# K-M木探索の範囲狭化による近傍文字識別の高速化

亀山博史† 鈴木 寿††

文字認識の実用化において，識別系の設計を効率良く行うことは重要である．最近傍識別は，識別能力が高く辞書の作成が容易であるという特長を持っているため，実用的に有効であると考えられる．しかし1つの難点として，参照パターンとの照合をいかに高速に行うかという問題がある．本論文では，K-M木とよばれる2分木を辞書のデータ構造として用い，木内の探索における距離計算の回数を削減する手法について述べる．K-M木へのデータの格納は高速に行えるため，多様な読み取り対象に対応して高性能な識別系を迅速に設計できることが期待される．本手法では，従来の三角不等式に基づいて探索範囲を狭化する条件にパラメータ $\alpha$ を導入することにより，探索範囲をより狭化できるようにした．数字，英大文字およびカタカナからなる4つの手書き文字のサンプルを対象とした実験を行った結果，適切にパラメータ $\alpha$ を設定することによって，高い正読率を保ちながら大幅な高速化が達成され，実用的に有効であることが示された．また，このようなパラメータの設定方法についても検討した．

# Speed-up of a Nearest-neighbor Character Classifier by Narrowing the Search Area over a K-M Tree

HIROFUMI KAMEYAMA† and HISASHI SUZUKI††

The nearest-neighbor classifier is effective for the practical use of character recognition, since its ability of discrimination is high and a discrimination dictionary can be made easily. However, it is a problem to speed up the search in a dictionary for the nearest pattern. This paper presents a method of eliminating the number of distance computations for the nearest-neighbor character classifier by using a K-M tree which is a binary tree proposed by Kalantari and McDonald. Since construction of a K-M tree is quickly executed, it can be expected that a high-performance classifier will be efficiently designed according to various character sets. In searching the nearest pattern over the K-M tree, the search area is narrowed by using a rule based on a triangle inequality. For further narrowing of the search area over the K-M tree, a parameter $\alpha$ is introduced into the rule. Experiments were made by using four kinds of hand-written character sets (two numeral sets, a capital alphabet set, and a katakana set). The results suggest a practical efficiency of the method: The classifier with a parameter $\alpha$ chosen appropriately keeps the correct recognition rate high, and a good number of distance computations are eliminated. A method of setting the parameter appropriately has also been presented.

## 1. はじめに

近年，文字読み取り手法の進歩[1)~4)]と高性能なハードウェアの低価格化によって，光学式文字読み取り（OCR）の事務処理における応用は様々な対象にまで広がってきている．ところが，認識率の良い悪いは，筆記具や字体の品質などに大きく依存する．それゆえメーカの開発者は，別の用途で設計した文字認識系を再調節したり，あるいは読み取り対象ごとに文字認識系を設計するといった手段によって対応している．

文字認識系は大まかには特徴抽出系と識別系から構成される．高性能な文字認識系を設計するためには，カテゴリー間を良く分離する特徴抽出系の設計と，それに適合した識別系の設計が必要となる．認識対象に応じて特徴量の種類や次元数が変更されたとき，識別系の設計をやり直さなければならない．したがって，識別系の設計を効率良く短時間に行うことが，文字認識の実用化においては重要である．

さて，最近傍識別法は，まずカテゴリーが付与された参照パターンの集合（識別辞書）から入力パターンとの距離が最も近いパターンを探索する．そしてこの

† グローリー工業株式会社中央研究所
Central Research Laboratory, GLORY Ltd.
†† 中央大学理工学部情報工学科
Department of Information and System Engineering, Faculty of Science and Engineering, Chuo University

パターンのカテゴリーを入力パターンのカテゴリーと見なす．この方法において，参照パターンの集合を大きくとれば，パターンの分布が分からなくてもベイズ識別に近い程度の高い識別能力を得られることが知られている[5]．また，認識対象から抽出された標本パターンの特徴量を，そのまま辞書に格納して参照パターンとして使用することが可能である．これらの特長により，高性能な識別系の効率的な設計という観点から，筆者らは最近傍識別を利用した文字認識法が実用的に有効であると考えている．しかし1つの難点として，参照パターンが増加するにつれ記憶容量と探索時間の増大が問題となる．

探索集合 $\boldsymbol{S}$ から被探索点 $\boldsymbol{x}$ に最も近い点を見つける問題に関して，データ $\boldsymbol{p} \in \boldsymbol{S}$ を木に格納し，木内の探索の際，三角不等式に基づいて距離計算の回数を削減する手法を提案したもの[6),7)]やその応用研究[8)]がある．Kalantari と McDonald[7)]が示した木（K-M木とよぶ）は，各ノードにデータを格納した2分木である．$N$ 個のデータを格納するK-M木を生成するための計算時間は $\mathcal{O}(N \log N)$ と速く，しかもデータを木へ連続的に追加・格納することが可能であるという特長がある．それゆえ，K-M木を最近傍識別のための辞書のデータ構造に採用することによって，多様な読み取り対象に識別系を迅速に対応させることが可能になると期待される．しかし，K-M木を用いた探索法では，探索に要する距離計算の回数はデータの次元数に大きく依存しており，次元数が増加してくると距離計算の削減効果は低下してくる．従来，高いパターン分離能力を持った特徴抽出法[9)~13)]が開発されており，それらによって文字の元パターンは数十～数百の次元数を持った特徴量に変換されるが，この次元数は文献7)，8)で検証された次元数より大幅に大きい．したがって，文字認識に，K-M木を用いた探索法[7)]をそのまま適用しても，処理の高速化は十分には達成できないと思われる．

文献1)では，最近傍識別の辞書を設計するために6万文字に及ぶ手書き数字を用いている．そこでは，標本パターンから得られた特徴量をそのまま参照パターンとして用いるのではなく，識別に有用でないパターンを削除する方法[14)]や，三角不等式の適用の際，基準点を効果的に配置する方法[15)]を採り入れることによって，記憶容量と計算時間の問題を解決している．また文献16)，17)では，訓練サンプルを用いて参照パターンを繰り返し修正することによって，少数の参照パターンで高い識別能力を得る手法が示されている．以上の手法は識別系の設計に十分な計算時間が用意できる場合には非常に有効である．$k$ 最近傍識別を適用した漢字の認識[18)]では，大分類による候補の削減と，距離計算の打ち切りによって処理量を低減している．

本稿では，識別系の設計を効率良く行うという観点から，Kalantari と McDonald[7)]の木に着目し，これを利用した最近傍識別の高速化について検討する．高速化の改善を次のようにして達成する：従来法は，三角不等式に基づいたルールによって，1つも近傍点が存在しないことが確実な $\boldsymbol{S}$ の部分集合のみを探索しないようにしていた．これに対して提案法は，探索から除外される部分集合が大きくなるようなパラメータをルールに導入することによって，高速化を達成する．パラメータの導入によって，入力パターンに最も近いパターンが探索される保証はなくなる．このことによる識別能力への影響と高速化の改善効果について，筆記条件あるいは字種の異なる4種類の手書き文字のサンプルを用いた実験を行って検討する．実験では，高いカテゴリー分離能力を持つことが報告されている[12),20)]輪郭の方向成分を反映した特徴を用いる．適切にパラメータを設定することによって，木探索における識別能力の低下は小さく抑えられたまま大幅な高速化が達成できることを，実験により示す．また，このようなパラメータの設定方法についても検討する．

## 2. 最近傍点の探索法

Kalantari と McDonald が提案したアルゴリズム（K-Mアルゴリズム）は，$n$ 次元空間上の $N$ 個の点の集合 $\boldsymbol{S} = \{\boldsymbol{p}_1, \cdots, \boldsymbol{p}_N\}$ の中で点 $\boldsymbol{x}$ に最も距離の小さい $\boldsymbol{q}^* \in \boldsymbol{S}$ を高速に見つける問題（最近傍点問題）の一解法を与える[7)]ものであるが，本章ではこのアルゴリズムについて述べる．

### 2.1 探索範囲の狭化

$\boldsymbol{S}$ の任意の部分集合 $\tilde{\boldsymbol{S}}$ に対して，1点 $\boldsymbol{a}$ から $\tilde{\boldsymbol{S}}$ の各点までの距離の最大値を $R$ とおく．

$$R = \max_{\boldsymbol{p}_i \in \tilde{\boldsymbol{S}}} d(\boldsymbol{a}, \boldsymbol{p}_i)$$

ここに，$d(\cdot,\cdot)$ はユークリッド距離を表す．

**定理1** ある $\boldsymbol{q}' \in \boldsymbol{S}$ に対し，もし

$$d(\boldsymbol{a}, \boldsymbol{x}) - R \geq d(\boldsymbol{x}, \boldsymbol{q}') \qquad (1)$$

ならば（**図1**を参照），集合 $\tilde{\boldsymbol{S}}$ の中に，$\boldsymbol{q}'$ より $\boldsymbol{x}$ に近い点は1つも存在しない．

この定理は，不等式(1)が成立することが分かれば，部分集合 $\tilde{\boldsymbol{S}}$ 内を探索する必要がないことを示唆している．

### 2.2 K-Mアルゴリズム

K-Mアルゴリズムは，探索に先だって，K-M木と

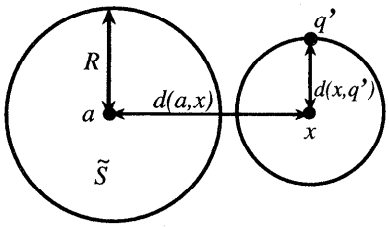



図1 探索範囲の狭化
Fig. 1 Narrowing of search area.

呼ばれる次のような2分木に集合 $\boldsymbol{S}$ を蓄える：K-M木 $T$ のノード $N_i$ $(i=0,1,2,\cdots,N)$ には，データ $\boldsymbol{p}_i$ $(i=1,2,\cdots,N)$（$N_0$ にはデータ $\boldsymbol{p}_i$ は格納されていない）および左右それぞれの子ノードを示すポインタ $L_{\text{child}}$，$R_{\text{child}}$ と，さらに左右それぞれの部分木の勢力半径

$$R_{\text{L}} = \max_{\boldsymbol{w}\in N_{\text{L}}} d(\boldsymbol{p}_{\text{L}}, \boldsymbol{w}), \tag{2}$$

$$R_{\text{R}} = \max_{\boldsymbol{w}\in N_{\text{R}}} d(\boldsymbol{p}_{\text{R}}, \boldsymbol{w}) \tag{3}$$

が格納してある（**図2**を参照）．ここに，$\boldsymbol{p}_{\text{L}}$ および $\boldsymbol{p}_{\text{R}}$ はそれぞれ左および右の子ノードに格納されたデータを表し，$N_{\text{L}}$ および $N_{\text{R}}$ はそれぞれ左および右の子ノードを頂点とする部分木を表す．K-M木の生成は，連続して与えられるデータを1個ずつ木の終端に追加していくことによって行われる．付録A.1にアルゴリズムを示す．

例として，9個の2次元のデータ $\boldsymbol{p}_1(0,8)$, $\boldsymbol{p}_2(30,8)$, $\cdots$, $\boldsymbol{p}_9(43,6)$ が順次与えられたときに生成されるK-M木を，図2に示す．ノード $N_0$ の左子ノード $N_{\text{L}}$ は，データ $\boldsymbol{p}_1$ が格納されている $N_1$ となり，右子ノード $N_{\text{R}}$ は，データ $\boldsymbol{p}_2$ が格納されている $N_2$ となる．各ノードの中段の数値は左右の部分木の勢力半径を示している．ノード $N_0$ の $R_{\text{L}}$ は，式(2)に従って，$N_1$ に格納されたデータ $\boldsymbol{p}_1$ から $N_1$ を頂点とする部分木に格納されたデータ $\boldsymbol{p}_1, \boldsymbol{p}_3, \boldsymbol{p}_4, \boldsymbol{p}_7, \boldsymbol{p}_8$ までの距離の最大値で，$R_{\text{L}} = \max(d(\boldsymbol{p}_1,\boldsymbol{p}_1), d(\boldsymbol{p}_1,\boldsymbol{p}_3), d(\boldsymbol{p}_1,\boldsymbol{p}_4), d(\boldsymbol{p}_1,\boldsymbol{p}_7), d(\boldsymbol{p}_1,\boldsymbol{p}_8)) = d(\boldsymbol{p}_1,\boldsymbol{p}_4) = \sqrt{281}$ となる．この最大値は，付録A.1に示すアルゴリズムにおいてステップ(2.1)の(ii)を実行することによって得られる．同様に $R_{\text{R}} = \max(d(\boldsymbol{p}_2,\boldsymbol{p}_2), d(\boldsymbol{p}_2,\boldsymbol{p}_5), d(\boldsymbol{p}_2,\boldsymbol{p}_6), d(\boldsymbol{p}_2,\boldsymbol{p}_9)) = d(\boldsymbol{p}_2,\boldsymbol{p}_9) = \sqrt{173}$ となる．

生成されたK-M木を，次のように定理1を利用して部分木の探索を削減しながら探索することによって，被探索点 $\boldsymbol{x}$ の最近傍点がノードの個数以下の距離計算回数で得られる．まず，ルートノードを訪れる．$\boldsymbol{x}$ と左右の子ノードに格納されたデータとの距離を計算し，近い方のデータが格納された子ノードから先に深さ優先で探索する．ただし，探索しようとする子ノードが終端ノードであるか，あるいは左右の子ノードに対してそれぞれ

$$d(\boldsymbol{p}_{\text{L}}, \boldsymbol{x}) - R_{\text{L}} \geq d(\boldsymbol{x}, \boldsymbol{q}') \tag{4}$$

あるいは

$$d(\boldsymbol{p}_{\text{R}}, \boldsymbol{x}) - R_{\text{R}} \geq d(\boldsymbol{x}, \boldsymbol{q}') \tag{5}$$

が成立すれば，対応する子ノードの探索は行わないで終了とし，他方の未探索の子ノードを探索する．両方の子ノードの探索が済めば1つ上の親ノードに戻る．この探索アルゴリズムの全体を付録A.2（パラメータ $\alpha$ を1にとった場合）に示す．

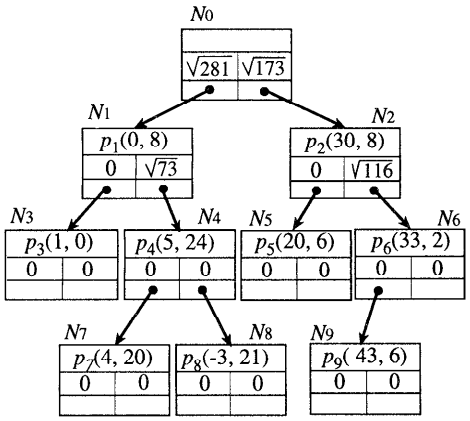



図2 K-M木の例
Fig. 2 An example of K-M tree.

例として，点 $(14,0)$ を被探索点 $\boldsymbol{x}$ として前述の9個のデータが格納された図2のK-M木を探索した過程を次に示す．ポインタ $ptr$ は最初にルートノード $N_0$ を訪れた後，順次 $N_1$, $N_0$, $N_2$, $N_6$, $N_2$ を訪れ，最後に $N_0$ に戻って，探索が終了する．$ptr$ が $N_1$ を訪れているとき，$N_1$ の右子ノード $N_4$ は終端ノードではないが，$N_1$ の右子ノード $N_4$ に対して式(5)，すなわち

$$d(\boldsymbol{p}_4, \boldsymbol{x}) - d(\boldsymbol{p}_4, \boldsymbol{p}_8) \geq d(\boldsymbol{x}, \boldsymbol{p}_3) \tag{6}$$

が探索打ち切りの条件として成り立つため，$ptr$ は $N_4$ を訪れずに親ノード $N_0$ に戻る．これによって，$\boldsymbol{p}_7$ および $\boldsymbol{p}_8$ との距離計算が省かれることになって，7回の距離計算で最近傍点 $\boldsymbol{p}_5$ が得られる．

**図3**に，次元数 $n$ とK-Mアルゴリズムにおける距離計算の回数との関係を調べた実験の一例を示す．疑似乱数関数を用いて $n$ 次元の10,000個のデータを2セット発生させた．$n$ 次元のデータの各要素は，区間 $[0,255]$ で均一に分散した整数で，要素間に相関はな

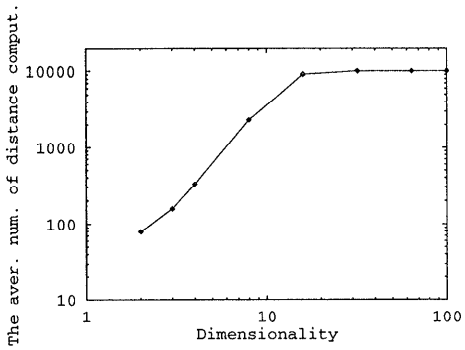



図3 次元数と距離計算回数の平均値との関係
Fig. 3 The relation between the dimension and the average number of distance computations.

い．一方のセットを K-M 木に格納し，他方のセットの各データを被探索点として K-M 木を探索したときの距離計算回数の平均値を示した．32 次元のときにはまったく距離計算回数の削減はなされなかった．図 3 は，次元数が大きくなってくると，定理 1 による探索範囲狭化の効果が低下してくることを示している．

K-M アルゴリズムは探索を行うために準備すべきデータ構造を高速に生成できるため，識別系を迅速に設計するという観点から有効であると考えられる．しかし，実際の文字認識で用いられる特徴量の次元数は数十～数百次元と高く，このような次元での K-M アルゴリズムの高速化が課題となる．これについて次章で考察し，高速化の方法を提案する．

## 3. 高速化手法の提案

本章では，探索を高速化するための手法について述べる．

### 3.1 探索範囲についての解析

$$\eta = (d(\boldsymbol{a},\boldsymbol{x}) - d(\boldsymbol{x},\boldsymbol{q}'))/R \tag{7}$$

とおくと，定理 1 の不等式 (1) は

$$\eta \geq 1 \tag{8}$$

と書き換えられる．このとき，定理 1 より集合 $\tilde{\boldsymbol{S}}$ の中に $\boldsymbol{q}'$ より $\boldsymbol{x}$ に近い点は 1 つも存在しない．

一方，

$$0 \leq \eta < 1 \tag{9}$$

のときは，集合 $\tilde{\boldsymbol{S}}$ の中に $\boldsymbol{q}'$ より $\boldsymbol{x}$ に近い点がどのくらい存在するのだろうか．

図 4 に，$n$ 次元空間内で点 $\boldsymbol{a},\boldsymbol{x},\boldsymbol{q}'$ が不等式 (9) を満たしているときの様子を示す．点 $\boldsymbol{a}$ を中心として半径が $R$ の球を $SP_a$，被探索点 $\boldsymbol{x}$ を中心として半径が $d(\boldsymbol{x},\boldsymbol{q}')$ の球を $SP_x$ で表す．集合 $\tilde{\boldsymbol{S}}$ に属する点は球 $SP_a$ の内部および球面上に存在する．そのうち，$\boldsymbol{q}'$ より $\boldsymbol{x}$ に近い点は，球 $SP_a$ と球 $SP_x$ が交差した領域にのみ存在する．この部分の体積 $V_c$ が球 $SP_a$ の体積に占める割合を次のようにして見積もる．

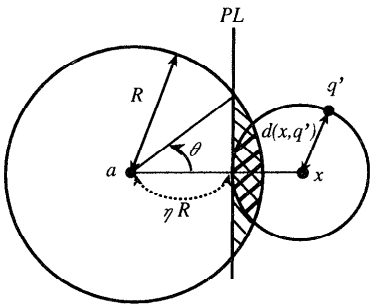



図 4 探索範囲についての解析
Fig. 4 Analysis on search area.

点 $\boldsymbol{a}$ と $\boldsymbol{x}$ を結ぶ直線に垂直で球 $SP_x$ に接する平面を $PL$ と表す．$V_c$ の代わりに球 $SP_a$ が平面 $PL$ で切り取られた部分の体積 $V_c'(\geq V_c)$ を計算する．いま，点 $\boldsymbol{a}$ と平面 $PL$ との距離は $\eta R$ である．球 $SP_a$ の体積 $V_a$ は

$$V_a = \frac{(\sqrt{\pi}R)^n}{\Gamma(n/2+1)} \tag{10}$$

で与えられる．また，

$$V_c' = \int_{\eta R}^{R} \frac{(\sqrt{\pi})^{n-1}}{\Gamma((n-1)/2+1)} \left(\sqrt{R^2-x^2}\right)^{n-1} dx$$
$$= \frac{(\sqrt{\pi})^{n-1}R^n}{\Gamma((n+1)/2)} \int_0^{\arcsin\sqrt{1-\eta^2}} \sin^n\theta d\theta \tag{11}$$

となる．したがって，部分集合 $\tilde{\boldsymbol{S}}$ の中で $\boldsymbol{q}'$ より $\boldsymbol{x}$ に近い点が存在する空間の体積が球 $SP_a$ に占める割合は

$$V_c'/V_a = \frac{\Gamma((n+2)/2)}{\sqrt{\pi}\Gamma((n+1)/2)} \int_0^{\arcsin\sqrt{1-\eta^2}} \sin^n\theta d\theta$$

より大きくないということがいえる．

図 5 に，次元数を 3，16，100 としたときの $V_c'/V_a$ の計算値を示した．$V_c'/V_a$ は $\eta$ に関して単調減少で，100 次元では $\eta = 0.5$ のとき $10^{-7}$ 程度である．このことから，集合 $\tilde{\boldsymbol{S}}$ の大きさや分布には依存するものの，$\eta$ が 1 に近い場合，$\boldsymbol{q}'$ より $\boldsymbol{x}$ に近い点が $\tilde{\boldsymbol{S}}$ の中に存在する可能性は次元が高いときわめて小さいということが予想される．

### 3.2 探索範囲のさらなる狭化

3.1 節の考察から予想されるように，不等式 (8) が成り立たない一部

$$\alpha \leq \eta < 1, \quad \text{ただし } 0 \leq \alpha < 1 \tag{12}$$

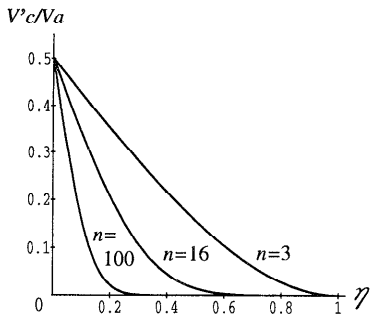



図5 分離された領域の体積比 $V_c'/V_a$
Fig. 5 Volume ratio $V_c'/V_a$ of the separated parts.

のケースに部分集合 $\tilde{S}$ の探索を打ち切っても，$\tilde{S}$ 内の $q'$ より近い点が見過ごされる可能性はきわめて小さいであろう．

そこで本研究では，K-M 木の探索において部分木の探索範囲を狭化する条件として，従来の不等式 (1) の代わりに，

$$d(a,x)-\alpha R \geq d(x,q') \quad (0 \leq \alpha < 1) \qquad (13)$$

を利用することにした．条件 (13) が成り立つのは，従来用いられていた条件 (1) が成り立つケースと不等式 (12) のケースの和となっている．それゆえ部分集合の探索の打ち切りは行われやすくなり高速化が期待できる．

この条件 (13) を利用して K-M 木を探索する例を示す．図 2 の 2 分木を $\alpha = 0.9$ に設定して探索すると，$ptr$ は $N_0$，$N_1$，$N_0$，$N_2$ と進む．$N_2$ の右子ノード $N_6$ に対して探索打ち切り条件が成り立ち，$ptr$ は親ノード $N_0$ へ戻って，探索は終了となる．この探索に要した距離計算回数は，$\alpha = 1$ のときの 7 回から 6 回に減少する．最近傍点は $N_5$ が正しく得られる．

しかし，$\alpha = 0.3$ に設定したときには，$ptr$ はルートノード $N_0$ の次に $N_1$ を訪れた後 $N_0$ へ戻り，未探索の右子ノード $N_2$ への訪問は打ち切られて，探索は終了する．要した距離計算回数は 4 回へと減少するが，近傍点として $p_3$ が得られ探索に誤りが生じる．

以上のように，$\alpha$ を小さく設定すると不等式 (13) は成立しやすくなって探索範囲が狭化されるので，K-M 木探索の距離計算回数が減少する反面，探索もれが生じやすくなる．

## 4. 文字認識実験

K-M 木を条件 (13) のもとで探索することによる高速化の効果と，そのとき発生する探索もれが識別能力へ及ぼす影響を調べるため，文字認識実験を行った．

### 4.1 使用したサンプル

第 1 回および第 2 回の郵政研究所文字認識技術コンテストにおいて学習用サンプルとして公開された手書き郵便番号のサンプル[19]を用いる．各データは実際の郵便はがきから収集された 3 桁の数字が記入されている横 240 × 縦 120 ドットの 2 値化画像である．はがきに印刷されている郵便番号枠はドロップアウトされているが，郵便番号枠の 4 つの頂点に対応する画像上の点の座標は固定値で与えられている．

実験には，この画像から次のように文字を 1 文字ずつ切り出して用いる．各 2 値化画像中の黒画素領域の輪郭線を 8 連結で追跡し，幅および高さが閾値以下の小さい連結成分をノイズとして除去する．そして，残った輪郭連結成分の中で，重心位置が，郵便番号枠の内部に対応する画像上の領域にあるものを集めて，1 つの数字と見なすことによって，切り出し処理を行う．この処理によって得られた輪郭成分がちょうど 1 つの文字を構成しているかを目視によって確認し，そうでない場合は実験データから除外する．公開されている 9,500 枚のデータのうち，画像の収集状態がきわめて不良な 500 枚のセット（data3）は本実験には使用せず，学習用として前半の 7,000 枚（data1, data2, data4）を，評価用として後半の 2,000 枚（data5）を使用する．

このほかに，電総研の文字データベース ETL-1 に収められている手書き数字，手書き英大文字（26 カテゴリー）および手書きカタカナ（47 カテゴリー）も用いる．ETL1 の場合，横 64× 縦 63 ドットの 16 階調の画像に，原則的には 1 個の文字が記入されているが，上述と同様にしてノイズ除去処理を施した後，抽出された文字が不良であればそれを除外する．学習用として前半に収められている 1,000 人分を，残りの 445 人分を評価用に用いる．

以上 4 種類の手書き文字のサンプルを実験に用いるが，以後郵政研究所提供の数字のサンプルを IPTP-N とよび，ETL-1 データベース中の数字，英大文字，カタカナのサンプルをそれぞれ ETL1-N，ETL1-A，ETL1-K とよぶ．実験に用いる各サンプルの文字数を**表 1** に示す．

### 4.2 特徴抽出法

本研究では，加重方向ヒストグラム特徴[12]を用いる．この特徴を最近傍識別と組み合わせることによって高い認識精度を達成できることが報告[20]されている．

次の要領で 100 次元の特徴量を抽出する．まず，8 連結の輪郭追跡によって得られる 8 方向のチェインコードから，向きの情報を取り除いた 4 方向の指数を，

**表1** 4つのサンプルにおける文字数と K-M 木の生成時間
Table 1 The number of characters and the time of K-M tree construction in four samples.

| サンプル | 学習用文字数 | 評価用文字数 | K-M 木の生成時間 |
|---|---|---|---|
| IPTP-N | 20735 | 5948 | 14.0s |
| ETL1-N | 9893 | 4383 | 5.7s |
| ETL1-A | 25757 | 10586 | 17.4s |
| ETL1-K | 45202 | 18725 | 32.4s |

各輪郭点に与える．1文字の外接矩形を小領域に分割し，各小領域に対する方向指数の頻度を計数する．そして，各方向指数ごとに，計数値を，対応する小領域に割り当てた方向指数特徴面をつくる．さらに，4枚の方向指数特徴面別に，縦・横5×5の格子点の各々で，重なりのある2次元ガウスフィルタによるぼかし処理を行う．このようにして得られた $4\times(5\times5)=100$ 次元の特徴ベクトルの各成分に対し，指数関数（指数を0.5とする）による変換を施す．

### 4.3 識別処理

学習用サンプルの各画像から抽出した特徴量に文字カテゴリーを表すラベルを付与した後，K-M 木に格納する．各学習用サンプルから得られる特徴量とラベルのデータ全部を K-M 木に格納する処理時間を表1に示す．実験に使用したプログラム言語はC言語であり，計算機は HP712（80 MHz）である．

IPTP-N の学習用サンプルから生成した K-M 木を用いて，IPTP-N の評価用サンプルの 5948 文字について $\alpha$ を 1.0, 0.95, $\cdots$ と 0.05 刻みに減少させつつ識別処理を行った．識別処理では，K-M 木を探索することによって得られる近傍点の持つカテゴリーを，識別結果として出力する．その他の3つのサンプルについても同様に識別処理を行った．**図6**，**図7** にそれぞれ（1文字あたりの）距離計算回数の平均値と正読率を示す．また，**図8** に距離誤差比 $d(\hat{q}-q^*)/d(x-q^*)$ の平均値を示す．ここに，$q^*$ は全数照合によって得られる最近傍点を表し，$\hat{q}$ は提案法によって得られる近傍点を表す．また $x$ は被探索点つまり評価サンプルの各データを表す．

4種類のどのサンプルについても，距離計算回数の平均値の対数は，ほぼ $\alpha$ に比例している．一方，$\alpha$ を1から0.6まで変化させても，正読率の低下は4サンプルとも 0.01%以下である．距離誤差比は，4サンプルとも $\alpha$ の減少とともに単調に増加している．

以上から，適切に $\alpha$ を設定しさえすれば，提案法によって，識別能力の低下を抑えつつ処理の高速化を達成できることが分かった．そのような適切なパラメータ $\alpha$ の一設定法を次節で示す．

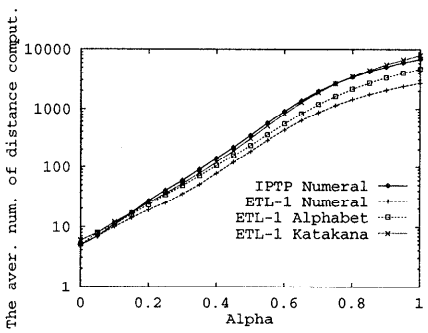



**図6** $\alpha$ と距離計算回数の平均値との関係
Fig. 6 The relation between $\alpha$ and the average number of distance computations.

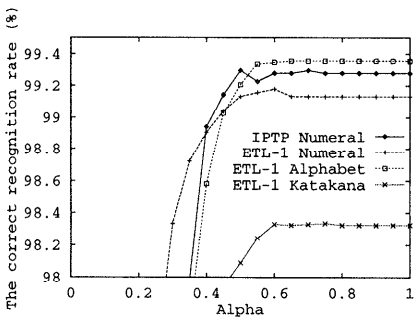



**図7** $\alpha$ と正読率との関係
Fig. 7 The relation between $\alpha$ and the correct recognition rate.

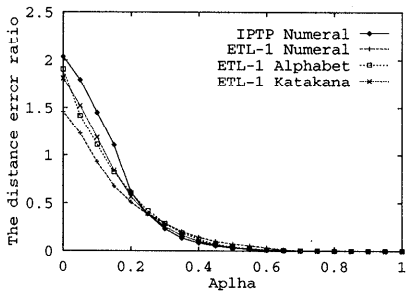



**図8** $\alpha$ と距離誤差比との関係
Fig. 8 The relation between $\alpha$ and the distance error ratio.

### 4.4 パラメータ $\alpha$ の設定法

正読率の低下を $\varepsilon$%以内と指定したとき，この範囲で距離計算回数が最小となる $\alpha$ の値 $\alpha^*$ をもって適切なパラメータ値と見なす．すなわち，パラメータ $\alpha$ に対する正読率および距離計算回数の平均値を，それぞれ $S(\alpha)$ および $T(\alpha)$ で表すと，

**表2** 識別処理時間の比較
Table 2 Comparison of the time required for search processing.

| 郵政研提供手書き数字（IPTP-N） | | | |
|---|---|---|---|
| | 提案法 | K-M 法 | 全数照合 |
| $\alpha$ | 0.5 | 1 | — |
| 距離計算回数の平均値 | 350 | 6989 | 20735 |
| 正読率 (%) | 99.29 | 99.28 | 99.28 |
| 識別処理時間の平均値 (ms) | **9.2** | 176 | 414 |

| ETL-1 手書き数字（ETL1-N） | | | |
|---|---|---|---|
| | 提案法 | K-M 法 | 全数照合 |
| $\alpha$ | 0.5 | 1 | — |
| 距離計算回数の平均値 | 185 | 2722 | 9893 |
| 正読率 (%) | 99.13 | 99.13 | 99.13 |
| 識別処理時間の平均値 (ms) | **4.7** | 69 | 197 |

| ETL-1 手書き英大文字（ETL1-A） | | | |
|---|---|---|---|
| | 提案法 | K-M 法 | 全数照合 |
| $\alpha$ | 0.55 | 1 | — |
| 距離計算回数の平均値 | 358 | 4747 | 25757 |
| 正読率 (%) | 99.34 | 99.36 | 99.36 |
| 識別処理時間の平均値 (ms) | **9.2** | 119 | 514 |

| ETL-1 手書きカタカナ（ETL1-K） | | | |
|---|---|---|---|
| | 提案法 | K-M 法 | 全数照合 |
| $\alpha$ | 0.6 | 1 | — |
| 距離計算回数の平均値 | 822 | 8041 | 45202 |
| 正読率 (%) | 98.33 | 98.32 | 98.32 |
| 識別処理時間の平均値 (ms) | **21.9** | 202 | 901 |

$$T(\alpha^*) = \min_{S(1)-S(\alpha)\leq\varepsilon} T(\alpha) \qquad (14)$$

である．

4.3 節の IPTP-N に関する実験結果を式 (14) にあてはめることによって，$\varepsilon = 0.05\%$ に対し $\alpha^* = 0.5$ が得られる．同様に，ETL-1，ETL1-A，ETL1-K に関する実験結果から，$\varepsilon = 0.05\%$に対し $\alpha^* = 0.5$，0.55，0.6 が得られる．

以上の要領で設計した $\alpha$ の最適値における距離計算回数の平均値，正読率および識別処理時間を**表2**に示す．比較のために，K-M 法と全数照合による結果もあわせて示す．4 種類のすべてのサンプルについて，K-M 法にくらべ提案法の識別処理は大幅に高速化されることが分かる．

### 4.5 考　　察

第 1 に，$\alpha$ を 1 から減少させてゆくと距離誤差比は単調に変化する（図 8 参照）にもかかわらず，図 7 および表 2 によって示されるように，$1 \geq \alpha \geq \alpha^*$ において正読率は $\alpha$ が 1 のときに計測される値のまわりをわずかに上下している．この現象は，探索もれによって，入力パターンのカテゴリーへの割当てがクラス間の境界付近で不安定になっているために生じたと思われる．

第 2 に，IPTP-N は，ボールペンや筆などのさまざまな筆記具で記入された実際の郵便番号から集めた手書き数字のサンプルであり，ETL1 は，鉛筆で実験用に記入された手書き文字のサンプルである．IPTP-N の学習用サンプルから生成した K-M 木を用い，$\alpha = 1$ において ETL1-N の評価用サンプルを認識したときの正読率は 97.08%となり，4.3 節の実験値 99.13%より 2.05 ポイント低下した．この結果は，読み取り対象に応じて識別系を再設計することの必要性を示唆している．K-M 木を利用した文字認識法では，表 1 に示すように，K-M 木への参照パターンの格納すなわち辞書の構成が高速に行える．したがって，読み取り対象に応じて迅速に識別系を設計できるという点から，提案法を利用した文字認識法は実用的に有効であるといえる．

第 3 に，本識別処理を実装するにあたって，$N$ 個の $n$ 次元の参照パターンを格納するための記憶容量と，K-M 木のデータ構造を記述するための $5(N+1)$ 個のアドレスおよび浮動小数点を格納する記憶容量が必要である．文献 15) でも三角不等式に基づいて照合回数を削減しているが，参照パターンのほかに少なくとも $(n+1)N$ 個の浮動小数点を格納するための記憶容量が必要であり，提案法より大幅に多い．IPTP-N の 20,735 個の手書き数字に対して本手法を実装するには，参照パターン用に 2.1M バイトと，K-M 木のデータ構造用に 415K バイトがあればよい．このようにカテゴリー数の少ない認識対象を扱うときには，現在の半導体技術では実用的に問題になることは少ない．一方，文字パターンにノイズが多く含まれていたり，用いる特徴が読み取り対象に適さず分離性が低くなっている場合には，参照パターン数の増大や探索もれによる識別能力の低下が問題となる恐れがある．また，漢字の認識となると最近傍識別法では記憶容量に関して現実的でなく，パラメトリックな識別手法の方が有効であろう．

## 5. ま と め

識別系を迅速に設計するという観点から K-M 木のデータ構造に着目した．高次元のデータに対して K-M 木の探索を高速化するため，三角不等式に基づいて探索範囲を狭化する条件に，パラメータ $\alpha$ を導入することを提案した．4 種類の手書き文字のサンプルに対して，提案した探索法を用いた文字認識実験を行った結果，$\alpha$ を適切に設定することによって最近傍識別と同様の高い認識率を保ちながら大幅な高速化が達成で

きることが示された.

本手法による探索時間や探索もれの頻度および識別性能を理論的に分析することは，残された課題ではあるが，実際のデータの分布に強く依存するので困難と思われる．むしろ，本来の最近傍識別の長所は，分布未知のパターンを取り扱い得る点，または，パターン分布があらかじめ仮定されたモデルから外れた場合にも高い識別能力が得られる点にこそある．現在筆者らは，提案法を用いた文字認識システムやパターン判別装置を様々な対象に適用することを計画しており，有効な知見が得られれば今後それらを報告したい．また，漢字のようなカテゴリー数の大きい認識対象にも本手法が適用できるように記憶容量を削減することも，今後の課題として取り組みたい．

## 参考文献


1) Sabourin, M., et al.: Classifier Combination for Hand-printed Digit Recognition, *2nd Int. Conf. on Document Analysis and Recognition*, pp.163–166, IEEE Computer Society Press (1993).
2) 丹羽寿男，山本浩司，小島良宏，〆木泰治，丸野 進，萱嶋一弘：パターンと記号の統合化処理による文字認識，信学論（D-II），Vol.J78-D-II, No.2, pp.263–271 (1995).
3) 田中明道，中村 修，北村 正：文字サイズ変動に適応する文字認識法，信学論（D-II），Vol.J76-D-II, No.12, pp.2547–2555 (1993).
4) 進藤宣博，阿曽弘具，木村正行：低品質印刷文字を高精度に識別する複合認識アルゴリズム，情報処理学会論文誌，Vol.35, No.9, pp.1714–1721 (1994).
5) 鳥脇純一郎：認識工学，コロナ社 (1993).
6) Fukunaga, K. and Narenda, P.M.: A Branch and Bound Algorithm for Computing $k$-Nearest Neighbours, *IEEE Trans. Comput.* Vol.C-24, pp.750–753 (1975).
7) Kalantari, I. and McDonald, G.: A Data Structure and Algorithm for the NearestPoint Problem, *IEEE Trans. Softw. Eng.*, Vol.SE-9, No.5, pp.631–634 (1983).
8) 湯浅哲也，有本 卓：K-M アルゴリズムのベクトル量子化への応用，信学論（A），Vol.J75-A, No.9, pp.1496–1502 (1992).
9) 岡 隆一：セル特徴を用いた手書き漢字の認識，信学論（D），Vol.J66-D, No.1, pp.17–24 (1983).
10) 萩田紀博，内藤誠一郎，増田 功：外郭方向寄与度特徴による手書き漢字の識別，信学論（D），Vol.J66-D, No.10, pp.1185–1192 (1983).
11) 萩田紀博，増田 功：手書き漢字認識のための方向寄与度特徴の次元圧縮，信学技報，Vol.PRL85-36, pp.13–22 (1985).
12) 若林哲史，鶴岡信治，木村文隆，三宅康二：特徴量の次元数増加による手書き数字認識の高精度化，信学論（D-II），Vol.J77-D-II, No.10, pp.2046–2053 (1994).
13) 孫 寧，安倍正人，根本義章：改良型方向線素特徴量および部分空間法を用いた高精度な手書き文字認識システム，信学技報，PRU94-31, pp.33–40 (1994).
14) Toussaint, G.T., Bhattacharya, B.K., and Poulsen, R.S.: The Application of Voronoi Diagrams to Nonparametric Decision Rules, *Computer Science and Statistics: The Interface*, Billard, L. (Ed.), pp.97–108, Elsevier Science Publishers (1985).
15) Ramasubramanian, V. and Paliwal, K.K.: An Efficient Approximation-elimination Algorithm for Fast Nearest-neighbour Search Based on a Spherical Distance Coordinate formulation, *Pattern Recogn. Letters*, Vol.13, No.7, pp.471–480 (1992).
16) Yan, H.: Handwritten Digit Recognition using an Optimized Nearest Neighor Classifier, *Pattern Recogn. Letters*, Vol.15, No.2, pp.207–211 (1994).
17) Sabourin, M. and Mitiche, A.: Modeling and Classification of Shape using a Kohonen Associative Memory with Selective Multiresolution, *Neural Networks*, Vol.6, No.2, pp.275–283 (1993).
18) 仙田修司，美濃導彦，池田克夫：高速な大規模マルチテンプレート手書き文字認識，信学技報，PRU95-116, pp.79–84 (1995).
19) 松井俊弘，山下郁生，若原 徹，吉室 誠：文字認識アルゴリズムの複合化手法の検討—第1回文字認識技術コンテストの結果より，信学技報，PRU92-33, pp.65–72 (1992).
20) 西川修史，若林哲史，木村文隆，三宅康二，堤田敏夫：手書き数字認識における特徴量の合成，信学技報，PRU95-114, pp.67–72 (1995).


## 付 録

### A.1 K-M 木の生成アルゴリズム

K-M 木の生成アルゴリズムを示す.

(1) ルートノードの $L_{\text{child}}$，$R_{\text{child}}$ をヌルにする.

(2) $\boldsymbol{p}=\boldsymbol{p}_1,\cdots,\boldsymbol{p}_N$ に対して順次ステップ (2.1)〜(2.3) を適用する.

(2.1) ルートノードから終端ノードまでたどる．(i) ポインタ $ptr$ をルートノードにセットする．(ii) $ptr$ が指すノード $N_{ptr}$ の左右の子ノードがともに存在する（$L_{\text{child}}$，$R_{\text{child}}$ がともにヌルでない）間，次の操作を行う．$\boldsymbol{p}_{\text{L}}$，$\boldsymbol{p}_{\text{R}}$ は左，右の子ノードに格納されたデータを

表すとする．もし $d(\boldsymbol{p},\boldsymbol{p}_{\rm L}) < d(\boldsymbol{p},\boldsymbol{p}_{\rm R})$ ならば，$R_{\rm L}$ を更新したのち $ptr$ を左の子ノード $L_{\rm child}$ に動かす．そうでなければ，$R_{\rm R}$ を更新したのち $ptr$ を右の子ノード $R_{\rm child}$ に動かす．更新は，$d(\boldsymbol{p},\boldsymbol{p}_{\rm L}) > R_{\rm L}$ または $d(\boldsymbol{p},\boldsymbol{p}_{\rm R}) > R_{\rm R}$ のときのみ，それぞれ $R_{\rm L} = d(\boldsymbol{p},\boldsymbol{p}_{\rm L})$ および $R_{\rm R} = d(\boldsymbol{p},\boldsymbol{p}_{\rm R})$ とする．(iii) ステップ (2.2) へ進む．

(2.2) ノード $N_{ptr}$ の左右とも子ノードがないとき，新しいノードを $L_{\rm child}$ として作り $\boldsymbol{p}_{\rm L}$ に $\boldsymbol{p}$ をセットする．$N_{ptr}$ の $R_{\rm L}$ と $R_{\rm R}$ は 0 にセットしておく．

(2.3) ノード $N_{ptr}$ が左の子ノードだけを持つとき，新しく右の子ノード $R_{\rm child}$ を作り $\boldsymbol{p}_{\rm R}$ に $\boldsymbol{p}$ をセットする． □

### A.2 K-M 木の探索アルゴリズム

K-M 木の探索アルゴリズムを示す．ただし，$\alpha = 1$ のときが Kalantari と McDonald の提案法に相当し，$0 \le \alpha < 1$ のときが筆者らの提案法に相当する．

(1) 被探索点 $\boldsymbol{x}$ を入力する．

(2) 最近傍点 $\boldsymbol{q}'$ を空に，最短距離 $NDIST$ を $\infty$ に初期化する．

(3) ポインタ $ptr$ をルートノードにセットして，手続き SEARCH($ptr$) を再帰的に呼び出す．

(4) $\boldsymbol{q}'$ を最近傍点（の候補）として出力する． □

**手続き SEARCH($ptr$)**

(i) もし $ptr$ がヌルならば，ステップ (vii) へ進む．

(ii) 距離 $d(\boldsymbol{x},\boldsymbol{p}_{\rm L})$ および $d(\boldsymbol{x},\boldsymbol{p}_{\rm R})$ を計算（ただし，$\boldsymbol{p}_{\rm R}$ が存在しないときは距離値を無限大にセット）し，それぞれスタック $DL$ および $DR$ に積む．以後，$DL$ および $DR$ は最後に積まれたスタックの値とする．

(iii) 最短距離 $NDIST$ に $\min(NDIST, DL, DR)$ をセットする．同時に最近傍点 $\boldsymbol{q}'$ も更新する．

(iv) もし $DL < DR$ ならば (v.a)～(v.b) に，そうでなければ (vi.a)～(vi.b) に進む．

(v) 左部分木を先に探索する：

(v.a) $DL - \alpha R_{\rm L} \ge NDIST$ ならば，(v.b) へ進む（左部分木探索の打ち切り）．そうでなければ，左部分木の探索を始めるために SEARCH($N_{\rm L}$) を呼び出す．SEARCH から戻った後は (v.b) へ進む．

(v.b) $DR - \alpha R_{\rm R} \ge NDIST$ ならば，(vii) へ進む（右部分木探索の打ち切り）．そうでなければ，右部分木の探索を始めるために SEARCH($N_{\rm R}$) を呼び出す．SEARCH から戻った後は (vii) へ進む．

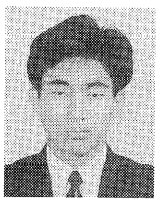
(vi) 右部分木を先に探索する：

(vi.a) $DR - \alpha R_{\rm R} \ge NDIST$ ならば，(vi.b) へ進む（右部分木探索の打ち切り）．そうでなければ，SEARCH($N_{\rm R}$) を呼び出す．SEARCH から戻った後は (vi.b) へ進む．

(vi.b) $DL - \alpha R_{\rm L} \ge NDIST$ ならば，(vii) へ進む（左部分木探索の打ち切り）．そうでなければ，SEARCH($N_{\rm L}$) を呼び出す．SEARCH から戻った後は (vii) へ進む．

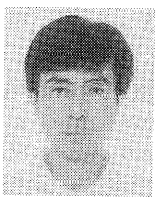
(vii) スタック $DL$ および $DR$ をポップし，$ptr$ を親ノードに戻して，SEARCH が呼び出された直後に復帰する． □

(平成 8 年 7 月 1 日受付)
(平成 9 年 2 月 5 日採録)



**亀山 博史**（正会員）

昭和 59 年大阪大学工学部電気工学科卒業．昭和 61 年同大大学院修士課程修了．同年グローリー工業（株）に入社．以来，文字認識の研究およびその応用システムの開発に従事．平成 2～3 年東京大学工学部計数工学科受託研究員．現在グローリー工業中央研究所技師．工博．IEEE 会員．



**鈴木 寿**

中央大学理工学部助教授．情報工学科知能情報制御研究室．半順序関係を考慮した 2 分木データ構造，ブール多値論理系などについて研究．専門分野は情報理論，機械知能，それらの学際領域．大阪大学大学院基礎工学研究科物理系専攻博士課程修了，工博．IEEE，情報理論とその応用学会，日本ロボット学会の会員．