

部首単位標準パターンとキューブサーチに基づく筆順フリーな オンライン文字認識アルゴリズム

蔡 文杰[†] 内田 誠一[†] 迫江 博昭[†]

An Efficient Stroke-Order-Free On-Line Character Recognition Algorithm
Based on Radical Reference Pattern

Wenjie CAI[†], Seiichi UCHIDA[†], and Hiroaki SAKOE[†]

あらまし グラフサーチにより最適画間対応を定めて筆順自由性を実現するオンライン文字認識法であるキューブサーチ法の動作速度と認識精度の改善を検討した。筆順変動を部首内の変動と部首間の変動に分離して、部首単位標準パターンに基づく2段階のキューブサーチアルゴリズムを構成した。併せて、処理量最小化条件を含む部首単位分割の指針を示した。教育漢字を対象とする画数固定条件での認識実験により、速度、精度両面での改善が確認され、併せて、処理量最小化部首分割条件の妥当性が確認された。

キーワード オンライン文字認識, 筆順フリー, 部首単位, キューブサーチ, 高速化

1. ま え が き

ペンの動きを時系列として取り込むオンライン文字認識では、漢字のような多画文字の場合、画間の筆順が問題になる。これに対して筆順変動を許した文字認識手法が検討されてきた。当初は文字の画数 N に関して $O(N!)$ の計算量が必要と考えられたが、各種数理計画法の導入等の工夫により、高速化の成果が得られている [1]~[4]。

迫江らの提案したキューブサーチ法 [1], [2] は入力パターンと標準パターンの間の画の対応を定義する全単写を N 次元キューブグラフ上の最適経路問題として定式化し、DP (Dynamic Programming) により探索することによって $O(2^N)$ のアルゴリズムとなっている。しかし、漢字は画数が多く、しかも多字種であることから、よりいっそうの高速化が望まれている。

本論文は漢字が部首構造^(注1)を有する点と、キューブサーチがグラフサーチである点とに着目して高速な認識アルゴリズムの構築を試みたものである。周知のように漢字は部首の複合体として構成されており、実

際の筆記の視察によると、部首の途中から別の部首に遷移するという筆順は極めてまれである。また同じ部首が多くて文字に共通して現れる場合が少なくない。これらのことを利用して、キューブサーチに次のような改良を行う。

(1) 筆順変動を部首内の筆順変動と部首単位での筆順変動に分離することにより、部首の内部から他の部首に遷移する筆順変動を事前に排除して探索空間を圧縮する。

(2) 部首に関する処理結果(相異度)を、部首を共有する文字間で共通利用する。

この(1)によっては不自然な筆順でのマッチングを排除できることから、また(2)によっては文字標準パターン間の部首変形の影響を排除できることから認識精度の向上も期待できる。

これらを実現するため、処理は画単位での筆順自由条件で部首単位の構造解析を行う部首レベル処理とそれらの結果をもとに部首単位での筆順自由条件で文字の構造解析をする文字レベル処理よりなる。このアルゴリズムは連続音声認識における2段DPマッチング法 [5] と同じ思想に立ったものである。部首を単位と

[†]九州大学大学院システム情報科学府, 福岡市
Graduate School of Information Science and Electrical Engineering, Kyushu University, 6-10-1 Hakozaeki, Higashi-ku, Fukuoka-shi, 812-8581 Japan

(注1): ここでの“部首”は必ずしも辞書の意味での部首を意味しない。高速化あるいは高精度化の目的をもって定義される画の複合体を含むものとする。

する漢字認識は種々試みられているが [6] ~ [8], 最適な画対応を探索するという枠組みで行われたものは見当たらない.

なお, 本論文では画数固定の枠組みで検討した. この成果をもとにした画数フリー化の拡張は次の段階と考えている. 以下, 2. では本研究の出発点となるキューブサーチの概要を述べる. 3. では部首単位処理の考え方と, 具体的アルゴリズムを説明する. 4. では部首セットの設計方針に関して検討する. 5. では実験により従来のキューブサーチ法に対する優位性を示す.

2. 基本的なキューブサーチの原理 [1], [2]

2.1 問題の設定と準備

N 画のオンライン入力文字を次のように筆記順の画の系列として表現する.

$$A = A_1 A_2 \dots A_k \dots A_N \quad (1)$$

ここに A_k は第 k 画で, 文字の局所的特徴 a_{ik} (例えば筆線方向, xy 座標等) の時系列表現である.

$$A_k = a_{1k} a_{2k} \dots a_{ik} \dots a_{lk}$$

同様に標準パターンを次のように表現する.

$$B = B_1 B_2 \dots B_l \dots B_N \quad (2)$$

$$B_l = b_{1l} b_{2l} \dots b_{jl} \dots b_{Jl}$$

ここでパターン長 I, J は正確には $I(k), J(l)$ と表記すべきであるが, 本論文では単に I, J と略記する.

画 A_k と B_l の間の相異度 (例えばチェーンコードの DP マッチング距離) を

$$\delta(k, l) = D(A_k, B_l) \quad (3)$$

と示し, 画間相異度と呼ぶ. すべての (k, l) の組に関して画間相異度を求めることとし, これらをテーブルに記憶する. 入力第 k 画に対応づけられる標準パターンの画を $l(k)$ とする. 最適な画対応 $l(k)$ の評価基準として, 画間相異度 $\delta(k, l(k))$ の総和を最小とする. すなわち, 次の最小化問題として定式化される.

$$D(A, B) = \min_{\{l(k)\}} \left[\sum_{k=1}^N \delta(k, l(k)) \right] \quad (4)$$

画対応 $l(k)$ は, すべての画をもれなく一意に対応づける必要があるため, 全単射であるとする.

2.2 キューブサーチアルゴリズム

全単射条件を保証するため, 式 (4) を図 1 のようなコスト付きグラフの最適経路問題として計算する. このグラフはキューブ状で各ノード (状態) は, そこに至るまでに既対応の画で 1, 未対応の画で 0 となるフラグ (ビット) の組となっている. 最下位ビットから

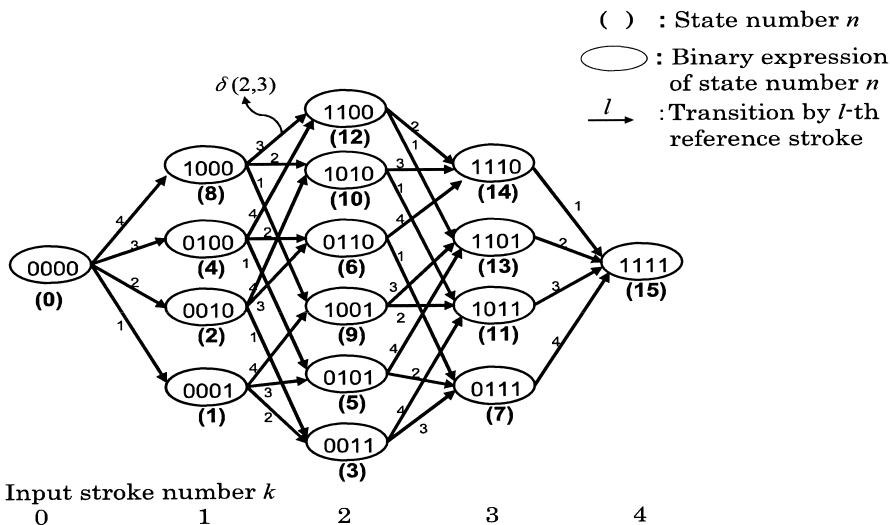


図 1 キューブサーチグラフ ($N = 4$)
Fig. 1 Cube search graph ($N = 4$).

$1, 2, \dots, N$ と番号づけられるビット位置は標準パターンの $1, 2, \dots, N$ 画に対応する. $k-1$ 段から k 段への遷移で第 l ビットが $0 \rightarrow 1$ と反転するエッジのコストとして画間相異度 $\delta(k, l)$ が付随する. 画対応決定問題はこのキューブ状グラフで初期ノード $0 = (00 \dots 0)$ から最終ノード $2^N - 1 = (11 \dots 1)$ に至る最適経路の探索問題になる. これにより生成される経路では, $(11 \dots 1)$ に到達することにより全部の画に対する画が行われる. また, 一度 1 となったフラグに対応する画が再び現れることはない. すなわち, 全単射性が保証される画対応をすべて生成できるのである.

上記最適経路問題は, 次のような DP によって計算することができる.

$$G(n) = \min_m [G(m) + \delta(k, l)] \quad (5)$$

ここに m は第 $k-1$ 段のノード, n は第 k 段のノードである. 具体的な計算は次のような前向き計算によっている.

(1) 画順サーチ処理の初期化

$$G(n) = \infty, n = 1, 2, \dots, 2^N - 1.$$

$$G(0) = 0.$$

(2) 画順サーチ処理 ($k = 1, 2, \dots, N$)

(2-1) 第 $k-1$ 段の各ノード n を生成しながら

(2-2) ~ (2-4) を実行.

(2-2) n のビットのうち 0 であるものを 1 個を 1 に変え $\rightarrow nw$.

そのビットの位置を $\rightarrow l$.

(2-3) $\min[G(n) + \delta(k, l), G(nw)] \rightarrow G(nw)$.

(2-4) 1 を立てるビットの位置を変えながら

(2-2) 以下を繰り返す.

(3) 終結

入力 A と標準 B のマッチング結果として文字間相異度

$$D(A, B) = G(2^N - 1)$$

を得る.

以上のアルゴリズムでは, ワークエリア記憶量はノード数 $O(2^N)$ であり, 計算量はエッジ数で $O(N \cdot 2^{N-1})$ となる. 以上によって, 画対応に全単射性を満たしつつ, 基本単位である画を標準パターンへ対応づけるという構造解析処理を実現するのが, キューブサーチ法である.

2.3 ビームサーチ DP

画数 N が 20 を超える漢字が少なくないので, 上記の計算量は小さいとはいえない. これに対して迫江,

慎 [1], [2] は DP 計算 (2-1) ~ (2-4) に枝刈りを組み込んだビームサーチ DP による高速化を試みている. 実験により一応の実用的な実行速度を確認しているが, 経済性を考えるとといった高速化が望まれる. また, DP ワークエリア $G(n)$ の記憶量 $O(2^N)$ はビームサーチを導入しても低減されず, 問題を残している. そもそも枝刈りはある程度最適パスを見失ってしまう危険性を覚悟して利用されるもので, 一定個数のノードを残すという基準で行うと, 原探索空間が大きいほど, 危険性も増す. $N = 20$ ともなるとキューブグラフの $k = 10$ でのノード数は 1.8×10^5 のオーダーになる. 枝刈りの高速化効果と危険度抑圧の両立は困難であり, 原サーチ空間の圧縮が望まれる.

2.4 キューブサーチの利点

グラフサーチとして定式化され実行されるキューブサーチには, 同じくグラフサーチで構築された音声認識技術との連想から, 次のような利点があるといえる.

(1) 単語標準パターンを用いて文音声を認識する連続単語列認識技術と同様に, 文字標準パターンを用いた枠なし文字列認識への拡張の可能性.

(2) サブワード標準パターン (例えば CVC 連鎖) を用いた単語音声認識と同様に, 部首単位標準パターンに基づく文字認識の可能性.

(3) 確率的モデル (HMM) の導入による高精度化.

(4) これらに関係して音声認識で用いられる高速サーチ手法の利用 (2.3 のビームサーチ DP はこの一環である).

本論文で検討するのは (2) の方向である.

3. キューブサーチの部首単位分解による高速化

2.3 で述べたように画を基本単位として一気に文字単位の構造解析を実行するキューブサーチでは大画数の場合, 計算量, 記憶量の双方に関して問題を残していた. そこでキューブサーチ処理を分解して, 画を基本単位として部首単位の構造解析を実行する部首レベルのキューブサーチと, 部首を基本単位として文字単位の構造解析を実行する文字レベルキューブサーチに分解することを考える.

例を $N = 10$ 画の文字 “桜” にとって説明する. 記憶量 (ノード数) は $2^N = 1024$, 計算量は $10 \cdot 2^9 = 5120$ である. この文字を 4 画の部首 “木”, 3 画の部首 “ツ”, 3 画の部首 “女” に分解する (図 2). 各部首レベルでは基本単位である画の対応を探索するためのキューブ

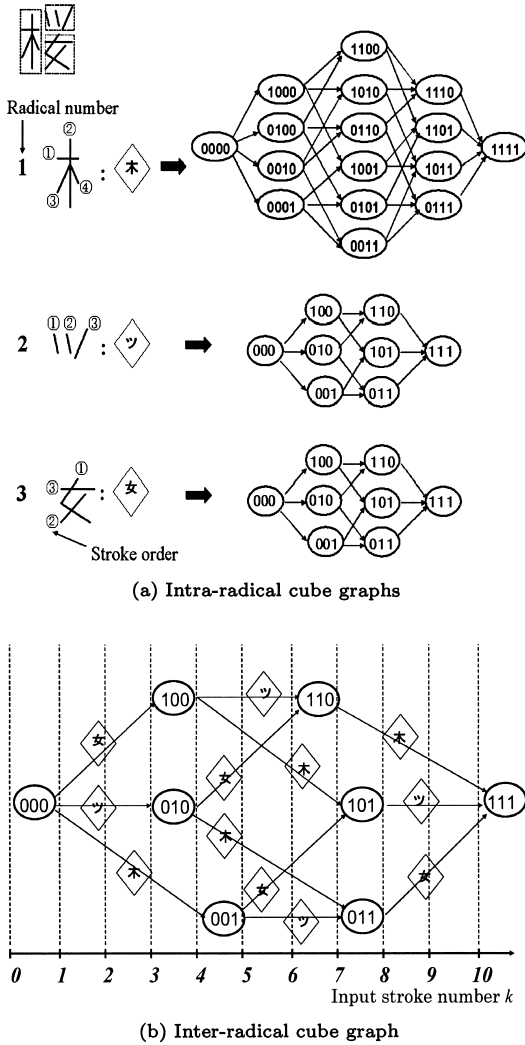


図2 部首に基づくキューブサーチグラフ(パターン“桜”, $N = 10, m = 3, n_1 = 4, n_2 = n_3 = 3$)
 Fig. 2 Radical-based cube search graph (pattern“桜”, $N = 10, m = 3, n_1 = 4, n_2 = n_3 = 3$).

サーチが行われる(同図(a)). それらのノード数は16, 8, 8, エッジ数は32, 12, 12となる. これらの上位に部首を基本単位として部首間の対応を探索するキューブサーチが行われる(同図(b)). このグラフにおいてはノード数は8, エッジ数は12である. この文字レベルのキューブサーチにおける各エッジにいずれかの部首が均等に対応するので, 全体としてのノード数は $12 \times (16 + 8 + 8)/3 = 128$, エッジ数は $12 \times (32 + 12 + 12)/3 = 224$ となり, もとのノード数1024, エッジ数5120に比べて大幅に圧縮される.

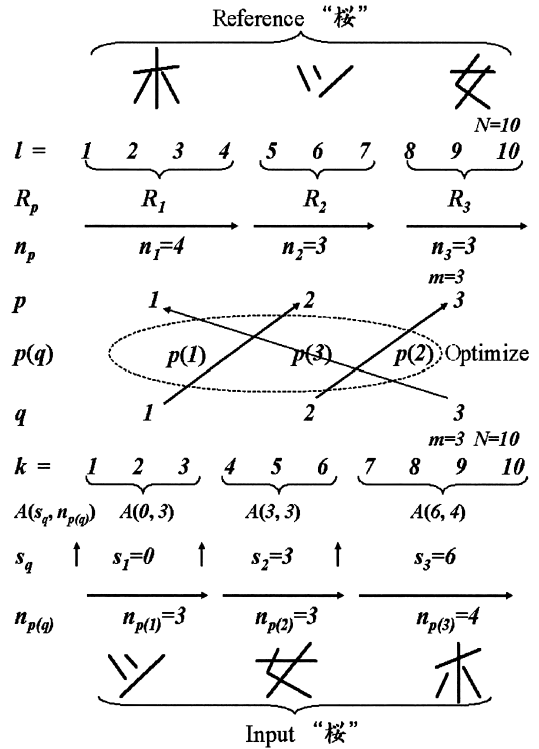


図3 部首単位キューブサーチの原理
 Fig. 3 Principle of radical based cube search.

なお, これらの効果は「部首間にわたる筆順変動がない」ことを前提としている. すなわち部首筆記が未完のままで別の部首筆記に移ることはないとしたものである. これは本論文での前提となっている. 実際, 部首間にわたる筆順変動はまれである. 比較的多発して無視できないそのような筆順変動に対しては4.で現実的な対策を検討する.

本章では, このような部首分解によるキューブサーチの高速化について論じる. 2段DPマッチング[5]の考え方にに基づき, 処理は部首レベル処理と文字レベル処理よりなる. 図2と図3を参照しながら部首単位キューブサーチの考え方を説明する.

3.1 標準パターンの部首単位分割

標準パターン B を次のように筆記順の m 個の部首の系列として再定義する.

$$B = R_1 R_2 \dots R_p \dots R_m \tag{6}$$

ここに R_p は p 番目の部首を意味する. n_p を部首 R_p の画数, N を全体の画数とすると,

$$\sum_{p=1}^m n_p = N \quad (7)$$

となる．図 2，図 3 の例では， $N = 10$ ， $m = 3$ ， $n_1 = 4$ ， $n_2 = 3$ ， $n_3 = 3$ である．

3.2 部首レベル処理

入力パターンの第 $s+1$ 画から始まる長さ n_p 画の系列を部分パターン $A(s, n_p)$ と呼ぶ．

$$A(s, n_p) = A_{s+1}A_{s+2}\dots A_{s+n_p} \quad (8)$$

この部分パターンと部首標準パターン R_p の間で画を単位としたキューブサーチを行い部首間相異度

$$\lambda(s, p) = D(A(s, n_p), R_p) \quad (9)$$

を得る．これは入力パターン中に部首と仮定される部分内で標準パターン画との間で最適画対応を行って評価した局所的な相異度（画間相異度和）である．この計算は部分パターンのすべての始点 s ，及びすべての部首パターン R_p に対して行う必要はない．図 2 の例でいえば，ノードが存在する $k = s$ で，そのノードに後続可能な部首パターン R_p に対して行えばよい．例えば，同図 (b) の $k = 3$ におけるノード (010) に後続する部首パターンは“女”と“木”しか許されない．よって，“ツ”はこのノードにおける部首レベル処理から省略できる．

3.3 文字レベル処理

図 2 (b) の探索問題を解く．言葉で記述すると，入力画の系列 $1, 2, \dots, k, \dots, N$ を連続した m 個の区間 $1, 2, \dots, q, \dots, m$ に分割する．第 q 区間を標準パターンの第 p 部首パターンに対応づける関数を $p = p(q)$ とする．この対応づけは全単射条件を満たすものとする．このとき，次の最小化問題を解いて文字間相異度 $D(A, B)$ を得る．

$$D(A, B) = \min_{p=p(q)} \left[\sum_{q=1}^m \lambda(s, p) \right] \quad (10)$$

制約条件は $p = p(q)$ が全単射で，区間長の和が

$$\sum_{q=1}^m n_{p(q)} = N \quad (11)$$

を満たすことである．この 2 種の制約は図 2 (b) のグラフの拘束を言い換えたものである．

3.4 効果

以上の原理によると，次のような効果が得られると期待される．実行速度に関しては，

(1) 3. の冒頭に図 2 の例で説明したように，エッジ数の大幅低減による高速化が期待できる．この効果は部首レベル処理量の低減，及びそれに含まれる，画間相異度計算回数の低減の二重の効果がある．

(2) 部首が複数の文字で共通に現れることが少なくない．一度計算した部首間相異度 $\lambda(s, p)$ をテーブルに保存しておき参照することにより，部首間相異度計算を効率化できる．

ワークエリア記憶量に関しては

(3) N 画の場合，元来 $O(2^N)$ の記憶量が必要であった．部首分解をすると部首レベルのキューブサーチで $O(2^{\max(n_p)})$ のメモリ 1 組と，文字レベルのための $O(2^m)$ のメモリ 1 組とがあればよい (m は文字中の部首数)．これらは文字ごとに異なるが，全文字の標準パターンを設計し終わった時点では，全標準パターンの中に出現する最大部首画数と，最大部首数とから確定することができる．

認識性能に関しては (2) の効果として

(4) 部首間にわたる筆順変動の排除による過整合の抑制効果と，文字間での部首形状のばらつき除去とにより，認識精度の向上が期待される．

以下，理論的に予測されるこれらの効果を実現するための標準パターン設計指針を 4. に，実行アルゴリズムを 5. に述べる．

4. 標準パターンの設計指針

まず，計算量（エッジ数最小）の観点から最適な部首分割を考察する． N 画の文字を n_1 画， n_2 画， \dots ， n_m 画の m 個の部首に分解する．部首レベルと文字レベルのキューブグラフで必要とする総エッジ数は

$$M = 2^{m-1} \cdot (n_1 \cdot 2^{n_1-1} + n_2 \cdot 2^{n_2-1} + \dots + n_m \cdot 2^{n_m-1})$$

これを式 (7) の制約のもとに最小化すると

$$m = n_1 = n_2 = \dots = n_m = \sqrt{N} \quad (12)$$

が得られる． \sqrt{N} を近似する整数を N_R とするとき，部首数，各部首の画数ともに N_R に近くとればよいことになる．その場合のエッジ数の近似値は $N \cdot 2^{N_R-1}$ となる．

これを参考にして次のような方針で部首単位標準パ

ターンを設計する．

(1) 辞書の意味での部首を基本とする．

(2) 部首の途中で他の部首に遷移するという部首間の筆順変動が多発する文字では、その前後に分解してそれぞれ新たな部首とする．

(3) 多画の部首は分割して式 (12) の条件に近づけるようにする．

実際には、以上の方針で部首単位標準パターンを設計後、予備実験による試行錯誤が必要であろう．部首分解が確定すると、対応する文字レベルキューブグラフは 2.2 のルールによって機械的に定まる．部首分解と、対応する文字レベルキューブグラフの例を付録に示す．

5. 実行アルゴリズム

部首標準パターンの共通化 (3.4 の効果 (2)) を組み込んだ実行アルゴリズムを図 4 に示す．部首標準パターンは文字間の重複を整理して全文字共通のテーブル $S(r)$ に、それらの画数はテーブル $L(r)$ に、それぞれ用意される．文字標準パターン (式 (6)) の $r = R_p$ はこれを参照するポインタとなっている．式 (9) の部首レベル処理は部首標準パターン $S(r)$ と入力パターン上に仮定される部首区間 $A(s, L(r))$ との部首間相異度を計算する関数 $\Lambda(s, r)$ で示されている．部首標準パターン共通化による高速化を実現するため、テーブル $dist(s, r)$ を用いる．このテーブルは入力パターンが与えられ認識処理が起動される時点ですべて十分大きな値 ∞ でリセットされる．以後各文字の標準パターンとの間でマッチング処理が繰り返される過程で、部首レベルのキューブサーチで部首間相異度 $d = \Lambda(s, r)$ が計算されたとき、 $dist(s, r)$ ($r = R_p$) として保存する (10)．以後、別の標準パターンとのマッチングで同一の s と r の組合せが発生したときは $d = dist(s, r)$ ($r = R_p$) と参照することにより再度の部首レベルキューブサーチを省略する (11)．初期設定した ∞ は $dist(s, r)$ が未計算であることを示すマークになっている．

ビームサーチの組込み、及び②におけるビットパターンの生成法に関しては、文献 [2] の図 2 を参照されたい．

6. 実 験

6.1 概 要

旧教育漢字 882 種を認識対象とした．30 名の筆記

[Radical reference tables]

$S(r)$: common radical reference

$L(r)$: number of strokes in $S(r)$

B : reference pattern of Eq.(6) ($r=R_p$ is a pointer to $S(r)$ and $L(r)$)

[Other variables]

$\Lambda(s, r)$: a function which returns radical distance between $A(s, L(r))$ and $S(r)$

t, v : m bit digit for identifying node

$G(t)$: DP work area

$u(t)$: input stroke position $s=k$ of node t

$dist(s, r)$: common radical reference table(reset by ∞ in advance of starting recognition)

[Cube search]

[Initialization]

$G(v) \leftarrow \infty, v = 1, 2, \dots, 2^m - 1$

$G(0) \leftarrow 0$

$u(0) \leftarrow 0$

[Radical order search by DP]

① for $q = 1$ to m do

② for all t with $(q-1)$ bits are '1' do

③ $s \leftarrow u(t)$

④ for $p = 1$ to m do

⑤ if p -th bit of t is '0' then

⑥ $v \leftarrow$ change p -th bit of t to '1'

⑦ $r \leftarrow R_p$

⑧ if $dist(s, r) = \infty$ then

⑨ $d \leftarrow \Lambda(s, r)$

⑩ $dist(s, r) \leftarrow d$

⑪ else $d \leftarrow dist(s, r)$

⑫ $G(v) \leftarrow \min [G(v), G(t) + d]$

⑬ $u(v) \leftarrow u(t) + L(r)$

⑭ end

⑮ end

⑯ end

[Termination]

⑰ $D(A, B) \leftarrow G(2^m - 1)$

図 4 部首単位キューブサーチアルゴリズム

Fig. 4 Radical-based cube search algorithm.

データから画数変動のない計 17983 データを選択して実験対象とした．前処理として位置・大きさの正規化とスムージングを行った後、 xy 座標情報と筆点方向情報 (8 方向量子化) をもった特徴の時系列とした．

部首標準パターンとしては 4. の (1)(2)(3) の

表 1 実験結果
Table 1 Experimental results.

Algorithm	Stroke base	Radical base
Recog. rate (%)	99.15	99.31
Average recog. time (ms)	82	40

表 2 認識時間の内訳 (ms)
Table 2 Recognition time details (ms).

Algorithm	Stroke base	Radical base
Calculation time of stroke distances	39	30
Cube search time	43	10
Total (Recog. time)	82	40

方針によって総計 1009 個を作成し、予備実験によって補正した。5 画前後のものが多数を占め、最大のものは 9 画であった (例えば、部首“重”)。これらによって各文字の標準パターンを 1~4 個の部首に分解した。

実験は部首分割による高速化効果と認識率の動向に関して行った。認識は画数固定の条件で行った。すなわち、入力パターンの画数を用いて予備選択を行って同一の画数の標準パターンとのみマッチングを行い、文字間相異度最小の基準で強制判定を行った。高精度認識を前提として、画間距離としては実績のある DP マッチング距離 [1], [2] を用いた。実験に用いたのは DELL Precision WorkStation 530 (1.7 GHz) である。ビームサーチ条件 [1], [2] は予備実験により最適点を定めた。

6.2 実験及び結果

上記の 17983 データに関して平均認識率と平均認識時間を従来の画単位キューブサーチと本提案の部首単位キューブサーチの双方に対して測定した。結果を表 1 に示す (ビームサーチ利用)。これにより認識率、処理速度の双方での改善が確認された。

なお、従来法で誤認識され、本手法で改善されたのは 78 例、逆に従来法で正しく認識され、本手法で誤認識となったのは 50 例であった。

表 2 には認識に必要な処理時間の内訳を示している (ビームサーチ利用)。キューブサーチに要する時間は部首分割によって大幅に短縮されている。この結果画間相異度の計算時間が支配的となっており、全体としては 2 倍の高速化にとどまっている。この上は、画間相異度の定義 (現行は DP マッチング距離) を簡素化することによって、全体として高速化を図る必要があるといえる。

表 3 入力“興”の認識に必要なとされたキューブサーチ時間 (ms)

Table 3 Cube search time for recognizing an input “興” (ms).

Beam search	without	with
Stroke base	3430	195
2 radicals	68	22
4 radicals	26	16

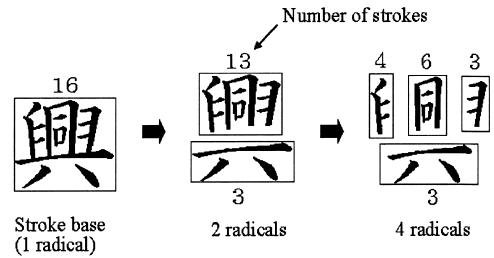


図 5 パターン“興”の 3 種の部首分解
Fig.5 Three ways of radical decomposition of pattern “興”.

計算量の見地から 4. で述べた部首分解条件 (式 (12)) の妥当性を示す実験結果を表 3 に示した。16 画の文字“興” ($N_R = 4$) を図 5 のように 3 種の部首分解をした場合のそれぞれについてキューブサーチ時間を測定したものである。 $N_R = 4$ より定まる、4 部首分割、各部首 4 画という最適条件に近い分割で最高速度が得られている。これにより、部首分解方針 (3) の妥当性が確認された。

3.4 (2) の効果を見るため、部首間相異度 $\lambda(s, p)$ の保存・再利用を除外して認識時間を測定したところ 46 ms であった。これは表 1 の 40 ms に対応するもので、結果として (2) の効果はあるものの、大きいとはいえない。これは同一部首パターンが多数回出現してもその位置 s が異なり、再利用できる回数がそれほど多くないためと考えられる。

キューブサーチのワークエリア記憶量を、3.4 の (3) の解析に基づいて、実験に用いた標準パターンに関して測定した。文字単位の場合最大画数は $N = 20$ で、ワークエリアサイズは $O(2^{20}) = 10^6$ であった。部首単位の場合、最大部首画数は $n_p = 9$ (“重”), 最大部首数は $m = 4$ で、ワークエリアサイズは $O(2^9) + O(2^4) = 528$ であった。記憶量低減効果は極めて大きい。

7. むすび

筆順自由なオンライン文字認識アルゴリズムである

キューブサーチ法に、部首単位処理の過程を組み込んで2段階化し、処理速度と認識精度の向上を図った。実験では処理時間、認識率ともに改善が確認された。特にキューブサーチに必要とされる処理時間は大幅に改善された。その結果、キューブサーチ以前の過程として必要な画間距離計算の処理時間が全体の認識時間に対して支配的となった。現段階の検討では高水準の認識精度を重視し、画間距離としては実績のある座標情報と方向情報の時系列に関するDPマッチング距離を用いている。今後、認識速度に重点を置く場合には、認識精度に一定程度の妥協を持ち込んでの画間距離定義の見直しが必要であろう。

部首単位標準パターン設計指針を検討し、計算量の見地からの最適分割条件を示したが、実験により妥当性が確認された。また、キューブサーチ記憶量の大幅な低減も確認された。

標準パターンの実験的改良によって更なる認識精度改善と高速化の余地が残されている。他方、最適な部首分解指針の確立と自動化は、残された興味ある課題といえる。また、[2]の考えに基づいた画数自由化への拡張が必要と考えている。

謝辞 本研究の一部は総務省戦略的情報通信研究開発推進制度の支援を受けた。

文 献

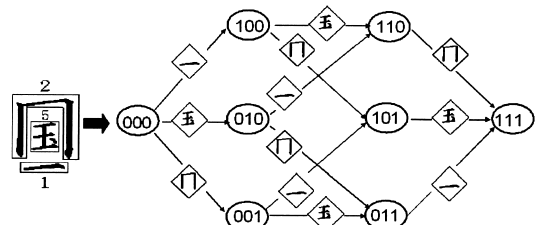
- [1] 迫江博昭, 慎 重弼, “筆順フリーなオンライン文字認識のための画対応サーチアルゴリズム”九州大学大学院システム情報科学研究科報告, vol.2, no.1, pp.99-104, March 1997.
- [2] 慎 重弼, 迫江博昭, “筆順・画数自由オンライン文字認識のための画対応決定法—多層キューブサーチ”信学論(D-II), vol.J82-D-II, no.2, pp.230-239, Feb. 1999.
- [3] T. Wakahara, A. Suzuki, N. Nakajima, S. Miyahara, and K. Odaka, “Stroke-number and stroke-order free on-line Kanji character recognition as one-to-one stroke correspondence problem,” IEICE Trans. Inf. & Syst., vol.E79-D, no.5, pp.529-534, May 1996.
- [4] 横田登志美, 葛貫壮四郎, 郡司圭子, 桂 晃洋, 浜田長晴, 福永 泰, “筆順画数同時フリーを実現する楔文字モデルによるオンライン日本文字認識方式”情処学論, vol.44, no.3, pp.980-990, March 2003.
- [5] H. Sakoe, “Two-level DP-matching—A dynamic programming-based pattern matching algorithm for connected word recognition,” IEEE Trans. Acoust. Speech Signal Process., vol.ASSP-27, no.6, pp.588-595, Dec. 1979.
- [6] 未代誠仁, 中川正樹, “オンライン手書き日本語文字認識における構造化字体表現に対するプロトタイプ学習”信学論(D-II), vol.J86-D-II, no.1, pp.1-11, Jan. 2003.
- [7] C.L. Liu, S. Jaeger, and M. Nakagawa, “Online recog-

inition of Chinese characters: The state-of-the-art,” IEEE Trans. Pattern Anal. Mach. Intell., vol.26, no.2, pp.198-213, Feb. 2004.

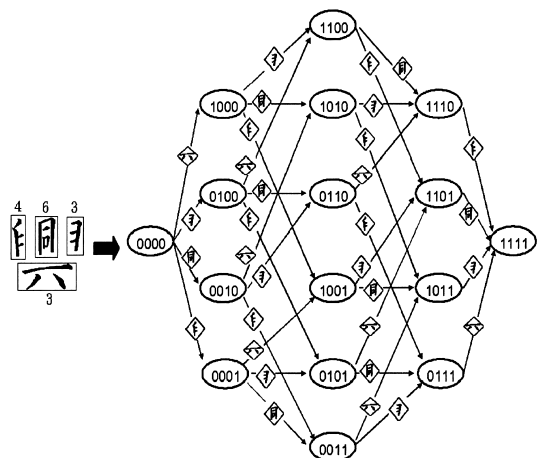
- [8] X.H. Xiao and R.W. Dai, “On-line handwritten Chinese character recognition directed by components with dynamic templates,” Int’l J. Pattern Recognition and Artificial Intelligence, vol.12, no.1, pp.143-157, 1998.

付 録

標準パターンの部首分解と、対応する文字レベルキューブグラフの例を示す。いずれも4.の部首分解方針(1)~(3)を考慮したものである。(a)“国”の例では構え“口”の最終画“一”を部首“玉”の筆記の後で書く場合と“玉”に先立って各場合の両方に対処できるようにになっている。(b)“興”の例は図5の4部首分解に対応する。キューブグラフは部首分解が確定すると2.2のルールによって機械的に定まるものである。



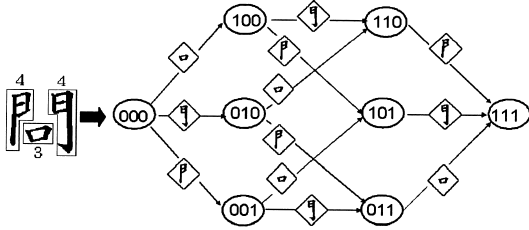
(a) Radical decomposition of pattern “国” ($N = 8, m = 3, n_1 = 2, n_2 = 5, n_3 = 1$)



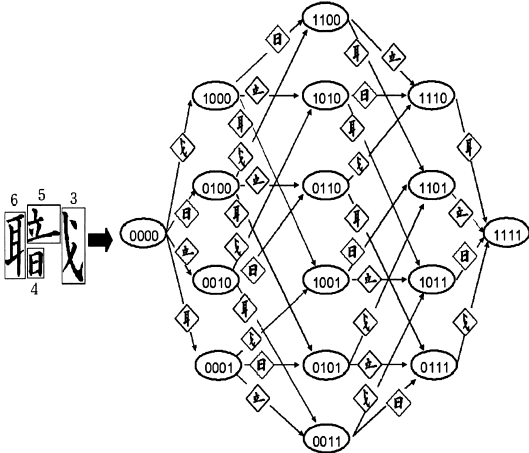
(b) Radical decomposition of pattern “興” ($N = 16, m = 4, n_1 = 4, n_2 = 6, n_3 = n_4 = 3$)

図 A・1 分解例と対応するキューブグラフ
Fig.A・1 Examples of radical decomposition and inter-radical cube graph.

(平成 16 年 10 月 25 日受付, 17 年 2 月 7 日再受付)



(c) Radical decomposition of pattern “問” ($N = 11, m = 3, n_1 = n_2 = 4, n_3 = 3$)



(d) Radical decomposition of pattern “職” ($N = 18, m = 4, n_1 = 6, n_2 = 5, n_3 = 4, n_4 = 3$)

図 A.1 (つづき)
Fig. A.1 (Continued)



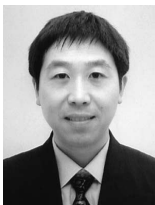
内田 誠一 (正員)

平 2 九大・工・電子卒．平 4 同大大学院修士課程（情報）了．セコム（株）勤務を経て，現在，同大システム情報科学研究院知能システム学部助教．博士（工学）．画像パターンの解析・認識，及び音声の認識・合成に関する研究に従事．IEEE，日本音響学会，情報処理学会，映像情報メディア学会各会員．



迫江 博昭 (正員：フェロー)

昭 41 九工大・制御卒．昭 43 九大大学院修士課程（電子）了．同年日本電気（株）入社．平元，九大に移り，現在システム情報科学研究院知能システム学部助教．工博．音声認識及び画像パターン解析の研究に従事．IEEE ASSP Senior Award，本会業績賞，本会米沢ファウンダーズメダル受賞記念特別賞及び論文賞（社）日本発明協会・内閣総理大臣発明賞，九工大嘉村記念賞各受賞．IEEE，日本音響学会，情報処理学会，映像情報メディア学会各会員．



蔡 文杰 (学生員)

平 2 中国武漢大・理工・電子卒．平 8 同大大学院修士課程（情報）了．現在，九大大大学院博士後期課程在学中．文字認識に関する研究に従事．