

パターン類似度法による 手書き平仮名文字認識の実験

Experiment on Machine Recognition of Handprinted "HIRAGANA" Characters
by Pattern Matching Method

村瀬 洋 木村 文隆 吉村 ミツ 三宅 康二
Hiroshi MURASE Fumitaka KIMURA Mitsu YOSHIMURA Yasuji MIYAKE

名古屋大学 工学部 三重大学 工学部

Faculty of Engineering, Nagoya University Fac. of Eng., Mie Univ.

This paper describes methods on pattern matching basis for recognizing handprinted Japanese characters called "HIRAGANA". The preprocessing consists of normalizing the size and position of an input pattern and gaussian filtering for noise reduction. The similarity to each category is calculated from a formula involving the eigen vectors of the correlation matrix derived from learning pattern vectors of the same category. We examined the effects of increasing the number of eigen vectors used and limiting the value of inner products of the eigen vectors and input pattern vector. In a recognition experiment, correct recognition rate of 99.8% was obtained for learning data and that of 96.7% for test data.

1 まえがき

パターン類似度法は、印刷文字OCR等で一般的に用いられる手法である。特に飯島の提案した複合類似度法⁽¹⁾は、印刷文字の位置ずれや太さの変化等の難者に対し高い性能が示され、ASPET/71⁽²⁾をはじめ2000字種を含む印刷漢字OCR⁽⁴⁾にも生かされている。

パターン類似度法では、その処理システムが空間回路論によって構成できるという特徴があげられる。すなわち他の逐次的、直列的な処理を多く含む文字認識手法に比較し並列的でありより高速な認識が可能となる。

ここではこの類似度法を手書き平仮名文字⁽⁵⁾に適用し、その効果を調べた。手書き平仮名文字は印刷文字に比較し極端な変形を持つている。又カタカナ文字等が高い認識率が得られている位相線分法⁽⁶⁾およびストローク抽出法⁽⁷⁾でも複雑な特徴抽出操作が必要となり、より困難な認識対象となっている。類似度法の場合、高度な特徴抽出機構は不用であるが、文字の重なり⁽⁸⁾に注目した前処理が重要となる。今回は前処理として

印刷文字にない変動、特に大きさの正規化を行った。それに加えASPET/71開発時に確立された文字認識技術であるボケ操作および標準化操作を行った。類似度の計算には複合類似度を基本として、特に特徴部分空間の拡大、高次軸での成分値の制限等を行い、その結果学習データで96.7%の認識率が得られた。

2 実験データ

認識実験には電子技術総合研究所作製の常用手書きひろが打文字データベースを用いた。これは、所定の用紙の縦7mm×横5mmの枠内に「見本を見て、丁寧に書いて下さい。」という指示のもとに書かれた文字を、顕微鏡装置で64×63メッシュに標本化し濃度値を16レベルに量子化したものである。

今回は48カテゴリのうち"・"、"。"を除く46カテゴリを使用し、学習データとして最初の9200文字を、未知データとして残りから2300文字を採用した。

3 学習・認識 システム 概要

学習・認識システムの流れを図1に示す。

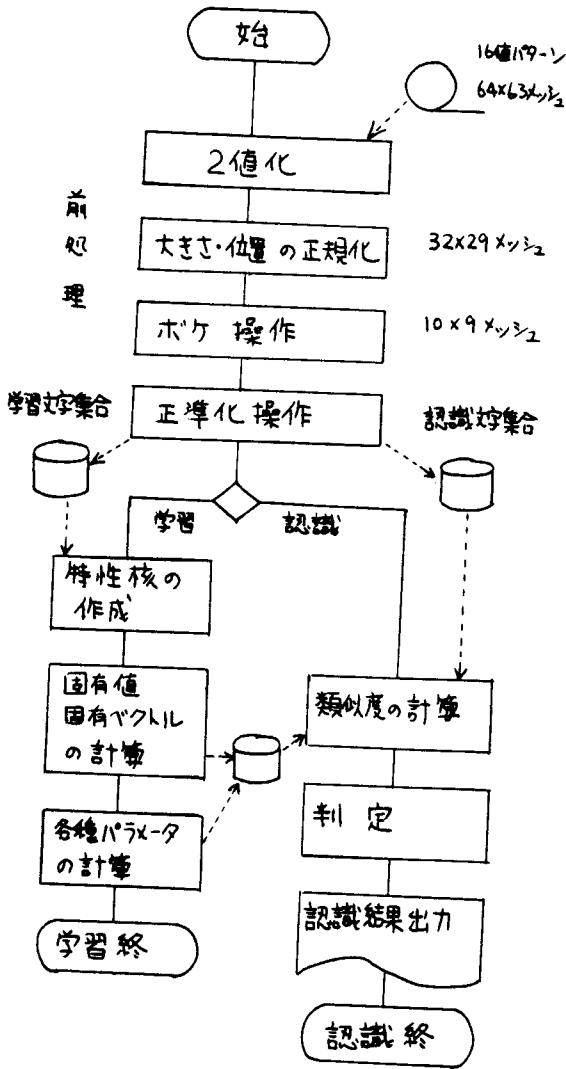


図1 学習・認識システム

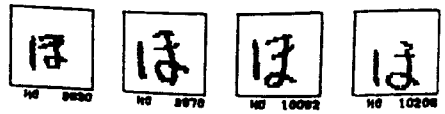


図2 文字例「ほ」(2値化パターン)

字の大きさの変動が認識に悪影響を与えることが推測される。そこでこのような変形を吸収する目的で1次変換の範囲で大きさ及び位置の正規化を行った[式(1),(2)]。

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} X' \\ Y' \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \quad \dots\dots(1)$$

$$f^*(X', Y') = f(X, Y) \quad \dots\dots(2)$$

但し X, Y 元の座標系

X', Y' 変換後の座標系

$f(X, Y)$ 原文字パターン

$f^*(X, Y)$ 変換後の文字パターン

ここで $a_{ij}, b_k (1 \leq i, j, k \leq 2)$ の適当な設定により文字の回転までを含む大きさ及び位置の正規化が可能となる。しかし、ここでは処理が単純で直感的に類似度法に合理的と思われる手法として $a_{12} = a_{21} = 0$ とした次の2つの方法を用いた。

(1) 正規化1 (図3)

文字を縦方向および横方向に伸縮して出力画面いっぱいにする。すなわち a_{11}, a_{22} を独立に決める。

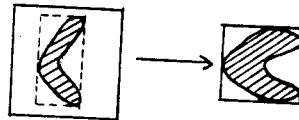


図3 正規化1の例

(2) 正規化2 (図4)

文字を相似的に伸縮させて、文字を囲む長方

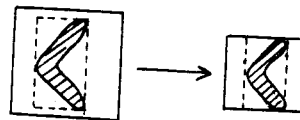


図4 正規化2の例

3.1 正規化

手書き文字は同じく内に書かれた文字であっても、人により、あるいは条件により、図2の2値化パターンの例にもあるようにわくの隅に書かれる場合や、大小さまざまに書かれる場合が多い。パターン類似度法の場合、類似度は文字を構成する線分等の絶対的な位置に着目し全体として未知パターンと標準パターンがどれ程よく重なるかに依存するため、位置ずれや文

形の中心を出力画面の中心と一致させその条件で出力画面のわくに文字が落ちるようにする。この場合 $a_{11} = a_{22}$ となる。

図5は正規化1を施した文字パターン("す")を重ね合せたものである。

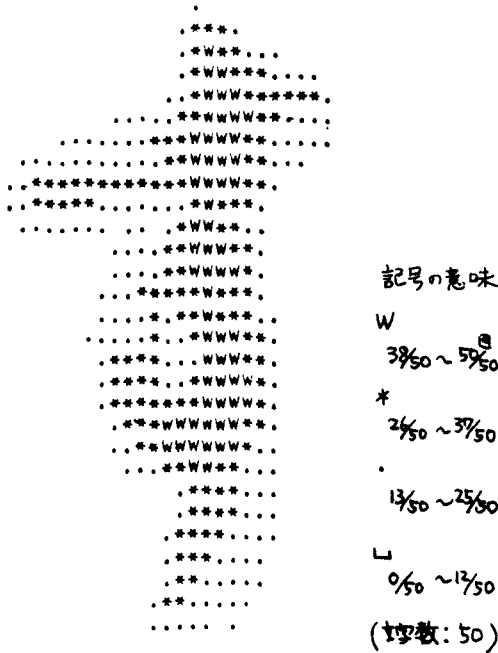


図5 大きさを正規化した文字の重ね合せ例

この例から分かるように、位置および大きさを正規化した文字パターンは比較的良好に重なる。

3.2 ボケ操作および正規化操作

ボケ操作および正規化操作は、試作印刷文字 OCR ASPET/71 で用いられた技術である。ボケ・正規化操作の効果については次のように説明されている。

図形が受ける変動、特に不規則性変動には多くの種類がある。しかしこれらに共通していることは、文字図形の形状に比べて変化の激しい分布形状をなし、高い周波数成分中に含まれていることである。ボケ操作の目的はこのノイズ成分を抑制し本質的な図形成分を浮きよらせることにある。又正規化はボケ操作による一様化したパターンベクトルに対し異なるパターン間での分離をよくするために必要な操作である。

ボケ操作はガウス形重み関数と原文字パターンとのたたみ込み積分で計算されるもので、一般的には式(3)で表される。

$$f(w) = \frac{1}{4\pi\sqrt{T}} \int \exp\left\{\frac{1}{4}(w-w')T^{-1}(w-w')\right\} \times g(w') dw'$$

但し g : 原文字パターン ----- (3)

f : ボケパターン

T : ガウス形分布の形状を表す量

又、正規化は式(4)で表現される。

$$f(w) = f(w) - \left\{ (f, f_c) / \|f_c\|^2 \right\} f_c(w) \text{ ----- (4)}$$

但し f : 原文字パターン

f_c : 正規化パターン

f_c : 位置 w に無関係なパターン

次に述べる予備実験1では印刷文字で効果のあったこれらの操作が手書き文字でどの程度の効果があるかについて調べた。

予備実験1

大きさと位置の正規化だけを施したパターンと、それにボケ・正規化操作を施したパターン[図6]について、単純類似度法[式(9)]による認識率を求めた。表1にその結果を示す。

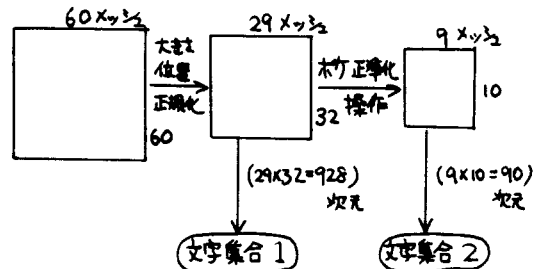


図6 単純類似度法による予備実験1

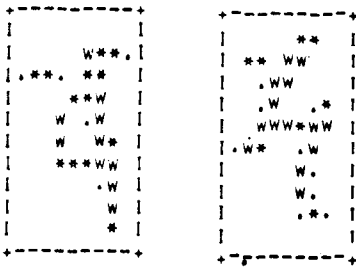
表1 予備実験1の結果

	文字集合1	文字集合2
認識率	84.7%	85.8%

同表が示すように、ボケ及び正規化操作をした文字集合2は、文字集合1に比べて、パターン次元数がかなり低下しているにもかかわらずより良い認識率を与えた。単純類似度で良い結果を与えた前処理が必ずしも他の類似度法の前処理として良いとはいえないが、評価の目安と

はなるであろう。

ボケ・正規化操作を施した文字パターン例を図7に示す。



〔最大値から最小値までのレベルを4等分しそれぞれを"W", "*", ".", " "とした〕

図7 前処理後の文字パターン例 ("す", "ぞ")

3.3 正規化

正規化①に加えてボケ・正規化操作が終った段階で文字線幅(要素の値が正の部分の太さ)がある許容値以下の場合に元の文字パターンにもどって太め操作を施し、単純に太さを正規化する方法を正規化ると名づける。

3.4 類似度

(1) 文字表現

文字パターンは次のような濃度の位置に関する関数 $f(x)$ で表わせるが、ここでは x を量子化しその各点の濃度から作られる正規化ベクトル \mathbf{f} ($\|\mathbf{f}\|=1$) を考える(図8)。

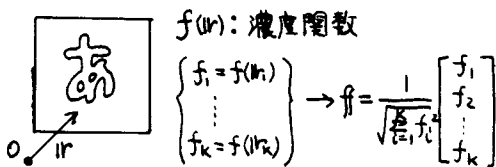


図8 文字パターンベクトルの意味

(2) 特徴空間

文字パターンベクトル \mathbf{f} の次元 n は普通極めて高い(ボケ操作による第2標本化で次元を落しても、今回の場合 $n=90$ である)。そこでパターンの特徴を変更しないうちに、なるべく次元を減らすことが考えられる。各カテゴリごとにパターンベクトルの分布を調べ、低次元化された空間がそのカテゴリの特徴をよく示

す部分空間といえ、この空間を特徴空間と呼ぶことにする。

その空間の選び方としては、仮書らによつて示されているモード関数系によつて得られるものを用いた。すなわちカテゴリ i の特徴空間は、カテゴリ i の特性核 [式(5)] の固有値

$$K^{(i)} = \frac{1}{n} \sum_{j=1}^n g_j^{(i)} g_j^{(i)T} \quad \text{----- (5)}$$

但し $g_j^{(i)}$: j 番目の学習パターン

n : 学習文字数

$\lambda_k^{(i)}$ ($i=1, 2, \dots, K$; 但し $\lambda_1^{(i)} \geq \lambda_2^{(i)} \geq \dots \geq \lambda_k^{(i)}$)

及び、それに対応する固有ベクトル $\varphi_k^{(i)}$

[$\{\varphi_k^{(i)}\}_{k=1, 2, \dots, K}$ はモード関数系 (Karhunen-Loeve 直交系ともいう) と呼ばれる] のうち、固有値の大きい順に m 個選んだ固有ベクトル $\{\varphi_k^{(i)}\}_{k=1, 2, \dots, m}$ によつて張られる空間とする。
($m \leq K$)

具体例として、テストパターンに1次元変動を加えたときの $\varphi_1^{(i)}$ 、 $\varphi_2^{(i)}$ を図8に示す。位置変動及び線幅変動に対応する変動を一様分布で与えたときの固有ベクトルの概略図である。
 $\varphi_2^{(i)}$ 、 $\varphi_3^{(i)}$ では変動部分で高い成分値を持つことか分かる。

位置変動

線幅変動

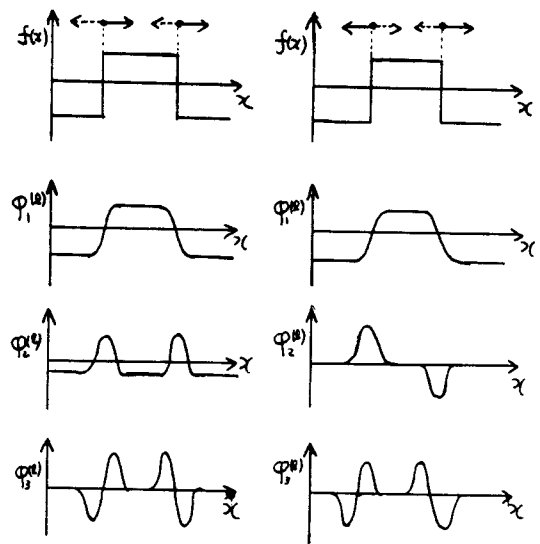


図8 1次元変動における固有ベクトルの例

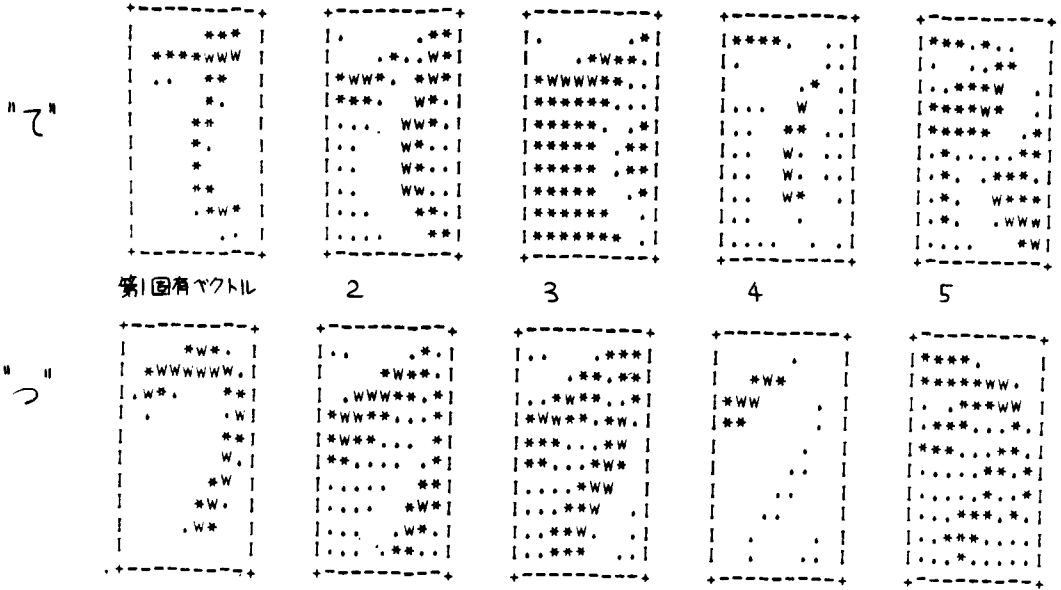


図9 平仮名文字パターンの固有ベクトル例：「て」と「つ」
(パターンの記号(W, *, .,)の意味は図7と同じ)

図9に平仮名文字パターンの固有ベクトルの例を示す。

一般に第1固有ベクトルは平均パターンに極めて近く、第2, 3固有ベクトルは変動部分に絶対値の高い成分をもつパターン、あるいは少数の変形パターンが現れる。

(3) 類似度法

類似度 $S^{(k)}(f)$ は未知パターンとカテゴリ- l の標準パターンがどの程度、似ているかを示す量であり、これを用いて f のカテゴリ-を次のように判定する。

$$S^{(l)}(f) > S^{(k)}(f) \longrightarrow \begin{array}{l} f \text{ のカテゴリ-を} \\ l \text{ と判定} \end{array}$$

(k は l 以外のすべてのカテゴリ-)

(4) 実験に用いた類似度

カテゴリ-別の部分空間に基づく類似度は、基本的には未知パターンベクトルが、各カテゴリ-の特徴空間内に持っている成分の大きさであるが、実験ではこの類似度を式(6)のように変形したものを用了。

$$\textcircled{\text{C}} S^{(l)}(f) = \left\{ \sum_{i=1}^m \alpha_i A_i^{(l)}(f)^2 \right\}^{\frac{1}{2}} \quad \dots (6)$$

但し

$$A_i^{(l)}(f) = \min \{ L_i^{(l)}, |(\varphi_i^{(l)}, f)| \} \quad \dots (7)$$

$$L_i^{(l)} = \max_j \{ |(\varphi_i^{(l)}, g_j)| \} \quad \dots (8)$$

$A_i^{(l)}(f)$ は i 番目の特徴軸との類似度、 α_i はその荷重値、 $\varphi_i^{(l)}$ は特性核 $K^{(l)}$ の i 番目の固有ベクトル、 $L_i^{(l)}$ は i 軸の制限値、 g_j は j 番目の学習パターン ($1 \leq j \leq m$) である。

予備実験2

実際の文字パターンが各カテゴリ-の特徴空間内でどのような分布を持っているかについて調べるため、文字パターン 50個を用いて、各固有ベクトルとの内積値の平均および最大値を調べた。

表2は、カテゴリ-「あ」と「い」の特徴空間内で、文字「あ」が持つ各軸平均を示したものである。この例から分かるように自カテゴリ-空間内では、低次の軸程大きな成分をもっているのに対し、他カテゴリ-空間ではこの傾向がほとんど認められない。

又図10は、各軸成分の上限を、自カテゴリ-空間内、他カテゴリ-空間内に分けて調べた結果を示したものである。

表2 各軸成分の平均値

軸番号	"a" 特徴空間	"i" 特徴空間
1	37.8	6.62
2	12.0	4.18
3	9.1	6.27
4	6.2	2.82
5	5.9	10.37

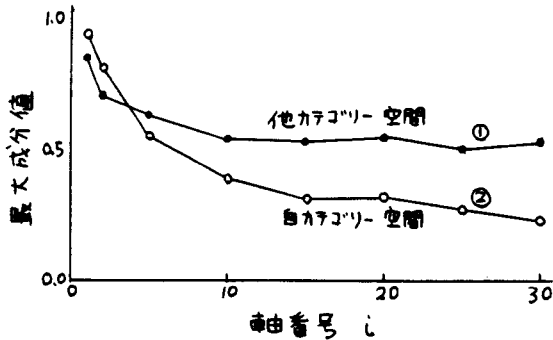


図10 各軸における文字パターン成分の最大値

この予備実験2から分かるように、文字認識のために重要な情報は、ほとんど低次の教軸に属していると思われる。しかし少数であるが極端な変形をもつ文字に対しては類似度をあげるためには、高次の軸まで使用する必要がある。高次の軸までとったとき同時に高くなりかちな他カテゴリ-空間における類似度を下げる目的でここでは制限値 $L_i^{(1)}$ を設けた。 $L_i^{(1)}$ を 0.1 の値と一致させたとき、自カテゴリ-空間との類似度をほとんど下げずに他カテゴリ-空間との類似度を下げることが期待できる。

ここでは、この $S_i^{(1)}(f)$ 以外に比較の目的で典型的な類似度として知られている次の $S_0^{(1)}(f)$ と $S_1^{(1)}(f)$ をも実験に用いた。

$$\textcircled{2} S_0^{(1)}(f) = (f, \varphi^{(1)}) \quad \dots\dots (9)$$

ここで $\varphi^{(1)}$ はカテゴリ- l の標準パターンであり、実験では平均パターンを用いた。この類似度は単純類似度とも呼ばれる。

$$\textcircled{3} S_1^{(1)}(f) = \left\{ \frac{1}{n} \sum_{i=1}^m (f, \varphi_i^{(1)})^2 \right\}^{\frac{1}{2}} \quad \dots\dots (10)$$

$$S_i^{(1)}(f) \approx \left\{ \sum_{j=1}^m \lambda_j^{(i)} (f, \varphi_j^{(i)})^2 \right\}^{\frac{1}{2}} \quad \dots\dots (11)$$

但し、 $\varphi_j^{(i)}$ はカテゴリ- l の j 番目の学習パターン、 n は学習文字数、又 $\lambda_j^{(i)}$ ($i=1, 2, \dots, m$; $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_m$)、 $\varphi_j^{(i)}$ はそれぞれ特性核 $K^{(i)}$ の固有値、固有ベクトルである。

式(10)で示される手法は、学習パターンすべてとの内積の2乗和によって類似度を計算する手法であり、今回使用した類似度 $S_i^{(1)}(f)$ の基礎となった方法である。この類似度は(11)式で示すように特性核の少数の固有値、固有ベクトルを用いて近似的に表現できる。特に実際のパターンで計算してみると、固有値の2番目以降の値の収束が極めて速いため、教次元程度までとることによって、ほとんど(10)式の値と(11)式の値は等しくなる。そこで実験では(11)式を用いた。

4 実験結果 および考察

4.1 正規化の比較

正規化の1~3を用いて認識実験を行、たところ、表3に示す結果を得た。又文字の分布状況の目安となる固有値の大きさを表4に示す。

表3 正規化による認識率の比較

次元数	正規化1	正規化2	正規化3
10	94.96%	95.00	95.30
15	95.22	95.22	95.65
20	95.91	95.57	95.74
25	96.13	95.61	95.87

表3から分かるように次元数を増加していったとき正規化1の場合に最も認識率が高くなるので、以後の認識では正規化1を使用することとする。

固有値の収束性の速さは、その文字パターンがどの程度小さな次元内に存在しているかを示す。縦横方向に別々に伸縮を行、て大きさを正規化する操作(正規化1)に対し、相似性を保つ伸縮(正規化2)、あるいは積分の大きさをあ

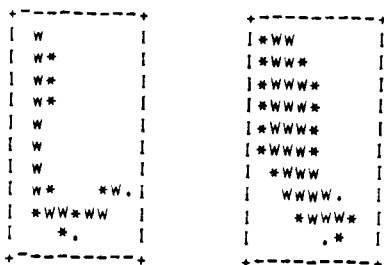
表4 カテゴリー"あ"の正規化による固有値の違い

軸番号	正規化1	正規化2	正規化3
1	0.560	0.593	0.665
2	0.085	0.070	0.070
3	0.045	0.038	0.043
4	0.026	0.030	0.024
5	0.022	0.021	0.018
6	0.020	0.020	0.015

許容値までたらず操作(正規化3)によって固有値の収束がやや良くなっていることが表4から分かる。すなわち、正規化2及び3によって文字パターンが比較的安定化したといえる。これに対応して、特徴空間の次元が少なくなるときの例えば10次元程度では、表3に示すように正規化2, 3のほうが認識率がやや高くなっている。

特徴空間の次元数を増加していったとき、正規化1のほうが認識率が高くなるのは、正規化2, 3の場合より、高次の軸(固有ベクトル)の異なるカテゴリー間での類似性が少ないためと思われる。

正規化1が正規化2に比較して文字の安定性が悪いのは、図11に示すような縦方向あるいは横方向に極端に長い文字の場合、処理(縦横独立にその伸縮比を決める)によって文字パターンが異常に広がることがあるためであると思われる。



普通に縮尺した"し" 異常に縮尺した"し"

図11 正規化1による異常例

4.2 各類似度法の比較

3.4 (4) で示した類似度 $S^{(1)}(f)$, $S_0^{(1)}(f)$, 及び $S_1^{(2)}(f)$ を用いて認識した結果を図12に示す。

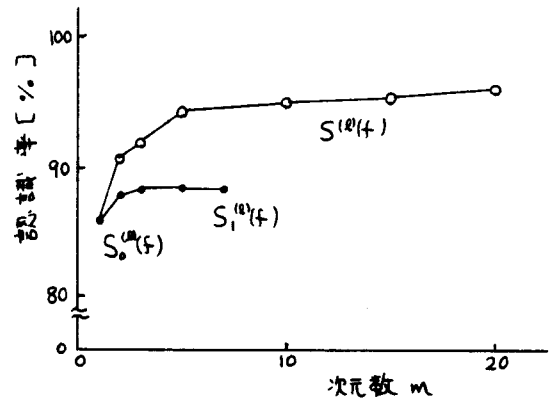


図12 特徴空間の次元に対しての各手法の比較

図12で示されるように、いずれの類似度を用いる場合も、単純類似度 $S_0^{(1)}(f)$ を用いる場合と比べると特徴空間の次元数を増加させるによって認識率が向上している。 $S_1^{(2)}(f)$ において、3次元程度で認識率が飽和しているのは、特性核の固有値の収束が速く、4次元以上の特徴軸(固有ベクトル)を使用してもほとんど類似度が変化しないためである。

$S^{(1)}(f)$ では高次の特徴軸の荷重値が高いため次元数を増大し過ぎると、他カテゴリー空間との類似度も高くなり認識率が低下する。文字パターン空間が90次元である今回の場合、特徴空間の次元数を30程度としたとき最も認識率が高くなることが分った。

4.3 各軸の類似度制限の効果

各軸との類似度上限を設けることが認識向上にどの程度役立つかを調べるため、下記の条件で実験を行った。

(1) 制限を付けない場合

式(5)の $\alpha_i = 1.0$ ($i=1, 2, \dots, 90$), 式

(6)の $L_i^{(1)} = 1.0$ ($i=1, 2, \dots, 90$) とする。

(2) 制限を付けた場合

式(5)の $\alpha_i = 1.0$ ($i=1, 2, \dots, 90$), 式

(6) の $L_i^{(1)}$ を式(7) で求めた量とする。

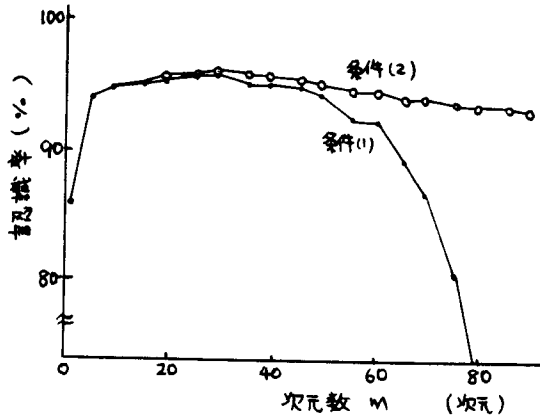


図 13 制限の影響

実験結果を図 13 に示す。同図が示すように特徴空間の次元数が低いとき、例えば 1~8 次元程度ではほとんど認識率が変らないのに対し、次元数を十分大きくするとやや向上する。これは、低次元軸では上限値が大きいので他カテゴリ空間上でのその軸の成分が大きくても、制限効果が顕著に現れないためであろう。

この条件のもとでは特徴空間の次元数が 30 のとき認識率が最大となり、制限しないときに比較して認識率で約 0.4% 向上した。

4.4 学習量増加の効果

現在使用している学習文字数は、1 カテゴリ一きり 200 文字である。手持のデータ数の都合で実験にはやや少ないが、学習文字数を 250 に増加して実験を行った。その結果を表 5 に示す。

表 5 学習量増加による認識率

学習量 次元数	200 文字	250
10 次元	94.96%	95.22%
15	95.22%	95.52%
20	95.91%	95.91%
25	96.13%	96.09%

10~20 次元程度で認識率がやや高くなっていることが分かる。更に学習量を増加したとき認識率がどのように変化するかについてはここでは分からない。

4.5 誤読パターンについて

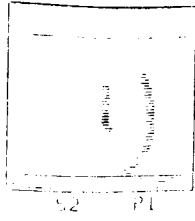
表 6 に、未学習データ 2300 文字を認識させたときの誤り文字を示す。

表 6 誤読表 [() 内はその個数]

入力	誤読結果	入力	誤読結果
あ	め(1), わ(1)	か	ぬ(1)
い	り(1)	の	へ(1), づ(1)
う		は	ほ(1), に(1)
え	ん(2)	ひ	む(1), れ(1)
お	よ(1)	ふ	
か	ろ(1), な(1), す(1), お(1), は(1)	へ	はい(1)
き	さ(3), ま(1)	ほ	ほ(4)
く		ま	よ(1)
け		み	れ(1)
こ		む	も(4)
さ	こ(1), き(1)	め	わ(1), あ(1)
し		も	ぎ(1), た(1), ま(1)
す		や	れ(1)
せ	と(1), や(1), て(1)	ゆ	み(1), ま(1), か(1)
そ	と(1)	よ	は(1), あ(1), ま(1), は(1)
た		ろ	ろ(1), ち(2), む(1), よ(1)
ち	や(1), ろ(3), も(1)	り	み(1)
つ		る	よ(1), ま(2)
て	そ(2)	れ	わ(1)
と	し(1), て(1)	ろ	ろ(1), ち(2)
な		わ	
に	す(1)	を	と(1), ら(1)
ぬ	ね(2)	ん	も(1)

図 14 に実験の誤読パターンの例を示す。誤読の原因を厳密に分類するのは困難であるが、ここでは前処理の失敗と、それ以後の失敗とに大別した。前処理の失敗は 2 値化の失敗、切出しの失敗、大きさの正規化による極端な変形等があげられる。又、それ以後の失敗は極端な文字変形、人間が見てもまぎらわしい、文字の多くの部分が重なる、と分類してみた。

図 15 に未知文字におけるカテゴリ一別の認識率を示す。



52 P1



17 M1



26 M0



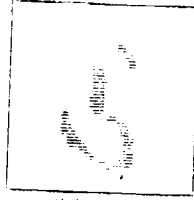
71 M0

(a) 2値化の失敗例

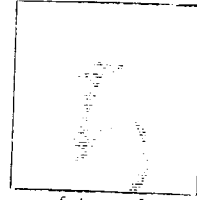
(b) セグメント抽出の失敗例



37 M0



42 M0



67 M1



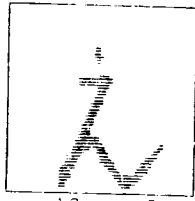
113 M1

(c) 極端な文字変形例

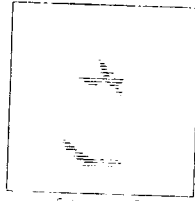
"ち" → "ろ"

"め" → "わ"

(d) 人間が見てもまぎらわしい例



10 E



31 SR

"え" → "ん"

"せ" → "こ"

(e) 文字の多くの部分が重なった例

図14 誤読パターン例

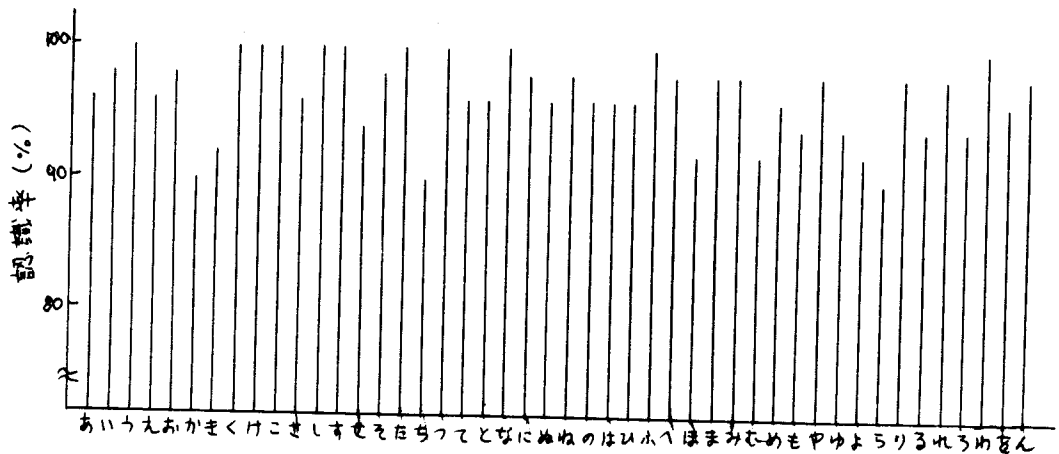


図15 カテゴリ別の認識率

5 むすび

パターン類似度を用いて、手書き平仮名文字を認識した結果を表7に示す。

表7 類似度法による認識率(棄却=0)

類似度	未知文字	学習文字
$S_0^{(4)}$	85.8%	90.0
$S_1^{(4)}$ _{10次元}	88.0	
$S_2^{(4)}$ _{35次元}	96.5	99.8

又、2値化の失敗による誤読文字6例のうち4例は、4近傍の孤立点を除去することによって正読となった。これにより、未知文字データの認識において96.7%の正読率が得られ、パターン類似度は比較的単純な処理の割に高い認識率を与えることが分かった。

今後の課題としては、前処理について更に詳細な検討を加えることによつて、より適切な正規化方法を確立すること、および文字パターン次元数を適切なレベルにおさえることなどがある。手持ちのデータ数の関係で学習量増大の効果を十分調べる事ができないが、未知文字と学習文字の認識率が極めて異なることから考えてこの効果は興味深い。特徴空間の次元数は、実際面では認識時間と記憶容量に直接影響を与える量であるため、更に次元数を減らそうような効果的な特徴空間の決定などを考えていきたい。

謝辞

日頃御指導いただき本学工学部、本多俊雄教授、ならびに熱心に討論して下さいた研究室諸氏に厚くお礼申し上げます。又、FACOM 230-38の使用に際しいろいろと便宜をはかって下さり、情報工学科の方々にも感謝の意を表します。

文献

- (1)飯島「文字読取装置 ASPET/71」テレビジョン Vol 27, NO3, 1973
- (2)飯島「混合類似度による識別理論(複合類似度法の一一般化)」PRL 74-24 (P45)
- (3)飯島, 森「人間の識別能力に迫るOCR」ASPET/71 日経エレクトロニクス 1972-5-22
- (4)森, 坂井「2000字種を100字/秒で読む印刷OCRの開発」日経エレクトロニクス 1977-10-31
- (5)村瀬, 杉浦, 吉村, 三宅「手書き平仮名文字認識における複合類似度の手法の応用効果」昭53 東海支那連大 NO378
- (6)村瀬, 吉村, 三宅「パターン類似度法を用いた手書き平仮名文字認識の一実験」昭54 通信全大 NO1333
- (7)山本, 森, 山田「凹凸構造抽出による手書きひらがな文字認識」PRL 78-38 (P1)
- (8)藤井, 吉村, 三宅, 市川「ストローク抽出法による手書き平仮名文字認識(Ⅱ)」PRL 78-8 (P21)