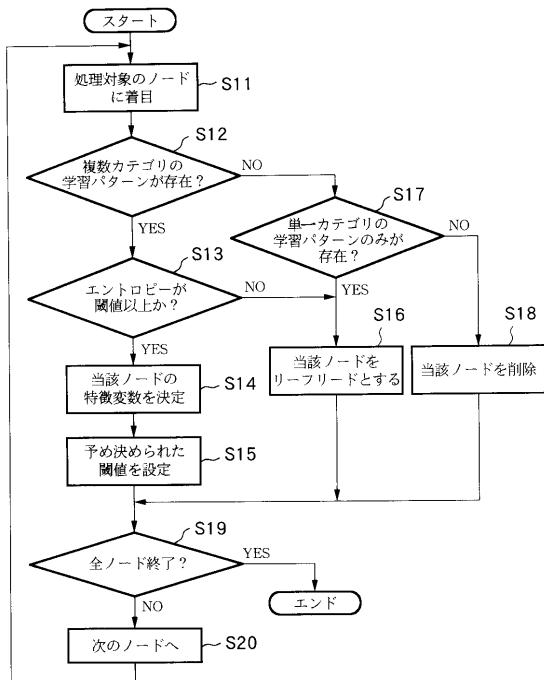
【図 4】



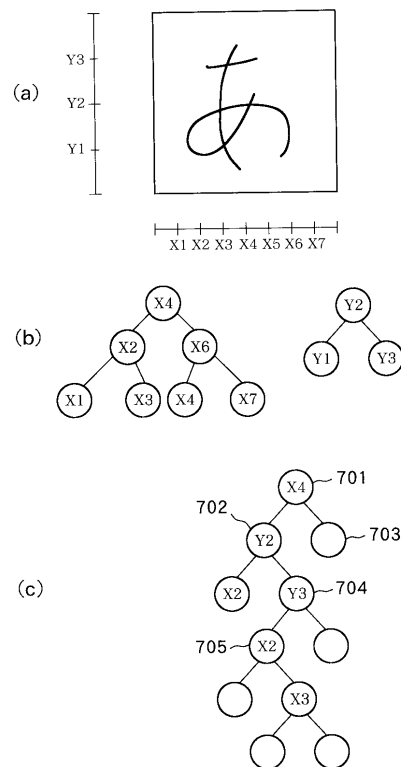
【図 5】



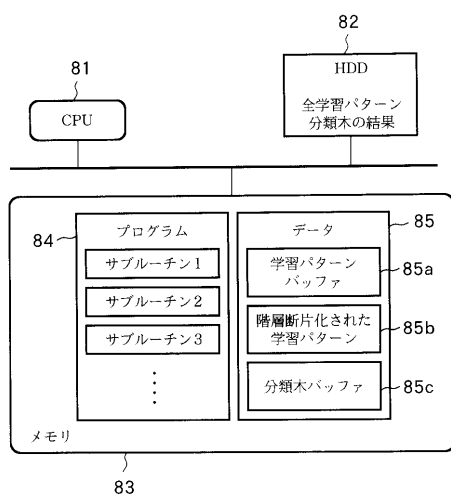
【図 6】



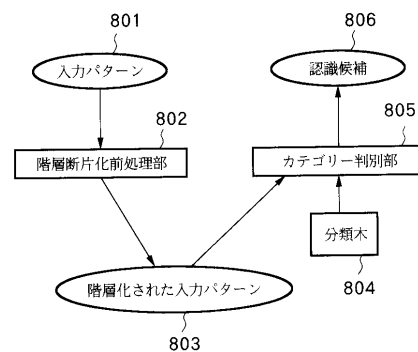
【図 7】



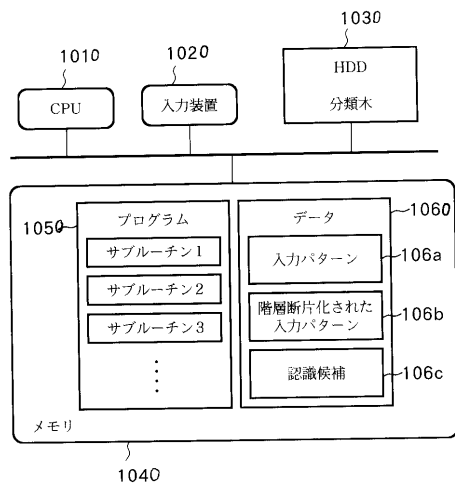
【図 8】



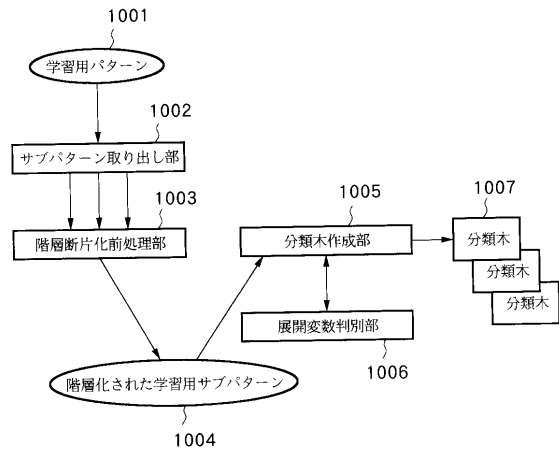
【図 9】



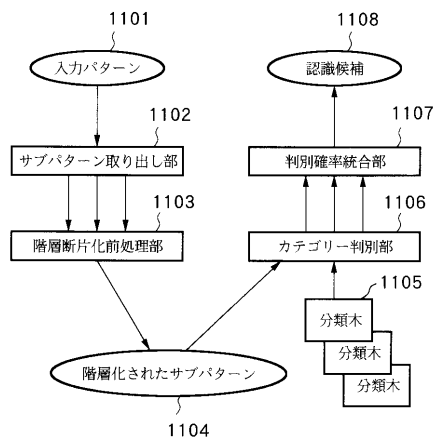
【図 10】



【図 11】



【図 12】



フロントページの続き

- (56)参考文献 特開平08-123913(JP,A)
特開平02-125391(JP,A)
特開昭62-198993(JP,A)

(58)調査した分野(Int.Cl.⁷, DB名)

G06K 9/00~9/76
G06T 7/00~7/60
G10L 3/00