

再構成されたパターンに関する一考察

瀬 良 豊 士*

Consideration on the Reformed Pattern.

Toyoshi SERA

Abstract

The basic conception concerned with reduction of processing-time and memory-capacity in the use of a digital computer has already been shown by the author in the paper "Pattern Recognition by the codes of 3×3 elements and Karhunen Lóeve Orthogonal System." First of all, this paper says that a quantized pattern of 225 dimensions is transformed into a pattern vector of 9 dimensions, and secondly it shows the investigation of the relation among each vector.

Furthermore, the author researches that the degree of the dependence I_{ij} is defined on a set of the codes formed with 3×3 elements which construct pattern vector of low dimensions and that the degree of the dependence is related with the reformed pattern vector.

1. 緒 言

パターン認識の際、与えられたパタンの次元を低次元のパタン（再構成されたパタン）に変換することにより、計算機の処理時間を短縮し、かつメモリー容量を小さくする基本的な考えはすでに示されている^{1,2)}。ここでは、225次元（ 15×15 ）に量子化されたパタンを9次元という低次元のパタンに変換し、これを用いてパタンに関する考察を行なう。さらに、ここでは低次元のパタンを構成するに必要な 3×3 素子コードの集合上で従属の割合 I_{ij} を定義し、再構成されたパタンとの関係をあわせて考察する。以上において、ここで取り扱う考察点を示したが、これらの点を考察することは *Karhunen Lóeve* 直交系を用いたパターン認識にとって非常に重要なことである。

2. 3×3 素子コード

与えられたパタンのもつ情報をすべて用いることによりパターン認識を行なうとすれば、処理する情報量は大変なものとなり認識ということは不可能に近くな

る。たとえば、*Karhunen Lóeve* 直交系を用いたパターン認識において、与えられたパタンを 15×15 の(0.1)要素で量子化してさへも、処理過程における相関行列は 225×225 の正方行列となり処理する情報量は大変なものとなる。したがって、合理的にパターン認識を行なうには認識に必要な情報を取り出し、処理して認識を行なう必要がある。そこで、ここでは、与えられたパタンから必要な情報を取り出す 3×3 素子コード集合について述べる。

3×3 素子コード集合は 3×3 次元の(0.1)要素からなるマトリックスで、与えられたパタンから幾何学的な特徴、端点、2分岐点、3分岐点を抽出するように構成されている。この 3×3 素子コードは多くのパタン集合をもとに経験的に選ばれたものであり、ここでは、22個の 3×3 素子コードを特徴抽出（情報抽出）のために用いる。 3×3 素子コードによる特徴抽出の方法はすでに報告されているので^{1,3)}、ここでは 3×3 素子コードの集合のみを Fig. 1 に示す。この図で、端点を抽出する 3×3 素子コードの集合は Fig. 2 に示すような8方向の端点を抽出するように工夫されている。

* 工業短期大学部電気工学科

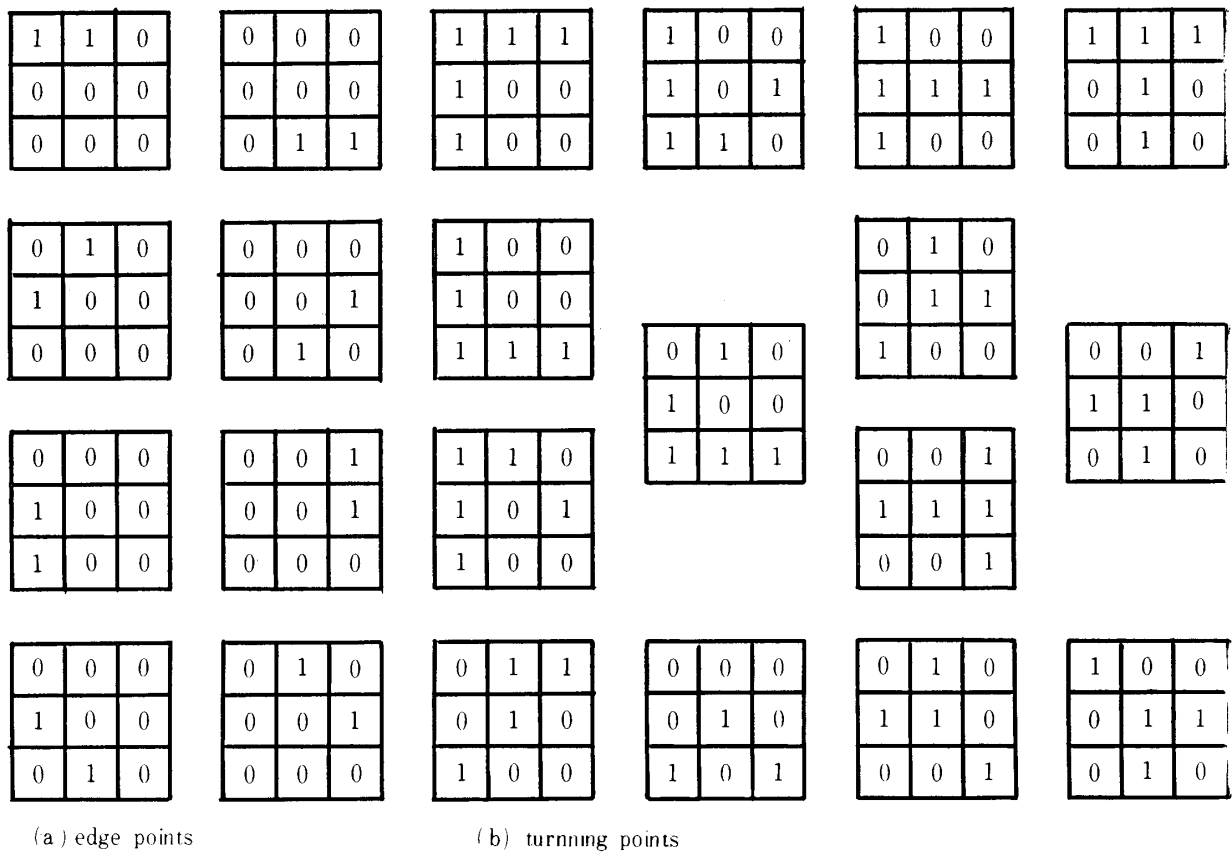


Fig. 1 A Set of 3 × 3 Codes, Case (I)

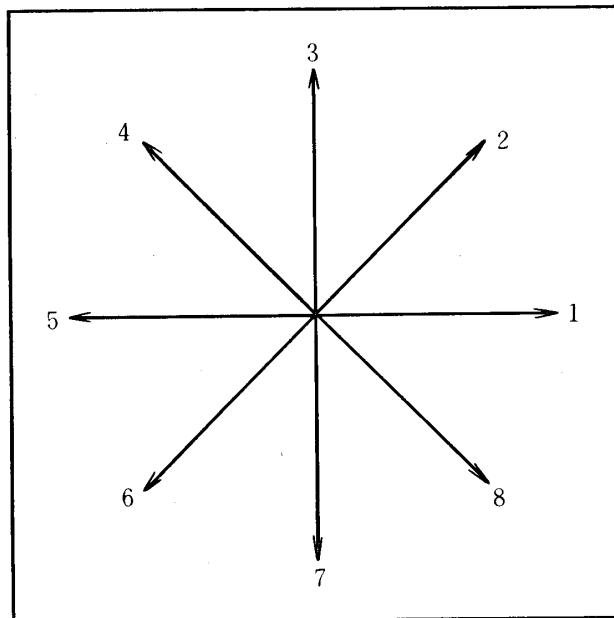


Fig. 2 Direction of 8 edges

3. 再構成されたパターン

再構成されたパターン (パターンベクトル) を構成するため 3 × 3 素子コード集合により抽出される特徴の集

合を F とすると,

$$F = \{f^1, f^2, \dots, f^i, \dots, f^{22}\} \dots (1)$$

ただし,

$$f^i = \begin{pmatrix} f_{11}^i & f_{12}^i & f_{13}^i \\ f_{21}^i & f_{22}^i & f_{23}^i \\ f_{31}^i & f_{32}^i & f_{33}^i \end{pmatrix}$$

$$f_{ij}^i : 0 \sim 1 \text{ の値. } i, j = 1 \sim 3.$$

となる. したがって任意の未知パターン x から抽出された特徴の集合を $F(x)$ とすれば,

$$F(x) = \{f^1(x), f^2(x), \dots, f^i(x), \dots, f^k(x)\} \dots (2)$$

$f^i(x)$: パターン x から抽出された特徴 f^i

と表される. この結果, $F(x)$ はパターン x のもつ固有の属性の集合であり, もとのパターン x を認識するために必要な情報を含んでいることがわかる. つぎに, この $F(x)$ の情報をもとに数学的処理 (Karhunen Lóeve 直交系) を施して認識を行なうため, この $F(x)$ 低次元のベクトルに変換することを考える¹⁾. この変換は集合 $F(x)$ のマトリックス要素を適当な低次元のベ

クトルに置きかえる変換である。ここでは、9次元のベクトルその要素 y_i に変換する場合を考える。変換方法は $f^i(x)$ の第1例を y_i の前の3つの要素に、第2行を中の3つの要素に、そして第3行を後の3つの要素に置きかえる方法であり、 $y_i(x)$ は

$$y_i(x) = (f_{11}^i(x), f_{12}^i(x), f_{13}^i(x), \dots, f_{31}^i(x), f_{32}^i(x), f_{33}^i(x)) \\ = (y_{i1}(x), y_{i2}(x), y_{i3}(x), \dots, y_{i7}(x), y_{i8}(x), y_{i9}(x)) \dots (3)$$

ただし、

$y_{ij} : f^i(x)$ の j 番目の要素を書きかえたものとなる。この $y_i(x)$ は $f^i(x)$ と同じものであるが、ベクトル Y (行列) を

$$Y(x) = \begin{pmatrix} y_1(x) \\ y_2(x) \\ \vdots \\ y_i(x) \\ \vdots \\ y_k(x) \end{pmatrix} = \begin{pmatrix} y_{11}(x), y_{12}(x), \dots, y_{19}(x) \\ y_{21}(x), y_{22}(x), \dots, y_{29}(x) \\ \vdots \\ y_{k1}(x), y_{k2}(x), \dots, y_{k9}(x) \end{pmatrix} \dots (4)$$

を構成すると、 Y の各列の要素が1となる度はパターン x 固有のものとなり、 $F(x)$ と同様、認識を行なうに必要な情報を含むことが分かる。しかし、抽出される特徴の数によって、行列 Y の各列の1になる度合いが左右されるので、マトリックスの各和の列和 X_i をつぎのように求め、

$$X_i = \sum_{l=1}^k y_{li}(x) \dots (5)$$

これを要素とする新しいベクトル X を

$$X = \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_9 \end{pmatrix} = \begin{pmatrix} \sum_{l=1}^k y_{l1}(x) \\ \sum_{l=1}^k y_{l2}(x) \\ \vdots \\ \sum_{l=1}^k y_{l9}(x) \end{pmatrix} \dots (6)$$

と構成し、さらに

$$X^t \cdot X = 1 \dots (7)$$

$t : X$ の行ベクトルを表わす。

さるよう正規化すれば、抽出された特徴の数によ

て左右されないパターンベクトルが構成される。この結果、このパターンベクトル X は構成方法から、もとのパターン x を認識するために必要なパターン固有の情報を含んだベクトルであることが分かる。そして、このベクトル X を再構成されたパターンベクトルと呼ぶ。Fig. 3 にパターンベクトルが再構成される過程を示している。

4. 従属の割合の定義

パターンから特徴を抽出する 3×3 素子コードの集合 F の要素が互いに独立であれば、独立していない場合に比べて、より多くの情報を抽出することができる。したがって要素が互いに独立であるコード集合から構成されたパターンベクトルは、より多くの情報を含んだベクトルとなり認識に有利になると推測できる。しかし、 3×3 素子コードはほんの9つの要素から構成されているにすぎない。さらに、コード集合内には同一視できるコードも含まれている。したがって、すべてのコードを独立にかつ必要な情報を抽出するように構成することは不可能である。そこで、ここでは従属の割合 I_{ij} を定義し、この値が小さいコード集合と大きいコード集合を構成することにより、 I_{ij} の値が小さいとき、即ちコード集合の要素間がより独立であるとき、推測通りよりよい認識が行なわれることを示す。

従属の割合 I_{ij} を定義するため、 3×3 素子の集合 F の内で、端点を抽出するコードの集合 C_1 、2分岐点を抽出するコードの集合を C_2 、3分岐点を抽出するコードの集合を C_3 とすると、

$$F = \{ C_1, C_2, C_3 \} \dots (8)$$

$$C_1 = \{ y_1, y_2, \dots, y_8 \} \dots (9)$$

$$C_2 = \{ y_9, y_i, \dots, y_{15} \} \dots (10)$$

$$C_3 = \{ y_{16}, y_{17}, \dots, y_{22} \} \dots (11)$$

ただし、

$y_i : f^i$ を事前に 1×9 行ベクトルに変換したものである。 $i = 1 \sim 22$ 。

と表わされる。そこで、コード y_i とコード y_j 間の従属の割合 I_{ij} を

$$I_{ij} = (y_i \cdot y_j) = \sum_{l=1}^9 y_{li} \cdot y_{lj} \dots (12)$$

$i, j = 1 \sim 22$

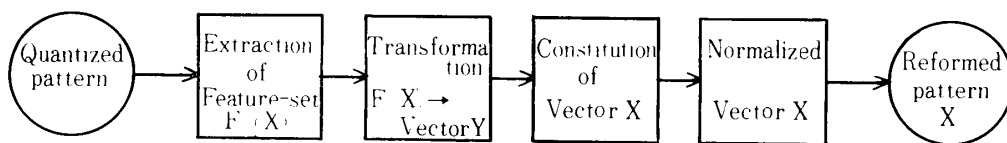


Fig. 3 Process of Constituting Reformed Pattern

と定義する。(12)式で y_i, y_j を正規化すれば、

$$I_{ij} = \frac{\sum_{l=1}^9 y_{il} \cdot y_{jl}}{\sqrt{\sum_{l=1}^9 y_{il}^2} \cdot \sqrt{\sum_{l=1}^9 y_{jl}^2}} \dots\dots\dots (13)$$

となり、 I_{ij} の値は

$$0 \leq I_{ij} \leq 1 \dots\dots\dots (14)$$

の範囲の値を取ることが分かる。ここで、 $I_{ij} = 0$ は完全独立、 $I_{ij} = 1$ は完全従属を表わす。

つぎに、集合 C_1, C_2, C_3 の間の I_{ij} を $I(C_\alpha, C_\beta)$ とすると、

$$I(C_\alpha, C_\beta) = \begin{cases} \sum_{i=1}^{l_1} \sum_{j=1}^{l_2} I_{ij} / l_1 \cdot l_2 & : \text{if } \alpha < \beta \\ \dots\dots\dots & \dots\dots\dots (15) \\ \sum_{i=1}^{l_1} \sum_{j=1}^{l_2} I_{ij} / \{l_1 \cdot (l_1 - 1) / 2\} & : \text{if } \alpha = \beta \\ & i < j \end{cases}$$

ただし、

$l_1, l_2 : C_\alpha, C_\beta$ の要素の数、 $\alpha, \beta = 1 \sim 3$ 。

となる。さらに、 $\alpha = 1$ としたときの平均の従属割合 I_s は

$$I_s = \frac{1}{3} \sum_{\beta=1}^3 I(C_1, C_\beta) \dots\dots\dots (16)$$

と表わせる。

5. 類似度

再構成されたパタンを用いて、パタン間、パタン類間、従属の割合 I_{ij} 、および認識率との関係を考察するため類似度 L_m^n 、および g_{ij} を定義する⁴⁾。そのためまず、パタンの集合 X^s を、

$$X^s = \{X_m^n\} \dots\dots\dots (17)$$

n : 各クラスに含まれる任意に選ばれる代表パタンの数、 $n = 1 \sim N$ 。

m : クラスの数、 $m = 1 \sim M$

とする。つぎに平均パタンベクトル M_m を

$$M_m = \sum_{n=1}^N P_m(n) \cdot X_m^n \dots\dots\dots (18)$$

$P_m(n)$: クラス m 内でパタン X_m^n が生起する確率。

と定めると、 M_m とクラス m 内のそれぞれのパタンとの類似度 L_m^n は

$$L_m^n = (M_m \cdot X_m^n) / \|M_m\| \cdot \|X_m^n\| \dots\dots\dots (19)$$

として定義できる。この式で M_m, X_m^n を正規化しておけば、

$$L_m^n = (M_m \cdot X_m^n) \dots\dots\dots (20)$$

となる。この値 L_m^n によって各クラスのパタンが平均

のパタンにどの程度近いものであるかがわかる。さら

に、 L_m^n の確率平均 L_m を

$$L_m = \sum_{n=1}^N P_m(n) \cdot L_m^n \dots\dots\dots (21)$$

とすれば、この値から各クラスのパタンの均一の度合がわかる。

同様の考え方で、クラス間の類似度 g_{ij} を平均パタンベクトル上で考えると

$$g_{ij} = (M_i \cdot M_j) \dots\dots\dots (22)$$

$i, j = 1 \sim M$

と定義できる。

6. 実験方法

さきの説明の考えにもとづいた結果を示すため、まず、パタン集合として A, B, C, D, E, F, G, H の 8 個の文字からなるパタン集合を用い、これらを 15×15 次元の (0.1) 要素のマトリックスに量子化した後、パタンベクトルを構成することにした。

つぎに、 I_{ij} について考察を行なうため、Fig. 1 で与えられたコード集合の中で、端点を抽出するコード集合のみが異なった別のコードを構成した。そして両者を区別するため、Fig. 1 で与えられたコード集合を **Case (I)**、他を **Case (II)** として表わすことにした。最後に、(17)式のパタン集合 X^s は、 $N=27$ とし

$$X^s = \{X_m^n\} \dots\dots\dots (17)$$

$n = 1 \sim 27, m = 1 \sim 8$ 。

を考察することにした。

7. 実験結果

Table 1 に再構成されたパタンベクトルの例を示している。この表で最後の列は正規化の要素を表わしている。

Table 2 には **Case (I)** の集合 C_1 と C_2 の要素間の I_{ij} の計算結果を示している。さらに、Table 3 には、(15)、(16)式から求めた **Case (I)**、**Case (II)** の $I(C_\alpha, C_\beta)$ 、および I_s の値を示している。

Table 4 には、(19)式から求めた L_m^n の値を示している。そして、Table 5 には、(20)式から求めた g_{ij} の計算結果を示している。Fig. 4 には、**Case (I)**、**Case (II)** で再構成されたパタン C の L_m^n の値、および L_m の値を示している。

Table 1 Example of reformed patterns

Dimension Input pattern	Dimension									Normalized factor $\sum_{l=1}^9 x_{li}^2$
	1	2	3	4	5	6	7	8	9	
A	0	6	2	2	5	4	3	0	2	98
	1	3	1	3	4	4	2	1	2	61
B	3	2	1	3	3	1	2	2	2	45
	1	2	2	2	4	1	4	1	2	51
C	1	1	0	0	1	0	1	0	1	5
	2	3	1	2	0	0	1	0	0	19
D	2	1	1	2	0	0	2	1	1	16
	1	0	0	1	1	1	2	1	1	10
E	4	5	1	3	1	1	2	3	1	67
	5	6	2	2	2	1	4	1	1	92
F	3	3	2	1	2	2	2	1	0	36
	3	2	1	1	1	2	1	0	0	21
G	2	3	2	0	2	2	1	2	2	30
	1	1	2	1	1	1	0	1	0	10
H	1	0	3	4	2	4	3	0	1	56
	0	1	3	3	2	4	3	0	1	49

Table 2 Result of calculation I_{ij}

$C_1 \backslash C_2$	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8
y_9	0.6325	0.6325	0.6325	0.3162	0.0000	0.0000	0.3162	0.3162
y_{10}	0.3162	0.3162	0.6325	0.6325	0.6325	0.3162	0.0000	0.0000
y_{11}	0.6325	0.6325	0.6325	0.3162	0.6325	0.3162	0.3162	0.6325
y_{12}	0.3536	0.3536	0.3536	0.0000	0.0000	0.0000	0.3536	0.3536
y_{13}	0.3162	0.3162	0.6325	0.6325	0.0000	0.6325	0.0000	0.3162
y_{14}	0.3162	0.6325	0.6325	0.6325	0.6325	0.3162	0.0000	0.3162
y_{15}	0.0000	0.0000	0.4082	0.0000	0.4082	0.0000	0.0000	0.0000

Table 3 Result of calculation $I(C_\alpha, C_\beta)$

Case	$I(C_\alpha, C_\beta)$						I_s
	$I(C_1, C_1)$	$I(C_1, C_2)$	$I(C_1, C_3)$	$I(C_2, C_2)$	$I(C_2, C_3)$	$I(C_3, C_3)$	
A	0.2143	0.4110	0.2499	0.5245	0.4697	0.5183	0.2917
B	0.5000	0.2835	0.4681	0.5245	0.4697	0.5183	0.4172

Table 4 Result of calculation L_m

		Value of L_m			
Case	Pattern				
Case (I)	A	0.937804	Case (II)	A	0.950881
	B	0.852655		B	0.848007
	C	0.743770		C	0.843751
	D	0.830081		D	0.818651
	E	0.929348		E	0.917792
	F	0.932322		F	0.944962
	G	0.837808		G	0.875225
	H	0.947533		H	0.957003

Table 5 Result of calculation g_{ij}

Pattern		Value of g_{ij}							
Case	Pattern	A	B	C	D	E	F	G	H
I	A	1.00000							
	B	0.826975	1.00000						
	C	0.744302	0.775796	1.00000					
	D	0.637831	0.923123	0.715970	1.00000				
	E	0.642088	0.842221	0.919334	0.852593	1.00000			
	F	0.767883	0.842115	0.889943	0.793459	0.937773	1.00000		
	G	0.880523	0.841484	0.807819	0.751519	0.782346	0.877966	1.00000	
	H	0.873826	0.868129	0.650401	0.767876	0.620101	4.744501	0.800620	1.00000
II	A	1.00000							
	B	0.825012	1.00000						
	C	0.851029	0.850049	1.00000					
	D	0.610382	0.929199	0.676566	1.00000				
	E	0.790572	0.947194	0.910463	0.812593	1.00000			
	F	0.891050	0.913606	0.927063	0.722432	0.963413	1.00000		
	G	0.927418	0.862750	0.924601	0.652143	0.885852	0.950301	1.00000	
	H	0.903988	0.802368	0.881491	0.592309	0.834820	0.907739	0.948198	1.00000

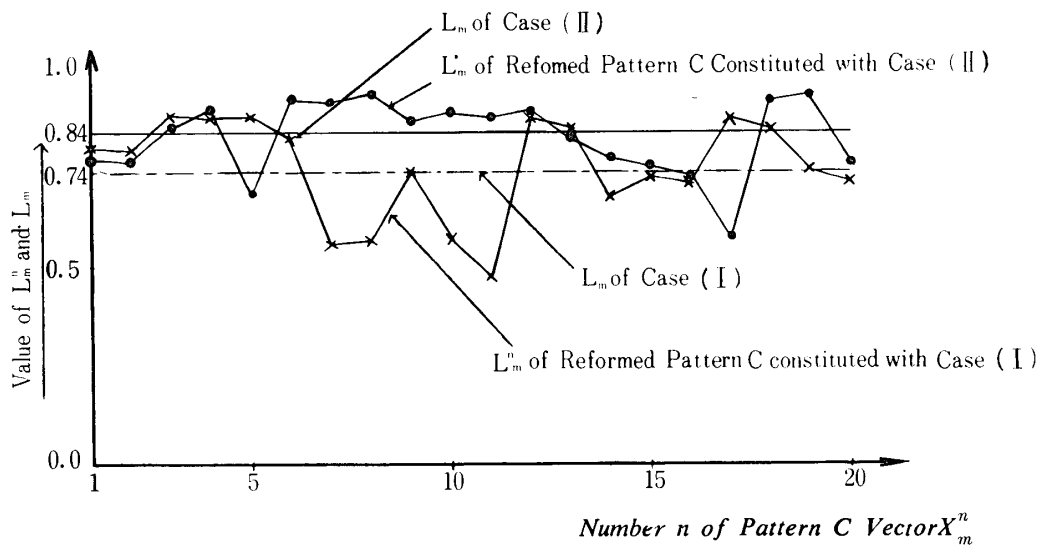


Fig. 4 Variation of L_m^n

8. 考 察

Table 1, 2, 3, 4, 5 および Fig. 4 で, 再構成されたパタン, 類似度 L_m^n , L_m , g_{ij} および 3×3 素子コード集合上での I_{ij} の計算結果を示した. ここではコード集合 Case (I), Case (II) を用いて行なった認識実験の結果のもとにして全体の考察を行なう認識方法としては, *Karhunen Lóeve* 法を用い64個の未知文字を認識させた. この結果, Case (I) の場合認識率 87.5%, Case (II) の場合 43.7% という認識率を得た. これより, さきに述べたように I_{ij} の値 (Table 3 の I_s 参考) が小さいとき, よい結果が得られることが示された. ところが, Table 4, Fig. 4 を見ると L_m^n , L_m の値に関しては, Case (I), Case (II) 両者にはほとんど差がなく, どちらかと言えば Case (II) の方が値が大きく, パタン類内のパタンは均一であることがわかる. したがって, Case (II) による認識結果の方が良いはずであることになり, 先のことと矛盾する. しかし, Table 5 の g_{ij} の値を見ると Case (I) の場合は Case (II) の場合に比べて全体的に小さい. 即ち, パタン類間の類似度が小さいことを示している. このことはパタン認識を行なう際に, 認識システムがパタンを判別しやすいことを示している. この結果が 3×3 素子コードの従属の割合 I_{ij} の値の差, および認識率の差となって表われることになり, 矛盾のないことがわかる.

9. 結 論

以上の考察から

- (1) 3×3 素子コード集合上で I_{ij} の値を小さくすることは, パタン類間の類似度を小さくする.
- (2) その結果, 認識システムはパタンを判別しやすくなる.
- (3) パタン類内の均一度が大きいことよりも, パタン類間の類似度が小さいことが認識に際しては大切であると考えられる.
- (4) I_{ij} の値は 3×3 素子コード集合を構成する方法の大きな指針となる

という結論が得られた. 今後は, 3×3 素子コードをもう少し次元の高いベクトルに変換した場合の従属の割合 I_{ij} , 類似度, および認識率の関係を調べ, パタン類が多くなっても低次元の立場で *Karhunen Lóeve* 法が使用できることを示す考えである.

最後に, 日頃から何かと御指導頂く同大学工学部平田威彦助教授に深く謝意を表します. また, 計算機使用に際して手助けしていただいた西村女史に感謝します.

参 考 文 献

- 1) 瀬良豊士: 情報処理学会誌 (1971年4月号) 掲載予定
- 2) 瀬良豊士: 電気四学会中国支部大会, 223, 15 (1971)
- 3) 瀬良豊士: 山口大工学部研究報告, 22, 55 (1971)
- 4) 野口, 飯島: 情報処理, 12, 26~35 (1971)

(昭和46年12月2日受理)