

文字認識用
プログラムパッケージ
SMART

利用手引書

サブルーチン及び関数篇

豊田工業高等専門学校 情報工学科

野澤繁之
竹下鉄夫
岡部直木

三重大学 工学部 電子工学科
鶴岡信治

三重大学 工学部 情報工学科
木村文隆

中部大学 経営情報学部
吉村ミツ

平成8年3月

文字認識用プログラムパッケージ S M A R T

機能と（サブルーチン及び関数の）登録個数一覧

機能別	登録個数
1. 前処理のためのサブルーチン	計 24
1) 雑音の消去	3
2) 2値化	1
3) 位置、大きさの正規化	20
2. 特徴抽出のためのサブルーチン	計 17
4) 重み付き縮小	8
5) ヒストグラム	1
6) 方向セグメント抽出	4
7) 外郭方向寄与度抽出	4
3. 辞書作成のためのサブルーチン	計 6
8) 統計量計算	6
4. 識別に用いる関数	計 11
9) 識別関数	11
5. 出力のためのサブルーチン	計 17
10) 印刷、プロッター、 ファックスなどへの出力ルーチ	17
6. その他のサブルーチン	計 5
11) その他	5
登録サブルーチン及び関数	総計 80

S M A R T サブルーチン及び関数 目次

名 前	機 能	ページ
0. サブルーチン及び関数の名前と関数一覧	 1
1. 前処理のためのサブルーチン		
1) 雑音の消去 F I L T M 4 F I L T M 8 S M O O T H	4 近傍による平均値フィルタ 8 近傍による平均値フィルタ 非線形平滑化 5 7 9
2) 2 値化 B I N A R Y	多値文字画像の 2 値化 1 2
3) 位置、大きさの正規化 P S I Z E B P S I Z E M N S I Z E B N S I Z E M G C E N T B G C E N T M N T R N S 1	2 値文字画像について 指定領域の大きさ (右端、左端、上端、下端) を求める 多値文字画像について 指定領域の大きさ (右端、左端、上端、下端) を求める 2 値文字画像の大きさ (右端、左端、上端、下端) を求める 多値文字画像の大きさ (右端、左端、上端、下端) を求める 2 値文字画像の重心の位置を 求める N O R M B 2 の子ルーチン 多値文字画像の重心の位置を 求める N O R M M 2 の子ルーチン 2 値文字画像の伸縮変換 (N O R M B 1 の子ルーチン 1 3 1 5 1 7 1 9 2 1 2 3 2 5

名 前	機 能	ペー ジ
N T R N S 2	多値文字画像の伸縮変換 (N O R M B 2 の子ルーチン)	… … 2 7
N T R N S 3	2 値文字画像の伸縮変換 (N O R M B 7 の子ルーチン)	… … 2 9
N O R M B 1	2 値文字画像の枠合わせ による位置、大きさの正規化	… … 3 1
N O R M B 2	2 値文字画像の重心合わせ による位置、大きさの正規化	… … 3 3
N O R M B 3	2 値文字画像の重心・枠合わ による位置、大きさの正規化	… … 3 5
N O R M B 4	2 値文字画像の枠合わせ による位置、大きさの正規化 (小領域の平均値)	… … 3 7
N O R M B 7	2 値文字画像の枠合わせ による位置、大きさの正規化 (縦横比を保持)	… … 3 9
N O R M M 1	多値文字画像の枠合わせ による位置、大きさの正規化	… … 4 1
N O R M M 2	多値文字画像の重心合わせ による位置、大きさの正規化	… … 4 3
N O R M M 3	多値文字画像の 重心・枠合わせによる 位置、大きさの正規化	… … 4 5
N O R M N L	2 値文字画像の線密度 イコライゼーションによる 非線形正規化法	… … 4 7
N O R M N 2	2 値文字画像の線密度 イコライゼーションによる 非線形正規化法 2	… … 4 9
C A N C A L	ベクトルの正純化	… … 5 1

名 前	機 能	ペ ー ジ
2. 特徴抽出のためのサブルーチン		
4) 重み付き縮小 W E I G T 1	文字画像データの第2標本化 (その1)	… 5 2
W E I G T 2	文字画像データの第2標本化 (その2)	… 5 5
W E I 2 2 4	2(横) × 2(縦) × 4(方向) への第2標本化処理	… 5 9
W E I 4 4 4	4(横) × 4(縦) × 4(方向) への第2標本化処理	… 6 1
W G T I N 1	重み関数を求める(その1)	… 6 3
W G T I N 2	重み関数を求める(その2)	… 6 5
W G T I N 3	重み関数を求める(その3)	… 6 7
W G T I N 4	重み関数を求める(その4)	… 6 8
5) ヒストグラム H I S T S S	小領域内の4方向の ヒストグラム作成	… 7 0
6) 方向セグメント抽出 D S E G M T	文字画像から4方向の線素 (4方向セグメント)を抽出	… 7 2
D C O D E 4	8方向コーディング	… 7 4
P T G E N 4	方向パターンの生成	… 7 6
J O I N T 4	方向セグメントパターン の結合	… 7 8
7) 外郭方向寄与度抽出 H K I Y O	文字画像の外郭方向寄与度 特徴の計算	… 8 0
T K I Y O	水平方向の指定外郭深度 までの輪郭黒点を求める	… 8 2
X K I Y O	+45度方向の 指定外郭深度までの 輪郭黒点を求める	… 8 4

名前	機能	ページ
K I Y O D O	指定した点の方向寄与度 の計算	… 8 6
3. 辞書作成のためのサブルーチン		
8) 統計量計算		
M S T A T 1	特徴ベクトルの 平均を求める	… 8 7
M S T A T 2	特徴ベクトルの平均と 共分散行列を求める	… 8 9
M S T A T 3	特徴ベクトルの平均と 自己相関行列を求める	… 9 1
M S T A T 4	M S T A T 3 により 特徴ベクトルの平均と 自己相関行列を求めて、 その固有値、 固有ベクトルを求める (ベクトル版あり)	… 9 3
M S T A T 5	M S T A T 2 により 特徴ベクトルの平均と 共分散行列を求めて、 その固有値、 固有ベクトルを求める (ベクトル版あり)	… 9 5
M S T A T 6	等分散仮定の下で、 各クラスの平均ベクトルと 共分散行列から線形識別 関数の重みベクトルと 定数項を求める	… 9 7
4. 識別に用いる関数		
9) 識別関数		
C D 1	シティ距離	… 9 9
E D 1	二乗ユークリッド距離	… 1 0 0
E D 2	ユークリッド距離	… 1 0 1

名 前	機 能	ペ ー ジ
L D 1	線形識別関数	… 1 0 2
M D 1	マハラノビス距離 (ベクトル版あり)	… 1 0 3
M Q D 1	修正2次識別関数1 (ベクトル版あり)	… 1 0 5
M Q D 2	修正2次識別関数2 (ベクトル版あり)	… 1 0 7
M S 1	複合類似度 (ベクトル版あり)	… 1 0 9
Q D 1	2次識別関数 (ベクトル版あり)	… 1 1 1
S S 1	単純類似度1	… 1 1 3
S S 2	単純類似度2	… 1 1 4
5. 出力のためのサブルーチン		
10) 印刷、プロッター、ファックスなどへの出力ルーチン		
P O U T 1 6	16値文字画像の ラインプリンター出力	… 1 1 5
P O U T 2 B	2値文字画像の ラインプリンター出力	… 1 1 6
M S W A K U	2次元配列の周囲、 上下左右に'0'を付ける	… 1 1 7
P L T B I A	多値画像のプロッター出力	… 1 1 8
P L T B 4 5	2値画像20文字分の プロッター出力	… 1 2 1
P L T H 4 A	4方向指數ヒストグラムの プロッター出力	… 1 2 3
P L T H 4 B	小領域の4方向指數 ヒストグラムの プロッター出力	… 1 2 5
P L T H 4 5	4方向指數ヒストグラム 20文字分プロッター出力	… 1 2 7
P L T 2 T I	2値画像のプロッター出力	… 1 2 9

名 前	機 能	ペ ー ジ
D S P G S P	文字画像の グラフィック出力	… 1 3 1
I N T D S P	サブルーチン G S P D S P の初期化	… 1 3 4
E N D D S P	サブルーチン G S P D S P の終了処理	… 1 3 5
P L L I N E	文字画像をグラフィックに 出力するための データを作成	… 1 3 6
F X P P L C	F A X, P P L 出力のため のパラメータのチェック	… 1 3 8
F X P P L O	F A X, P P L へ出力	… 1 4 1
F X P P L P	F A X, P P L 出力のため のパラメータ、 タイトル設定	… 1 4 3
F X P P L R	F A X, P P L 出力のため の文字画像の入力と パッキング	… 1 4 7
6. その他のサブルーチン		
11) その他		
B I T S E T	1 バイトの文字型データの ビットパターンをセット	… 1 5 1
E L B L N K	2 値文字画像の周囲の 不要な空白を除去	… 1 5 3
R W H I S T	パッキングデータの横方向 のヒストグラムを求める	… 1 5 5
C L H I S T	パッキングデータの縦方向 のヒストグラムを求める	… 1 5 7
S O U K A N	1 次元配列の自己相関係数 を求める	… 1 5 9

S M A R T サブルーチン及び関数 名前と引数一覧

[前処理のためのサブルーチン]

(1) 雑音の消去

F I L T M 4 (IA, I1, J1, IB)
F I L T M 8 (IA, I1, J1, IB)
S M O O T H (IA, I1, J1, IS, IE, JS, JE, IB, IT)

(2) 2 値化

B I N A R Y (IA, I1, J1, IB, IT)

(3) 位置、大きさの正規化

P S I Z E B (IA, I1, J1, IS, IE, JS, JE, IL, IR, JT, JB, ILL)
P S I Z E M (IA, I1, J1, IS, IE, JS, JE, IL, IR, JT, JB, IT, ILL)
N S I Z E B (IA, I1, J1, IL, IR, JT, JB, ILL)
N S I Z E M (IA, I1, J1, IL, IR, JT, JB, IT, ILL)
G C E N T B (IA, I1, J1, IGX, IGV, SVX, SVY, ILL)
G C E N T M (IA, I1, J1, IGX, IGY, SVX, SVY, IT, ILL)
N T R N S 1 (IA, I1, J1, IB, I2, J2, C11, IL, C21, JT)
N T R N S 2 (IA, I1, J1, IB, I2, J2, C11, C12, C21, C22)
N T R N S 3 (IA, I1, J1, IB, I2, J2, IL, IR, JU, JD)
N O R M B 1 (IA, I1, J1, IB, I2, J2, ILL)
N O R M B 2 (IA, I1, J1, IB, I2, J2, CX, CY, ILL)
N O R M B 3 (IA, I1, J1, IB, I2, J2, ILL)
N O R M B 4 (IA, I1, J1, IB, I2, J2, ILL)
N O R M B 7 (IA, I1, J1, IB, I2, J2, ILL)
N O R M M 1 (IA, I1, J1, IB, I2, J2, IT, ILL)
N O R M M 2 (IA, I1, J1, IB, I2, J2, CX, CY, IT, ILL)
N O R M M 3 (IA, I1, J1, IB, I2, J2, IT, ILL)
N O R M N L (IA, I1, J1, IB, I2, J2, HX, HY, NX, NY, IX, IY)
N O R M N 2 (IA, I1, J1, IB, I2, J2, HX, HY, NX, NY, IX, IY)
C A N C A L (A, IS, B)

[特徴抽出のためのサブルーチン]

(4) 重み付き縮小

WEIGT1	(IA, I1, J1, B, I2, J2, W, K1, K2, IP, JP, IS, JS, ILL)
WEIGT2	(IA, I1, J1, B, I2, J2, W, K1, K2, IP, JP, IS, JS, ILL)
WEI224	(IA, OB, W)
WEI444	(IA, OB, W)
WTIN1	(W, K1, L1, ILL)
WTIN2	(W, K1, K2, ILL)
WTIN3	(W)
WTIN4	(W, K1, K2, ILL)

(5) ヒストグラム

HISTSS	(IX, IY, IN, IS, IS1, IHIST, IH, JH, I1, J1)
--------	--

(6) 方向セグメント抽出

DSEGMENT	(IA, I1, J1, IB, IWORKE, IWORKC)
CODE4	(E, F, C, IMAX, JMAX)
PTGEN4	(F, D, IMAX, JMAX)
JOINT4	(F, D, IMAX, JMAX)

(7) 外郭方向寄与度抽出

HKIYO	(IA, I1, J1, H, IZDIV, NS, ITDIV, IERR)
TKIYO	(IA, I1, J1, J, DN, NS, ERR)
XKIYO	(IA, I1, J1, I, J, DN, NS, ERR)
KIYODO	(IA, I1, J1, I, J, D, ILL)

[辞書作成のためのサブルーチン]

(8) 統計量計算

M S T A T 1	(LSP, MV, N, M)
M S T A T 2	(LSP, MV, COV, N, M)
M S T A T 3	(LSP, MV, ACOR, N, M)
M S T A T 4	(LSP, EVAL, WORK, EVEC, N, M, ESP, ILL)
M S T A T 5	(LSP, MV, EVAL, WORK, EVEC, N, M, ESP, ILL)
M S T A T 6	(MV, COV, W, W0, N, L)

[識別に用いる関数]

(9) 識別関数

C D 1	(X, MV, N)
E D 1	(X, MV, N)
E D 2	(X, MV, N)
L D 1	(X, W, W0, N)
M D 1	(X, MV, EVAL, EVEC, N)
M Q D 1	(X, MV, EVAL, EVEC, N, H)
M Q D 2	(X, MV, EVAL, EVEC, N, K, H)
M S 1	(X, EVEC, N, K)
Q D 1	(X, MV, EVAL, EVEC, N)
S S 1	(X, MV, N)
S S 2	(X, NMV, N)

[出力のためのサブルーチン]

(10) 印刷、プロッター、ファックスなどへの出力ルーチン

P O U T 1 6	(IA, I1, J1, ILL)
P O U T 2 B	(IA, I1, J1, ILL)
M S W A K U	(IA, I1, J1, IB, M)
P L T B I A	(IA, I1, J1, X0, Y0, DX, DY, IT, MNO, MCA)
P L T B 4 5	(IA, I1, J1, MESG, MOJINO, MOJICA, MOJINN, MP)
P L T H 4 A	(HIST4, IHH, JHH, MESG, MOJINN, MOJICA, MOJINN, MP)
P L T H 4 B	(HIST4, IHH, JHH, I, J, R, CX, CY)
P L T H 4 5	(HIST4, IHH, JHH, MESG, MOJINO, MOJICA, MOJINN, MP)
P L T 2 T I	(IA, I1, J1, X0, Y0, DX, DY, ITHRSH)
D S P G S P	(IA, I1, J1, AISTEP, AJSTEP, IMOJI, JMOJI, IMOJIC, JMOJIC, FRAME, ILL)
I N T D S P	(DPTYPE, ILL)
E N D D S P	(ENDDSP)
P L L I N E	(IA, I1, J1, NXP, NYP, NX, NY, NYY, NL, NP)
F X P P L C	(I1, J1, SIZE, I2, J2, AISTEP, AJSTEP, IMOJI, JMOJI, IMOJIC, JMOJIC, FRAME, IWIDE, KSCALE, ILL)
F X P P L O	(SIZE, IB, I2, J2, FAXPPL, ILL)
F X P P L P	(I1, J1, SIZE, IB, I2, J2, AISTEP, AJSTEP, IMOJI, JMOJI, FRAME, IWIDE, KSCALE, TITLE, ILL)
F X P P L R	(IA, I1, J1, SIZE, IB, I2, J2, IWORK, AISTEP, AJSTEP, IMOJI, JMOJI, IMOJIC, JMOJIC, FRAME, IWIDE, KSCALE, ICONT, ILL)

[その他のサブルーチン]

(11) その他

B I T S E T	(BITPAT, BYTPAT)
E L B L N K	(IA, I1, J1, IB, I2, J2, ILL)
R W H I S T	(IA, I1, J1, ISTROW, IENDRW, IBHIST, ILL)
C L H I S T	(IA, I1, J1, JSTCLM, JENDCL, IBHIST, ILL)
S O U K A N	(IHIST, IMAX, AUTCOR, ILL)
L I N E S G	

F I L T M 4

4 近傍による平均値フィルタ

作成	三浦重広、掛布英辰	1985年 7月
形式	サブルーチン	言語: FORTRAN サイズ: 51行

(1) 概要

画像データに4近傍による平均値フィルタをかける。

(2) 使用法

CALL FILTM4(IA, I1, J1, IB)

引数	型と種類	属性	内 容
IA	整数型 2次元配列	入力	入力画像
I1	整数型	入力	IA及IBの整合寸法（第1添字の値）
J1	整数型	入力	IA及IBの整合寸法（第2添字の値）
IB	整数型 2次元配列	出力	出力画像

(3) 計算方法

入力画像IAの各要素IA(I, J)についてその要素と4近傍の要素の値を加え5でわったものを四捨五入して整数化する。フィルタをかける範囲は、I = 2, I1 - 1, J = 2, J1 - 1である。

出力画像IBの要素のうち一番外側の要素にはその1つ内側の要素を代入する。

	1	—	I	—	I	1	
1	0	0	0	0	0	0	0
J	0						0
IA(I, J)							0
1	0						0
J	0						0
J1	0	0	0	0	0	0	0

入力画像

	1	—	I	—	I	1	
1							
J							
IB(I, J)							
1							
J							
J1							

出力画像

(4) 子ルーチン
なし。

F I L T M 8

8 近傍による平均値フィルタ

作成	三浦重広、掛布英辰	1985年 7月
形式	サブルーチン	言語 : FORTRAN サイズ : 51行

(1) 概要

入力画像に8近傍による平均値フィルタをかける。

(2) 使用法

CALL FILTM8(IA, I1, J1, IB)

引数	型と種類	属性	内 容
IA	整数型 2次元配列	入力	入力画像
I1	整数型	入力	IA及IBの整合寸法（第1添字の値）
J1	整数型	入力	IA及IBの整合寸法（第2添字の値）
IB	整数型 2次元配列	出力	出力画像

(3) 計算方法

入力画像IAの各要素IA(I, J)についてその要素と8近傍の要素の値を加え9でわったものを四捨五入して整数化する。ただし、IAの要素のうち一番外側の要素には0がはいってるものとし、フィルタをかける範囲は、I = 2, I1 - 1, J = 2, J1 - 1である。

出力画像IBの要素のうち一番外側の要素にはその1つ内側の要素を代入する。

	1	—	I	—	I	1	
1	0	0	0	0	0	0	0
J	0						0
	0		IA(I, J)				0
1	0						0
J	0						0
	0						0
J1	0	0	0	0	0	0	0

入力画像

	1	—	I	—	I	1	
1							
J							
			IB(I, J)				
1							
J							
J1							

出力画像

(4) 子ルーチン
なし。

SMOOTH

Smoothing

非線形平滑化

作成	鶴岡信治	1985年 7月
形式	サブルーチン	言語 : FORTRAN サイズ : 35行

(1) 概要

2値図形、濃淡画像の非線形平滑化を行う。4近傍に非0要素が3個以上ある0要素は、1要素の平均濃度値をその点の濃度値とする。したがって2値図形において2ビットまでの穴は除去することができ、1ビットの輪郭部のへこみを取り除ける。8近傍に0要素が8つある1要素は、0要素にする。すなわち、孤立点を除去する。

(2) 使用法

```
CALL SMOOTH(IA, I1, J1, IS, IE, JS, JE, IB, IT)
```

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	処理したい画像
I1	整数型	入力	IAの整合寸法 (第1添字の値)
J1	整数型	入力	IAの整合寸法 (第2添字の値)
IS	整数型	入力	平滑化する範囲 (X方向の下限)
IE	整数型	入力	平滑化する範囲 (X方向の上限)
JS	整数型	入力	平滑化する範囲 (Y方向の下限)
JE	整数型	入力	平滑化する範囲 (Y方向の上限)

I B	整数型 2次元配列	出力	処理結果の画像
I T	整数型	入力	I Aを2値化するしきい値

(3) 計算法

概要で述べたことを並列演算で行う。IA=IBとすると、逐次演算で行うことになり、多少並列演算と結果が異なるが、画像配列の節約になる。

(1) 近傍 (IA→f(x), IB→g(x)とする)

$$\begin{aligned} 4 \text{近傍} : N^4(x) &= \{X_1, X_3, X_5, X_7\} \\ &= \{X_k \mid k \subset S_1\} \end{aligned}$$

$$\begin{aligned} 8 \text{近傍} : N^8(x) &= \{X_1, X^2, \dots, X^8\} \\ &= \{X_k \mid k \subset S_1\} \end{aligned}$$

但し、 S_1 、 S はそれぞれ添字集合
 $\{1, 3, 5, 7\}$, $\{1, 2, \dots, 8\}$

X ₄	X ₃	X ₂
X ₅	X	X ₁
X ₆	X ₇	X ₈

図1 要素Xとその近傍X_k

(2) 2値画像 (濃淡画像についても以下のように考える)

$$1 \cdots f(x) \geq IT$$

$$b(x) =$$

$$0 \cdots f(x) < IT$$

点集合 : B = {X | b(x)=1} , W = {X | b(x)=0}

(3) 穴の除去

$\sum b(X_k) \geq 3$ となる画素については

$$g(x) = \frac{1}{N} \sum_{k \subset S_1 \cap B} f(X_k)$$

但し、NはS₁∩Bを満たす要素の数

(4)孤立点の除去

$\sum_{k \in S} b(x_k) = 0$ となる画素については $g(x) = 0$ とする。

(5)その他の画素 $g(x) = f(x)$

(4) 子ルーチン

なし

B I N A R Y

画像データの2値化

作成	三浦重広、掛布英辰	1985年 7月
形式	サブルーチン	言語 : FORTRAN サイズ : 31行

(1) 概要

画像データを与えられた敷居値により2値化する。

(2) 使用法

CALL BINARY(IA, I1, J1, IB, IT)

引数	型と種類	属性	内 容
IA	整数型 2次元配列	入力	入力画像
I1	整数型	入力	IA及IBの整合寸法（第1添字の値）
J1	整数型	入力	IA及IBの整合寸法（第2添字の値）
IB	整数型 2次元配列	出力	出力画像
IT	整数型	入力	敷居値

(3) 計算方法

IA(I, J) が IT 以上のとき IB(I, J) は 1

IA(I, J) が IT 未満のとき IB(I, J) は 0

(4) 子ルーチン

なし。

P S I Z E B

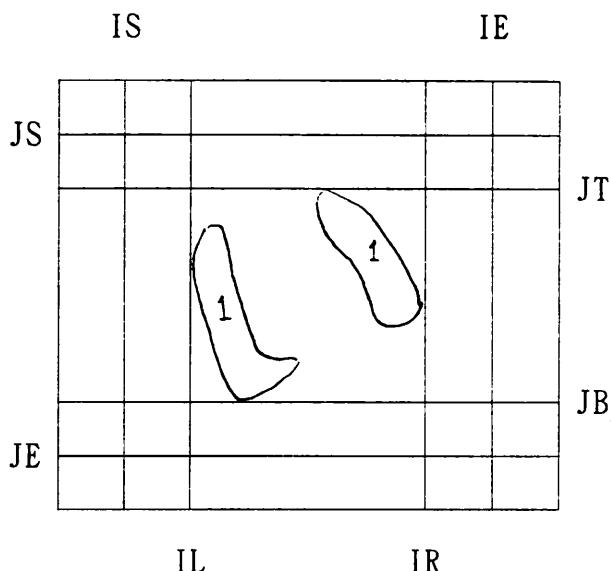
Object Size for binary picture

2値図形について図の大きさ（右端、左端、上端、下端）を求める

作成	鶴岡信治	1985年 7月
形式	サブルーチン	言語:FORTRAN サイズ: 46行

(1) 概要

IA(I1, J1)で与えられた画像の
特定の範囲 (IS, JS), (IE, JE)
内で 1 である場所の右端、左端、
上端、下端を求める。



(2) 使用法

CALL PSIZEB(IA, I1, J1, IS, IE, JS, JE, IL, IR, JT, JB, ILL)

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	処理したい画像
I1	整数型	入力	IA の整合寸法 (第 1 添字の値)
J1	整数型	入力	IA の整合寸法 (第 2 添字の値)
IS	整数型	入力	IA 中の検索する範囲 (左端)

I E	整数型	入力	I A 中の検索する範囲（右端）
J S	整数型	入力	I A 中の検索する範囲（上端）
J E	整数型	入力	I A 中の検索する範囲（下端）
I L	整数型	出力	I A 中の 1 要素の左端の位置
I R	整数型	出力	I A 中の 1 要素の右端の位置
J T	整数型	出力	I A 中の 1 要素の上端の位置
J B	整数型	出力	I A 中の 1 要素の下端の位置
I L L	整数型	出力	0 : 正常終了 1000: 全部 0 であり、上端が求まらない 2000: 下端が求まらない 3000: 左端が求まらない 4000: 右端が求まらない

(3) 計算法

画像配列 I A を左上から走査して、はじめて 1 となる行を求めて、これを画像の上端の位置(JT)とする。

他の端についても同様であり、走査の方向だけが異なる。

(4) 子ルーチン

なし

PSIZEM

Object Size for multilevel picture

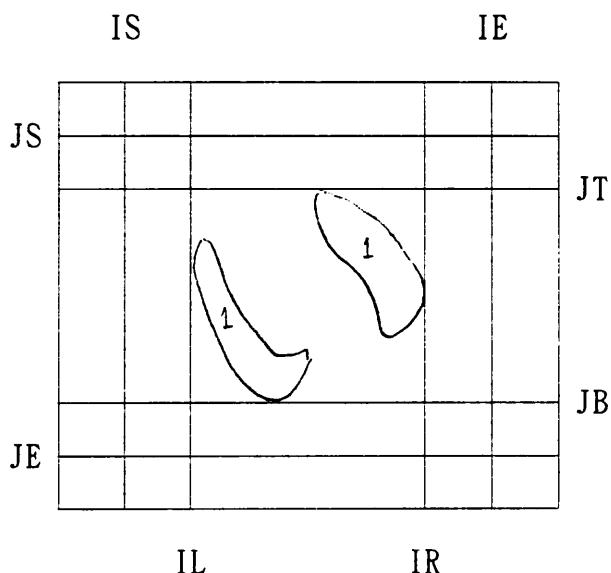
濃淡画像について図の大きさ（右端、左端、上端、下端）を求める

作成	鶴岡信治	1985年 7月
形式	サブルーチン	言語:FORTRAN サイズ: 46行

(1) 概要

IA(I1, J1)で与えられた画像の
特定の範囲(IS, JS), (IE, JE)
内で、しきい値以上の濃度であ
る場所の右端、左端、上端、
下端を求める。

PSIZEBの濃淡図形版。



(2) 使用法

CALL PSIZEM(IA, I1, J1, IS, IE, JS, JE, IL, IR, JT, JB, IT, ILL)

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	処理したい画像
I1	整数型	入力	IAの整合寸法（第1添字の値）
J1	整数型	入力	IAの整合寸法（第2添字の値）
IS	整数型	入力	IA中の検索する範囲（左端）

I E	整数型	入力	I A 中の検索する範囲（右端）
J S	整数型	入力	I A 中の検索する範囲（上端）
J E	整数型	入力	I A 中の検索する範囲（下端）
I L	整数型	出力	I A 中の 1 要素の左端の位置
I R	整数型	出力	I A 中の 1 要素の右端の位置
J T	整数型	出力	I A 中の 1 要素の上端の位置
J B	整数型	出力	I A 中の 1 要素の下端の位置
I T	整数型	入力	I A を 2 値化するしきい値
I L L	整数型	出力	0 : 正常終了 1000: 全部 0 であり、上端が求まらない 2000: 下端が求まらない 3000: 左端が求まらない 4000: 右端が求まらない

(3) 計算法

画像配列 I A を左上から走査して、はじめてしきい値以上となる行を求めて、画像の上端の位置(JT)とする。

他の端についても同様であり、走査の方向だけが異なる。

(4) 子ルーチン

なし

N S I Z E B

Object Size for Binary Picture

2 値文字画像の大きさ（左端、右端、上端、下端）を求める

作成	吉村ミツ	1985年 7月
形式	サブルーチン	言語:FORTRAN サイズ: 64行

(1) 概要

2 値文字画像の大きさ（左端、右端、上端、下端）を求める。

(2) 使用法

CALL NSIZEB(IA, I1, J1, IL, IR, JT, JB, ILL)

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	入力文字画像の配列 大きさは (I1, J1)
I1	整数型	入力	IAの整合寸法 (第1添字)
J1	整数型	入力	IAの整合寸法 (第2添字)
IL	整数型	出力	文字画像左端の位置
IR	整数型	出力	文字画像右端の位置
JT	整数型	出力	文字画像上端の位置
JB	整数型	出力	文字画像下端の位置
ILL	整数型	出力	0: 正常終了 900, 901, 902, 903 : 文字画像がない時(IA全て0)

(3) 計算法

画像配列 I A を左上から走査して、はじめて 1 となる行を求めて、これを画像の上端の位置(JT)とする。

他の端についても同様であり、走査の方向だけが異なる。

(4) 子ルーチン

なし

N S I Z E M

Object Size for Multi-Level Picture

濃淡文字画像の大きさ（左端、右端、上端、下端）をもとめる

作成	吉村ミツ	1985年 7月
形式	サブルーチン	言語：FORTRAN サイズ：65行

(1) 概要

濃淡文字画像の大きさ（左端、右端、上端、下端）をもとめる。

(2) 使用法

CALL NSIZEM(IA, I1, J1, IL, IR, JT, JB, IT, ILL)

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	入力文字画像の配列 大きさは (I1, J1)
I1	整数型	入力	IAの整合寸法 (第1添字)
J1	整数型	入力	IAの整合寸法 (第2添字)
IL	整数型	出力	文字画像左端の位置
IR	整数型	出力	文字画像右端の位置
JT	整数型	出力	文字画像上端の位置
JB	整数型	出力	文字画像下端の位置
IT	整数型	入力	IAの背景を0にするしきい値
ILL	整数型	出力	0: 正常終了

900, 901, 902, 903

: 文字画像がない時 (IA全てが0)

(3) 計算法

画像配列 IA を左上から走査して、はじめてしきい値以上となる行を求めて、画像の上端の位置(JT)とする。

他の端についても同様であり、走査の方向だけが異なる。

(4) 子ルーチン

なし

G C E N T B

Center of Gravity on the Binary Picture

2 値文字画像の重心位置を求める

作成	吉村ミツ	1985年 7月
形式	サブルーチン	言語:FORTRAN サイズ: 52行

(1) 概要

2 値文字画像の重心位置を求める。

(2) 使用法

CALL GCENTB(IA, I1, J1, IGX, IGV, SVX, SVY, ILL)

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	入力文字画像の配列 大きさは (I1, J1)
I1	整数型	入力	IAの整合寸法 (第1添字の値)
J1	整数型	入力	IAの整合寸法 (第2添字の値)
IGX	整数型	出力	横方向の一次モーメント (重心位置のY座標)
IGY	整数型	出力	縦方向の一次モーメント (重心位置のY座標)
SVX	実数型	出力	横方向の二次モーメントの平方根
SVY	実数型	出力	縦方向の二次モーメントの平方根
ILL	整数型	出力	0 : 正常終了

900:文字画像がない時(IA全て0)

(3) 計算法

$$IGX = \frac{\sum \sum (I * IA(I, J))}{M} \quad IGY = \frac{\sum \sum (J * IA(I, J))}{M}$$

$$SVX = \frac{\sum \sum ((I - IGX)^2 * IA(I, J))}{M}$$

$$SVY = \frac{\sum \sum ((J - IGY)^2 * IA(I, J))}{M}$$

ここで、Mは、濃度1の値を持つ点 IA(I, J) の和である。

(4) 子ルーチン

なし

G C E N T M

Center of Gravity on the Multi-Level Picture

濃淡文字画像の重心位置を求める

作成	吉村ミツ	1985年 7月
形式	サブルーチン	言語 : FORTRAN サイズ : 55行

(1) 概要

濃淡文字画像の重心位置を求める。

(2) 使用法

CALL GCENTM(IA, I1, J1, IGX, IGY, SVX, SVY, IT, ILL)

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	入力文字画像の配列 大きさは (I1, J1)
I1	整数型	入力	IAの整合寸法 (第1添字の値)
J1	整数型	入力	IAの整合寸法 (第2添字の値)
IGX	整数型	出力	横方向の1次モーメント (重心位置のX座標)
IGY	整数型	出力	縦方向の1次モーメント (重心位置のY座標)
SVX	実数型	出力	横方向の2次モーメントの平方根
SVY	実数型	出力	縦方向の2次モーメントの平方根
IT	整数型	入力	IAの背景を0にするしきい値

I L L	整数型	出力	0 : 正常終了 900 : 文字画像がない時 (IA 全てが 0)
-------	-----	----	---------------------------------------

(3) 計算法

$$IGX = \frac{\sum \sum (I * IA(I, J))}{M} \quad IGY = \frac{\sum \sum (J * IA(I, J))}{M}$$

$$SVX = \left[\frac{\sum \sum ((I - IGX)^2 * IA(I, J))}{M} \right]^{0.5}$$

$$SVY = \left[\frac{\sum \sum ((J - IGY)^2 * IA(I, J))}{M} \right]^{0.5}$$

ここで、Mは、しきい値 IT 以上の値を持つ IA(I, J) の和である。

(4) 子ルーチン

なし

N T R N S 1

Transformation of Picture Pattern NO. 1

文字画像の伸縮と位置の変換を行なう（その1）

作成	吉村ミツ	1985年 7月
形式	サブルーチン	言語:FORTRAN サイズ: 30行

(1) 概要

2値文字画像の伸縮と位置の変換を行なう。

(2) 使用法

```
CALL NTRNS1(IA, I1, J1, IB, I2, J2, C11, IL, C21, JU)
```

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	入力文字画像の配列 大きさは (I1, J1)
I1	整数型	入力	IAの整合寸法 (第1添字)
J1	整数型	入力	IAの整合寸法 (第2添字)
IB	整数型 2次元配列	出力	出力文字画像の配列 大きさは (I2, J2)
I2	整数型	入力	IBの整合寸法 (第1添字)
J2	整数型	入力	IBの整合寸法 (第2添字)
C11	実数型	入力	変換係数
IL	整数型	入力	伸縮を行う左端の座標

C21	実数型	入力	変換係数
J U	整数型	入力	伸縮を行う上端の座標

(3) 計算法

変換係数に基づき変換する。

$$IB(I, J) = IA(ID, JD)$$

$$\begin{aligned} \text{ここで } ID &= C11 \times (I - 1) + IL + 0.5 \\ JD &= C21 \times (J - 1) + JU + 0.5 \end{aligned}$$

$$I = 1, \dots, I2$$

$$J = 1, \dots, J2$$

(4) 子ルーチン

なし

N T R N S 2

Transformation of Picture Pattern No. 2

2 値文字画像の伸縮と位置の変換を行う（その 2）

作成	吉村ミツ	1985年 7月
形式	サブルーチン	言語 : FORTRAN サイズ : 30 行

(1) 概要

2 値文字画像の伸縮と位置の変換を行う。

(2) 使用法

CALL NTRNS2(IA, I1, J1, IB, I2, J2, C11, C12, C21, C22)

引数	型と種類	属性	内容
IA	整数型 2 次元配列	入力	入力文字画像の配列 大きさは (I1, J1)
I1	整数型	入力	IA の整合寸法 (第 1 添字)
J1	整数型	入力	IA の整合寸法 (第 2 添字)
IB	整数型 2 次元配列	出力	出力文字画像の配列 大きさは (I2, J2)
I2	整数型	入力	IB の整合寸法 (第 1 添字)
J2	整数型	入力	IB の整合寸法 (第 2 添字)
C11	実数型	入力	変換係数
C12	実数型	入力	変換係数

C 2 1	実数型	入力	変換係数
C 2 2	実数型	入力	変換係数

(3) 計算法

変換係数に基づき変換する。

$$IB(I, J) = IA(ID, JD)$$

$$\text{ここで } ID = C11 \times I + C12$$

$$JD = C21 \times J + C22$$

$$I = 1, 2, \dots, I2$$

$$J = 1, 2, \dots, J2$$

(4) 子ルーチン

なし

N T R N S 3

Transformation of Binary Image No. 3

2 値文字画像の縦横比を保持する位置・大きさの正規化

作成	竹下鉄夫	1988年 9月
形式	サブルーチン	言語 : FORTRAN サイズ : 80行

(1) 概要

縦横比を保ちながら 2 値文字画像を位置・大きさの正規化をするサブルーチン NORM7 の孫ルーチン、縦横比を保って実際の変換をしているサブルーチン。

(2) 使用法

```
CALL NTRNS3(IA, I1, J1, IB, I2, J2, IL, IR, JD)
```

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	入力文字画像の 2 次元配列 大きさは IA (I1, J1)
I1	整数型	入力	IA の整合寸法 (第 1 添字)
J1	整数型	入力	IA の整合寸法 (第 2 添字)
IB	整数型 2次元配列	出力	出力文字画像の 2 次元配列 大きさは IB (I2, J2)
I2	整数型	入力	IB の整合寸法 (第 1 添字)
J2	整数型	入力	IB の整合寸法 (第 2 添字)
IL	整数型	入力	入力文字画像の左端
IR	整数型	入力	入力文字画像の右端

J U	整数型	入力	入力文字画像の上端
J D	整数型	入力	入力文字画像の下端

(3) 計算法

入力文字画像の大きさを横 I (= I R - I L), 縦 J (= J D - J U) (但し I R, I L, J U, J D はサブルーチン N S I Z E B で使用している引数と同じでそれぞれ、I R : 右端, I L : 左端, J U : 上端, J D : 下端である) と計算し、縦横の縮尺は次のように求める。

$$\text{横の縮尺} = I / (I 2 - 1)$$

$$\text{縦の縮尺} = J / (J 2 - 1)$$

二つの縮尺の内、小さい方の縮尺を選んでその数値で縦横ともに変換する。以上の説明では縮小する場合のみを説明しているが大きく拡大する場合でも同様である。

実際の変換は、変換後の座標が変換前のどの座標に相当するか各点について計算しながら大きさの変換を行う。

NORMB1

Normalize Binary Image No.1

2値文字画像の枠合わせによる位置、大きさの正規化

作成	吉村ミツ	1985年 7月
形式	サブルーチン	言語:FORTRAN サイズ: 29行

(1) 概要

2値文字画像の外接長方形の中心を枠の中心に合わせて位置、大きさの正規化をする。

(2) 使用法

CALL NORMB1(IA, I1, J1, IB, I2, J2, ILL)

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	入力文字画像の配列 大きさは (I1, J1)
I1	整数型	入力	IAの整合寸法 (第1添字の値)
J1	整数型	入力	IAの整合寸法 (第2添字の値)
IB	整数型 2次元配列	出力	出力文字画像の配列 大きさは (I2, J2)
I2	整数型	入力	IBの整合寸法 (第1添字の値)
J2	整数型	入力	IBの整合寸法 (第2添字の値)
ILL	整数型	出力	0 : 正常終了 900 : 文字画像がない時(IA全て0)

(3) 計算法

2 値文字画像の大きさを求めた上で、以下に示す過程で、伸縮と位置の移動をする。

```
CALL NSIZEB(IA, I1, J1, IL, IR, JT, JB, ILL)
C11 = (IR - IL) / I2
C21 = (JT - JB) / J2
CALL NTRNS1(IA, I1, J1, IB, I2, J2, C11, IL, IL, C21, JT)
```

(4) 子ルーチン

NSIZEB (画像の外接長方形を求める)

NTRNS1 (画像の伸縮と位置の変換をする)

NORMB 2

Normalize Binary Image NO. 2

2 値文字画像の重心合せによる位置、大きさの正規化

作成 吉村ミツ 1985年 7月

形式 サブルーチン 言語 : FORTRAN サイズ : 31行

(1) 概要

2 値文字画像の重心を中心にして、縦横 2 次モーメントの平方根大の部分を与えた大きさに伸縮する。

(2) 使用法

CALL NORMB2(IA, I1, J1, IB, I2, J2, CX, CY, ILL)

引数	型と種類	属性	内容
IA	整数型 2 次元配列	入力	入力文字画像の配列 大きさは (I1, J1)
I1	整数型	入力	IA の整合寸法 (第 1 添字の値)
J1	整数型	入力	IA の整合寸法 (第 2 添字の値)
IB	整数型 2 次元配列	出力	出力文字画像の配列 大きさは (I2, J2)
I2	整数型	入力	IB の整合寸法 (第 1 添字の値)
J2	整数型	入力	IB の整合寸法 (第 2 添字の値)
CX	実数型	入力	横方向の切り出し係数
CY	実数型	入力	縦方向の切り出し係数

I L L	整数型	出力	0 : 正常終了 900 : 文字画像がない時 (IA全てが 0)
-------	-----	----	---

(3) 計算法

2値文字画像の重心を求めた上で、次に示すような伸縮変換をする。

```
CALL GCENTB(IA, I1, J1, IGX, IGY, SVX, SVY, ILL)
IXC = I2 ÷ 2
IYC = J2 ÷ 2
C11 = 2 × CX × SVX ÷ I2
C21 = 2 × CY × SVY ÷ J2
C12 = IGX - C11 × IXC
C22 = IGY - C21 × IYC
CALL NTRNS2(IA, I1, J1, IB, I2, J2, C11, C12, C21, C22)
```

(4) 子ルーチン

G C E N T B (2値文字画像の重心を求める)

N T R N S 2 (伸縮と位置の移動をする)

NORMB 3

Normalize Binary Image No. 3

2 値画像の外接枠・重心合わせによる位置・大きさの正規化

作成	鶴岡信治	1985年 7月
形式	サブルーチン	言語: FORTRAN サイズ: 75行

(1) 概要

文字の外接長方形と文字の重心を求め、外接長方形の中心に文字の重心が位置するように位置・大きさの正規化をする。

(2) 使用法

CALL NORMB3(IA, I1, J1, IB, I2, J2, ILL)

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	正規化前の2値画像
I 1	整数型	入力	IAの整合寸法（第1添字の値）
J 1	整数型	入力	IAの整合寸法（第2添字の値）
IB	整数型 2次元配列	出力	位置・大きさの正規化済の画像
I 2	整数型	入力	IBの整合寸法（第1添字の値）
J 2	整数型	入力	IBの整合寸法（第2添字の値）
ILL	整数型	出力	0 : 正常終了 0以外 : 図の大きさが求まらない

(3) 計算法

入力画像IAにおいて文字部分の外接長方形、重心を求め、出力画像の中心に文字の重心がくるように、4つの小領域で別々に縦、横の伸縮率を計算し、位置・大きさの正規化をする。

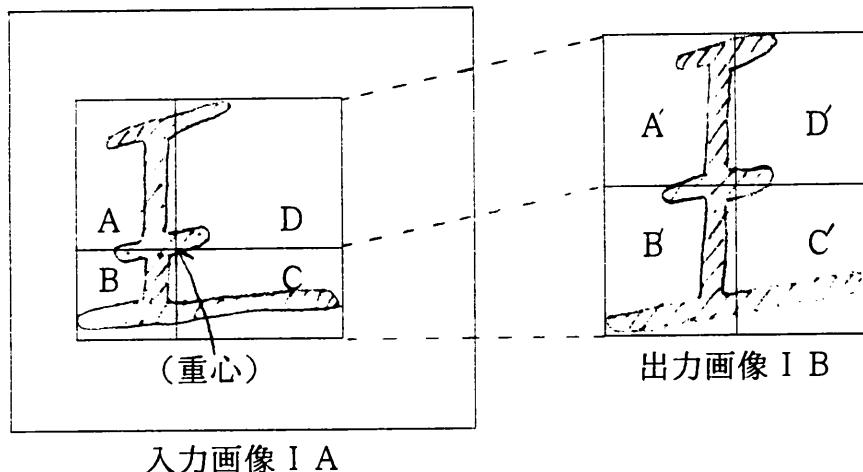


図1 位置・大きさの正規化

(4) 子ルーチン

N S I Z E B (入力2値画像の外接長方形を求める)

G C E N T B (入力2値画像の重心座標を求める)

(5) 備考

従来から外接枠合わせ、重心合わせの正規化法はよく用いられているが、本方法は三重大学で開発された方法であり、両者の中間の性質を示すと考えられる。特に文字線分の飛び出し、つき出し等について有効と考えられる。

NORMB4

Normalize Binary Image N0.4

2値画像の外接枠合わせによる位置・大きさの正規化（小領域の平均値）

作成 鶴岡信治 1987年 9月

形式 サブルーチン 言語 : FORTRAN サイズ : 57行

(1) 概要

文字の外接長方形を求めて、縦、横独立に伸縮率を求め、小領域の平均値を出力画像の点の値とするので、縮小率が大きいときに滑らかな変換が行える。

(2) 使用法

CALL NORMB4(IA, I1, J1, IB, I2, J2, ILL)

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	入力2値画像 IA(I1, J1)
I1	整数型	入力	IAの整合寸法（第1添字の値）
J1	整数型	入力	IAの整合寸法（第2添字の値）
IB	整数型 2次元配列	出力	出力2値画像 IB(I2, J2)
I2	整数型	入力	IBの整合寸法（第1添字の値）
J2	整数型	入力	IBの整合寸法（第2添字の値）
ILL	整数型	出力	0:正常終了 900, 901, 902, 903: IAがすべて0のとき

(3) 計算法

まず、文字領域の外接長方形枠を求める(NSIZEB)。次に、枠の横の長さと出力画像の横の長さの比（横の縮小率）と、枠の縦の長さと出力画像の縦の長さの比（縦の縮小率）を求め、入力画像を出力画像の画素数の小領域に分割し、小領域内の画素の濃度値の平均値を求め、それが0.5以上のときは1、0.5未満のときは0を出力画像の濃度値とする。

出力画像1点と入力画像の小領域を対応づけて縮小しているので、計算時間はかかるが、縮小率が1.5以上のときには滑らかな画像を得ることができる。

(4) 子ルーチン

NSIZEB (文字の外接長方形枠の大きさを求める)

NORMB7

Normalize Binary Image No. 7

2値文字画像の位置・大きさの正規化（縦横比を保持）

作成	竹下 鉄夫	1987年 9月
形式	サブルーチン	言語 : FORTRAN サイズ : 25行

(1) 概要

2値文字画像を縦横比を保ちながら位置・大きさの正規化をする。出力する枠の中心と文字の中心を一致させる。縦または横のサイズのうち大きい方で決まる比率で全体を縮小または拡大する。したがって、縦方向が枠一杯に正規化されるときには横方向は両サイドに空白が生じる。反対に横方向が枠一杯に正規化されるときは縦方向に上下に空白の部分が生じる。

(2) 使用法

```
CALL NORMB7(IA, I1, J1, IB, I2, J2, ILL)
```

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	入力文字画像の2次元配列 大きさはIA(I1, J1)
I1	整数型	入力	IAの整合寸法（第1添字の値）
J1	整数型	入力	IAの整合寸法（第2添字の値）
IB	整数型 2次元配列	出力	出力文字画像の2次元配列 大きさはIB(I2, J2)
I2	整数型	入力	IBの整合寸法（第1添字の値）
J2	整数型	入力	IBの整合寸法（第2添字の値）

I L L	整数型	出力	I L L = 0 : 正常終了 I L L ≠ 0 : 入力文字画像がオール 0
-------	-----	----	--

(3) 計算法

入力文字画像の大きさを横 I (= I R - I L), 縦 J (= J D - J U) (但し I R, I L, J U, J D はサブルーチン N S I Z E B で使用している引数と同じでそれぞれ、I R : 右端, I L : 左端, J U : 上端, J D : 下端である) と計算し、縦横の縮尺は次のように求める。

$$\text{横の縮尺} = I / (I_2 - 1)$$

$$\text{縦の縮尺} = J / (J_2 - 1)$$

二つの縮尺の内、小さい方の縮尺を選んでその数値で縦横ともに変換する。以上の説明では縮小する場合のみを説明しているが大きく拡大する場合でも同様である。

(4) 子ルーチン

N S I Z E B (外接長方形の大きさを求める)

N T R N S 3 (入力文字の縦横比を変えないように、伸縮と位置の移動をする)

(5) 注意

本サブルーチンは孫ルーチンとして N S I Z E B (大きさの測定ルーチン), N T R N S 3 (サイズ変換ルーチン) を含む。

(6) 使用例 (サンプルプログラム)

(a) 市松模様を生々し、その後周囲にゼロを付加するサブルーチン M S W A K U を使い不要なゼロをつける。しかる後、N O R M B 7 サブルーチンを呼び縦横比を保持する正規化を実行している。実行されている様子を印刷サブルーチン P O U T 2 B を使って確認している。

(b) 使用法

ジョブ制御文のデータセットが EXAMPLE. CNTL(NORMB7) に用意されているので SUBMITすればよい。

プログラム本体は EXAMPLE. FORT(NORMB7) に格納されている。

(c) 使用しているサブルーチン

N O R M B 7, P O U T 2 B, M S W A K U

NORMM1

Normalize Multi-Level Image No. 1

濃淡文字画像の枠合わせによる位置、大きさの正規化

作成	吉村ミツ	1985年 7月
形式	サブルーチン	言語:FORTRAN サイズ: 30行

(1) 概要

濃淡文字画像の外接長方形の中心を枠の中心に合わせて位置、大きさの正規化をする。

(2) 使用法

```
CALL NORMM1(IA, I1, J1, IB, I2, J2, IT, ILL)
```

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	入力文字画像の配列 大きさは (I1, J1)
I1	整数型	入力	IAの整合寸法 (第1添字の値)
J1	整数型	入力	IAの整合寸法 (第2添字の値)
IB	整数型 2次元配列	出力	出力文字画像の配列 大きさは (I2, J2)
I2	整数型	入力	IBの整合寸法 (第1添字の値)
J2	整数型	入力	IBの整合寸法 (第2添字の値)
IT	整数型	入力	IAの背景を0にする数値
ILL	整数型	出力	0: 正常終了

		900 : 文字画像がない時(IA全て0)
--	--	-----------------------

(3) 計算法

文字画像の大きさを求めた上で伸縮する

```
CALL NSIZEM(IA, I1, J1, IL, IR, JT, JB, ILL)
C11 = (IR - IL) / I2
C21 = (JT - JB) / J2
CALL NTRNS1(IA, I1, J1, IB, I2, J2, C11, IL, IL, C21, JT)
```

(4) 子ルーチン

N S I Z E M (画像の外接長方形を求める)

N T R N S 1 (画像の伸縮と位置の変換をする)

NORMM2

Normalize Multi-Level Image NO. 2

濃淡文字画像の重心合せによる位置、大きさの正規化

作成	吉村ミツ	1985年 7月
形式	サブルーチン	言語 : FORTRAN サイズ : 32行

(1) 概要

濃淡文字画像の重心を中心にして、縦横2次モーメントの平方根大の部分を与えられた大きさに伸縮する。

(2) 使用法

CALL NORMM2(IA, I1, J1, IB, I2, J2, CX, CY, IT, ILL)

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	入力文字画像の配列 大きさは (I1, J1)
I1	整数型	入力	IAの整合寸法 (第1添字の値)
J1	整数型	入力	IAの整合寸法 (第2添字の値)
IB	整数型 2次元配列	出力	出力文字画像の配列 大きさは (I2, J2)
I2	整数型	入力	IBの整合寸法 (第1添字の値)
J2	整数型	入力	IBの整合寸法 (第2添字の値)
CX	実数型	入力	横方向の切り出し係数
CY	実数型	入力	縦方向の切り出し係数

I T	整数型	入力	I Aを2値化するしきい値
I L L	整数型	出力	0 : 正常終了 900 : 文字画像がない時 (I A 全てが 0)

(3) 計算法

文字画像の重心を求めた上で次に示すように伸縮する。

```
CALL GCENTM(IA, I1, J1, IGX, IGY, SVX, SVY, IT, ILL)
```

$$IXC = I2 \div 2$$

$$IYC = J2 \div 2$$

$$C11 = 2 \times CX \times SVX \div I2$$

$$C21 = 2 \times CY \times SVY \div J2$$

$$C12 = IGX - C11 \times IXC$$

$$C22 = IGY - C21 \times IYC$$

```
CALL NTRNS2(IA, I1, J1, IB, I2, J2, C11, C12, C21, C22)
```

(4) 子ルーチン

G C E N T M (2 値文字画像の重心を求める)

N T R N S 2 (伸縮と位置の移動をする)

NORMM3

Normalize Multi-Level Image No. 3

多値画像の外接重心合わせによる位置・大きさの正規化

作成	鶴岡信治	1985年 7月
形式	サブルーチン	言語: FORTRAN サイズ: 75行

(1) 概要

文字の外接長方形と文字の重心を求め、外接長方形の中心に文字の重心が位置するように位置・大きさの正規化をする。

(2) 使用法

CALL NORMM3(IA, I1, J1, IB, I2, J2, IT, ILL)

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	正規化前の2値画像
I1	整数型	入力	IAの整合寸法（第1添字の値）
J1	整数型	入力	IAの整合寸法（第2添字の値）
IB	整数型 2次元配列	出力	位置・大きさの正規化済の画像
I2	整数型	入力	IBの整合寸法（第1添字の値）
J2	整数型	入力	IBの整合寸法（第2添字の値）
IT	整数型	入力	IAを2値化するしきい値
ILL	整数型	出力	0 : 正常終了

0以外：図の大きさが求まらない

(3) 計算法

入力画像IAにおいて文字部分の外接長方形、重心を求め、出力画像の中心に文字の重心がくるように、4つの小領域で別々に縦、横の伸縮率を計算し、位置・大きさの正規化をする。

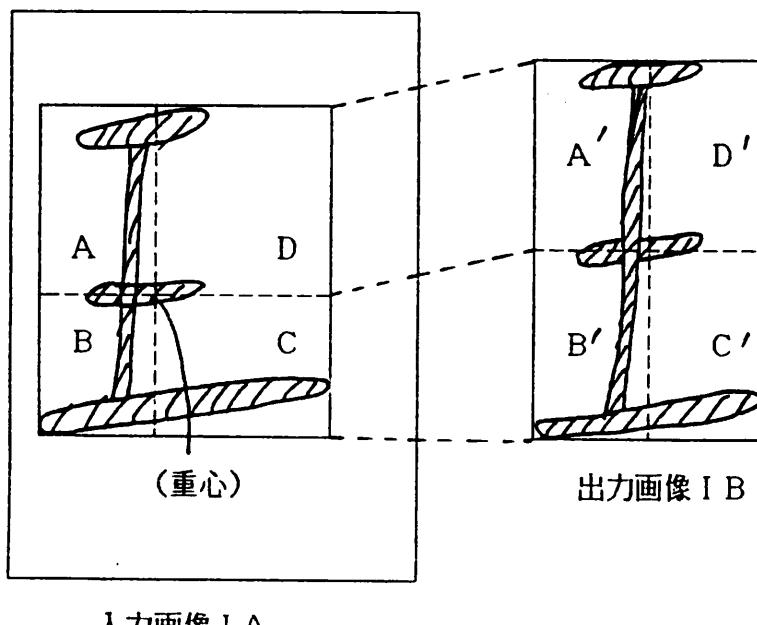


図1 位置・大きさの正規化

(4) 子ルーチン

N S I Z E M (外接長方形を求める)

G C E N T M (重心座標を求める)

(5) 備考

従来から外接枠合わせ、重心合わせの正規化法はよく用いられているが、本方法は三重大学で開発された方法であり、両者の中間の性質を示すと考えられる。特に文字線分の飛び出し、つき出し等について有効と考えられる。

NORMNL

Normalize Using the Non Linear Method

2値画像の線密度イコライゼーションによる非線形正規化法

作成	鶴岡信治, 扇田直樹	1990年 8月
形式	サブルーチン	言語: FORTRAN サイズ: 65行

(1) 概要

縦、横独立に文字線の本数の射影を求め、それと標本化間隔との積が一定になる
ように変換する。

(2) 使用法

```
CALL NORMNL(IA, I1, J1, IB, I2, J2, HX, HY, NX, NY, IX, IY)
```

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	入力2値画像 IA(I1, J1)
I1	整数型	入力	IAのX方向の大きさ
J1	整数型	入力	IAのY方向の大きさ
IB	整数型 2次元配列	出力	出力2値画像 IB(I2, J2)
I2	整数型	入力	IBのX方向の大きさ
J2	整数型	入力	IBのY方向の大きさ
HX	整数型 1次元配列	出力	X方向の線の本数の射影 HX(I1)

H Y	整数型 1 次元配列	出力	Y 方向の線の本数の射影 H Y (J 1)
N X	整数型	出力	H X の総和
N Y	整数型	出力	H Y の総和
I X	整数型 1 次元配列	出力	X 方向の新座標と旧座標の対応 I X (I 2)
I Y	整数型 1 次元配列	出力	Y 方向の新座標と旧座標の対応 I Y (J 2)

(3) 参考文献

- (i) 山田博三, 斎藤泰一, 山本和彦：“線密度イコライゼーション－相関法のための非線形正規化法－”，電子情報通信学会論文誌（D），J 67-D, 11(1984).

NORMN 2

Normalize Using the Non Linear Method NO. 2

2 値画像の線密度イコライゼーションによる非線形正規化法（その 2）

作成	鶴岡信治, 原田泰明	1990年 8月
形式	サブルーチン	言語 : FORTRAN サイズ : 159行

(1) 概要

縦、横独立に文字線の本数の射影を求め、それと標本化間隔との積が一定になる
ように変換する。その 1 の方法より、適切な非線形正規化ができる。

(2) 使用法

CALL NORMN2(IA, I1, J1, IB, I2, J2, HX, HY, NX, NY, IX, IY)

引数	型と種類	属性	内容
IA	整数型 2 次元配列	入力	入力 2 値画像 IA (I 1, J 1)
I 1	整数型	入力	IA の X 方向の大きさ
J 1	整数型	入力	IA の Y 方向の大きさ
IB	整数型 2 次元配列	出力	出力 2 値画像 IB (I 1, J 1)
R H X	実数型 1 次元配列	出力	X 方向の線の本数の射影 R H X (I 1)
R H Y	実数型 1 次元配列	出力	Y 方向の線の本数の射影 R H Y (J 1)
R N X	実数型	出力	H X の総和

R N Y	実数型	出力	H Yの総和
I X	整数型 1次元配列	出力	X 方向の新座標と旧座標の対応 I X (I 1)
I Y	整数型 1次元配列	出力	Y 方向の新座標と旧座標の対応 I Y (J 1)
L X	整数型 1次元配列	出力	X 方向の新座標と旧座標の対応 L X (I 1, J 1)
L Y	整数型 1次元配列	出力	Y 方向の新座標と旧座標の対応 L Y (I 1, J 1)
R H 0	実数型 2次元配列	出力	各画素の線密度 R H 0 (I 1, J 1)

(3) 参考文献

- (i) 山田博三, 齊藤泰一, 山本和彦：“非線形正規化法の改良”，昭和63年度電子情報通信学会春期全国大会, D-439(1988).

C A N C A L

Canonicalization

正純化操作を施す

作成 吉村ミツ	1985年 7月
形式 サブルーチン	言語:FORTRAN サイズ: 27行

(1) 概要

原点を通る長さ 1 のベクトルにする。

(2) 使用法

CALL CANCEL(A, IS, B)

引数	型と種類	属性	内容
A	実数型 1次元配列	入力	1次元ベクトル 大きさは (IS)
IS	整数型	入力	A, B の添字
IB	実数型 1次元配列	出力	1次元ベクトル 大きさは (IS)

(3) 計算法

$$IB(I) = (IA(I) - m) / \sigma$$

$$m = \sum IA(I) / IS, \quad \sigma = \{\sum (IA(I)^2 - m)\}^{0.5}$$

WE I G T 1

Weighting Filter 1

画像データの第2標本化

作成	山川洋一郎、掛布英辰	1985年 7月
形式	サブルーチン	言語: FORTRAN サイズ: 116行

(1) 概要

各種重み係数により画像データの第2標本化を行う。

(2) 使用法

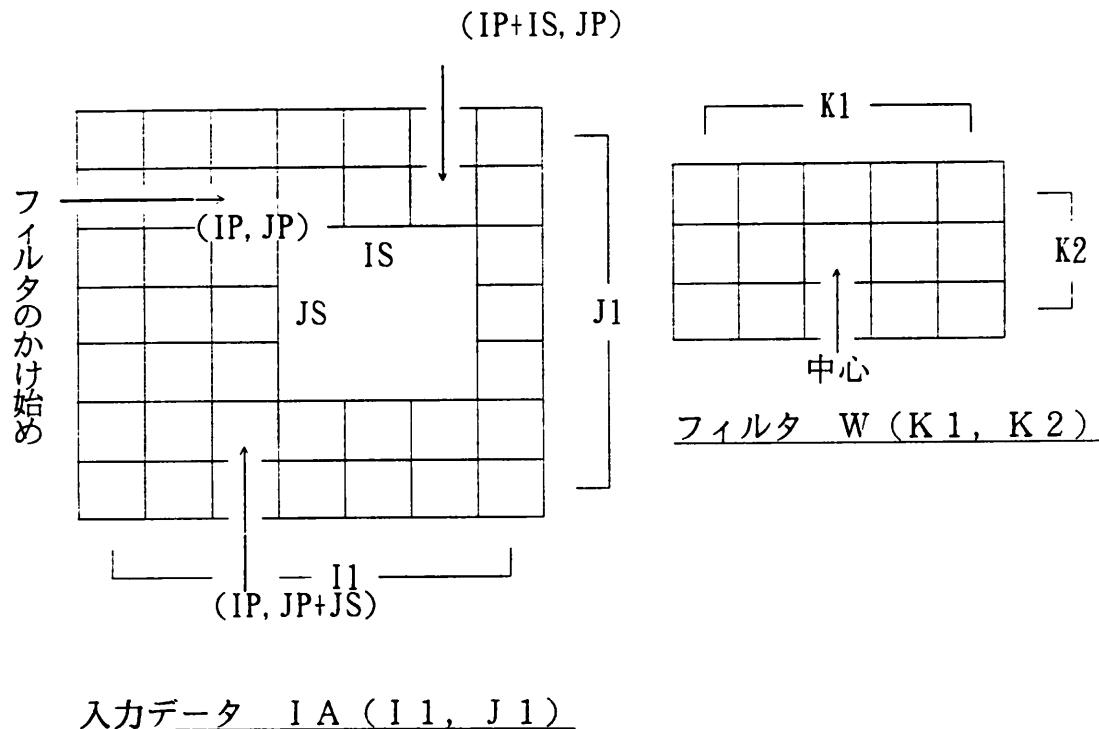
CALL WEIGT1(IA, I1, J1, B, I2, J2, W, K1, K2, IP, JP, IS, JS, ILL)

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	処理を行う画像
I 1	整数型	入力	IAの整合寸法 (第1添字の値)
J 1	整数型	入力	IAの整合寸法 (第2添字の値)
B	実数型 2次元配列	出力	出力画像
I 2	整数型	入力	Bの整合寸法 (第1添字の値)
J 2	整数型	入力	Bの整合寸法 (第2添字の値)
W	実数型 2次元配列	入力	重み係数
K 1	整数型	入力	Wの整合寸法 (第1添字の値)

K 2	整数型	入力	Wの整合寸法（第2添字の値）
I P	整数型	入力	フィルターの横方向スタートポイント (フィルターの中心の位置)
J P	整数型	入力	フィルターの縦方向スタートポイント (フィルターの中心の位置)
I S	整数型	入力	横方向のサンプリング間隔
J S	整数型	入力	縦方向のサンプリング間隔
I L L	整数型	出力	I L L = 0 : 正常終了 I L L = 30000 : I 2, J 2 が不適当

(3) 計算法

フィルタのかけ始めの座標およびサンプリング間隔を与えることによりサンプリングポイントを計算し、フィルタをかけることにより、第2標本化を行う。



(4) 子ルーチン
なし

(5) 備考

出力画像の整合寸法 I_2, J_2 は、あらかじめ (1) 式により求めておく必要がある。

$$I_2 = (I_1 - IP) / IS + 1 \quad] \quad (1)$$

$$J_2 = (J_1 - JP) / JS + 1 \quad]$$

I_{LL} の判定式

$I_{LL} = 30000 \quad I_2, J_2$ が (1) 式を満足しない。

WE I G T 2

Weighting Filter 2

任意のWeighting Filter (重み係数) により第2標本化を行う

作成	竹下鉄夫	1985年	6月
形式	サブルーチン	言語 : FORTRAN	サイズ : 117行

(1) 概要

入力2次元配列IA(I1, J1)に対してスタートポイント, サンプリング間隔を指定し, 任意のWeighting Filter (重み係数) により第2標本化を行う
出力配列の大きさはB(I2, J2)である。

類似のサブルーチンWEIGT1が入力配列からWeighting Filterがはみ出した場合も第2標本化を行うのに対して, このWEIGT2は入力配列からWeighting Filterがはみ出た時には第2標本化をしない。
(詳細はパラメータの必要条件の項参照)

(2) 使用法

```
CALL WEIGT2(IA, I1, J1, B, I2, J2, W, K1, K2, IP, JP, IS, JS, ILL)
```

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	入力データの入っている2次元配列
I1	整数型	入力	IAの整合寸法 (第1添字の寸法)
J1	整数型	入力	IAの整合寸法 (第2添字の寸法)
B	実数型 2次元配列	出力	出力データの入る2次元配列
I2	整数型	入力	Bの整合寸法 (第1添字の寸法)

J 2	整数型	入力	Bの整合寸法（第2添字の寸法）
W	実数型 2次元配列	入力	Weighting Filter（重み係数）の入っている2次元配列
K 1	整数型	入力	Wの整合寸法（第1添字の寸法）
K 2	整数型	入力	Wの整合寸法（第2添字の寸法）
I P	整数型	入力	フィルターの横方向スタートポイント
J P	整数型	入力	フィルターの縦方向スタートポイント
I S	整数型	入力	横方向のサンプリング間隔
J S	整数型	入力	縦方向のサンプリング間隔
I L L	整数型	出力	I L L = 0 : 正常終了 I L L = 3 0 0 0 0 : パラメータが不適当。この時には計算しない, B = 0. 0

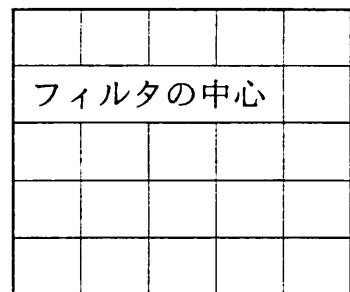
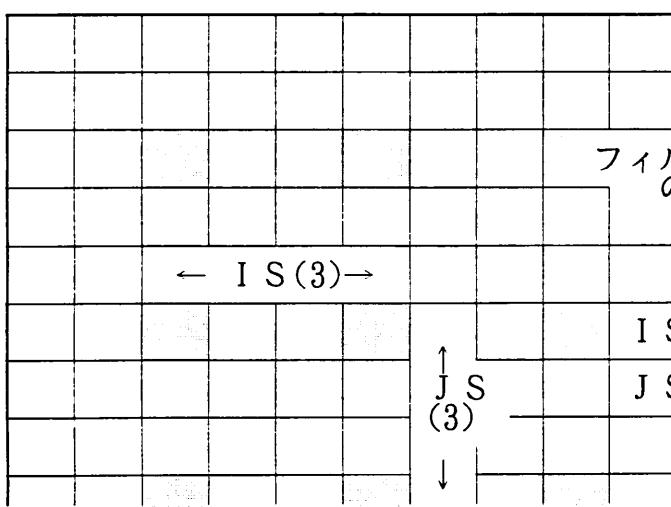
(a) K₁ または K₂ が奇数の場合

入力 I A (I 1, J 1)

Weighting Filter K₁=K₂=5

スタートポイント I P(3)

J P
(3)
→



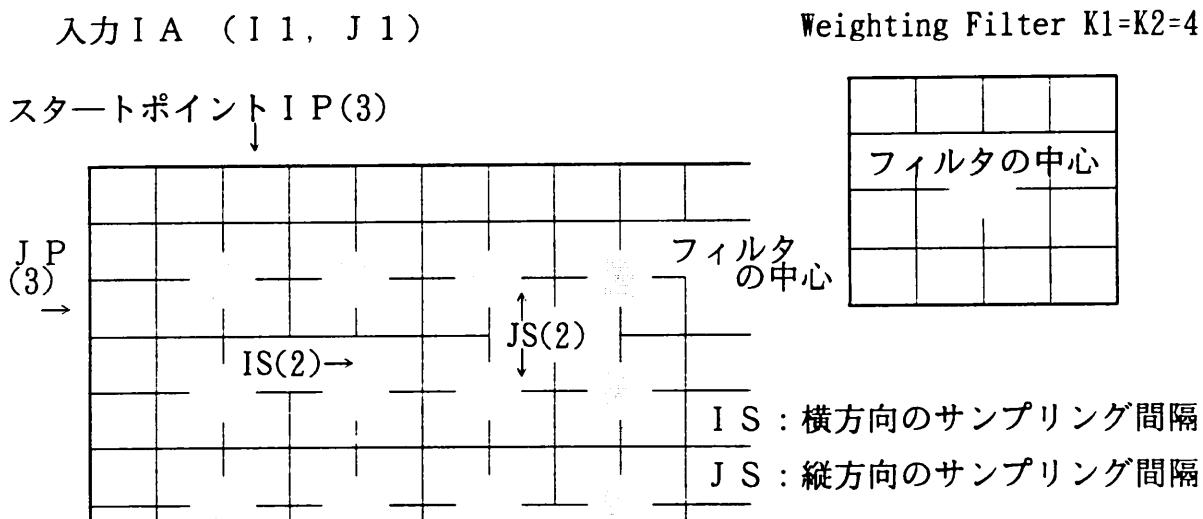
フィルタ
の中心

I S : 横方向のサンプリング間隔

J S : 縦方向のサンプリング間隔

↑
J S
(3)
↓

(b) K 1 または K 2 が偶数の場合



K 1 または K 2 が偶数の場合は上図の様にフィルターの中心が配列要素の上にこないため、スタートポイントは1つずれた点で定義する。

(3) パラメータの必要条件

(a) フィルターとサンプリング間隔の関係

1. 1 $K_1 \geq I_S$ (横方向)

1. 2 $K_2 \geq J_S$ (縦方向)

(b) フィルターサイズとスタートポイントの関係

2. 1 $I_P > K_1 / 2$ (フィルターが入力の左側に、はみ出さない)

2. 2 $J_P > K_2 / 2$ (フィルターが入力の上側に、はみ出さない)

(c) 入力の大きさの条件

3. 1 $I_1 \geq I_P + (I_2 - 1) * I_S + (K_1 - 1) / 2$

(フィルターが入力の右側に、はみ出さない)

3. 2 $J_1 \geq J_P + (J_2 - 1) * J_S + (K_2 - 1) / 2$

(フィルターが入力の下側に、はみ出さない)

(4) 使用例

上記のサブルーチンのテストを行うためのプログラム及び、結果は P O U T 2 B の場合と同じ。

WEI224

Weighting to $2 \times 2 \times 4$

2(横) \times 2(縦) \times 4(方向)への第2標本化処理

作成	鶴岡信治	1990年 9月
形式	サブルーチン	言語: FORTRAN サイズ: 44行

(1) 概要

7×7 の小領域の方向指數ヒストグラム(4方向)を第2標本化し、 2×2 の標本点の加重方向指數ヒストグラムを求める。

(2) 使用法

CALL WEI224(IA, OB, W)

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	方向指數ヒストグラム 大きさは(7, 7, 4)
OB	実数型 2次元配列	出力	加重方向指數ヒストグラム 大きさは(2, 2, 4)
W	実数型 2次元配列	出力	重み関数の入る配列 大きさは(13, 13)

(3) 計算法

重み関数を用いて、7（横）×7（縦）×4（方向）の方向指數ヒストグラムを第2標本化し、2（横）×2（縦）×4（方向）の加重方向指數ヒストグラムを求める。

このサブルーチンを呼ぶ前に、サブルーチンW G T I N 2を呼んで第2標本化処理の重み関数を求めておくこと。

(4) 参考文献

- (i) 鶴岡信治, 栗田昌徳, 原田智夫, 木村文隆, 三宅康二
：“加重方向指數ヒストグラム法による手書き漢字・ひらがな認識”
電子情報通信学会論文誌（D），J 7 0 - D，7, pp. 1390-1397(1987).

WEI444

Weighting to $4 \times 4 \times 4$

4(横) \times 4(縦) \times 4(方向)への第2標本化処理

作成	鶴岡信治	1990年 9月
形式	サブルーチン	言語: FORTRAN サイズ: 31行

(1) 概要

7×7 の小領域の方向指數ヒストグラム(4方向)を第2標本化し、 4×4 の標本点の加重方向指數ヒストグラムを求める。

(2) 使用法

CALL WEI444(IA, OB, W)

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	方向指數ヒストグラム 大きさは(7, 7, 4)
OB	実数型 2次元配列	出力	加重方向指數ヒストグラム 大きさは(4, 4, 4)
W	実数型 2次元配列	出力	重み関数の入る配列 大きさは(5, 5)

(3) 計算法

重み関数を用いて、7（横）×7（縦）×4（方向）の方向指數ヒストグラムを第2標本化し、4（横）×4（縦）×4（方向）の加重方向指數ヒストグラムを求める。

このサブルーチンを呼ぶ前に、サブルーチンW G T I N 3を呼んで第2標本化処理の重み関数を求めておくこと。

(4) 参考文献

- (i) 鶴岡信治, 栗田昌徳, 原田智夫, 木村文隆, 三宅康二
：“加重方向指數ヒストグラム法による手書き漢字・ひらがな認識”
電子情報通信学会論文誌(D), J70-D, 7, pp. 1390-1397(1987).

W G T I N 1

Weight in NO. 1

重み関数を求める（その1）

作成	吉村ミツ	1985年 7月
形式	サブルーチン	言語:FORTRAN サイズ: 32行

(1) 概要

重み関数を配列の中に入れる。

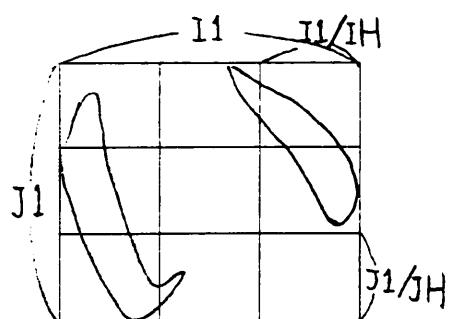
(2) 使用法

CALL WGTIN1(W, K1, L1, ILL)

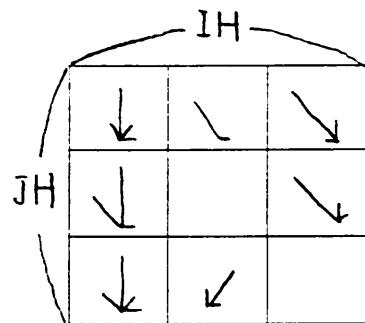
引数	型と種類	属性	内容
W	実数型 2次元配列	出力	重み関数の入る配列 大きさは(5, 5)
K 1	整数型	入力	Wの第1添字 (K 1 = 5)
L 1	整数型	入力	Wの第2添字 (L 1 = 5)
I L L	整数型	出力	0 : 正常終了(K1=L1=5) 3000 : それ以外

(3) 計算法

配列の中に重み関数を入れる。



輪郭点列 (X, Y) 座標



方向指数ヒストグラム

図2 方向指数ヒストグラムの作成

(4) プログラム例

```
PARAMETER(I1=66, J1=65, JBL=1000, IH=7, JH=7)
INTEGER IC(I1, J1), JBKX(JBL), JBLY(JBL), INDEX4(JBL)
INTEGER JHIST(IH, JH, 4)
:
:
CALL BDFL2(IC, IC, I1, J1, JBLX, JBLY, JBL, JBL1, NBD, 8, KERR)
IF(KERR).EQ.-1) WRITE(6,*)'BDFL2 ERR(JBL OVER)'
CALL HISTSS(JBLX, JBLY, INDEX4, JBL, JBL1, IHIST, IH, JH, I1, J1)
WRITE(6,6001) (((IHIST(I, J, K), I=1, IH), K=1, 4)J=1, JH)
6001 FORMAT(1H, 5X, 714, 2X, 714, 2X, 714)
:
:
```

W G T I N 2

Weight in NO. 2

第2標本化のときの重み関数を求める（その2）

作成	鶴岡信治	1990年 9月
形式	サブルーチン	言語：FORTRAN サイズ：49行

(1) 概要

7×7 の小領域の方向指數ヒストグラム（4方向）から第2標本化し、 2×2 の標本点の加重方向指數ヒストグラムを求めるときの重み関数を 13×13 の2次元配列に求める。

(2) 使用法

CALL WGTIN2(W, K1, K2, ILL)

引数	型と種類	属性	内容
W	実数型 2次元配列	出力	重み関数の入る配列 大きさは(13, 13)
K 1	整数型	入力	WのX方向の大きさ(K1=13)
K 2	整数型	入力	WのY方向の大きさ(K2=13)
ILL	整数型	出力	0 : 正常終了(K1=K2=13) 10000 : K 1 が13でない 20000 : K 2 が13でない

(3) 計算法

DATA文で定義してある重み関数の実数値を2次元配列に入れる。

このサブルーチンを呼んだあとに、サブルーチンWE I 2 2 4を呼ぶと第2標準化の処理ができる。

(4) 参考文献

- (i) 鶴岡信治, 栗田昌徳, 原田智夫, 木村文隆, 三宅康二
：“加重方向指数ヒストグラム法による手書き漢字・ひらがな認識”
電子情報通信学会論文誌(D), J70-D, 7, pp. 1390-1397(1987).

W G T I N 3

Weigt in NO.3

第2標本化のときの重み関数を求める（その3）

作成	鶴岡信治	1990年 9月
形式	サブルーチン	言語：FORTRAN サイズ：19行

（1）概要

7×7 の小領域の方向指數ヒストグラム（4方向）から第2標本化し、 4×4 の標本点の加重方向指數ヒストグラムを求めるときの重み関数を 5×5 の2次元配列に求める。

（2）使用法

CALL WGTIN3(W)

引数	型と種類	属性	内容
W	実数型 2次元配列	出力	重み関数の入る配列 大きさは(5, 5)

（3）計算法

DATA文で定義してある重み関数の実数值を2次元配列に入れる。

このサブルーチンを呼んだあとに、サブルーチンWEI444を呼ぶと第2標本化の処理ができる。

（4）参考文献

(i) 鶴岡信治, 栗田昌徳, 原田智夫, 木村文隆, 三宅康二

：“加重方向指數ヒストグラム法による手書き漢字・ひらがな認識”

電子情報通信学会論文誌(D), J70-D, 7, pp. 1390-1397(1987).

W G T I N 4

Weight in NO.4

重み関数を求める（その4）

作成	鶴岡信治	1985年 7月
形式	サブルーチン	言語:FORTRAN サイズ：18行

(1) 概要

重み関数（図1）を
配列の中に入れる。

.0	.0092	.0169	.0092	.0
.0092	.0573	.1054	.0573	.0092
.0169	.1054	.1939	.1054	.0169
.0092	.0573	.1054	.0573	.0092
.0	.0092	.0169	.0092	.0

図1 重み関数

(2) 使用法

CALL WGTIN4(W, K1, K2, ILL)

引数	型と種類	属性	内容
W	実数型 2次元配列	出力	重み関数の入る配列 大きさは(5, 5)
K 1	整数型	入力	Wの第1添字 (K 1 = 5)
K 2	整数型	入力	Wの第2添字 (K 2 = 5)
I L L	整数型	出力	0 : K 1 = 5, K 2 = 5 のとき 1000 : それ以外のとき

(3) 計算法

配列Wに図1の数値を代入する。

HISTSS

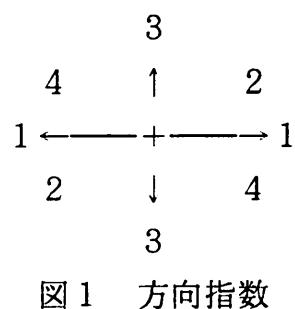
Histogram Sakusei

小領域内の4方向のヒストグラムを作成する

作成	鶴岡信治	1985年 7月
形式	サブルーチン	言語: FORTRAN サイズ: 36行

(1) 概要

輪郭線の点列 (X, Y座標値) が与えられたとき、4方向 (双方向、図1参照) の方向指数を求め、指定された分割数により作られる小領域内での方向指数のヒストグラムを作成する。



(2) 使用法

```
CALL HISTSS(IX, IY, IN, IS, IS1, IHIST, IH, JH, JH, I1, J1)
```

引数	型と種類	属性	内容
I X	整数型 1次元配列	入力	輪郭点列のX座標 (0が輪郭の区切り)
I Y	整数型 1次元配列	入力	輪郭点列のY座標 (0が輪郭の区切り)
I N	整数型 1次元配列	出力	各輪郭点の方向指数
I S	整数型	入力	I X, I Y, I Nの整合寸法
I S1	整数型	入力	I X, I Y, I Nの有効範囲

I H I S T	整数型 3次元配列	出力	各小領域の4方向の方向指数ヒストグラム (I H, J H, 4)
I H	整数型	入力	X方向の小領域の数
J H	整数型	入力	Y方向の小領域の数
I 1	整数型	入力	画像のX方向の大きさ
J 1	整数型	入力	画像のY方向の大きさ

(3) 計算法

まず輪郭点列 (I X, I Y) の各点について4方向の方向指数を1次元配列 I N に求め、小領域の数 (I H, J H) と画像の大きさ (I 1, J 1) により、小領域の大きさを計算し (I 1 / I H, J 1 / J H) 、各小領域における方向指数のヒストグラムを作成する。I X, I YについてはSPIDERの輪郭追跡プログラムBDFL2の結果を用いるとよい。

DSEGMENT

Direction Segments Extraction from Character Pattern

文字パターンより 4 方向の線素（方向セグメント）抽出する

作成	馬場口 登	1986年 7月
形式	サブルーチン	言語 : FORTRAN サイズ : 39行

(1) 概要

文字パターンより、垂直・水平・右斜め上・左斜め上の 4 方向の線素（4 方向セグメント）を細線化することなく抽出する。

2 値パターンの各黒点から、抽出する方向の幾何学的距離を測定し、最長の方向のコードにその画素値を変換した後、コード別に生成された方向画面内でセグメント結合を施し、方向セグメントを抽出する。

本方法の特徴は細線化を用いないこと、及びアナログ的な処理により任意の方向のセグメントが抽出可能であることである。

(2) 使用法

CALL DSEGMENT(IA, I1, J1, IB, IWORKE, IWORKC)

引数	型と種類	属性	内容
IA	整数型 2 次元配列	入力 出力	入力文字のパターン, IA (I1, J1) 出力時には 4 方向コードが入る
I1	整数型	入力	IA の整合寸法 (第 1 添字の寸法)
J1	整数型	入力	IA の整合寸法 (第 2 添字の寸法)
IB	整数型 3 次元配列	出力	出力パターン, IB (I1, J1, 4) 4 方向セグメントが出力される
IWORKE	整数型 2 次元配列	WORK	作業用配列, IWORKE (I1, J1)

I WORK C	整数型 3次元配列	WORK	作業用配列, I WORK C (I 1, J 1, 8)
----------	--------------	------	------------------------------------

(3) 出力

出力として得られる 4 方向コードならびに出力画像は次のとおりである。

4 方向コードは

- 1 : 右斜め上
- 2 : 垂直
- 3 : 左斜め上
- 4 : 水平 である。

出力画像に入る、各方向セグメントは

- I B (I 1, J 1, 1) : 右斜め上のセグメント
- I B (I 1, J 1, 2) : 垂直のセグメント
- I B (I 1, J 1, 3) : 左斜め上のセグメント
- I B (I 1, J 1, 4) : 水平のセグメント である。

本サブルーチンは孫ルーチンとして、D C O D E 4, P T G E N 4, J O I N T 4 という、3つのサブルーチンを呼んでいる。

(4) 使用例

上記のサブルーチンのテストを行うためのプログラムはD S E G M T C K にある。

(5) 使用上の注意

本ルーチン並びに孫ルーチンはいずれも大阪大学工学部、馬場口登氏より提供されたものである。使用の際はそのことを明記すること。

(6) 参考文献

馬場口 登他, 細線化を用いない文字パターンの方向セグメント抽出の一手法,
信学論 (D) J 6 5 - D, N o . 7, pp874-881,

D C O D E 4

4-Direction Coding

文字パターンより 4 方向セグメントを抽出するため 8 方向コードを計算する

作成	馬場口 登	1 9 8 6 年 7 月
形式	サブルーチン	言語 : FORTRAN サイズ : 1 2 4 行

(1) 概要

文字パターンより、垂直・水平・右斜め上・左斜め上の 4 方向の線素（4 方向セグメント）を細線化することなく抽出する、D S E G M T の孫ルーチンである。

2 値パターンの各黒点から、8 方向の幾何学的距離（連続する黒画素数）を測定する。その結果を C (I M A X, J M A X, 8) に格納する。

(2) 使用法

```
CALL DCODE4(E, F, C, IMAX, JMAX)
```

引数	型と種類	属性	内容
E	整数型 2 次元配列	WORK	作業用配列, E (I M A X, J M A X)
F	整数型 2 次元配列	入力 出力	入力文字のパターン, F (I M A X, J M A X)
C	整数型 3 次元配列	WORK	作業用配列, 方向コード毎の距離が出力される。C (I M A X, J M A X, 8)
I M A X	整数型	入力	F の整合寸法 (第 1 添字の寸法)
J M A X	整数型	入力	F の整合寸法 (第 2 添字の寸法)

(3) 出力

C (I MAX, J MAX, 8) に出力として得られる 8 方向は次のとおりである。

C (I MAX, J MAX, 1) : 右斜め上

C (I MAX, J MAX, 2) : 上向き垂直

C (I AMX, J MAX, 3) : 左斜め上

C (I AMX, J MAX, 4) : 左向き水平

C (I MAX, J MAX, 5) : 左斜め下

C (I MAX, J MAX, 6) : 下向き垂直

C (I MAX, J MAX, 7) : 右斜め下

C (I MAX, J MAX, 8) : 右向き水平 である。

各配列には、文字パターンの各黒画素から見た連続する黒画素数が入る。

(4) 使用例

上記のサブルーチンのテストを行うためのプログラムは D S E G M T C K にある。

(5) 使用上の注意

本ルーチンは大阪大学工学部、馬場口登氏より提供されたものである。使用の際はそのことを明記すること。

(6) 参考文献

馬場口 登他, 細線化を用いない文字パターンの方向セグメント抽出の一手法,
信学論 (D) J 65-D, No. 7, pp874-881,

P T G E N 4

Generation of Direction Pattern

方向パターンの生成

作成	馬場口 登	1986年 7月
形式	サブルーチン	言語 : FORTRAN サイズ : 31行

(1) 概要

文字パターンより、垂直・水平・右斜め上・左斜め上の4方向の線素（4方向セグメント）を細線化することなく抽出する、DSEGMENTの孫ルーチンである。

F (IMAX, JMAX) より、4方向パターンの出力画像D (IMAX, JMAX, 4) を生成する。

(2) 使用法

CALL PTGEN4(F, D, IMAX, JMAX)

引数	型と種類	属性	内容
F	整数型 2次元配列	入力 出力	入力文字のパターン、4方向コードを出力する。F (IMAX, JMAX)
D	整数型 3次元配列	出力	出力パターン、D (I1, J1, 4) 4方向セグメントが出力される
IMAX	整数型	入力	F の整合寸法（第1添字の寸法）
JMAX	整数型	入力	F の整合寸法（第2添字の寸法）

(3) 出力

出力として得られる4方向コードは次のとおりである。

4方向コードは、入力画像F (IMAX, JMAX) に入る。

- 1 : 右斜め上
- 2 : 垂直
- 3 : 左斜め上
- 4 : 水平 である。

(4) 使用例

上記のサブルーチンのテストを行うためのプログラムはSEGMENTCKにある。

(5) 使用上の注意

本ルーチンは大阪大学工学部、馬場口登氏より提供されたものである。使用の際はそのことを明記すること。

(6) 参考文献

馬場口 登他, 細線化を用いない文字パターンの方向セグメント抽出の一手法,
信学論 (D) J 65-D, No. 7, pp874-881,

JOINT4

Segments Combination on Direction Patterns

方向セグメントパターンの結合

作成	馬場口 登	1986年 7月
形式	サブルーチン	言語: FORTRAN サイズ: 93行

(1) 概要

文字パターンより、垂直・水平・右斜め上・左斜め上の4方向の線素（4方向セグメント）を細線化することなく抽出する、DSEGMENTの孫ルーチンである。

4方向パターンの入ったF(I MAX, J MAX)等を使い、4方向パターンの出力画像D(I MAX, J MAX, 4)を作成する。

(2) 使用法

```
CALL JOINT4(F, D, IMAX, JMAX)
```

引数	型と種類	属性	内容
F	整数型 2次元配列	入力 出力	入力文字のパターン、4方向コードを出力する。F(I MAX, J MAX)
D	整数型 3次元配列	出力	出力パターン、D(1, 1, 4) 4方向セグメントが出力される
I MAX	整数型	入力	Fの整合寸法（第1添字の寸法）
J MAX	整数型	入力	Fの整合寸法（第2添字の寸法）

(3) 出力

出力として得られる 4 方向コードならびに出力画像は次のとおりである。

F (I MAX, J MAX) に入る 4 方向コードは

- 1 : 右斜め上
- 2 : 垂直
- 3 : 左斜め上
- 4 : 水平 である。

出力画像の各方向セグメントは 2 値画像であり、以下の配列に入る。

- I B (I 1, J 1, 1) : 右斜め上のセグメント
- I B (I 1, J 1, 2) : 垂直のセグメント
- I B (I 1, J 1, 3) : 左斜め上のセグメント
- I B (I 1, J 1, 4) : 水平のセグメント である。

(4) 使用例

上記のサブルーチンのテストを行うためのプログラムは D S E G M T C K にある。

(5) 使用上の注意

本ルーチンは大阪大学工学部、馬場口登氏より提供されたものである。使用の際はそのことを明記すること。

(6) 参考文献

馬場口 登他、細線化を用いない文字パターンの方向セグメント抽出の一手法、
信学論 (D) J 6 5 - D, No. 7, pp874-881,

HKIYO

Calculation of Peripheral Direction Contributivity Feature

外郭方向寄与度特徴の計算

作成	村上 幸太郎	1987年12月
形式	サブルーチン	言語 : FORTRAN サイズ : 150行

(1) 概要

文字画像データ (IA (I1, J1)) の外郭方向寄与度特徴の計算。

(2) 使用法

CALL HKIYO (IA, I1, J1, H, IZDIV, NS, ITDIV, IERR)

引数	型と種類	属性	内 容
IA	整数型 2次元配列	入力	入力文字画像
I1	整数型定数	入力	IAの横サイズ (第1添え字)
J1	整数型定数	入力	IAの縦サイズ (第2添え字)
H	整数型 4次元配列	出力	外郭方向寄与度特徴計算結果 H(ITDIV, NS, M, IZDIV) M=1:水平方向の方向寄与度 M=2:右上がり45度方向の方向寄与度 M=3:垂直方向の方向寄与度 M=4:右下がり45度方向の方向寄与度
IZDIV	整数型定数	入力	投影軸分割数
NS	整数型定数	入力	外郭深度

ITDIV	整数型定数	入力	走査方向
IERR	整数型変数	出力	エラーコード IERR=0 : 正常終了 IERR=1 : IZDIVが制限を犯した IERR=2 : ITDIVが4又は8でない IERR=222: NSが制限を犯した

(3) 計算法

外郭方向寄与度特徴は、文字画像データを 96×96 で計算するため、本サブルーチン内部で 96×96 に正規化する。

次に、正規化された文字画像データを水平に外郭深度NSまで走査し、輪郭黒点における外郭方向寄与度特徴を計算する。続いて+45度方向に走査し外郭方向寄与度特徴を計算する。（ITDIV=4のときは、+45度方向の走査はしない。）

以上の計算が、終了したら文字画像データを90度右へ回転し同様に水平、+45度方向に走査し、外郭方向寄与度特徴を計算する。

文字画像データを更に2回、回転させ指定された全ての走査方向に対する外郭方向寄与度を計算する。

(4) 備考

1. ITDIV, NS, IZDIVの制限は、以下の通りである。（）内の数値であればよい。

$$IZDIV = (2, 4, 6, 8, 12, 16)$$

$$NS = (1, 2, 3)$$

$$ITDIV = (4, 8)$$

2. 文字画像サイズが 96×96 でないときの正規化方法は外郭長方形合わせによる。
3. 本サブルーチン内で水平方向の走査をする‘TKIYO’と45度方向の走査をする‘XKIYO’を使用している。

(5) 参考文献

- 荻田 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認識」電子情報通信学会技術研究報告 J 66-D, 10 pp. 1185-1192
(1983.10)

TKIYO

Search of Black-dots for Horizontal Direction

文字画像を水平に走査して指定された外郭深度までの輪郭黒点を求める

作成	村上 幸太郎	1987年12月
形式	サブルーチン	言語 : FORTRAN サイズ : 73行

(1) 概要

文字画像データ (IA (I1, J1)) の指定された行を走査して指定の外郭深度までの輪郭黒点を求める。

(2) 使用法

CALL TKIYO (IA, I1, J1, J, DN, NS, ERR)

引数	型と種類	属性	内 容
I A	整数型 2次元配列	入力	入力文字画像
I 1	整数型定数	入力	I A の横サイズ (第1添え字)
J 1	整数型定数	入力	I A の縦サイズ (第2添え字)
J	整数型定数	入力	走査する行
DN	実数型 2次元配列	出力	外郭方向寄与度計算結果
NS	整数型定数	入力	外郭深度
ERR	整数型変数	出力	エラーコード ERR=0 : 正常終了 ERR=111: J が制限を犯している。 ERR=222: NS が制限を犯している。

(3) 計算法

文字画像データの J で指定された行を水平方向に走査してドットをチェックする。黒点（文字画像データの 1 の部分）があれば、その黒点の方向寄与度を計算するサブルーチンをコールする。その点の方向寄与度が得られると指定された外郭深度まで走査して方向寄与度を求める。

(4) 備考

方向寄与度計算サブルーチン ‘K I Y O D O’ を含む。

X K I Y O

Search of Black-dots for +45 Degree Direction

文字画像データを +45 度方向に走査して外郭方向寄与度を求める

作成	村上 幸太郎	1987年12月
形式	サブルーチン	言語 : FORTRAN サイズ : 93行

(1) 概要

文字画像データ (IA (I1, J1)) の指定された点から +45 度方向に走査して指定の外郭深度までの輪郭黒点を求める。

(2) 使用法

```
CALL XKIY0 (IA, I1, J1, I, J, DN, NS, ERR)
```

引 数	型と種類	属性	内 容
I A	整数型 2 次元配列	入力	入力文字画像
I 1	整数型定数	入力	I A の横サイズ (第 1 添え字)
J 1	整数型定数	入力	I A の縦サイズ (第 2 添え字)
I	整数型定数	入力	走査する列
J	整数型定数	入力	走査する行
D N	実数型 2 次元配列	出力	外郭方向寄与度計算結果
N S	整数型定数	入力	外郭深度
E R R	整数型変数	出力	エラーコード ERR=0 : 正常終了 ERR=111: J が制限を犯している。

ERR=222: N S が制限を犯している。

(3) 計算法

文字画像データの I と J で指定された点から +45 度方向に走査してドットをチェックする。黒点（文字画像データの 1 の部分）があれば、その黒点の方向寄与度を計算するサブルーチンをコールする。その点の方向寄与度が得られると指定された外郭深度まで +45 度方向に走査して方向寄与度を求める。

(4) 備考

方向寄与度計算サブルーチン ‘K I Y O D O’ を含む。

K I Y O D O

Calculation of Peripheral Direction Contributivity

方向寄与度の計算

作成	村上 幸太郎	1987年12月
形式	サブルーチン	言語: FORTRAN サイズ: 116行

(1) 概要

指定した点の方向寄与度を求める。

(2) 使用法

CALL KIYODO (IA, I1, J1, I, J, D, ILL)

引 数	型と種類	属性	内 容
I A	整数型 2次元配列	入力	入力文字画像
I 1	整数型定数	入力	I A の横サイズ (第1添え字)
J 1	整数型定数	入力	I A の縦サイズ (第2添え字)
I	整数型定数	入力	指定する点
J	整数型定数	入力	指定する点
D	実数型 1次元配列	出力	指定された点の方向寄与度
ILL	整数型変数	出力	エラーコード ILL=0 : 正常終了 ILL=-1: 指定された点が 1 でない

(3) 計算法

求める。

指定された点から 8 方向に触手を伸ばしていき、その点から輪郭線までの長さを

M S T A T 1

Statistics Calculation Program(1)

統計量計算プログラム(1)

作成	木村文隆	1985年 7月
形式	サブルーチン	言語:FORTRAN サイズ: 30行

(1) 概要

学習サンプルから特徴ベクトルの平均を次式により計算する。

$$MV(I) = \frac{1}{M} \sum_{J=1}^M LSP(I, J) \quad (I = 1, 2, \dots, N)$$

(2) 使用法

CALL MSTAT1(LSP, MV, N, M)

引数	型と種類	属性	内容
LSP	実数型 2次元配列	入力	学習サンプル行列
MV	実数型 1次元配列	入力	平均ベクトル
N	整数型	入力	次元数 MVの整合寸法 LSPの整合寸法 (第1添字の寸法)
M	整数型	入力	学習サンプルサイズ LSPの整合寸法 (第2添字の寸法)

(3) その他

CD1, ED1, ED2, SS1, SS2で用いる平均ベクトルMVを計算するためなどに使用する。

M S T A T 2

Statistics Calculation Program(2)

統計量計算プログラム(2)

作成	木村文隆	1985年 7月
形式	サブルーチン	言語:FORTRAN サイズ: 47行

(1) 概要

学習サンプルから特徴ベクトルの平均と共分散を次式により計算する。

$$MV(I) = (1 / M) \sum_{K=1}^M LSP(I, K) \quad (I = 1, 2, \dots, N)$$

$$COV(I, J) = (1 / M) \sum_{K=1}^M \{ LSP(I, K) \cdot LSP(J, K) \} - MV(I) \cdot MV(J) \quad (I = 1, 2, \dots, N ; J = 1, 2, \dots, N)$$

(2) 使用法

CALL MSTAT2(LSP, MV, COV, N, M)

引数	型と種類	属性	内容
LSP	実数型 2次元配列	入力	学習サンプル行列
MV	実数型 1次元配列	入力	平均ベクトル
COV	実数型 2次元配列	出力	共分散行列
N	整数型	入力	次元数 MVの整合寸法

			L S P (第1添字)、C O V (第1 、第2添字)の整合寸法
M	整数型	入力	学習サンプルサイズ L S Pの整合寸法 (第2添字の寸法)

(3) その他

M S T A T 5, M S T A T 6 の内部で使用。

MSTAT3

Statistics Calculation Program(3)

統計量計算プログラム(3)

作成	木村文隆	1985年 7月
形式	サブルーチン	言語:FORTRAN サイズ: 47行

(1) 概要

学習サンプルから特徴ベクトルの平均と自己相関行列を次式により計算する。

$$MV(I) = \frac{1}{M} \sum_{K=1}^M LSP(I, K) \quad (I = 1, 2, \dots, N)$$

$$ACOR(I, J) = \frac{1}{M} \sum_{K=1}^M \{ LSP(I, K) \cdot LSP(J, K) \} \quad (I = 1, 2, \dots, N; J = 1, 2, \dots, N)$$

(2) 使用法

CALL MSTAT3(LSP, MV, ACOR, N, M)

引数	型と種類	属性	内容
LSP	実数型 2次元配列	入力	学習サンプル行列
MV	実数型 1次元配列	入力	平均ベクトル
ACOR	実数型 2次元配列	出力	行列
N	整数型	入力	次元数 MVの整合寸法

			L S P (第1添字)、A C O R (第1、第2添字)の整合寸法
M	整数型	入力	学習サンプルサイズ L S P の整合寸法 (第2添字の寸法)

(3) その他

M S T A T 4 の内部で使用。

M S T A T 4

Statistics Calculation Program(4)

統計量計算プログラム(4)

作成	木村文隆	1 9 8 5 年 7 月
形式	サブルーチン	言語:FORTRAN サイズ: 25行

(1) 概要

M S T A T 3 を用いて学習サンプルから自己相関行列を計算し、その固有値、固有ベクトルを求める。

(2) 使用法

```
CALL MSTAT4(LSP, EVAL, WORK, EVEC, N, M, ESP, ILL)
```

引数	型と種類	属性	内容
L S P	実数型 2次元配列	入力	学習サンプル行列
E V A L	実数型 1次元配列	入力	自己相関行列の固有値ベクトル
W O R K	実数型 1次元配列	作業用	
E V E C	実数型 2次元配列	入力	自己相関行列の固有ベクトル行列
N	整数型	入力	次元数 EVAL, WORKの整合寸法 LSP (第1添字)、EVEC (第1、第2添字) の整合寸法

M	整数型	入力	学習サンプルサイズ L S Pの整合寸法 (第2添字の寸法)
E S P	実数型	入力	収束判定用パラメータ
I L L	整数型	入力	エラーコード

(3) その他

複合類似度MS1で用いるE V E Cを計算するために使用する。

固有値、固有ベクトルを計算するために、SSLのHOQRUSを使用しており、ESP, ILLはそのパラメータである。HOQRUSについてはSSL利用の手引を参照せよ。

M S T A T 5

Statistics Calculation Program(5)

統計量計算プログラム(5)

作成	木村文隆	1 9 8 5 年 7 月
形式	サブルーチン	言語:FORTRAN サイズ: 28行

(1) 概要

M S T A T 2 を用いて学習サンプルから平均ベクトルと共分散行列を計算し、共分散行列の固有値、固有ベクトルを求める。

(2) 使用法

CALL MSTAT5(LSP, MV, EVAL, WORK, EVEC, N, M, ESP, ILL)

引数	型と種類	属性	内容
L S P	実数型 2次元配列	入力	学習サンプル行列
M V	実数型 1次元配列	入力	平均ベクトル
E V A L	実数型 1次元配列	入力	共分散行列の固有値ベクトル
W O R K	実数型 1次元配列	作業用	
E V E C	実数型 2次元配列	入力	共分散行列の固有ベクトル行列
N	整数型	入力	次元数 MV, EVAL, WORK,

AM
Note
for(j=0 ...) {
for(i=0 ...)
[jxm+i]}

			L S P (第1添字)、E V E C (第1、第2添字)の整合寸法
M	整数型	入力	学習サンプルサイズ L S Pの整合寸法 (第2添字の寸法)
E S P	実数型	入力	収束判定用パラメータ
I L L	整数型	入力	エラーコード

(3) その他

Q D 1, M D 1, M Q D 1, M Q D 2で用いるE V E Cを計算するために使用する。

固有値、固有ベクトルを計算するために、S S LのH O Q R U Sを使用しており、E S P, I L Lはそのパラメータである。H O Q R U SについてはS S L利用の手引を参照せよ。

M S T A T 6

Statistics Calculation Program(6)

統計量計算プログラム(6)

作成	木村文隆	1985年 7月
形式	サブルーチン	言語:FORTRAN サイズ: 50行

(1) 概要

等共分散性の仮定のもとで2次識別関数から導かれる線形識別関数の重みベクトルと定数項を各クラスの平均ベクトル、共分散行列から計算する。共分散行列は等重みでプールして用いられる。

(2) 使用法

CALL MSTAT6(MV, COV, W, W0, N, L)

引数	型と種類	属性	内容
M V	実数型 2次元配列	入力	各クラスの平均ベクトル
C O V	実数型 3次元配列	入力	各クラスの共分散行列
W	実数型 2次元配列	出力	各クラスに対する重みベクトルからなる重みベクトル行列
W O	実数型 1次元配列	出力	各クラスに対する定数項からなる定数項ベクトル
N	整数型	入力	次元数 M V (第1添字)、C O V (第1、第2添字)、W (第1添字) の整合寸法

L	整数型	入力	クラス数 MV（第2添字）、COV（第3添 字）、W（第2添字）、WOの整合 寸法
---	-----	----	--

(3) その他

MV、COVは、MSTAT2で計算したものを用いる。すべてのクラスに対する重みベクトルと定数項が一度に計算される点が他の統計量計算ルーチンと異なるので注意すること。MV、COVもすべてのクラスに対するものを配列に用意して与えることが必要である。

C D 1

Cityblock Distance

シティブロック距離

作成	木村文隆	1985年 7月
形式	実数関数	言語:FORTRAN サイズ: 21行

(1) 概要

シティブロック距離を次式により計算する。

$$CD1 = \sum_{I=1}^N |X(I)-MV(I)|$$

(2) 使用法

CD1 (X, MV, N)

引数	型と種類	属性	内容
X	実数型 1次元配列	入力	特徴ベクトル
MV	実数型 1次元配列	入力	平均ベクトル
N	整数型	入力	次元数 X, MVの整合寸法

(3) その他

MVは、M S T A T 1により学習サンプルから求める。

E D 1

Squared Euclid Distance

二乗ユークリッド距離

作成	木村文隆	1985年 7月
形式	実数関数	言語:FORTRAN サイズ: 21行

(1) 概要

二乗ユークリッド距離を次式により計算する。

$$ED1 = \sum_{I=1}^N (X(I) - MV(I))^2$$

(2) 使用法

ED1 (X, MV, N)

引数	型と種類	属性	内容
X	実数型 1次元配列	入力	特徴ベクトル
MV	実数型 1次元配列	入力	平均ベクトル
N	整数型	入力	次元数 X, MVの整合寸法

(3) その他

MVは、M S T A T 1により学習サンプルから求める。

ED 2

Euclid Distance

ユークリッド距離

作成	木村文隆	1985年 7月
形式	実数関数	言語:FORTRAN サイズ: 22行

(1) 概要

ユークリッド距離を次式により計算する。

$$ED2 = \sqrt{\sum_{I=1}^N (X(I)-MV(I))^2}$$

(2) 使用法

ED2 (X, MV, N)

引数	型と種類	属性	内容
X	実数型 1次元配列	入力	特徴ベクトル
MV	実数型 1次元配列	入力	平均ベクトル
N	整数型	入力	次元数 X, MVの整合寸法

(3) その他

MVは、M S T A T 1により学習サンプルから求める。

L D 1

Linear Discriminant Function

線形識別関数

作成	木村文隆	1985年 7月
形式	実数関数	言語:FORTRAN サイズ: 21行

(1) 概要

線形識別関数を次式により計算する。

$$LD1 = \sum_{I=1}^N \{W(I) \times X(I)\} + W0$$

(2) 使用法

LD1 (X, W, W0, N)

引数	型と種類	属性	内容
X	実数型 1次元配列	入力	特徴ベクトル
W	実数型 1次元配列	入力	係数ベクトル
W0	実数型	入力	定数
N	整数型	入力	次元数 X, Wの整合寸法

(3) その他

W, W0は、M S T A T 6により求める。引用するプログラムでREAL宣言すること。

MD 1

Mahalanobis Distance

マハラノビス距離

作成	木村文隆	1985年 7月
形式	実数関数	言語:FORTRAN サイズ: 25行

(1) 概要

マハラノビス距離を次式により計算する。

$$MD1 = \sum_{I=1}^N \{ \sum_{J=1}^N EVEC(J, I) \times (X(J) - MV(J)) \}^2 / EVAL(I)$$

(2) 使用法

MD1 (X, MV, EVAL, EVEC, N)

引数	型と種類	属性	内容
X	実数型 1次元配列	入力	特徴ベクトル
MV	実数型 1次元配列	入力	平均ベクトル
EVAL	実数型 1次元配列	入力	共分散行列の固有値ベクトル
EVEC	実数型 2次元配列	入力	共分散行列の固有ベクトル行列
N	整数型	入力	次元数 X、MV、EVALの整合寸法 EVECの整合寸法

(第1、第2添字の寸法)

(3) その他

MV, EVAL, EVECは、MSTAT5により学習サンプルから求める。引用するプログラムでREAL宣言すること。

M Q D 1

Modified Quadratic Discriminant Function(1)

修正 2 次識別関数(1)

作成	木村文隆	1 9 8 5 年 7 月
形式	実数関数	言語:FORTRAN サイズ: 31 行

(1) 概要

修正 2 次識別関数の値を次式により計算する。

$$MQD1 = \sum_{I=1}^N \left[\left\{ \sum_{J=1}^N EVEC(J, I) \times (X(J) - MV(J)) \right\}^2 / (EVAL(I) + H) \right] \\ + \ln \prod_{I=1}^N (EVAL(I) + H)$$

(2) 使用法

MQD1 (X, MV, EVAL, EVEC, N, H)

引数	型と種類	属性	内容
X	実数型 1 次元配列	入力	特徴ベクトル
M V	実数型 1 次元配列	入力	平均ベクトル
E V A L	実数型 1 次元配列	入力	共分散行列の固有値ベクトル
E V E C	実数型 2 次元配列	入力	共分散行列の固有ベクトル行列

N	整数型	入力	次元数 X、MV、EVALの整合寸法 EVECの整合寸法 (第1、第2添字の寸法)
H	実数型	入力	定数

(3) その他

MV、EVAL、EVECは、MSTAT5により学習サンプルから求める。引用するプログラムでREAL宣言すること。最適なHの値は学習サンプル数に依存し、学習サンプル数が少ないほど大きくなる。第10ないし第20固有値の中から実験的に決めるといい。

MQ D 2

Modified Quadratic Discriminant Function(2)

修正 2 次識別関数(2)

作成	木村文隆	1985年 7月
形式	実数関数	言語:FORTRAN サイズ: 34行

(1) 概要

修正 2 次識別関数の値を次式により計算する。

$$\begin{aligned}
 MQD2 = & \left[\sum_{I=1}^N (X(I) - MV(I))^2 - \sum_{I=1}^K \{(1 - H/EVAL(I)) \sum_{J=1}^N EVEC(J, I) \right. \\
 & \times (X(J) - MV(J))^2 \} \right] / H + H^{(N-K)} \ln \left(\prod_{I=1}^K EVAL(I) \right)
 \end{aligned}$$

(2) 使用法

MQD2 (X, MV, EVAL, EVEC, N, K, H)

引数	型と種類	属性	内容
X	実数型 1 次元配列	入力	特徴ベクトル
M V	実数型 1 次元配列	入力	平均ベクトル
E V A L	実数型 1 次元配列	入力	共分散行列の固有値ベクトル
E V E C	実数型 2 次元配列	入力	共分散行列の固有ベクトル行列

N	整数型	入力	次元数 X、MV、EVALの整合寸法 EVECの整合寸法 (第1、第2添字の寸法)
K	整数型	入力	使用する固有ベクトル数 EVECの整合寸法 (第2添字の寸法)
H	実数型	入力	定数

(3) その他

MV、EVAL、EVECは、MSTAT5により学習サンプルから求める。引用するプログラムでREAL宣言すること。最適なHの値は学習サンプル数に依存し、学習サンプル数が少ないほど大きくする。第10ないし第20固有値の中から実験的に決めるといい。同様にしてKの値も10ないし20のなかから実験的に決める。

MS 1

Multiple Similarity

複合類似度

作成	木村文隆	1985年 7月
形式	実数関数	言語:FORTRAN サイズ: 25行

(1) 概要

複合類似度を次式により計算する。

$$MS1 = \sum_{J=1}^K \{ \sum_{I=1}^N (X(I) \times EVEC(I, J)) \}^2$$

(2) 使用法

MS1 (X, EVEC, N, K)

引数	型と種類	属性	内容
X	実数型 1次元配列	入力	特徴ベクトル
E V E C	実数型 2次元配列	入力	自己相関行列の固有ベクトル行列
N	整数型	入力	次元数 Xの整合寸法 E V E Cの整合寸法 (第1添字の寸法)
K	整数型	入力	使用する固有ベクトル数 E V E Cの整合寸法 (第2添字の寸法)

(3) その他

E V E C は、 M S T A T 4 により学習サンプルから求める。 $\| X \| = 1$ のとき、
 $-1 \leq M S 1 \leq 1$ となる。引用するプログラムで R E A L 宣言すること。K は、 5
ないし 20 のなかから実験的に選ぶ。

Q D 1

Quadratic Discriminant Function

2 次識別関数

作成	木村文隆	1 9 8 5 年 7 月
形式	実数関数	言語:FORTRAN サイズ: 29行

(1) 概要

2次識別関数の値を次式により計算する。

$$QD1 = \sum_{I=1}^N \left\{ \sum_{J=1}^N EVEC(J, I) \times (X(J) - MV(J)) \right\}^2 / EVAL(I) + \ln \left\{ \prod_{J=1}^N EVAL(J) \right\}$$

(2) 使用法

QD1 (X, MV, EVAL, EVEC, N)

引数	型と種類	属性	内容
X	実数型 1次元配列	入力	特徴ベクトル
MV	実数型 1次元配列	入力	平均ベクトル
EVAL	実数型 1次元配列	入力	共分散行列の固有値ベクトル
EVEC	実数型 2次元配列	入力	共分散行列の固有ベクトル行列
N	整数型	入力	次元数 X, MV, EVAL の整合寸法 EVEC の整合寸法

(3) その他

MV, EVAL, EVECは、MSTAT5により学習サンプルから求める。
QD1の値は負になることもある。

S S 1

Simple Similarity

単純類似度

作成	木村文隆	1985年 7月
形式	実数関数	言語:FORTRAN サイズ: 26行

(1) 概要

単純類似度を次式により計算する。

$$SS1 = \frac{\sum_{I=1}^N \{X(I) \times MV(I)\}}{\sqrt{\sum_{I=1}^N X(I)^2 \times \sum_{I=1}^N MV(I)^2}}$$

(2) 使用法

SS1 (X, MV, N)

引数	型と種類	属性	内容
X	実数型 1次元配列	入力	特徴ベクトル
MV	実数型 1次元配列	入力	平均ベクトル
N	整数型	入力	次元数 X, MVの整合寸法

(3) その他

MVは、MSTAT1により学習サンプルから求める。 $-1 \leq SS1 \leq 1$ となる。

S S 2

Simple Similarity

単純類似度

作成	木村文隆	1985年 7月
形式	実数関数	言語:FORTRAN サイズ: 24行

(1) 概要

単純類似度を次式により計算する。

$$SS2 = \frac{\sum_{I=1}^N \{X(I) \times NMV(I)\}}{\sqrt{\sum_{I=1}^N X(I)^2}}$$

(2) 使用法

SS2 (X, NMV, N)

引数	型と種類	属性	内容
X	実数型 1次元配列	入力	特徴ベクトル
NMV	実数型 1次元配列	入力	正規化平均ベクトル
N	整数型	入力	次元数 X, NMV の整合寸法

(3) その他

MVは、MSTAT1により学習サンプルから求めたものを、ノルム1となるように正規化したもの。 $|NMV| = 1$ のとき、 $-1 \leq SS2 \leq 1$ となる。

POUT16

Print Out of 16(Hexadecimal) Level character data

16値化データのプリント出力

作成	竹下鉄夫	1985年 6月
形式	サブルーチン	言語: FORTRAN サイズ: 85行

(1) 概要

与えられた16値レベル(0~15)2次元配列の各配列要素の値に従って、まわりに枠をつけて2次元文字図形をプリント出力する。入力2次元配列IAの配列要素が0ならばブランク、配列要素が1~15ならば対応する16進数を出力する。IAの配列要素が0~15以外の時は'X'を出力する。

(2) 使用法

CALL POUT16(IA, I1, J1, ILL)

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	プリント出力すべき2次元配列
I1	整数型	入力	IAの整合寸法(第1添字の寸法) $I1 \leq 130$
J1	整数型	入力	IAの整合寸法(第2添字の寸法) $J1 \leq 200$
ILL	整数型	出力	$ILL = 0$: 正常終了 $ILL = 2000$: 配列要素に0~15 以外を含む。 $ILL = 30000$: 引数I1, J1が 不適当 この時はプリント出力しない。

(3) 使用例

テストプログラムの TSTPOU16 に使用法と使用例がある。

POUT2B

Print Out of 2(Binary) Level character data

2値化データのプリント出力

作成	竹下鉄夫	1985年 6月
形式	サブルーチン	言語: FORTRAN サイズ: 82行

(1) 概要

与えられた2次元配列の各配列要素の値に従って、まわりに枠をつけて2次元文字図形をプリント出力する。入力2次元配列IAの配列要素が0ならばブランク、1ならば'*'を出力する。0または1以外の時は'X'を出力する。

(2) 使用法

CALL POUT2B(IA, I1, J1, ILL)

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	プリント出力すべき2次元配列
I1	整数型	入力	IAの整合寸法(第1添字の寸法) $I1 \leq 130$
J1	整数型	入力	IAの整合寸法(第2添字の寸法) $J1 \leq 200$
ILL	整数型	出力	$ILL = 0$: 正常終了 $ILL = 2000$: 配列要素に0または1以外を含む。 $ILL = 30000$: 引数I1, J1が不適当 この時はプリント出力しない。

(3) 使用例

テストプログラムの TSTPOU2S に使用法と使用例がある。

MSWAKU

Wakutsuke

2次元配列の周囲、上下左右に'0'を付ける

作成	竹下鉄夫	1985年 6月
形式	サブルーチン	言語：FORTRAN サイズ：36行

(1) 概要

入力2次元配列IA(I1, J1)の周囲に上下左右それぞれ幅M個の0をつけて新しく2次元配列IBに格納する。

出力配列の大きさはIB(I1 + 2 × M, J1 + 2 × M)となる。

(2) 使用法

CALL MSWAKU(IA, I1, J1, IB, M)

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	入力データの入っている2次元配列
I1	整数型	入力	IAの整合寸法（第1添字の寸法）
J1	整数型	入力	IAの整合寸法（第2添字の寸法）
IB	整数型 2次元配列	出力	出力データの入る2次元配列 IB(I1 + 2 × M, J1 + 2 × M)
M	整数型	入力	上下左右につける'0'の幅

(3) 使用例

上記のサブルーチンのテストを行うためのプログラム及び、実行結果はテストプログラムの TSTPOU2S 及び TSTPOU16 にある。

PLTBIA

Line Drawing for Binary・Multi-Level Image Using XY-Plotter

XYプロッターによる 2 値・多値画像の線画表現

作成	鶴岡信治	1986年	7月
形式	サブルーチン	言語 : FORTRAN	サイズ : 69行

(1) 概要

文字画像を 2 値化して、文字領域を XY プロッターを用いて、複数の横線（一行に 1 本）で描く。入力画像は 2 値画像でも多値画像でも良い。

(2) 使用法

```
CALL PLTBIA(IA, I1, J1, X0, Y0, DX, DY, IT, MNO, MCA)
```

引数	型と種類	属性	内容
I A	整数型 2 次元配列	入力	表示したい画像
I 1	整数型	入力	I P の整合寸法（第 1 添字の値）
J 1	整数型	入力	I P の整合寸法（第 2 添字の値）
X 0	実数型	入力	紙上の原点の x 座標
Y 0	実数型	入力	紙上の原点の y 座標
D X	実数型	入力	紙上の文字枠の X 方向の大きさ
D Y	実数型	入力	紙上の文字枠の Y 方向の大きさ
I T	整数型	入力	2 値化のしきい値

M N O	整数型	入力	文字の番号（6桁以内の数字）
M C A	整数型	入力	字種の番号（6桁以内の数字）

(3) 計算法

文字を左上から水平に走査していき、しきい値 I T 以上の点を見つけると、線を引き始め（ペンダウンし）、しきい値以下である点を見つけるとペンアップする。

高速に線画を書くために、奇数行目では、左から右へ線を引き、偶数行目では、右から左へ線を引く、工夫がしてある。

文字枠の左下に数字M N Oをその右に数字M C Aをかく。A 4 サイズ(21.0mm×25.0mm)では、X 0を2.0, Y 0を21.0とし、DXを4.0, DYを4.0とすると、3×4 個の画像の表示ができる。

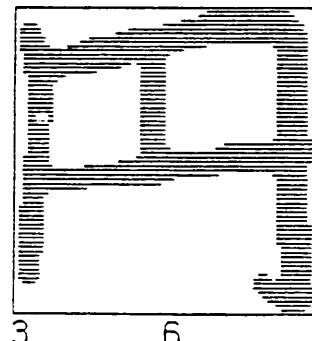
(4) 子ルーチン

F R A M E, N U M B E R, P L O T

(5) プログラム例

```
INTEGER IA(70,70)
C*** PLOTTER NO SYOKICHI NO SETTEI ***  
X0=2.0  
Y0=21.0  
CALL XINT  
CALL XVIEWP(0.0,0.0,21.0,25.0)  
DX=4.0  
DY=4.0  
ITHRSH=1  
C*** MOJI NITUITE NO LOOP ***  
MOJICA=2  
DO 1000 MOJINN=1,20  
READ(8) IA  
CALL PLTBIA(IA,70,70,X0,Y0,DX,DY,ITHRSH,MOJINN,MOJICA)  
C*** MOJI WAKU NO HIDARI SITA NO SEIJYO ***  
X0=X0+6.0
```

```
IF(X0.GT.16.0) THEN  
  X0=2.0  
  Y0=Y0-6.0  
  IF(Y0.LT.0.0) THEN  
C*** NEW PAGE ***  
  CALL XVIEWP(0.0,0.0,21.0,25.0)  
  X0=2.0  
  Y0=21.0  
  END IF  
  END IF  
1000 CONTINUE  
  CALL XEND  
  STOP  
END
```



出力例

P L T B 4 5

Line Drawing of 20 Binary Patterns Using XY-Plotter

X Y プロッターによる 20 個の 2 値パターンの線画出力

作成	鶴岡信治	1987年 6月
形式	サブルーチン	言語 : FORTRAN サイズ : 90 行

(1) 概要

文字画像の 2 値パターンを A4 版 1 枚に横 4 文字、 縦 5 文字、 計 20 文字、 X Y プロッターに横線で出力する。

(2) 使用法

CALL PLTB45(IA, I1, J1, MESG, MOJINO, MOJICA, MOJINN, MP)

引数	型と種類	属性	内容
IA	整数型 2 次元配列	入力	表示したい画像 IA (I1, J1)
I1	整数型	入力	IA の整合寸法 (第 1 添字の値)
J1	整数型	入力	IA の整合寸法 (第 2 添字の値)
MESG	文字型	入力	表示したい文字列 (8 文字以内)
MOJINO	整数型	入力	文字の通し番号 (7 衔以内)
MOJICA	整数型	入力	字種の番号 (7 衔以内)
MOJINN	整数型	入力	字種内の番号 (7 衔以内)
MP	整数型	入力	画面内の位置の番号 (1 - 20 の数字)

(3) 計算法

1画面に20文字分の2値パターンを画像の1行ごとに1本の横線（濃度値1の画素：線あり、0の画素：空白）でかき、表示位置はMPで制御できる。すなわち、MPの1-4は、1行目の左-右に対応し、5-8は2行目、9-12は3行目、13-16は4行目、17-20は5行目に対応する。またこのサブルーチンを呼ぶたびに、MPは自動的に1ずつ増加する。すなわち、最初に表示したい場所（左上はMP=1）を指定して置けば、次からは自動的に決定される。

文字枠の上にMSG（8文字以内の文字列）を、文字枠の下に左から順にMOJINO（7桁以内の数字）、MOJICA（7桁以内の数字）、MOJINN（7桁以内の数字）を書く。

(4) 子ルーチン

XVIEWP, FRAME, SYMBOL, NUMBER, PLT2TI

(5) プログラム例（メインプログラム）

テストプログラムの EXAMPLE.FORT(XY8HB) を参照のこと

PLTH4A

Line Drawing of 4-Direction Index Histogram Using XY-Plotter

XYプロッターによる4方向指数ヒストグラムの線画表現

作成	鶴岡信治	1986年	7月
形式	サブルーチン	言語 : FORTRAN	サイズ : 90行

(1) 概要

4方向指數ヒストグラムを、XYプロッターで各小領域の中心に放射状に描く。頻度はその方向の長さで表現されている。1画面に4文字分のヒストグラムをかく。

(2) 使用法

```
CALL PLTH4A(HIST4, IHH, JHH, MESG, MOJINN, MOJICA, MOJINN, MP)
```

引数	型と種類	属性	内容
HIST4	実数型 3次元配列	入力	表示したい4方向のヒストグラム (IHH, JHH, 4)
IHH	整数型	入力	HIST4の整合寸法 (第1添字の値)
JHH	整数型	入力	HIST4の整合寸法 (第2添字の値)
MESG	文字型	入力	表示したい文字列 (8文字以内)
MOJINO	整数型	入力	文字の通し番号 (7桁以内)
MOJICA	整数型	入力	字種の番号 (7桁以内)
MOJINN	整数型	入力	字種内の番号 (7桁以内)
MP	整数型	入力	画面内の位置の番号 (1-4の数字)

(3) 計算法

ヒストグラムの長さをSIN, COSで極座標に変換し、各小領域の中心位置に放射状に線をひく。長さの正規化はしていない。

1画面に4つのヒストグラムをかき、ヒストグラムの位置はMPで制御できる。すなわち、MPが1で紙面の左上、2で右上、3で左下、4で右下に表示する。またこのサブルーチンを呼ぶ度にMPは1ずつ増加する。すなわち、最初に表示したい場所を指定して置けば、次からは自動的に決定される。

文字枠の上にMESSAGEを、文字枠の下に左から順にMOJINO, MOJICA, MOJINNをかく。

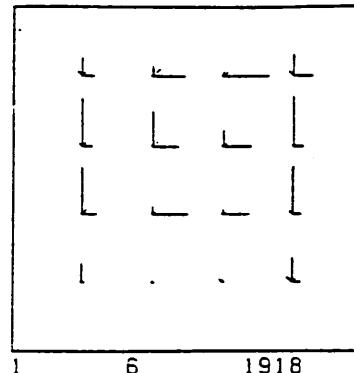
(4) 子ルーチン

PLTH4B (PLOT), FRAME, SYMBOL
NUMBER, XVIEWP

(5) プログラム例

```
REAL HIST4(4, 4, 4)
CHARACTER MSG*8
DATA MSG/'-4*4*4-'/
CALL XINT
MP=1
C*** MOJI NITUITE NO LOOP
DO 1000 MOJINO=1, 10
READ(8) NVOL, MOJICA, HIST4
CALL PLT4A(NVOL, MOJICA, MOJINO, HIST4, 4, 4, MSG, MP)
1000 CONTINUE
CALL XEND
STOP
END
```

- 4 * 4 * 4 -



出力例

P L T H 4 B

Line Drawing of 4-Direction Index Histogram Using XY-Plotter

XYプロッターによる4方向指数ヒストグラムの線画表現の子ルーチン

作成	鶴岡信治	1986年	7月
形式	サブルーチン	言語 : FORTRAN	サイズ : 25行

(1) 概要

小領域の4方向指数ヒストグラムを、XYプロッターで描く。頻度はその方向の長さで表現されている。

(2) 使用法

CALL PLTH4B(HIST4, IHH, JHH, I, J, R, CX, CY)

引数	型と種類	属性	内容
HIST4	実数型 3次元配列	入力	表示したい4方向のヒストグラム (IHH, JHH, 4)
IHH	整数型	入力	HIST4の整合寸法(第1添字の値)
JHH	整数型	入力	HIST4の整合寸法(第2添字の値)
I	整数型	入力	小領域の位置(1-IHH)
J	整数型	入力	小領域の位置(1-JHH)
R	実数型	入力	ヒストグラムの単位長さ当たりの半径
CX	実数型	入力	ヒストグラムの中心のX座標
CY	実数型	入力	ヒストグラムの中心のY座標

(3) 計算法

ヒストグラムの長さをSIN, COSで極座標に変換し、各小領域の中心位置に放射状に4本の線をひく。長さの正規化はしていないので、親ルーチンでRを調整すること。

(4) 子ルーチン

P L O T

PLTH45

Line Drawing of 20 4-Direction Index Histograms Using XY-Plotter

XYプロッターによる20個の4方向指數ヒストグラムの線画表現

作成	鶴岡信治	1987年 6月
形式	サブルーチン	言語 : FORTRAN サイズ : 80行

(1) 概要

文字パターンの加重方向指數ヒストグラム (IHH×JHHの小領域の4方向の成分) を A4版1枚に横4文字、縦5文字、計20文字分、XYプロッターに線画で出力する。

(2) 使用法

```
CALL PLTH45(HIST4, IHH, JHH, MESG, MOJINO, MOJICA, MOJINN, MP)
```

引数	型と種類	属性	内容
HIST4	実数型 3次元配列	入力	表示したい4方向のヒストグラム (IHH, JHH, 4)
IHH	整数型	入力	HIST4の整合寸法 (第1添字の値)
JHH	整数型	入力	HIST4の整合寸法 (第2添字の値)
MESG	文字型	入力	表示したい文字列 (8文字以内)
MOJINO	整数型	入力	文字の通し番号 (7桁以内)
MOJICA	整数型	入力	字種の番号 (7桁以内)
MOJINN	整数型	入力	字種内の番号 (7桁以内)
MP	整数型	入力	画面内の位置の番号 (1-20の数字)

(3) 計算法

1画面に20文字分のIHH×JHHの小領域の4方向のヒストグラム（線の長さでその方向の度数を表す放射状のグラフ）をかき、ヒストグラムの表示位置はMPで制御できる。すなわち、MPの1-4は、1行目の左-右に対応し、5-8は2行目、9-12は3行目、13-16は4行目、17-20は5行目に対応する。またこのサブルーチンを呼ぶたびにMPは自動的に1ずつ増加する。すなわち、最初に表示したい場所（左上はMP=1）を指定して置けば、次からは自動的に決定される。

文字枠の上にMESSAGE（8文字以内の文字列）を、文字枠の下に左から順にMOJINO（7桁以内の数字）、MOJICA（7桁以内の数字）、MOJINN（7桁以内の数字）を書く。

(4) 子ルーチン

XVIEWP, FRAME, SYMBOL, MEMBER, PLTH4B

(5) プログラム例（メインプログラム）

テストプログラムのEXAMPLE.FORT(XY8HIST)を参照

PLT2TI

Line Drawing of Binary Pattern Using XY-Plotter

XYプロッターによる2値パターンの線画出力

作成	鶴岡信治	1987年 6月
形式	サブルーチン	言語: FORTRAN サイズ: 90行

(1) 概要

文字画像をXYプロッターに横線（濃度値がしきい値以上の点：線あり、しきい値未満の点：空白）で出力する。PLTB45の小ルーチン。

(2) 使用法

```
CALL PLT2TI(IA, I1, J1, X0, Y0, DX, DY, ITHRSH)
```

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	表示したい画像 IA(I1, J1)
I1	整数型	入力	IAの整合寸法（第1添字の値）
J1	整数型	入力	IAの整合寸法（第2添字の値）
X0	実数型	入力	紙上の原点のX座標
Y0	実数型	入力	紙上の原点のY座標
DX	実数型	入力	紙上の文字枠のX方向の大きさ
DY	実数型	入力	紙上の文字枠のY方向の大きさ
ITHRSH	整数型	入力	2値化のしきい値

(3) 計算法

文字画像 I A を左上から水平に 1 行づつ走査していき、しきい値 I T H R S H 以上の点を見つけると、線を引き始め（ペンドウンし）、しきい値未満である点を見つけるとペンアップする。入力画像は多値画像でもよい。また、画像の縁の画素がしきい値以上であっても正常な動作をするので、大きさの正規化（N O R M B 3 など）をしたすぐ後にこのルーチンを用いても良い。

(4) 子ルーチン

P L O T

(5) プログラム例（親ルーチン）

P L T B 4 5

D S P G S P

Moji Pattern Display with GSP

G S Pを用いた文字パターンのグラフィックディスプレイ出力

作成	鈴木 隆之	1 9 8 6 年 3 月
形式	サブルーチン	言語 : FORTRAN サイズ : 1 6 8 行

(1) 概要

グラフィックディスプレイに出力したい文字パターンを IA (I 1, J 1) にセットして本ルーチンを呼ぶことにより IA (I 1, J 1) が IMOJIC, JMOJIC で指定した位置に出力される。

最初に横方向の出力文字位置指定を IMOJIC = 1, 縦方向の出力文字位置指定を JMOJIC = 1 と設定して、本ルーチンを呼べばあとは自動的に本ルーチン内で縦・横の位置指定パラメータ IMOJIC; JMOJIC を更新する。その時に、文字のセットする方向は横方向である。

(2) 使用法

CALL DSPGSP(IA, I1, J1, AISTEP, AJSTEP, IMOJI, JMOJI, IMOJIC, JMOJIC, FRAME, ILL)

引数	型と種類	属性	内容
IA	整数型 2次元 配列	入力	出力すべき文字パターン IA (I 1, J 1) $EP \geq 1.0$
I 1	整数型	入力	IA の整合寸法 (第 1 添字の寸法)
J 1	整数型	入力	IA の整合寸法 (第 2 添字の寸法)
A I S T E P	実数型	入力	横方向出力ステップ (枠の部分を除く) $I 1 / (I 1 + \text{横方向ブランク})$ $= A I S T E P \geq 1.0$

A J S T E P	実数型	入力	縦方向出力ステップ（枠の部分を除く） J 1 / (J 1 + 縦方向ブランク) = A J S T E P ≥ 1. 0
I M O J I	整数型	入力	横方向に出力する最大文字数
J M O J I	整数型	入力	縦方向に出力する最大文字数
I M O J I C	整数型	入出力	横方向の文字出力位置（本ルーチン内で自動的に更新する。）
J M O J I C	整数型	入出力	縦方向の文字出力位置（本ルーチン内で自動的に更新する。）
F R A M E	文字型	入力	文字の枠を指定。' YES' 枠有り ' NO' 枠無し
I L L	整数型	出力	I L L = 0 : 正常終了 I L L = 1 : 用紙が一杯になった。 I L L = 1 0 0 0 0 : パラメータが 不適当 IMOJIC > IMOJI OR JMOJIC > JMOJI

(3) IMOJIC, JMOJIC の更新規則

条件 ① IMOJIC = IMOJI

更新の結果

$$IMOJIC = 1, \quad JMOJIC = JMOJIC + 1$$

条件 上記①を満たさない時

更新の結果

$$IMOJIC = IMOJIC + 1, \quad JMOJIC = JMOJIC$$

(4) 使用例

上記のサブルーチンのテストを行うためのプログラムは、テストプログラムとして D I S P T S T にある。

(5) 注意

(a) 本サブルーチンはメモリ領域を大量に使用するため、LOGON時にSIZEパラメータを、S(100M)と指定するとよい。

例 LOGON TSS Z4999A/PASS SIZE(6144)

(b) 本サブルーチンは孫ルーチンとしてPLLINEをCALLしている。

INTDSP

Initialize for GSPDSP

サブルーチンGSPDSPの初期化

作成	鈴木 隆之	1986年 3月
形式	サブルーチン	言語: FORTRAN (GSPDSPのENTRY)

(1) 概要

グラフィックディスプレイに文字パターンを出力するGSPDSPの初期化ルーチンである。グラフィックディスプレイを使用するための領域を確保したり、グラフィックディスプレイのタイプを指定するためGSPDSPをCALLする前に必ず実行する必要がある。

(2) 使用法

CALL INTDGSP(DPTYPE, ILL)

引数	型と種類	属性	内容
DPTYPE	文字型 CHARACTER*5	入力	使用するグラフィックディスプレイのタイプを指定する。' T4014' または' F9431' を指定する
ILL	整数型	出力	ILL = 0 : 正常 ILL = 10000 : 異常 (DPTYPEの指定が正しくない)

E N D D S P

Close for GSPDSP

サブルーチン G S P D S P の終了処理

作成	鈴木 隆之	1 9 8 6 年 3 月
形式	サブルーチン	言語 : FORTRAN (G S P D S P の ENTRY)

(1) 概要

グラフィックディスプレイに文字パターンを出力する G S P D S P の終了処理ルーチンである。グラフィックディスプレイを使用するために確保した領域を解放する。

(2) 使用法

C A L L E N D D S P

P L L I N E

Generating PLOT's Data of Moji Pattern on Display

文字パターンをグラフィックディスプレイに出力するためのデータを作成する

作成	鈴木 隆之	1986年 3月
形式	サブルーチン	言語 : FORTRAN サイズ : 68行

(1) 概要

グラフィックディスプレイに出力すべき文字パターンを I A (I 1, J 1) にセットして本ルーチンを呼ぶことにより、 G S P の点表示ルーチン G P N T や線表示ルーチン G S L I N E のデータを作成する。

(2) 使用法

```
CALL PLLINE(IA, I1, J1, NXP, NYP, NX, NY, NYY, NL, NP)
```

引数	型と種類	属性	内容
I A	整数型 2次元配列	入力	出力すべき文字パターン I A (I 1, J 1)
I 1	整数型	入力	I A の整合寸法 (第 1 添字の寸法)
J 1	整数型	入力	I A の整合寸法 (第 2 添字の寸法)
N X P	整数型 1次元配列	出力	点表示データ (横方向座標)
N Y P	整数型 1次元配列	出力	点表示データ (縦方向座標)
N X	整数型 1次元配列	出力	線表示データ (横方向始点座標)

N Y	整数型 1次元配列	出力	線表示データ（縦方向始点座標）
N Y Y	整数型 1次元配列	出力	線表示データ（縦方向終点座標）
N L	整数型	出力	表示すべき線データの総数
N P	整数型	出力	表示すべき点データの総数

線は横方向に引くとしている。

(3) 注意

- (a)本サブルーチンはG S P D S Pの中からC A L Lされる孫ルーチンであり、一般の利用者はC A L Lする必要がない。
- (b)本サブルーチン内で使用する配列は、G S P D S P内で宣言している。表示することの出来る点・線の最大値はそれぞれ、点は5,000点であり、線は10,000本である。

F X P P L C

Moji Pattern Print Out by FAX or PPL , Parameter CHECK

ファクシミリ、PPL出力のためのパラメータのチェック

作成	竹下鉄夫	1986年 2月
形式	サブルーチン	言語：FORTRAN サイズ： 96行

(1) 概要

ファクシミリ、PPLに大量の文字パターンを出力するためのサブルーチンFX PPLRの種々のパラメータをチェックする。FX PPLRのパラメータICON Tの値により呼び出される。

ILL = 0 の場合のみ次のステップに進むことが出来る。

(2) 使用法

```
CALL FXPPLC(I1, J1, SIZE, I2, J2, AISTEP, AJSTEP,  
           IMOJI, JMOJI, IMOJIC, JMOJIC, FRAME, IWIDE, KSCALE, ILL)
```

引数	型と種類	属性	内容
I 1	整数型	入力	出力すべき文字パターンを2次元配列 IA (I 1, J 1)とした時の IAの整合寸法 (第1添字の寸法)
J 1	整数型	入力	IAの整合寸法 (第2添字の寸法)
S I Z E	文字型	入力	出力用紙の大きさ 'A4' または 'B4'
I 2	整数型	入力	出力用紙の横幅 'A4' = 1728 'B4' = 2048 これ以外は不可
J 2	整数型	入力	IBの整合寸法 (第2添字の寸法) 出力用紙の縦方向長さ。最大値は 'A'

			$4' = 2500, 'B4' = 3000$ (これより小さくても可)
A I S T E P	実数型	入力	横方向出力ステップ（枠の部分を除く） $I1 / (I1 + 横方向ブランク)$ $= AISTEP \geq 1.0$
A J S T E P	実数型	入力	縦方向出力ステップ（枠の部分を除く） $J1 / (J1 + 縦方向ブランク)$ $= AJSTEP \geq 1.0$
I M O J I	整数型	入力	横方向に出力可能な最大文字数
J M O J I	整数型	入力	縦方向に出力可能な最大文字数
I M O J I C	整数型	入力	横方向の文字出力位置
J M O J I C	整数型	入力	縦方向の文字出力位置
F R A M E	文字型	入力	文字の枠を指定。 'YES' 枠有り 'NO' 枠無し
I W I D E	整数型	入力	枠の幅（縦横共に同じ）
K S C A L E	整数型	入力	文字パターンの拡大率(KSCALE ≥ 1) 実際に用紙に出力される大きさは $IA(I1*KSCALE, J1*KSCALE)$ となる。
I L L	整数型	出力	$I L L = 0$: 正常終了 $I L L \neq 0$: パラメータが不適当。 この時には次のステップに進まないほうが良い。詳細は(3)項参照のこと

3 ILL の定義

- ① ILL = 0 正常
- ② ILL = 10000 I1 > I2 または J1 > J2
- ③ ILL = 12000 I1 * J1 > 1,000,000
- ④ ILL = 14000 FRAME が 'YES' または 'NO' でない。
- ⑤ ILL = 16000 SIZE が 'A4' または 'B4' でない。
- ⑥ ILL = 20000 KSCALE < 1 である。
拡大率は 1 以上でなければいけない。
- ⑦ ILL = 30000 横方向出力文字数 IMOJI がおかしい。
下記の式が成立しない。
- ① FRAME = 'YES' の場合 但し, [] はガウスの記号

$$\text{IMOJI} = [\text{I2}/(2*\text{IWIDE} - [\text{I1}*\text{AISTEP}]*\text{KSCALE})]$$
- ② FRAME = 'NO' の場合

$$\text{IMOJI} = [\text{I2}/([\text{I1}*\text{AISTEP}]*\text{KSCALE})]$$
- ⑧ ILL = 50000 縦方向出力文字数 JMOJI がおかしい。
下記の式が成立しない。
- ① FRAME = 'YES' の場合 但し, [] はガウスの記号

$$\text{JMOJI} = [(\text{J2}-\text{JTITLE})/(2*\text{IWIDE} + [\text{J1}*\text{AJSTEP}]*\text{KSCALE})]$$
- ② FRAME = 'NO' の場合

$$\text{JMOJI} = [(\text{J2}-\text{JTITLE})/([\text{J1}*\text{AJSTEP}]*\text{KSCALE})]$$
- ⑨ ILL = 60000 横方向出力文字位置を指定する IMOJI_C または
縦方向出力文字位置を指定する JMOJI_C がおかしい。
IMOJI_C > IMOJI または JMOJI_C > JMOJI である。

(4) 使用例

上記のサブルーチンのテストを行うためのプログラムは FAXTST1S にある。

F X P P L O

Moji Pattern Print Out by FAX or PPL , FAX or PPL Output

ファクシミリあるいは、PPLへ出力する

作成 竹下鉄夫 1986年 2月

形式 サブルーチン 言語 : FORTRAN サイズ : 72行

(1) 概要

サブルーチン F X P P L R によりパッキングした大量の文字データをファクシミリあるいは、FPLに出力する。

(2) 使用法

CALL FXPPLO(SIZE, IB, I2, J2, FAXPPL, ILL)

引数	型と種類	属性	内容
S I Z E	文字型	入力	出力用紙の大きさ ' A 4 ' または ' B 4 '
I B	整数型 2次元配列	出力	ファクシミリまたはPPLに出力される 1画素1ビットにパッキングされたデータを格納する2次元配列。 I B の大きさは I B (I 2 / 3 2 , J 2) である。
I 2	整数型	入力	出力用紙の横幅 ' A 4 ' = 1728 ' B 4 ' = 2048 これ以外は不可
J 2	整数型	入力	I B の整合寸法 (第2添字の寸法) 出力用紙の縦方向長さ。最大値は ' A ' 4 ' = 2500 , ' B 4 ' = 3000 (これより小さくても可)

FAXPPL	文字型	入力	出力装置を指定する。 'FAX' = ファクシミリ出力 'PPL' = Versatec Printer Plotter 出力
ILL	整数型	出力	ILL = 0 : 正常終了 ILL = 10000 : FAXPPLが 'FAX' 又は 'PPL' でない。 ILL ≠ 0 : 異常終了。

(3) 使用例

上記のサブルーチンのテストを行うためのプログラムはFAXTSTSにある。

参考文献(1)にあるように本ルーチンを含むプログラムを実行させる前にはTSS-MODEでIPPLと入力する必要がある。又、実際にPPLへ出力させる時にはTSS-MODEでOPPLE (ON)と入力する必要がある。

PPLに連続出力したいような時には、文献(1)を参考にして、適宜本ルーチンをモディファイしていただきたい。本ルーチンでは1枚しか出力できない。

(4) 参考文献

- (1)岡部直木：“画像入出力表示ルーチンと画像入力装置の性能”，名古屋大学大型計算機センターニュース，Vol. 14，No. 1，pp. 22～40 (1983)
- (2)“画像関係サブルーチンの追加登録について” 名古屋大学大型計算機センターニュース，Vol. 16，No. 3，pp. 290～293 (1985)

F X P P L P

Moji Pattern Print Out by FAX or PPL , Parameter & Title Input

ファクシミリ、PPL出力のためのパラメータ、タイトル設定

作成	竹下鉄夫	1986年 2月
形式	サブルーチン	言語：FORTRAN サイズ：134行

(1) 概要

ファクシミリ、PPLに大量の文字パターンを出力するための種々のパラメータ、タイトルを設定する。

特に横方向の出力文字数 (IMOJI) を指定して、拡大率 (KSCALE) を指定しない ($KSCALE < 1$) と、内部にて拡大率を算出する。逆に横方向出力文字数 (IMOJI) を指定しない ($IMOJI < 1$) で、拡大率を指定すれば、内部にて IMOJI を算出する。両者を指定した場合は横方向出力文字数を優先する。

一旦、設定したパラメータはサブルーチン FXPPLR はもとより、FXPPL0 を呼んで用紙に出力するまで変更してはならない。

(2) 使用法

```
CALL FXPPPLP(I1, J1, SIZE, IB, I2, J2, AISTEP, AJSTEP,  
IMOJI, JMOJI, FRAME, IWIDE, KSCALE, TITLE, ILL)
```

引数	型と種類	属性	内容
I1	整数型	入力	出力すべき文字パターンを 2 次元配列 IA (I1, J1) とした時の IA の整合寸法 (第 1 添字の寸法)
J1	整数型	入力	IA の整合寸法 (第 2 添字の寸法)
SIZE	文字型	入力	出力用紙の大きさ 'A4' または 'B4'

I B	整数型 2次元 配列	出力	ファクシミリまたはPPLに出力される 1画素1ビットにパッキングされたデータを格納する2次元配列。このルーチン 内で初期設定（すべて0）され、タイト ルがパッキングされる。I Bの大きさは I B (I 2 / 3 2, J 2) である。
I 2	整数型	入力	出力用紙の横幅 'A4' = 1728 'B4' = 2048 これ以外は不可
J 2	整数型	入力	I Bの整合寸法（第2添字の寸法） 出力用紙の縦方向長さ。最大値は 'A '4' = 2500, 'B4' = 3000 (これより小さくても可)
A I S T E P	実数型	入力	横方向出力ステップ（枠の部分を除く） $I1 / (I1 + \text{横方向ブランク})$ $= AISTEP \geq 1.0$ [IF(AISTEP ≤ 1.0) AISTEP=1.0]
A J S T E P	実数型	入力	縦方向出力ステップ（枠の部分を除く） $J1 / (J1 + \text{縦方向ブランク})$ $= AJSTEP \geq 1.0$ [IF(AJSTEP ≤ 1.0) AJSTEP=1.0]
I M O J I	整数型	入出力	横方向に出力可能な最大文字数
J M O J I	整数型	出力	縦方向に出力可能な最大文字数
F R A M E	文字型	入力	文字の枠を指定。 'YES' 枠有り 'NO' 枠無し
I W I D E	整数型	入力	枠の幅（縦横共に同じ）
K S C A L E	整数型	入出力	文字パターンの拡大率(KSCALE ≥ 1)

実際に用紙に出力される大きさは
IA(I1*KSCALE, J1*KSCALE) となる。

TITLE	文字型	入力	用紙上端のタイトル、出力可能な最大文字数は' A4' = 72, ' B4' = 85 出力可能な文字種別は0 ~ 9の数字、 A~Zの英大文字、- (マイナス) = (イコール) . (ピリオド) , (コンマ) _ (ブランク) の41字種のみ。これ以外の文字はブランクとなる。
ILL	整数型	出力	ILL = 0 : 正常終了 ILL ≠ 0 : パラメータが不適当。 この時には次のステップに進まないほうが良い。詳細は(6)項参照のこと

(3) KSCALEの算出法 ($IMOJI \geq 1$ の場合)

① FRAME = ' YES' の場合 但し, [] はガウスの記号

$$KSCALE=[I2/(2*IWIDE+[I1*AISTEP]*IMOJI)]$$

② FRAME = ' NO' の場合

$$KSCALE=[I2/([I1*AISTEP]*IMOJI)]$$

(4) IMOJIの算出法 ($KSCALE \geq 1$ の場合)

① FRAME = ' YES' の場合 但し, [] はガウスの記号

$$IMOJI=[I2/(2*IWIDE-[I1*AISTEP]*KSCALE)]$$

② FRAME = ' NO' の場合

$$IMOJI=[I2/([I1*AISTEP]*KSCALE)]$$

(5) JMOJIの算出法 ($KSCALE \geq 1$ は確定している)

① FRAME = ' YES' の場合 但し, [] はガウスの記号

$$JTITLE=32 \text{ (タイトルの縦幅)}$$

$$JMOJI=[(J2-JTITLE)/(2*IWIDE-[J1*AJSTEP]*KSCALE)]$$

② FRAME = ' NO' の場合

$$JMOJI=[(J2-JTITLE)/([J1*AJSTEP]*KSCALE)]$$

(6) ILLの定義

- ① ILL = 0 正常
- ② ILL = 10000 $I_1 > I_2$ または $J_1 > J_2$
- ③ ILL = 12000 $I_1 * J_1 > 1,000,000$
- ④ ILL = 14000 FRAMEが'YES'または'NO'でない。
- ⑤ ILL = 16000 SIZEが'A4'または'B4'でない。
- ⑥ ILL = 18000 I_2 がおかしい。
SIZE='A4'の時 $I_2 \neq 1728$
SIZE='B4'の時 $I_2 \neq 2048$
- ⑦ ILL = 20000 IMOJI < 1 かつ, KSCALE < 1 である。
横方向出力文字数か拡大率かどちらかを指定する必要がある。
- ⑧ ILL = 30000 横方向に1文字も出力出来ない。
KSCALE ≥ 1 の場合
下記の式で計算すると IMOJI < 1 となる。
① FRAME='YES'の場合 但し, [] はガウスの記号
 $IMOJI = [I_2 / (2 * IWIDE + [I_1 * AISTEP] * KSCALE)]$
- ② FRAME='NO'の場合
 $IMOJI = [I_2 / ([I_1 * AISTEP] * KSCALE)]$
- ⑨ ILL = 40000 横方向に1文字も出力出来ない。
IMOJI ≥ 1 の場合
下記の式で計算すると KSCALE < 1 となる。
① FRAME='YES'の場合 但し, [] はガウスの記号
 $KSCALE = [I_2 / (2 * IWIDE + [I_1 * AISTEP] * IMOJI)]$
- ② FRAME='NO'の場合
 $KSCALE = [I_2 / ([I_1 * AISTEP] * IMOJI)]$
- ⑩ ILL = 50000 縦方向に1文字も出力出来ない。
下記の式で計算すると JMOJI < 1 となる。
① FRAME='YES'の場合 但し, [] はガウスの記号
 $JMOJI = [(J_2 - JTITLE) / (2 * IWIDE + [J_1 * AJSTEP] * KSCALE)]$
- ② FRAME='NO'の場合
 $JMOJI = [(J_2 - JTITLE) / ([I_1 * AJSTEP] * KSCALE)]$

(7) 使用例

上記のサブルーチンのテストを行うためのプログラムはFAXPARASにある。

F X P P L R

Moji Pattern Print Out by FAX or PPL , Date Read & Packing

ファクシミリ、PPL出力のための文字パターンの入力とパッキング

作成	竹下鉄夫	1986年 2月
形式	サブルーチン	言語 : FORTRAN サイズ : 229行

(1) 概要

ファクシミリ、PPLに出力したい文字パターンをIA(I1, J1)にセットして本ルーチンを呼ぶことによりIA(I1, J1)がIWORK(I1, J1, IMOJIC)へ転送される。横1行分の転送が終了(IMOJI = IMOJICの時)した時点で、自動的にIWORK(I1, J1, IMOJI)が出力用2次元配列IB(I2/32, J2)にパッキングされる。

最初に横方向の出力文字位置指定をIMOJIC = 1、縦方向の出力文字位置指定をJMOJIC = 1と設定して、本ルーチンを呼べばあとは自動的に本ルーチン内で縦・横の位置指定パラメータIMOJIC, JMOJICを更新する。その時に、文字のセットする方向は横方向である。但し、最後に入力した1行についてはIMOJI = IMOJICでない限りパッキング(変換)をしないのでICONT制御フラッグにより、パッキングを指示する。

(2) 使用法

```
CALL FXPPLR(IA, I1, J1, SIZE, IB, I2, J2, IWORK, AISTEP, AJSTEP,  
IMOJI, JMOJI, IMOJIC, JMOJIC, FRAME, IWIDE, KSCALE, ICONT, ILL)
```

引数	型と種類	属性	内容
IA	整数型 2次元 配列	入力	出力すべき文字パターン IA(I1, J1)
I1	整数型	入力	IAの整合寸法(第1添字の寸法)
J1	整数型	入力	IAの整合寸法(第2添字の寸法)

S I Z E	文字型	入力	出力用紙の大きさ ' A 4' または ' B 4'
I B	整数型 2次元 配列	出力	ファクシミリまたはP P Lに出力される 1画素1ビットにパッキングされたデータを格納する2次元配列。 I B の大きさは I B (I 2 / 3 2, J 2) である。
I 2	整数型	入力	出力用紙の横幅 ' A 4' = 1 7 2 8 ' B 4' = 2 0 4 8 これ以外は不可
J 2	整数型	入力	I B の整合寸法(第2添字の寸法) 出力用紙の縦方向長さ。最大値は ' A 4' = 2 5 0 0, ' B 4' = 3 0 0 0 (これより小さくても可)
I W O R K	整数型	作業用	パッキングするための文字データを横1行分格納するための作業用3次元配列 I WORK (I 1, J 1, I M O J I) の大きさが必要
A I S T E P	実数型	入力	横方向出力ステップ(枠の部分を除く) I 1 / (I 1 + 横方向ブランク) = A I S T E P \geq 1. 0
A J S T E P	実数型	入力	縦方向出力ステップ(枠の部分を除く) J 1 / (J 1 + 縦方向ブランク) = A J S T E P \geq 1. 0
I M O J I	整数型	入力	横方向に出力可能な最大文字数
J M O J I	整数型	入力	縦方向に出力可能な最大文字数

I M O J I C	整数型	入出力	横方向の文字出力位置（本ルーチン内で自動的に更新する。）
J M O J I C	整数型	入出力	縦方向の文字出力位置（本ルーチン内で自動的に更新する。）
F R A M E	文字型	入力	文字の枠を指定。 ' YES' 枠有り ' NO' 枠無し
I W I D E	整数型	入力	枠の幅（縦横共に同じ）
K S C A L E	整数型	入力	文字パターンの拡大率($KSCALE \geq 1$) 実際に用紙に出力される大きさは $IA(I1*KSCALE, J1*KSCALE)$ となる。
I C O N T	整数型	入力	制御フラッグ パラメータのチェックと パッキングの実行を指示。その意味は ICONT パラメータチェック パッキング 0 しない しない 1 する しない 2 しない する 3 する する
I L L	整数型	出力	I L L = 0 : 正常終了 I L L = 1 : 用紙が一杯になった。もう これ以上は1枚に出力できない。 パラメータをチェックした時の I L L の 値については、FXPPLCの使用説 明書を参照のこと。

(3) I M O J I C, J M O J I C の更新規則

(1) I WORK をパッキングする場合

条件 ① I M O J I C = I M O J I

② I C O N T = 2 または I C O N T = 3

更新の結果

$$I M O J I C = 1 \quad , \quad J M O J I C = J M O J I C + 1$$

(2) I W O R K をパッキングしない場合

条件 上記①または②を満たさない時

更新の結果

$$I M O J I C = I M O J I C + 1 \quad , \quad J M O J I C = J M O J I C$$

(4) 使用例

上記のサブルーチンのテストを行うためのプログラムは F A X T S T 1 S にある。

BITSET

Bit & Byte Pattern set

文字型データにパッキングされている1バイトのデータのビットパターンをセットする（アンパックすることなくヒストグラムを求めるため）

作成	竹下 鉄夫	1987年 9月
形式	サブルーチン	言語：FORTRAN サイズ：173行

(1) 概要

文字型データに1バイト8ドット分パッキングされている、画像データのヒストグラムをアンパックすることなく求めるため1バイトの文字型データの、ビットパターンをセットする。

出力BITPATは、1バイトの文字型データを整数に変換したとき、その整数値を添字とする画像データの1バイト中に、いくつのビットに1がたっているかを示す。たとえば BITPAT(0)は0（オール0）、BITPAT(1)は1、BITPAT(2)も1、BITPAT(3)は2である。BITPAT(255)は8である。

出力BYTPATには、1バイトの文字型データを整数に変換したとき、その整数値を第2添字とする画像データの1バイト中に、第1添字で指定した位置のビットが0か1かを示す。なおこのとき、ビット1（LSB）は画像の最も左側、ビット8（MSB）が画像の最も右側とする。以下に、いくつかの例を示す。

BYTPAT(1, 0) = 0, …… BYTPAT(7, 0) = 0, BYTPAT(8, 0) = 0

BYTPAT(1, 1) = 0, …… BYTPAT(7, 1) = 0, BYTPAT(8, 1) = 1

BYTPAT(1, 2) = 0, …… BYTPAT(7, 2) = 1, BYTPAT(8, 2) = 0

BYTPAT(1, 3) = 0, …… BYTPAT(7, 3) = 1, BYTPAT(8, 3) = 1

…

…

…

BYTPAT(1, 128) = 1, …… BYTPAT(7, 128) = 0, BYTPAT(8, 128) = 0

…

…

…

BYTPAT(1, 255) = 1, …… BYTPAT(7, 255) = 1, BYTPAT(8, 255) = 1

(2) 使用法

CALL BITSET(BITPAT, BYTPAT)

引数	型と種類	属性	内容
B I T P A T	整数型 1次元配列	出力	出力すべき1ビットの数のパターン B I T P A T (0 : 255)
B Y T P A T	整数型 2次元配列	出力	出力パターン B Y T P A T (8, 0 : 255)

(3) 注意

パッキングされている画像データのヒストグラムアンパックしないで求めるRWHIST, CLHISTの中で、このBITPAT, BYTPATを使用しているので、RWHISTやCLHISTの中から呼び出される孫ルーチンである。

(4) 使用例

上記のサブルーチンのテストを行うためのプログラムはT S T H I S T S にある。

ELBLNK

Blank Elimination of Binary Image

2値文字画像の不要な空白を除去

作成	竹下 鉄夫	1988年 9月
形式	サブルーチン	言語 : FORTRAN サイズ : 61行

(1) 概要

文字の周囲の不要な空白を消去するためのサブルーチン。枠の中心と文字の中心を合わせる。文字のサイズはまったく変更しない。出力配列の大きさより大きい文字については何もしないでエラーコードを返す。

(2) 使用法

```
CALL ELBLNK(IA, I1, J1, IB, I2, J2, ILL)
```

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	入力文字画像の2次元配列 大きさは IA (I1, J1)
I1	整数型	入力	IAの整合寸法 (第1添字の値)
J1	整数型	入力	IAの整合寸法 (第2添字の値)
IB	整数型 2次元配列	出力	出力文字画像の2次元配列 大きさは IB (I2, J2)
I2	整数型	入力	IBの整合寸法 (第1添字の値)
J2	整数型	入力	IBの整合寸法 (第2添字の値)
ILL	整数型	出力	ILL = 0 : 正常終了

		I L L = 900, 901, 902, 903 : 入力文字画像がオール0 (N S I Z E Bによる) I L L = 10000 : I 2 < 横方向のサイズ I L L = 20000 : J 2 < 縦方向のサイズ I L L = 30000 : ILL=10000 & ILL=20000 I L L ≠ 0 のとき、出力画像は保証され ない
--	--	---

(3) 計算法

入力文字画像の大きさを N S I Z E B で求める。横 I (= I R - I L), 縦 J (= J D - J U) (但し I R, I L, J U, J D はサブルーチン N S I Z E B で使用している引数と同じでそれぞれ、I R : 右端, I L : 左端, J U : 上端, J D : 下端である) と計算し、縦横が新しい配列にはいるか否かチェックする。

チェックの式は 横 : $I < (I 2 - 1)$
縦 : $J < (J 2 - 1)$

上記二つの式を満たすときのみ本サブルーチンは動作する。

実際の変換は、変換後の座標が変換前のどの座標に相当するか各点について計算しながら大きさの変換を行う。

(4) 注意

本サブルーチンは孫ルーチンとして N S I Z E B を含む。

(5) 使用例 (サンプルプログラム)

(a)市松模様を生々し、その後周囲にゼロを付加するサブルーチン M S W A K U を使い不要なゼロをつける。しかる後、E L B L N K サブルーチンを呼び不要なゼロが除去されていることを印刷サブルーチン P O U T 2 B を使って確認している。

(b) 使用法

ジョブ制御文のデータセットが EXAMPLE.CNTL(ELBLNK)に用意されているので SUBMITすればよい。

プログラム本体は EXAMPLE.FORT(ELBLNK)に格納されている。

(c) 使用しているサブルーチン

E L B L N K, P O U T 2 B, M S W A K U (いずれも S M A R T に含まれるサブルーチン)

RWHIST

Row Histogram calculation

文字型データにパッキングされている画像データをアンパックすることなく、横(行)方向のヒストグラムを求める

作成	竹下 鉄夫	1987年 2月
形式	サブルーチン	言語 : FORTRAN サイズ : 39行

(1) 概要

文字型データに1バイト8ドット分パッキングされている、画像データのヒストグラムをアンパックすることなく求める。

入力画像の大きさは、IA(I1, J1)である。しかし、横方向には8ドットのデータが1バイトにパッキングされているため、実際の画像の大きさはIA(I1 × 8, J1)である。

ファクシミリ装置で入力した画像のヒストグラムを計算するときに使用すると便利なルーチンである。

このルーチンでは、I1, ISTROW, IENDRW等、横方向はバイト単位(画像では8ドットに相当)にカウントしているので注意が必要である。

(2) 使用法

```
CALL RWHIST(IA, I1, J1, ISTROW, IENDRW, IBHIST, ILL)
```

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	ヒストグラムを計算すべき画像パターン IA(I1, J1), 実際の画像の大きさはIA(I1 × 8, J1)である
I1	整数型	入力	IAの整合寸法(第1添字の寸法)
J1	整数型	入力	IAの整合寸法(第2添字の寸法)
ISTROW	整数型	入力	ヒストグラムのカウントを始める横方向

位置（実際の画像では、ISTROW×8 ドットの位置である）

I E N D R W	整数型	入力	ヒストグラムのカウントを終わる横方向位置（実際の画像では、IENDRW×8 ドットの位置である）
I B H I S T	整数型 1 次元配列	出力	横（行）方向ヒストグラム I B H I S T (J 1)
I L L	整数型	出力	I L L = 0 : 正常終了 I L L = 1 0 0 0 0 : ISTROW < 1 又は ISTROW > I1 I L L = 2 0 0 0 0 : IENDRW < ISTROW 又は IENDRW > I1

(3) 注意

パッキングされている画像データ IA (I 1, J 1) のヒストグラムをアンパックしないで求めるため、ビットパターンBITPATを使用しているので、このRWHISTの中からサブルーチンBITSETを呼び出している。

(4) 使用例

上記のサブルーチンのテストを行うためのプログラムはTSTHISTSにある。

C L H I S T

Column Histogram calculation

文字型データにパッキングされている画像データをアンパックすることなく、縦(列)方向のヒストグラムを求める

作成	竹下 鉄夫	1987年 2月
形式	サブルーチン	言語 : FORTRAN サイズ : 42行

(1) 概要

文字型データに1バイト8ドット分パッキングされている、画像データのヒストグラムをアンパックすることなく求める。

入力画像の大きさは、IA(I1, J1)である。しかし、横方向には8ドットのデータが1バイトにパッキングされているため、実際の画像の大きさはIA(I1 × 8, J1)である。

ファクシミリ装置で入力した画像のヒストグラムを計算するときに使用すると便利なルーチンである。

(2) 使用法

```
CALL CLHIST(IA, I1, J1, JSTCLM, JENDCL, IBHIST, ILL)
```

引数	型と種類	属性	内容
IA	整数型 2次元配列	入力	ヒストグラムを計算すべき画像パターン IA(I1, J1), 実際の画像の大きさはIA(I1 × 8, J1)である
I1	整数型	入力	IAの整合寸法(第1添字の寸法)
J1	整数型	入力	IAの整合寸法(第2添字の寸法)
JSTCLM	整数型	入力	ヒストグラムのカウントを始める縦方向位置(列番号)

JENDCL	整数型	入力	ヒストグラムのカウントを終わる縦方向位置（列番号）
IBHIST	整数型 1次元配列	出力	縦（列）方向ヒストグラム IBHIST (I1 × 8)
ILL	整数型	出力	ILL = 0 : 正常終了 ILL = 10000 : JSTCLM < 1 又は JSTCLM > J1 ILL = 20000 : JENDCL < JSTCLM 又は JENDCL > J1

(3) 注意

パッキングされている画像データIA(I1, J1)のヒストグラムをアンパックしないで求めるため、ビットパターンBITPAT, BYTPATを使用しているので、このCLHISTの中でサブルーチンBITSETを呼び出している。

(4) 使用例

上記のサブルーチンのテストを行うためのプログラムはTSHISTSにある。

SOUKAN

Auto-Correlation Coefficient Calculation

1次元配列の自己相関係数を求める

作成	竹下 鉄夫	1987年 2月
形式	サブルーチン	言語 : FORTRAN サイズ : 38行

(1) 概要

1次元配列の自己相関係数AUTCOR(i)を求める。計算式は下記による。

$$R(i) = \sum_{j=1}^{I_{\text{MAX}}-i+1} X(j) \times X(j+i-1)$$

$$\text{AUTCOR}(i) = R(i) / R(1)$$

(2) 使用法

CALL SOUKAN(IHIST, IMAX, AUTCOR, ILL)

引数	型と種類	属性	内容
I H I S T	整数型 1次元配列	入力	自己相関を計算すべきヒストグラム I H I S T (IMAX)
I M A X	整数型	入力	I H I S T の整合寸法
A U T C O R	実数型 1次元配列	出力	自己相関係数 AUTCOR(IMAX)
I L L	整数型	出力	$I L L = 0$: 正常終了 $I L L = 1 0 0 0 0$: AUTCOR(1) ≤ 0 I L L = 上記以外 : AUTCOR(1) \neq MAX $A U T C O R (I L L) = M A X$

(3) 使用例

上記のサブルーチンのテストを行うためのプログラムはT S T S O U K N にある。

索 引

B

B I N A R Y 1 2 B I T S E T 1 5 1

C

C A N C A L 5 1 C D 1 9 9
C L H I S T 1 5 7

D

D C O D E 4 7 4 D S E G M T 7 2
D S P G S P 1 3 1

E

E D 1 1 0 0 E D 2 1 0 1
E L B L N K 1 5 3 E N D D S P 1 3 5

F

F I L T M 4 5 F I L T M 8 7
F X P P L C 1 3 8 F X P P L O 1 4 1
F X P P L P 1 4 3 F X P P L R 1 4 7

G

G C E N T B 2 1 G C E N T M 2 3

H

H I S T S S 7 0 H K I Y O 8 0

I

I N T D S P 1 3 4

J

J O I N T 4 7 8

K

K I Y O D O 8 6

L

L D 1 1 0 2

M

M D 1 1 0 3 M Q D 1 1 0 5
M Q D 2 1 0 7 M S 1 1 0 9
M S T A T 1 8 7 M S T A T 2 8 9

M

M S T A T 3	9 1	M S T A T 4	9 3
M S T A T 5	9 5	M S T A T 6	9 7
M S W A K U	1 1 7			

N

N O R M B 1	3 1	N O R M B 2	3 3
N O R M B 3	3 5	N O R M B 4	3 7
N O R M B 7	3 9	N O R M M 1	4 1
N O R M M 2	4 3	N O R M M 3	4 5
N O R M N L	4 7	N O R M N 2	4 9
N S I Z E B	1 7	N S I Z E M	1 9
N T R N S 1	2 5	N T R N S 2	2 7
N T R N S 3	2 9			

P

P L L I N E	1 3 6	P L T 2 T I	1 2 9
P L T B 4 5	1 2 1	P L T B I A	1 1 8
P L T H 4 5	1 2 7	P L T H 4 A	1 2 3
P L T H 4 B	1 2 5	P O U T 1 6	1 1 5
P O U T 2 B	1 1 6	P T G E N 4	7 6
P S I Z E B	1 3	P S I Z E M	1 5

Q

Q D 1	1 1 1
-------	------	-------

R

R W H I S T	1 5 5
-------------	------	-------

S

S M O O T H	9	S O U K A N	1 5 9
S S 1	1 1 3	S S 2	1 1 4

T

T K I Y O	8 2
-----------	------	-----

W

W E I 2 2 4	5 9	W E I 4 4 4	6 1
W E I G T 1	5 2	W E I G T 2	5 5
W G T I N 1	6 3	W G T I N 2	6 5
W G T I N 3	6 7	W G T I N 4	6 8

X

X K I Y O	8 4
-----------	------	-----

文字認識用
プログラムパッケージ
S M A R T

利用手引書

テストプログラム篇

豊田工業高等専門学校 情報工学科

野澤繁之
竹下鉄夫
岡部直木

三重大学 工学部 電子工学科
鶴岡信治

三重大学 工学部 情報工学科
木村文隆

中部大学 経営情報学部
吉村ミツ

平成8年3月

SMARTテストプログラム 目次

プログラム名	処理内容	ページ
* テストプログラムに含まれるサブルーチン及び関数一覧		1
* サブルーチン・関数とテストプログラム		4
1. 前処理、特徴抽出用のテストルーチン		7
人工パターン		7
T S T P O U 2 S	2値文字画像をLP（ラインプリンタ）に 1H*、1Hで出力する。	7
T S T P O U 1 6	16値文字画像をLPに16進数で出力する。 (ただし0はブランク)	10
S T W G T 2	重み関数フィルタによる第2標本化処理結果 の出力作成までの各ルーチンの処理結果の出力	12
文字パターン		14
R D S A M P B I	2値画像からパターンマッチング法の特徴 ベクトル作成までの各ルーチンの処理結果の出力	14
R D S A M P H X	16値画像からパターンマッチング法の特徴 ベクトル作成までの各ルーチンの処理結果の出力	18
R D S A M P K T	2値画像からパターンマッチング法の特徴 ベクトル作成までの各ルーチンの処理結果の出力	20
P R E M A T C H	パターンマッチング法の特徴ベクトル作成まで の各ルーチンの処理結果の出力	21
F E A T U R E B	2値画像から加重方向ヒストグラム法の特徴 ベクトル作成までの各ルーチンの処理結果の出力	22
F E A T U R E H	16値画像から加重方向ヒストグラム法の特徴 ベクトル作成までの各ルーチンの処理結果の出力	23
D S E G M T C K	2値文字画像から4方向セグメントを求め、 結果をプリンタ又はディスプレイに表示する。	25
H K I Y O T S T	2値文字画像から外郭方向寄与度を求め、 結果をプリンタ又はディスプレイに表示する。	29
E L B L N K	2値画像にゼロを付加し後にゼロを除去する 各処理結果を出力する。	31
N O R M B 7	2値画像にゼロを付加し、後に正規化する。 各処理結果を出力する。	32

(次ページに続く)

プログラム名	処理内容	ページ
2. 特徴ベクトル作成プログラム例		
CREATEBI	2値画像からパターンマッチング法の特徴ベクトルを作成する。	34
CREATEHX	16値画像からパターンマッチング法の特徴ベクトルを作成する。	36
CREATEKT	2値画像からパターンマッチング法の特徴ベクトルを作成(サンプル数が少ない時の例)	38
CREFILEB	2値画像から加重方向ヒストグラム法の特徴ベクトルを作成する。	40
CREFILEH	16値画像から加重方向ヒストグラム法の特徴ベクトルを作成する。	42
3. 辞書作成及び識別テストルーチン		44
RECTEST0	16値文字画像の標本化を行い、加重方向指數ヒストグラム法の特徴ベクトルを求め、辞書作成のためのファイルを作成する。	44
RECTEST1	文字認識(小規模実験用) RECTEST0で求めた特徴ベクトルを用いてMSTAT4、MSTAT5により識別用辞書を作成、識別関数CD1、MQD2、SS1、MS1の値を計算、それをLPに出力する。	46
RECTEST2	認識用辞書作成(大規模実験用) RECTEST0で求めた特徴ベクトルを用いてMSTAT5により識別用辞書を作成し、ファイルに出力する。	49
RECTEST3	文字認識(大規模実験可) RECTEST0で求めた特徴ベクトルを用い、RECTEST2を用いて識別用辞書を作成する テスト用の文字データを読み、加重方向指數を求めて、作成済みの識別用辞書との間で、修正 2次識別関数MQD2により識別実験を行い、誤読文字のリストと正読率をLPに出力する。	51

(次ページに続く)

プログラム名	処理内容	ページ
4. 印刷、プロッター、ファックスなどへの出力テストルーチン		53
D I S P T S T	文字画像をグラフィックディスプレイに出力	53
X Y 8 H B	2値画像をA4版に横4*縦5計20文字 プロッター、O P R等に出力する。	54
X Y 8 H I S T	加重方向指数ヒストグラムをA4版大で 横4*縦5計20文字分プロッター、O P R 等に出力する。	57
X Y 8 H I S T B	2値画像と加重方向指数ヒストグラムを A4版大で横4*縦5計20文字分プロッター、 O P R等に出力する。	60
F A X P A R A S	パラメータセットルーチンF X P P L P の使用説明	63
F A X T S T I S	FAX、VERSATEC出力ルーチン F X P P L P、F X P P L R、F X P P L O の使用説明（その1）	65
F A X T S T 2 S	FAX、VERSATEC出力ルーチン F X P P L P、F X P P L R、F X P P L O の使用説明（その2）	66
R D T E S T B I	C R E A T E B Iで作成した特徴ベクトル を読み出して出力する。	67
R D T E S T H X	C R E A T E H Xで作成した特徴ベクトル を読み出して出力する。	69
R D T E S T K T	C R E A T E K Tで作成した特徴ベクトル を読み出して出力する。	70
5. その他のサービスルーチンのテストルーチン		71
T S T H I S T S	パッキング画像データのヒストグラムを求 めて、出力する。	71
T S T S O U K N	画像の横方向ヒストグラムの自己相関係数 を求めて出力する。	72
L I N E S G	特徴点を表す座標点列から2値の線文字 画像を求めて出力する。	73

SMART テストプログラム（コンプリートプログラム） に含まれるサブルーチン及び関数一覧表

注意

下線のあるルーチンはSMARTに含まれていないもの、即ち、SPIDER、SLIP、名古屋大学大型計算機センターに登録されている、等のルーチンである。
(配列はテストプログラム名のアルファベット順)

テストプログラム名	引用されるサブルーチン又は関数名		
CREATEBI	-WEIGT1 -WGTIN1 -MSWAKU	-RETL -NORMB1 -SMOOTH	-FILTM4 -SMOOTH
CREATEHX	-WEIGT1 -HIST1 -WGTIN1	-NORMM1 -RETL -SMOOTH	-THDS2 -FILTM8 -MSWAKU
CREATEKT	-WEIGT2 -WGTIN1 -MSWAKU	-RETL -NORMB1	-FILTM4 -SMOOTH
CREFILEB	-WEIGT1 -FILTM8 -SMOOTH	-RETL -WGTIN4 -HISTSS	-BDFL2 -MSWAKU
CREFILEH	-WEIGT1 -BDFL2 -WGTIN4 -HISTSS	-NORMM3 -RETL -MSWAKU	-BINARY -FILTM8 -SMOOTH
DISPTST	-RETL -DSPGSP	-INTDSP	-ENDDSP
DSEGMTCK	-POUT2B -DSEGMENT -MSWAKU	-RETL -NORMB1	-FILTM4 -SMOOTH
ELBLNK	-POUT2B (-NSIZEB -POUT2B)	-ELBLNK	-MSWAKU
FAXPARAS	-FXPPLP (-PACKL -UNPKL)	-FXPPLC	
FAXTSTS	-FXPPLO -FXPPLR	-FXPPLP -FXPPLC	-RETL

テストプログラム名	引用されるサブルーチン又は関数名		
FAXTST2S	-FXPPLO -FXPPLC	-FXPPLP <u>-RETL</u>	-FXPPLR
FEATUREB	-WEIGT1 -BDFL2 -SMOOTH	-POUT2B -WGTIN4 -HISTSS	<u>-RETL</u> -MSWAKU -NORMB4
FEATUREH	-WEIGT1 -NORMM3 <u>-RETL</u> -MSWAKU	-POUT2B -BINARY -FILTM8 -SMOOTH	-POUT16 -BDFL2 -WGTIN4 -HISTSS
HKIYOTST	<u>-RETL8B</u> -SMOOTH	-HKIYO -MSWAKU	-FILTM8
LINESG	-LINESG	-POUT16	
NORMB7	-NORMB7 (-NSIZEB -NTRNS3)	-POUT2B	-MSWAKU
PREMATCH	-WEIGT1 -BINARY <u>-HIST1</u> -NORMB1	-POUT2B -FILTM4 <u>-RETL</u> -MSWAKU	-POUT16 -THDS2 -WGTIN1
RDSAMPB1	-WEIGT1 -FILTM4 -SMOOTH	-POUT2B -WGTIN1 -MSWAKU	<u>-RETL</u> -NORMB1
RDSAMPHX	-WEIGT1 <u>-THDS2</u> -WGTIN1 -MSWAKU	-NORMM1 <u>-HIST1</u> -FILTM8	-POUT16 <u>-RETL</u> -SMOOTH
RDSAMPKT	-WEIGT2 -FILTM4 -SMOOTH	-POUT2B -WGTIN1 -MSWAKU	<u>-RETL</u> -NORMB1
RDSAMPKT	-WEIGT2 -FILTM4 -SMOOTH	-POUT2B -WGTIN1 -MSWAKU	<u>-RETL</u> -NORMB1
RDTESTB1	-なし		

テストプログラム名	引用されるサブルーチン又は関数名		
R D T E S T H X	-なし		
R D T E S T K T	-なし		
R E C T E S T 0	-WE I G T 1	-B I N A R Y	-R E T L
	-B D F L 2	-T H D S 2	-H I S T 1
	-F I L T M 8	-W G T I N 4	-M S W A K U
	-S M O O T H	-N O R M B 3	-H I S T S S
R E C T E S T 1	-M S 1	-M S T A T 4	-M S T A T 5
	-M Q D 2	-C D 1	-S S 1
	(-M S T A T 2	-M S T A T 3	-H O Q R U
R E C T E S T 2	-M S T A T 5	(-M S T A T 2	-H O Q R U
R E C T E S T 3	-M Q D 2	-S O R T S I	
T S T H I S T S	-R W H I S T	-C L H I S T	
T S T P O U 1 6	-P O U T 1 6	-M S W A K U	
T S T P O U 2 S	-P O U T 2 B	-M S W A K U	
T S T S O U K N	-R W H I S T	-S O U K A N	
T S T W G T 2	-W E I G T 2	-P O U T 2 B	-W G T I N 1
	-M S W A K U		
X Y 8 H B	-P L T B 4 5	-B I N A R Y	-R E T L
	-F I L T M 8	-X V I E W P	-X E N D
	-X I N T	-S M O O T H	-N O R M B 3
X Y 8 H I S T	-W E I G T 1	-B I N A R Y	-B D F L 2
	-R E T L	-F I L T M 8	-X V I E W P
	-X E N D	-W G T I N 4	-X I N T
	-P L T H 4 5	-M S W A K U	-S M O O T H
	-N O R M B 3	-H I S T S S	
X Y 8 H I S T B	-W E I G T 1	-P L T B 4 5	-B I N A R Y
	-B D F L 2	-R E T L	-F I L T M 8
	-X V I E W P	-X E N D	-W G T I N 4
	-X I N T	-P L T H 4 5	-M S W A K U
	-S M O O T H	-N O R M B 3	-H I S T S S

サブルーチン・関数とテストプログラム

以下の表は、SMARTに登録されているサブルーチン、あるいは関数が、どのテストプログラムに含まれているかを示すものである。この表にないサブルーチンあるいは関数は、テストプログラムでは引用していない。

サブルーチン 又は関数名	テストプログラム名
B I N A R Y	B → C R E F I L E H, → F E A T U R E H, → P R E M A T C H, → T E S T K 0, → X Y 8 H B, → X Y 8 H I S T, → X Y 8 H I S T B
C D 1	C → T E S T K 1
C L H I S T	→ T S T H I S T S
D S E G M T	D → D S E M T C K
D S P G S P	→ D S P T S T
E L B L N K	E → E L B L N K
E N D D S P	→ D S P T S T
F I L T M 4	F → C R E A T E B I, → C R E A T E K T, → D S E G M T C K, → P R E M A T C H, → R D S A M P B I, → R D S A M P K T
F I L T M 8	→ C R E A T E K X, → C R E F I L E B, → C R E F I L E H, → F E A T U R E B, → H K I Y O T S T, → R D S A M P H X, → T E S T K 0, → X Y 8 H B, → X Y 8 H I S T, → X Y 8 H I S T B
F X P P L C	→ F A X P A R A S, → F A X T S T 1 S, → F A X T S T 2 S

(次ページに続く)

サブルーチン 又は関数名	テストプログラム名
F X P P L O	F → F A X T S T 1 S, → F A X T S T 2 S
F X P P L P	→ F A X P A R A S, → F A X T S T 1 S, → F A X T S T 2 S
F X P P L R	→ F A X T S T 1 S, → F A X T S T 2 S
H I S T S S	H → C R E F I L E B, → C R E F I L E H, → F E A T U R E B, → F E A T U R E H, → T E S T K 0, → X Y 8 H I S T, → X Y 8 H I S T B
H K I Y O	→ H K I Y O T S T
I N T D S P	I → D I S P T S T
M Q D 2	M → T E S T K 2, → T E S T K 3
M S 1	→ T E S T K 1
M S T A T 2	→ T E S T K 1, → T E S T K 2
M S T A T 3	→ T E S T K 1
M S T A T 4	→ T E S T K 1
M S T A T 5	→ T E S T K 1, → T E S T K 2
M S W A K U	→ C R E A T E B I, → C R E A T E H X, → C R E A T E K T, → C R E F I L E B, → C R E F I L E H, → D S E G M T C K, → E L B L N K, → F E A T U R E B, → F E A T U R E H, → H K I Y O T S T, → N O R M B 7 → P R E M A T C H, → R D S A M P B I, → R D S A M P H X.

(次ページに続く)

サブルーチン 又は関数名	テストプログラム名
MSWAKU	<p style="text-align: center;">M</p> → RDSAMPKT, → TSTPOU16, → TSTPOU2S, → XY8HISTB
NORMB1	<p style="text-align: center;">N</p> → CREATEBI, → DSEGMTCK, → RDSAMPB1.
NORMB3	→ TESTK0, → XY8HIST,
NORMB4	→ FEATUREB
NORMB7	→ NORMB7
NORMM1	→ CREATEHX, → RDSAMPHX
NORMM3	→ CREFILEH, → FEATUREH
NSIZEB	→ ELBLNK, → NORMB7
NTRNS3	→ NORMB7
PLTB45	<p style="text-align: center;">P</p> → XY8HB, → XY8HISTB
PLTH45	→ XY8HIST, → XY8HISTB
POUT16	→ FEATUREH, → PREMATCH, → TSTPOU16
POUT2B	→ DSEGMTCK, → FEATUREB, → NORMB7, → RDAMPB1, → TSTPOU2S,
	→ ELBLNK, → FEATUREH, → PREMATCH, → RDAMPKT, → TSTWGT2

(次ページに続く)

マップルーチン 又は関数名	テストプログラム名	
RWHIST	R → T S T H I S T S ,	→ T S T S O U K N
SMOOTH	S → C R E A T E B I , → C R E A T E K T , → C R E F I L E H , → F E A T U R E B , → H K I Y O T S T , → R D S A M P H X , → T E S T K 0 , → X Y 8 H I S T ,	→ C R E A T E H X , → C R E F I L E B , → D S E G M T C K , → F E A T U R E H , → R D S A M P B I , → R D S A M P K T , → X Y 8 H B , → X Y 8 H I S T B
SOUKAN	→ T S T S O U K N	
SS1	→ T E S T K 1	
WEIGT1	W → C R E A T E B I , → C R E F I L E B , → F E A T U R E B , → P R E M A T C H , → R D S A M P H X , → X Y 8 H I S T ,	→ C R E A T E H X , → C R E F I L E H , → F E A T U R E H , → R D S A M P B I , → T E S T K 0 , → X Y 8 H I S T B
WEIGT2	→ C R E A T E K T , → T S T W G T 2 → R D S A M P B I , → R D S A M P K T ,	→ R D S A M P K T , → P R E M A T C H , → R D S A M P H X ,
WTIN1	→ C R E A T E B I , → C R E A T E K T , → R D S A M P B I , → R D S A M P K T .	→ C R E A T E H X , → P R E M A T C H , → R D S A M P H X , → T S T W G T 2
WTIN4	→ C R E F I L E B , → F E A T U R E B , → T E S T K 0 , → X Y 8 H I S T B	→ C R E F I L E H , → F E A T U R E H , → X Y 8 H I S T .

(終)

T S T P O U 2 S

Test Program for POUT2B subroutine

POUT2Bサブルーチンのテストプログラム

作成	竹下鉄夫	1985年 6月
形式	コンプリート・プログラム	言語:FORTRAN サイズ: 70行

(1) 概要

2値画像を人工的に発生し（横12×縦12の市松模様）、MSWAKUにより周囲に幅3の'0'をつけ、POUT2Bにより各処理毎の途中結果をラインプリンタに出力する。

TSSモードでの実行では、出力結果は端末のディスプレイ画面上になる。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(TSTPOU2S)に用意されているのでSUBMITすればよい。

プログラム本体はEXAMPLE.FORT(TSTPOU2S)に格納されている。

EXAMPLE.FORT(TSTPOU2S)をTSSモードで実行することもできる。

この場合には、TSS端末より以下のようにコマンドを入力すればよい。

FORT EXAMPLE.FORT(TSTPOU2S) GO(NOAE)

また、PFILE等のエディタでTSTPOU2Sを編集中であれば、上部のコマンド入力欄に、単に

FORT GO(NOAE)

あるいは、

RUN FORT FI

と入力すればよい。

(3) 使用しているサブルーチン名

MSWAKU, POUT2B

(4) 実行結果

以下の通り（次ページ）

(T S T P O U 2 S 實行結果)

T S T P O U 1 6

Test Program for POUT16 subroutine

P O U T 1 6 サブルーチンのテストプログラム

作成 竹下鉄夫 1985年 6月

形式 コンプリート・プログラム 言語:FORTRAN サイズ: 65行

(1) 概要

16値画像を人工的に発生し（横16×縦16の模様）、MSWAKUにより周囲に幅4の'0'をつけ、POUT16により各処理毎の途中結果をラインプリンタに出力する。

TSSモードでの実行では、出力結果は端末のディスプレイ画面上になる。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(TSTPOU16)に用意されているのでSUBMITすればよい。

プログラム本体はEXAMPLE.FORT(TSTPOU16)に格納されている。

EXAMPLE.FORT(TSTPOU16)をTSSモードで実行することもできる。

この場合の実行方法は、前記、TSTPOU2S の場合と全く同様である。

(3) 使用しているサブルーチン名

MSWAKU, POUT16

(4) 実行結果

以下の通り（次ページに続く）

```
*** DATA NO INSATSU ***
*** DATA WO 'Z1' NO KEISIKI DE INSATSU ***
0123456789ABCDEF
123456789ABCDEFO
23456789ABCDEF01
3456789ABCDEF012
456789ABCDEF0123
56789ABCDEF01234
6789ABCDEF012345
789ABCDEF0123456
89ABCDEF01234567
9ABCDEF012345678
ABCDEF0123456789
BCDEF0123456789A
CDEF0123456789AE
DEF0123456789ABC
EFO123456789ABCD
FO123456789ABCDF
```

TS TIPOL 16 実行結果（続き）

*** WAKU WO TSUKETE INSATSU SURU ***

```
1 123456789ABCDEF1  
1 123456789ABCDEF1  
1 123456789ABCDEF11  
1 13456789ABCDEF121  
1 1456789ABCDEF1231  
1 156789ABCDEF12341  
1 16789ABCDEF123451  
1 1789ABCDEF1234561  
1 189ABCDEF12345671  
1 19ABCDEF123456781  
1 1ABCDEF1234567891  
1 1BCDEF123456789A1  
1 1CDEF123456789AB1  
1 1DEF123456789ABC1  
1 1EF123456789ABCD1  
1 1F123456789ABCDE1
```

*** IAI(J1,J1) NO MAWARI NI 4 KO NO 'O' WO TUKETA
*** ATARASHII HAIRETSU WA IB(I1+2*M,J1+2*M) ***
*** DATA WO 'Z1' NO KEISIKI DE INSATSU ***

*** WAKU WO TSUKETE INSATSU SURU ***

123456789ABCDEF
123456789ABCDEF
23456789ABCDEF 1
3456789ABCDEF 12
456789ABCDEF 123
56789ABCDEF 1234
6789ABCDEF 12345
789ABCDEF 123456
89ABCDEF 1234567
9ABCDEF 12345678
ABCDEF 123456789
BCDEF 123456789A
CDF 123456789AB
DEF 123456789ABC
EF 123456789ABCD
F 123456789ABCDE

T S T W G T 2

Test Program for WEIGHT2 subroutine

WEIGHT2サブルーチンのテストプログラム

作成	竹下鉄夫	1985年 6月
形式	コンプリート・プログラム	言語:FORTRAN サイズ: 123行

(1) 概要

2値画像を人工的に発生し（横12×縦12の市松模様）、MSWAKUにより周囲に幅1の'0'をつけ、POUT2Bにより各処理毎の途中結果をラインプリンタに出力する。

WTIN1により重み付けフィルタをセットし、WEIGHT2により横4×縦4の大きさに第2標本化を行う。各処理毎の途中結果をラインプリンタに出力する。

TSSモードでの実行では、出力結果は端末のディスプレイ画面上になる。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(TSTWGT2S)に用意されているのでSUBMITすればよい。

プログラム本体はEXAMPLE.FORT(TSTWGT2S)に格納されている。

EXAMPLE.FORT(TSTWGT2S)をTSSモードで実行することもできる。

この場合の実行方法は、前記、TSTPOU2Sの場合と全く同様である。

(3) 使用しているサブルーチン名

MSWAKU, POUT2B, WTIN1, WEIGHT2

(4) 実行結果

以下の通り（次ページ）

(T S T W G T 2 S 実行結果)

*** ITIMATSU MOYOU NO INSATSU ***
 *** DATA WO 'Z1' NO KEISIKI DE INSATSU ***

000111000111
 000111000111
 000111000111
 111000111000
 111000111000
 111000111000
 000111000111
 000111000111
 000111000111
 111000111000
 111000111000
 111000111000
 *** WAKU WO TSUKETE INSATSU SURU ***

***	***
***	***
***	***
***	***
***	***
***	***
***	***
***	***
***	***
***	***
***	***
***	***
***	***
***	***
***	***
***	***

*** IA(I1,J1) NO MAWARI NI 1 KO NO 'O' WO TUKETA
 *** ATARASHII HAIRETSU HA IB(I1+2*M,J1+2*M) ***
 *** DATA WO 'Z1' NO KEISIKI DE INSATSU ***

00000000000000
 00001110001110
 00001110001110
 00001110001110
 01110001110000
 01110001110000
 01110001110000
 00001110001110
 00001110001110
 00001110001110
 00001110001110
 01110001110000
 01110001110000
 01110001110000
 00000000000000
 *** WAKU WO TSUKETE INSATSU SURU ***

***	***
***	***
***	***
***	***
***	***
***	***
***	***
***	***
***	***
***	***
***	***
***	***
***	***
***	***
***	***
***	***

***** JIKKOU NO KAKUNIN *****
 *** WEIGHTING FILTER

0.000	0.024	0.030	0.024	0.000
0.024	0.060	0.080	0.060	0.024
0.030	0.080	0.110	0.080	0.030
0.024	0.060	0.080	0.060	0.024
0.000	0.024	0.030	0.024	0.000
*** DAI 2 HYOUHONKA GO NO DATA				
0.156	0.670	0.234	0.670	
0.670	0.312	0.670	0.234	
0.234	0.670	0.312	0.670	
0.670	0.234	0.670	0.156	

!!! SEIJYOU SYUURYOU !!!

R D S A M P B I

Read Sample Data & Feature Extraction for Binary Image

2 値文字画像の特徴抽出

作成	竹下鉄夫	1985年 4月
形式	コンプリート・プログラム	言語:FORTRAN サイズ: 128行

(1) 概要

2 値文字画像（電総研手書き文字データベース、ETL-8B：原データのサイズは横64×縦C3）から特徴ベクトル（濃度変数：サイズは横14×縦15）を得るまでの各処理毎の途中結果をラインプリンタに出力する。

TSSモードでの実行では、出力結果は端末のディスプレイ画面上になる。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(RDSAMPB1)に用意されているのでSUBMITすればよい。

プログラム本体はEXAMPLE.FORT(RDSAMPB1)に格納されている。

特定の字種（変数：MOJICA）の指定した順序の文字（変数：MOJINN）の特徴ベクトルの得られる様子を出力することができる。

EXAMPLE.FORT(RDSAMPB1)をTSSモードで実行することもできる。この場合には、使用データセットとして、ETL-8B を選択するものとすると、以下のように実行に先立ち、ALLOC コマンドを実行する必要がある。

```
ALLOC FI(ETLDA) DA(ETL8.BVOL01) SHR  
FORT EXAMPLE.FORT(RDSAMPB1) GO(NOAE)
```

といった具合に実行する。

(3) 使用しているサブルーチン名

M SWAKU, P OUT 2 B, FILT M4, SMOOTH
NORM B1, WGT IN1, WEIGT 1

(4) 実行結果

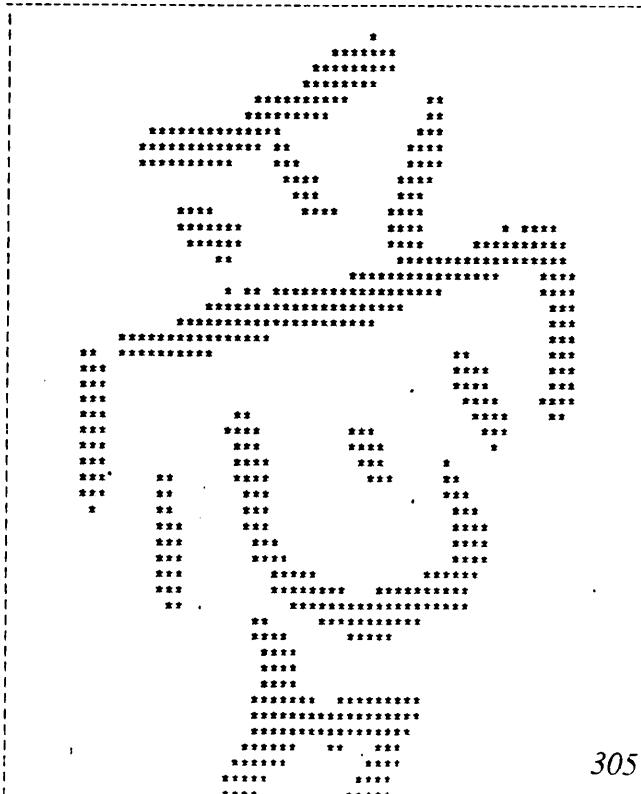
次ページ以下、3ページに亘って示す。

*** MOJI CATEGORY = 76
*** MOJI NUMBER = 10
*** GEN DATA

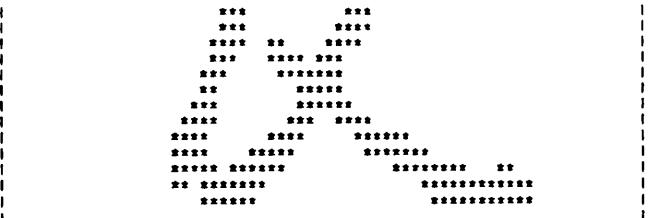
(R D S A M P B I 実行結果 その 1)



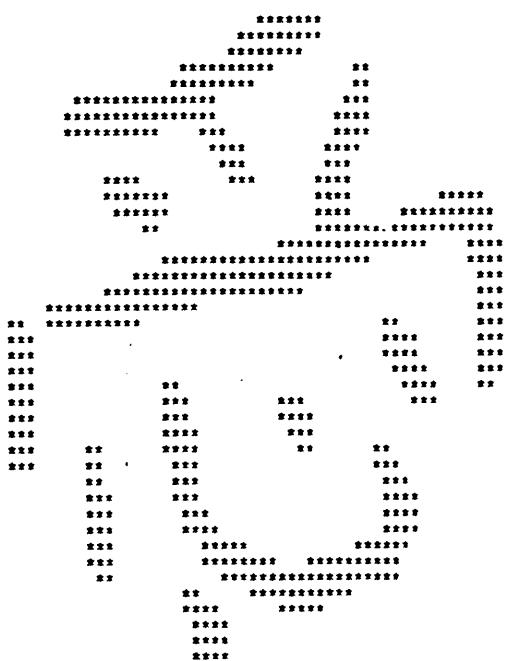
*** MSWAKU NO KAKUNIN



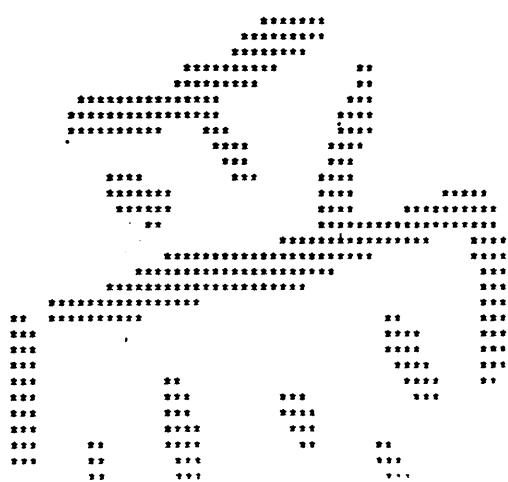
(R D S A M P B I 実行結果 その 2)



*** FILTH4 NO KAKUNIN



*** SMOOTH NO KAKUNIN



(R D S A M P B I 実行結果 その3)

*** SEIKI KA NO KAKUNIN

*** WEIGHTING FILTER

	0.000	0.024	0.030	0.024	0.000		0.000	0.024	0.000	0.000	0.000	0.000		0.000	0.000	0.000	0.000	
0.000	0.024	0.060	0.080	0.060	0.024	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.024	0.024	0.060	0.080	0.060	0.024	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.030	0.080	0.110	0.080	0.060	0.030	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.024	0.060	0.080	0.060	0.060	0.024	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.024	0.030	0.024	0.024	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.024	0.302	0.326	0.406	0.832	0.650	0.272	0.078	0.418	0.078	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.024	0.326	0.602	0.240	0.272	0.414	0.024	0.242	0.716	0.024	0.000	0.000	0.000	0.000	0.078	0.000	0.000	0.000
0.000	0.000	0.024	0.356	0.322	0.078	0.132	0.162	0.544	0.796	0.578	0.518	0.614	0.352	0.352	0.000	0.000	0.000	0.000
0.194	0.188	0.278	0.608	0.602	0.578	0.578	0.602	0.412	0.102	0.212	0.078	0.102	0.578	0.578	0.000	0.000	0.000	0.000
0.578	0.132	0.078	0.078	0.194	0.048	0.000	0.024	0.024	0.000	0.498	0.496	0.102	0.444	0.444	0.000	0.000	0.000	0.000
0.524	0.078	0.194	0.024	0.626	0.302	0.000	0.218	0.444	0.024	0.242	0.278	0.024	0.024	0.024	0.024	0.024	0.024	0.024
0.054	0.000	0.632	0.242	0.272	0.412	0.054	0.000	0.054	0.024	0.656	0.218	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.438	0.218	0.024	0.576	0.578	0.492	0.522	0.578	0.498	0.048	0.048	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.024	0.686	0.296	0.216	0.320	0.108	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.192	0.850	0.626	0.682	0.852	0.298	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.626	0.350	0.024	0.490	0.546	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.192	0.552	0.384	0.596	0.652	0.048	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.054	0.736	0.264	0.468	0.412	0.352	0.554	0.298	0.078	0.078	0.054	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.024	0.412	0.578	0.272	0.024	0.000	0.078	0.248	0.578	0.578	0.384	0.000	0.000	0.000	0.000	0.000	0.000

R D S A M P H X

Read Sample Data & Feature Extraction for 16-Level Image

1 6 値文字画像の特徴抽出

作成	竹下鉄夫	1 9 8 5 年 5 月
形式	コンプリート・プログラム	言語:FORTRAN サイズ: 1 5 2 行

(1) 概要

1 6 値文字画像（電総研手書き教育漢字、ETL-8H：原データのサイズは横 1 2 8 × 縦 1 2 7）から特徴ベクトル（濃度変数：サイズは横 1 4 × 縦 1 5）を得るまでの各処理毎の途中結果をラインプリンタに出力する。

T S S モードでの実行では、出力結果は端末のディスプレイ画面上になる。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(RDSAMPHX)に用意されているのでSUBMITすればよい。

プログラム本体はEXAMPLE.FORT(RDSAMPHX)に格納されている。

特定の字種（変数：M O J I C A）の指定した順序の文字（変数：M O J I N N）の特徴ベクトルの得られる様子を出力することができる。

EXAMPLE.FORT(RDSAMPHX)をT S S モードで実行することもできる。この場合には、使用データセットとして、ETL-8 を選択するものとすると、以下のように実行に先立ち、ALLOC コマンドを実行する必要がある。

```
ALLOC FI(ETLDA) DA(ETL8.VOL01) SHR
```

```
FORT EXAMPLE.FORT(RDSAMPHX) GO(NOAE)
```

といった具合に実行する。

(3) 使用しているサブルーチン名

```
MSWAKU, POUT16, FILTM8, SMOOTH, NORMM1  
WTIN1, WEIGHT1, HIST1, THDS2 (SPIDERより)
```

(4) 実行結果

前記 2 値化データの場合、即ち、R D S A M P B I を実行した場合と全く同様に 16 値で一たについて、以下の順で出力する。

- (1) 原データの 16 値表示 (128×127)
- (2) 上記に枠をつけて表示)
- (3) 8 近傍による平均値フィルタ (F I L T M 8) をかけた画像
- (4) 非線形平滑化 (S M O O T H) を行った画像
- (5) 位置と大きさの正規化 (この場合 N O R M M 1) を行った後の画像
- (6) 重み関数を求めて表示)
- (7) 第 2 標本化を行った結果

結果の図形表示は、ページが多くなるので省略。

R D S A M P K T

Read Sample Data & Feature Extraction for KATAKANA Character

2値文字画像の特徴抽出

作成	竹下鉄夫	1985年 4月
形式	コンプリート・プログラム	言語:FORTRAN サイズ: 128行

(1) 概要

2値文字画像（片仮名ETL-1B：原データのサイズは横60×縦60）から特徴ベクトル（濃度変数：サイズは横7×縦7）を得るまでの各処理毎の途中結果をラインプリンタに出力する。

TSSモードでの実行では、出力結果は端末のディスプレイ画面上になる。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(RDSAMPKT)に用意されているのでSUBMITすればよい。

プログラム本体はEXAMPLE.FORT(RDSAMPKT)に格納されている。

特定の字種（変数：MOJICA）の指定した順序の文字（変数：MOJINN）の特徴ベクトルの得られる様子を出力することができる。

EXAMPLE.FORT(RDSAMPKT)をTSSモードで実行することもできる。この場合には、使用データセットとして、ETL-1Bを選択するものとすると、以下のように実行に先立ち、ALLOCコマンドを実行する必要がある。

```
ALLOC FI(ETLDA) DA(ETL1.BVOL02) SHR  
FORT EXAMPLE.FORT(RDSAMPKT) GO(NOAE) といった具合に。
```

(3) 使用しているサブルーチン名

MSWAKU, POUT2B, FILTM4, SMOOTH
NORMB1, WGTIN1, WEIGT1

(4) 実行結果

前記 2値化データの場合、即ち、RDSAMPBIを実行した場合と全く同様であるので省略。

P R E M A T C H

Pre-process for 16-Level Image

多値文字画像の前処理

作成	掛布英辰	1985年 10月
形式	コンプリート・プログラム	言語:FORTRAN サイズ: 111行

(1) 概要

多値文字画像(ETL-8H)に対して文字認識に必要な前処理、即ち、枠付け、平均値フィルタ、2値化、大きさの正規化第二標本化等を行う。

T S S モードでの実行では、出力結果は端末のディスプレイ画面上になる。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(PREMATCH)に用意されているのでSUBMITすればよい。

プログラム本体はEXAMPLE.FORT(PREMATCH)に格納されている。

EXAMPLE.FORT(PREMATCH)をT S S モードで実行することもできる。

この場合の実行方法は、R D S A M P B I と全く同様である。

(3) 使用しているサブルーチン名

P O U T 1 6 , M S W A K U , F I L T M 4 , H I S T 1 (S P I D E R より)
T H D S 2 (S P I D E R より) , B I N A R Y , P O U T 2 B
N O R M B 1 , W G T I N 1 , W E I G T 1

(4) 実行結果

以下の順序で文字パターンがラインプリンタあるいはディスプレイ上に表示される。

- ①文字の原パターン (16進数表示)
- ②周囲に0の枠を付けた、原パターン (16進数表示)
- ③4近傍による平均値フィルタをかけたパターン
- ④しきい値を決めて2値化した2値化パターン
- ⑤大きさの正規化 (この場合N O R M B 1) を実行したパターン
- ⑥第2標本化を行った結果

FEATUREB

Feature Extraction for Binary Image

2 値文字画像の特徴抽出

作成	鶴岡信治	1985年 10月
形式	コンプリート・プログラム	言語: FORTRAN サイズ: 140行

(1) 概要

2 値文字画像(ETL-8B)から特徴ベクトル(加重方向指数ヒストグラム)を得るまでの各処理毎の途中結果をラインプリンタに出力する。

TSS モードでの実行では、出力結果は端末のディスプレイ画面上になる。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(FEATUREB)に用意されているのでSUBMITすればよい。

プログラム本体はEXAMPLE.FORT(FEATUREB)に格納されている。

EXAMPLE.FORT(FEATUREB)をTSSモードで実行することもできる。

(3) 使用しているサブルーチン名

POUT2B, NORMB3, SMOOTH, MSWAKU
BDFL2 (SPIDERより), HISTS, WGTIN4, WEIGHT1

(4) 実行結果

以下の順序で文字パターンがラインプリンタあるいはディスプレイ上に表示される。

- ①文字の原パターン(64×63の文字部分を*で表示)
- ②位置と大きさの正規化(この場合NORMB3)を実行したパターン
- ③平滑化を行った2値パターン
- ④周囲に枠を付けた2値化パターン
- ⑤計算して求めた方向指数及び加重方向指数

(5) 参考文献

- 1)原田, 鶴岡, 木村, 三宅：“加重方向指数ヒストグラムと疑似ベイズ識別法を用いた手書き漢字・ひらがな認識”, 信学技報, PRL83-68(1984)

FEATUREH

Feature Extraction for 16-level Image

多値文字画像の特徴抽出

作成	鶴岡信治	1985年 10月
形式	コンプリート・プログラム	言語: FORTRAN サイズ: 150行

(1) 概要

多値文字画像(ETL-8H)から特徴ベクトル(加重方向指数ヒストグラム)を得るまでの各処理毎の途中結果をラインプリンタに出力する。

TSSモードでの実行では、出力結果は、端末のディスプレイ画面上になる。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(FEATUREH)に用意されているのでSUBMITすればよい。

プログラム本体はEXAMPLE.FORT(FEATUREH)に格納されている。

EXAMPLE.FORT(FEATUREH)をTSSモードで実行することもできる。

この場合の実行方法は、RD SAMP B I 等と全く同様である。

(3) 使用しているサブルーチン名

P O U T 1 6 , F I L T M 8 , B I N A R Y , S M O O T H
N O R M M 3 , M S W A K U , B D F L 2 (S P I D E Rより)
H I S T S S , W G T I N 4 , W E I G T 1

(4) 実行結果

FEATUREBと全く同様に、以下の順序で文字パターンがラインプリンタあるいはディスプレイ上に表示される。

- ①文字の原パターン (128×127の文字部分を16進数表示)
- ②8近傍による平均値フィルタ (FILTM8) をかけたパターン (16進数表示)
- ③しきい値を決めて2値化した2値化パターン (文字部分を*表示)
- ④平滑化を行った2値化パターン
- ⑤位置と大きさの正規化 (この場合NORMM3) を実行したパターン

(大きさは 128×127 から 70×70 となる)

⑥再度平滑化を行った2値化パターン

⑦計算して求めた方向指数及び加重方向指数

結果の図形表示は、ページが多くなるので省略。

(4) 参考文献

- 1) 原田, 鶴岡, 木村, 三宅：“加重方向指数ヒストグラムと疑似ベイズ識別法を用いた手書き漢字・ひらがな認識”，信学技報，PRL83-68(1984)

D S E G M T C K

Directional Segment of Binary Image Check Program

2 値画像の 4 方向セグメント表示

作成	竹下鉄夫	1 9 8 6 年	9 月
形式 コンプリート・プログラム 言語:FORTRAN サイズ: 44 行			

(1) 概要

2 値文字画像（電総研手書き文字データベース、ETL-8B より平仮名：原データのサイズは横 64 * 縦 63）から 4 方向セグメントを求め、処理結果をライププリンタに出力する。

TSS モードでの実行では、出力結果は端末のディスプレイ画面上になる。

(2) 使用法

ジョブ制御文のデータセットが SMART.CNTL (DSEGMTCK) に用意されているので SUBMIT すればよい。

プログラム本体は EXAMPLE.FORT (DSEGMTCK) に格納されている。

EXAMPLE.FORT (DSEGMTCK) を TSS モードで実行することもできる。この場合には、使用データセットとして、ETL-8B を選択するものとすると、以下のように実行に先立ち、ALLOC コマンドを実行する必要がある。

```
ALLOC FI(ETLDA) DA(ETL8.BVOL01) SHR  
FORT EXAMPLE.FORT (DSEGMTCK) GO(NOAE)
```

といった具合に実行する。

(3) 使用しているサブルーチン名

DSEGMENT, SMOOTH, MSWAKU, NORMB1, FILTM4,
POUT2B

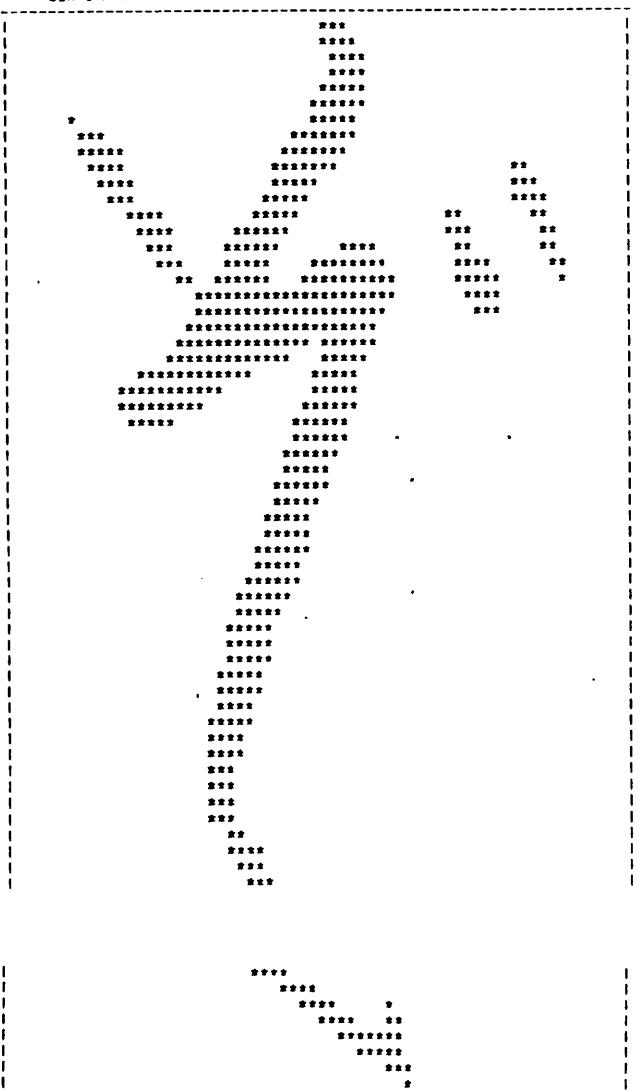
(4) 実行結果

以下の通り

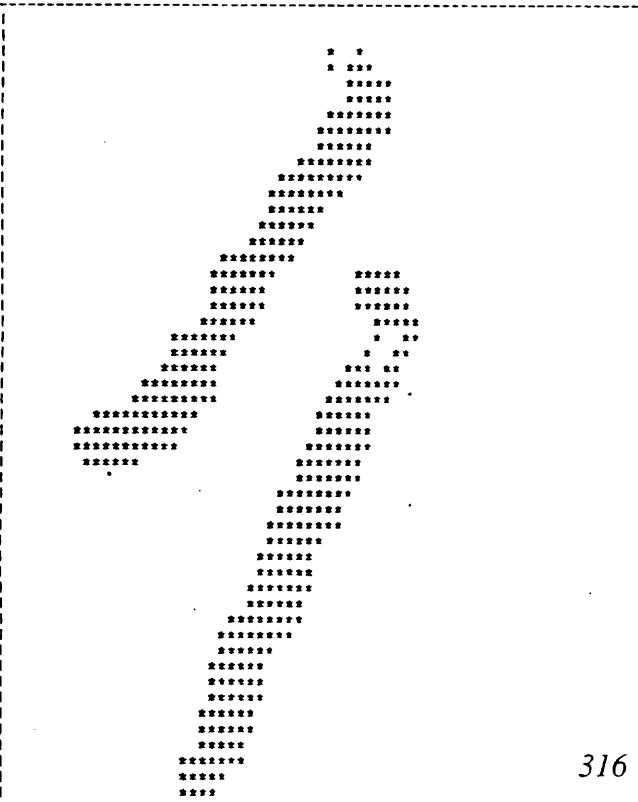
次ページ以下 3 ページに亘って示す。

*** MOJI CATEGORY = 26
*** MOJI NUMBER = 1
*** GEN DATA

(D S E G M T C K 実行結果 その 1)



*** AFTER DSEGMENT DIRECTION = 1 (RIGHT-UP)



(DSEGMENTCK 実行結果 その2)

*** AFTER DSEGMNT DIRECTION = 2 (HORIZONTAL)

二

*** AFTER DSEGMNT DIRECTION = 3 (LEFT-UP)

(D S E G M T C K 実行結果 その 3)

*** AFTER DSEGMENT DIRECTION = 4 (VERTICAL)

The pattern consists of a series of asterisks arranged in a triangular shape pointing downwards. It starts with a single asterisk at the top, followed by two on the second line, three on the third, four on the fourth, five on the fifth, and ends with three on the eighth line. There are two blank lines above this pattern.

The pattern consists of a series of asterisks arranged in a triangular shape pointing upwards. It starts with a single asterisk at the top, followed by two on the second line, three on the third, four on the fourth, five on the fifth, and ends with three on the eighth line. There are two blank lines below this pattern.

H K I Y O T S T

Test Program for HKIY0

H K I Y O のサンプルプログラム

作成	村上 幸太郎	1 9 8 7 年 1 2 月
形式	コンプリート・プログラム	言語 : FORTRAN サイズ : 8 2 行

(1) 概要

文字画像データ (IA (I 1, J 1)) の外郭方向寄与度特徴の計算するH K I Y Oの使用法を説明するテストプログラムである。

T S S モードでの実行では、出力結果は、端末のディスプレイ画面上になる。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(HKIYOTST)に用意されているのでSUBMITすればよい。

プログラム本体はEXAMPLE.FORT(HKIYOTST)に格納されている。

EXAMPLE.FORT(HKIYOTST)をT S S モードで実行することもできる。この場合には、使用データセットとして、ETL-8 を選択するものとすると、以下のように実行に先立ち、ALLOC コマンドを実行する必要がある。

```
ALLOC FI(ETLDA) DA(ETL8.VOL01) SHR  
FORT EXAMPLE.FORT(HKIYOTST) GO(NOAE)
```

といった具合に実行する。

(3) 使用しているサブルーチン名

M S W A K U, F I L T M 8, S M O O T H, H K I Y O

(4) 実行結果

以下(次ページ)に示すように、計算した外郭方向寄与度特徴が印字される。

IDIV =8 MS =3 M =4 IDIV =16
*** MOJI CATEGORY = 1
*** MOJI NUMBER = 10
*** MOJIMO =10 II=64 JI=63
SUIHEI

(H K I Y O T S T 実行結果)

		+45 DO	SUITYOKU	-45 DO
1 HOUKOU	0.980 0.000 0.000	0.143 0.000 0.000	0.095 0.000 0.000	0.078 0.000 0.000
	0.937 0.000 0.000	0.234 0.000 0.000	0.165 0.000 0.000	0.185 0.000 0.000
	0.912 0.239 0.000	0.299 0.203 0.000	0.161 0.156 0.000	0.228 0.297 0.000
	0.918 0.263 0.000	0.242 0.268 0.000	0.202 0.502 0.000	0.229 0.431 0.000
	0.283 0.000 0.000	0.278 0.000 0.000	0.805 0.000 0.000	0.203 0.000 0.000
	0.677 0.399 0.000	0.295 0.240 0.000	0.343 0.068 0.000	0.320 0.107 0.000
	0.901 0.585 0.000	0.291 0.470 0.000	0.159 0.442 0.000	0.273 0.355 0.000
	0.789 0.186 0.000	0.197 0.267 0.000	0.513 0.472 0.000	0.197 0.297 0.000
	0.405 0.608 0.459	0.349 0.295 0.489	0.314 0.319 0.518	0.590 0.644 0.373
	0.425 0.786 0.457	0.319 0.254 0.272	0.635 0.221 0.393	0.414 0.369 0.670
	0.530 0.948 0.931	0.406 0.176 0.232	0.555 0.212 0.164	0.369 0.130 0.226
	0.650 0.717 0.931	0.348 0.304 0.226	0.471 0.197 0.164	0.346 0.312 0.232
	0.550 0.243 0.725	0.179 0.246 0.280	0.185 0.408 0.320	0.723 0.655 0.433
	0.467 0.401 0.818	0.260 0.306 0.435	0.190 0.489 0.249	0.680 0.449 0.189
	0.297 0.522 0.718	0.451 0.222 0.285	0.547 0.258 0.412	0.467 0.548 0.269
	0.498 0.380 0.115	0.295 0.192 0.123	0.517 0.503 0.249	0.455 0.225 0.113
2 HOUKOU	0.000 0.000 0.000	0.000 0.000 0.000	0.000 0.000 0.000	0.000 0.000 0.000
	0.000 0.000 0.000	0.000 0.000 0.000	0.000 0.000 0.000	0.000 0.000 0.000
	0.026 0.000 0.000	0.018 0.000 0.000	0.007 0.000 0.000	0.163 0.000 0.000
	0.267 0.216 0.000	0.590 0.131 0.000	0.279 0.177 0.000	0.419 0.772 0.000
	0.151 0.147 0.023	0.906 0.178 0.014	0.332 0.120 0.019	0.127 0.477 0.077
	0.324 0.138 0.055	0.235 0.263 0.131	0.281 0.092 0.049	0.824 0.075 0.050
	0.239 0.244 0.161	0.254 0.418 0.316	0.506 0.242 0.128	0.612 0.366 0.285
	0.340 0.341 0.366	0.666 0.302 0.501	0.347 0.190 0.228	0.393 0.819 0.407
	0.490 0.301 0.309	0.529 0.310 0.396	0.372 0.114 0.485	0.415 0.827 0.585
	0.267 0.261 0.189	0.389 0.771 0.237	0.398 0.191 0.058	0.691 0.284 0.251
	0.302 0.543 0.000	0.219 0.601 0.000	0.224 0.371 0.000	0.889 0.361 0.000
	0.335 0.426 0.039	0.338 0.177 0.044	0.365 0.084 0.039	0.692 0.153 0.044
	0.439 0.019 0.000	0.272 0.014 0.000	0.183 0.019 0.000	0.700 0.078 0.000
	0.296 0.326 0.116	0.596 0.366 0.164	0.614 0.269 0.089	0.281 0.722 0.210
	0.210 0.112 0.030	0.403 0.209 0.029	0.230 0.208 0.005	0.277 0.202 0.072
	0.000 0.000 0.000	0.000 0.000 0.000	0.000 0.000 0.000	0.000 0.000 0.000
3 HOUKOU	0.758 0.000 0.000	0.262 0.000 0.000	0.495 0.000 0.000	0.191 0.700 0.000
	0.579 0.383 0.328	0.425 0.455 0.046	0.308 0.591 0.022	0.500 0.339 0.031
	0.336 0.396 0.613	0.662 0.552 0.462	0.197 0.396 0.321	0.386 0.304 0.376
	0.709 0.449 0.964	0.462 0.516 0.197	0.239 0.396 0.107	0.224 0.349 0.136
	0.470 0.515 0.848	0.463 0.595 0.467	0.419 0.329 0.146	0.315 0.282 0.166
	0.464 0.289 0.128	0.460 0.273 0.060	0.485 0.637 0.026	0.244 0.158 0.084
	0.244 0.155 0.255	0.162 0.205 0.161	0.937 0.107 0.059	0.169 0.177 0.126
	0.247 0.660 0.778	0.241 0.291 0.289	0.760 0.352 0.156	0.563 0.680 0.500
	0.330 0.771 0.000	0.513 0.471 0.000	0.564 0.264 0.000	0.197 0.224 0.000
	0.369 0.696 0.000	0.372 0.451 0.000	0.492 0.346 0.000	0.483 0.397 0.000
	0.668 0.845 0.284	0.503 0.271 0.122	0.245 0.220 0.032	0.268 0.201 0.083
	0.534 0.967 0.000	0.678 0.090 0.000	0.340 0.088 0.000	0.319 0.219 0.000
	0.515 0.716 0.000	0.700 0.367 0.000	0.236 0.269 0.000	0.266 0.427 0.000
	0.649 0.551 0.000	0.549 0.320 0.000	0.244 0.388 0.000	0.244 0.588 0.000
	0.543 0.494 0.000	0.472 0.440 0.000	0.554 0.510 0.000	0.319 0.421 0.000
	0.259 0.000 0.000	0.266 0.000 0.000	0.834 0.000 0.000	0.246 0.000 0.000
4 HOUKOU	0.180 0.052 0.000	0.258 0.037 0.000	0.137 0.051 0.000	0.316 0.139 0.000
	0.653 0.329 0.000	0.232 0.587 0.000	0.384 0.199 0.000	0.396 0.198 0.000
5 HOUKOU	0.496 0.649 0.000	0.273 0.462 0.000	0.192 0.293 0.000	0.610 0.323 0.000
	0.662 0.756 0.000	0.355 0.287 0.000	0.265 0.208 0.000	0.336 0.286 0.000
	0.321 0.466 0.496	0.546 0.613 0.239	0.578 0.166 0.196	0.321 0.294 0.287
	0.631 0.466 0.644	0.348 0.639 0.339	0.342 0.311 0.133	0.496 0.317 0.312
	0.677 0.572 0.104	0.242 0.344 0.646	0.273 0.358 0.108	0.536 0.394 0.193
	0.683 0.666 0.400	0.295 0.458 0.354	0.219 0.224 0.486	0.368 0.218 0.526
	0.363 0.582 0.171	0.738 0.385 0.947	0.272 0.391 0.179	0.371 0.291 0.170
	0.573 0.426 0.498	0.704 0.180 0.161	0.206 0.182 0.363	0.211 0.752 0.729
	0.375 0.364 0.420	0.738 0.300 0.486	0.450 0.315 0.219	0.221 0.742 0.526
	0.285 0.234 0.103	0.455 0.255 0.385	0.279 0.128 0.078	0.377 0.358 0.174
	0.000 0.000 0.000	0.000 0.000 0.000	0.000 0.000 0.000	0.000 0.000 0.000
	0.000 0.000 0.000	0.000 0.000 0.000	0.000 0.000 0.000	0.000 0.000 0.000
	0.000 0.000 0.000	0.000 0.000 0.000	0.000 0.000 0.000	0.000 0.000 0.000
6 HOUKOU	0.269 0.465 0.099	0.318 0.447 0.090	0.541 0.253 0.297	0.589 0.250 0.059
	0.493 0.360 0.376	0.341 0.689 0.270	0.631 0.412 0.528	0.378 0.335 0.453
	0.916 0.570 0.486	0.190 0.603 0.361	0.211 0.330 0.438	0.220 0.318 0.423
	0.655 0.826 0.529	0.230 0.376 0.368	0.206 0.367 0.364	0.618 0.195 0.548
	0.785 0.638 0.395	0.333 0.480 0.304	0.235 0.340 0.388	0.456 0.333 0.552
	0.692 0.392 0.391	0.567 0.188 0.328	0.208 0.416 0.458	0.291 0.656 0.521
	0.721 0.592 0.962	0.491 0.378 0.180	0.236 0.268 0.152	0.252 0.396 0.113
	0.791 0.348 0.629	0.228 0.339 0.423	0.358 0.262 0.341	0.281 0.659 0.258
	0.273 0.167 0.000	0.331 0.147 0.000	0.744 0.393 0.000	0.210 0.352 0.000
	0.257 0.935 0.000	0.241 0.208 0.000	0.430 0.108 0.000	0.691 0.219 0.000
	0.864 0.389 0.000	0.159 0.157 0.000	0.152 0.105 0.000	0.315 0.149 0.000
	0.314 0.000 0.000	0.367 0.000 0.000	0.708 0.000 0.000	0.300 0.000 0.000
	0.345 0.694 0.000	0.344 0.160 0.000	0.763 0.161 0.000	0.358 0.256 0.000
	0.639 0.478 0.000	0.173 0.078 0.000	0.272 0.055 0.000	0.391 0.110 0.000
	0.973 0.000 0.000	0.121 0.000 0.000	0.112 0.000 0.000	0.159 0.000 0.000
	0.983 0.000 0.000	0.062 0.000 0.000	0.078 0.000 0.000	0.145 0.000 0.000
7 HOUKOU	0.000 0.000 0.000	0.000 0.000 0.000	0.000 0.000 0.000	0.000 0.000 0.000
	0.158 0.129 0.038	0.154 0.217 0.069	0.242 0.101 0.008	0.518 0.258 0.027
	0.456 0.379 0.130	0.221 0.675 0.228	0.375 0.374 0.127	0.730 0.391 0.124
	0.522 0.026 0.000	0.224 0.079 0.000	0.200 0.008 0.000	0.742 0.011 0.000
	0.391 0.345 0.031	0.481 0.206 0.011	0.303 0.167 0.008	0.581 0.180 0.076
	0.452 0.580 0.000	0.768 0.379 0.000	0.161 0.258 0.000	0.187 0.588 0.000
	0.360 0.400 0.199	0.874 0.441 0.111	0.194 0.294 0.157	0.155 0.605 0.255
	0.270 0.587 0.335	0.209 0.428 0.659	0.769 0.212 0.277	0.383 0.689 0.437
	0.326 0.350 0.438	0.724 0.424 0.374	0.350 0.144 0.248	0.348 0.748 0.463
	0.274 0.182 0.205	0.386 0.280 0.242	0.658 0.146 0.083	0.380 0.552 0.292
	0.332 0.115 0.027	0.533 0.169 0.042	0.514 0.039 0.087	0.382 0.202 0.094
	0.144 0.243 0.012	0.111 0.483 0.081	0.186 0.132 0.012	0.965 0.108 0.008
	0.077 0.127 0.000	0.089 0.743 0.000	0.141 0.113 0.000	0.982 0.161 0.000
	0.020 0.000 0.000	0.011 0.000 0.000	0.006 0.000 0.000	0.165 0.000 0.000
	0.000 0.000 0.000	0.000 0.000 0.000	0.000 0.000 0.000	0.000 0.000 0.000
	0.000 0.000 0.000	0.000 0.000 0.000	0.000 0.000 0.000	0.000 0.000 0.000

E L B L N K

Test Program for ELBLNK

ELBLNKのサンプルプログラム

作成 竹下鉄夫 1988年 9月

形式 コンプリート・プログラム 言語:FORTRAN サイズ: 61行

(1) 概要

市松模様を生成し、その後周囲にゼロを付加するサブルーチンMSWAKUを使い不要なゼロをつける。しかる後、ELBLNKサブルーチンを呼び不要なゼロが除去されていることを印刷サブルーチンPOUT2Bを使って確認している。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(ELBLNK)に用意されているのでSUBMITすればよい。プログラム本体はEXAMPLE.FORT(ELBLNK)に格納されている。

(3) 使用しているサブルーチン名

ELBLNK, MSWAKU, POUT2B

(4) 實行結果

以下に示す。 (E L B L N K 塞行結果)

NORMB 7

Test Program for NORMB7

NORMB 7 のサンプルプログラム

作成	竹下鉄夫	1988年	9月
形式	コンプリート・プログラム	言語:FORTRAN	サイズ: 61行

(1) 概要

市松模様を生成し、その後周囲にゼロを付加するサブルーチンMSWAKUを使い不要なゼロをつける。しかる後、NORMB 7 サブルーチンを呼び縦横比を保持する正規化を実行している。実行されている様子を印刷サブルーチンPOUT2Bを使って確認している。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(NORMB7)に用意されているのでSUBMITすればよい。プログラム本体はEXAMPLE.FORT(NORMB7)に格納されている。

(3) 使用しているサブルーチン名

NORMB 7, MSWAKU, POUT2B

(4) 実行結果

次ページに示す。

(N O R M B 7 実行結果)

*** IA1(I1,J1) NO MAWARI NI M KO NO 'O' WO TUKETA
*** WAKU WO TSUKETE IA2 WO INSATSU SURU ***

**** *
*** * *
** * * *
* * * * *
* * * * *

*** IA2 WO TATE YOKO HI WO TAMOTTE SEIKIKA SURU
*** WAKU WO TSUKETE IA WO INSATSU SURU ***

*****|
**** |
*** |
** |
* |
*****|
**** |
*** |
** |
* |
*****|
**** |
*** |
** |
* |
*****|
**** |
*** |
** |
* |

CREATEBI

Feature Extraction & File Creation for Binary Image

2 値文字画像の特徴（濃度変数）抽出とファイルの作成

作成	竹下鉄夫	1985年 11月
形式	コンプリート・プログラム	言語:FORTRAN サイズ: 101行

(1) 概要

2 値文字画像（電総研手書き文字データベース、ETL-8B：原データのサイズは横 64 × 縦 63）から特徴ベクトル（濃度変数：サイズは横 14 × 縦 15）を抽出し、ファイル（順編成データセット）をディスクに作成する。（ここでは、10字種・50文字／字種、合計 500 文字分）

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(CREATEBI)に用意されているのでSUBMITすればよい。

プログラム本体はEXAMPLE.FORT(CREATEBI)に格納されている。

作成ファイルはGO.FT03F001で定義しているので、データセット名(DSN)、及びファイルの大きさを設定する、スペース パラメータ (SPACE: 作成文字数によって異なる) は利用者が適宜、指定する。

EXAMPLE.FORT(CREATEBI)をTSSモードで実行することもできる。この処理はプリンタに多くの出力をしないので、TSSモードでの実行が簡単である。

この場合には、使用データセットとして、ETL-8B を選択するものとすると、以下のように実行に先立ち、ALLOC コマンドを実行する必要がある。

また、出力データセットをDD名をFT03F001として、ALLOCとATTR コマンドを用いるか、ALLOC2 コマンドを用いて確保しておく必要がある。

例えば

```
ALLOC FI(ETLDA) DA(ETL8.BVOL01) SHR
ATTR SM RECFM(VBS) LRECL(852) BLKSIZE(26000)
ALLOC US(SM) FI(FT03F001) DA(CREATEBI.DATA) SP(15,1) T REL NEW
FORT EXAMPLE.FORT(CREATEBI) GO(NOAE)
```

といった具合に実行する。

(3) 使用しているサブルーチン名

W G T I N 1 , M S W A K U , F I L T M 4
S M O O T H , N O R M B 1 , W E I G T 1

(4) 実行結果

出力は作成データセット上に行われるので、プリンタあるいはディスプレイ上には、以下のような簡単なメッセージだけが出力される。

MOJI CATEGORY = 1
MOJI CATEGORY = 2
MOJI CATEGORY = 3
MOJI CATEGORY = 4
MOJI CATEGORY = 5
MOJI CATEGORY = 6
MOJI CATEGORY = 7
MOJI CATEGORY = 8
MOJI CATEGORY = 9
MOJI CATEGORY = 10

CREATEHX

Feature Extraction & File Creation for 16-Level Image

1 6 値文字画像の特徴（濃度変数）抽出とファイルの作成

作成	竹下鉄夫	1985年 11月
形式	コンプリート・プログラム	言語:FORTRAN サイズ: 119行

(1) 概要

16 値文字画像（手書き教育漢字ETL-8H：原データのサイズは横 128 × 縦 127）から特徴ベクトル（濃度変数：サイズは横 14 × 縦 15）を抽出し、ファイル（順編成データセット）をディスクに作成する。（ここでは、10 字種・50 文字／字種、合計 500 文字分）

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(CREATEHX)に用意されているのでSUBMITすればよい。

プログラム本体はEXAMPLE.FORT(CREATEHX)に格納されている。

作成ファイルはG0.FT03F001で定義しているので、データセット名(DSN)、及びファイルの大きさを設定する、スペース パラメータ (SPACE: 作成文字数によって異なる) は利用者が適宜、指定する。

EXAMPLE.FORT(CREATEHX)をTSSモードで実行することもできる。この処理はプリンタに多くの出力をしないので、TSSモードでの実行が簡単である。

この場合には、使用データセットとして、ETL-8H を選択するものとすると、以下のように実行に先立ち、ALLOC コマンドを実行する必要がある。

また、出力データセットをDD名をFT03F001として、ALLOCとATTR コマンドを用いるか、ALLOC2 コマンドを用いて確保しておく必要がある。

例えば

```
ALLOC FI(ETLDA) DA(ETL8.VOL01) SHR
ATTR SM RECFM(VBS) LRECL(852) BLKSIZE(26000)
ALLC2
"(DSN=CREATEBI.DATA), FT03F001, PS, FB, 266, 2660, PUB, TRK, 10, 10"
FORT EXAMPLE.FORT(CREATEHX) GO(NOAE)
```

といった具合に実行する。

(3) 使用しているサブルーチン名

WGTIN1, MSWAKU, FILTM8, SMOOTH, NORMM1
WEIGT1, HIST1, THDS2 (SPIDERより)

(4) 実行結果

出力は作成データセット上に行われるので、プリンタあるいはディスプレイ上には、以下のような簡単なメッセージだけが出力される。

```
*** MOJI CATEGORY = 1
*** MOJI CATEGORY = 2
*** MOJI CATEGORY = 3
*** MOJI CATEGORY = 4
*** MOJI CATEGORY = 5
*** MOJI CATEGORY = 6
*** MOJI CATEGORY = 7
*** MOJI CATEGORY = 8
*** MOJI CATEGORY = 9
*** MOJI CATEGORY = 10
```

CREATEKT

Feature Extraction & File Creation for KATAKANA Character

2値文字画像の特徴（濃度変数）抽出とファイルの作成

作成	竹下鉄夫	1985年 11月
形式	コンプリート・プログラム	言語:FORTRAN サイズ: 159行

(1) 概要

2値文字画像（片仮名ETL-1B：原データのサイズは横60×縦60）から特徴ベクトル（濃度変数：サイズは横7×縦7）を抽出し、ファイル（順編成データセット）をディスクに作成する。（ここでは、10字種・50文字／字種、合計500文字分）

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(CREATEKT)に用意されているのでSUBMITすればよい。

プログラム本体はEXAMPLE.FORT(CREATEKT)に格納されている。

作成ファイルはGO.FT03F001で定義しているので、データセット名(DSN)、及びファイルの大きさを設定する、スペース パラメータ (SPACE: 作成文字数によって異なる) は利用者が適宜、指定する。

EXAMPLE.FORT(CREATEKT)をTSSモードで実行することもできる。この処理はプリンタに多くの出力をしないので、TSSモードでの実行が簡単である。

この場合には、使用データセットとして、ETL-1B を選択するものとすると、以下のように実行に先立ち、ALLOC コマンドを実行する必要がある。

また、出力データセットをDD名をFT03F001として、ALLOCとATTR コマンドを用いるか、ALLOC2 コマンドを用いて確保しておく必要がある。

例えば

```
ALLOC FI(ETLDA) DA(ETL1.BVOL02) SHR
ATTR SM RECFM(VBS) LRECL(208) BLKSIZE(26050)
ALLC US(SM) F(FT03F001) DA(CREATEKT.DATA) SP(7,1) T REL NEW
FORT EXAMPLE.FORT(CREATEKT) GO(NOAE)
```

といった具合に。

(3) 使用しているサブルーチン名

W G T I N 1 , M S W A K U , F I L T M 4 , S M O O T H
N O R M B 1 , W E I G T 2

(4) 実行結果

出力は作成データセット上に行われるので、プリンタあるいはディスプレイ上には、以下のような簡単なメッセージだけである。

MOJI CATEGORY = 1
MOJI CATEGORY = 2
MOJI CATEGORY = 3
MOJI CATEGORY = 4
MOJI CATEGORY = 5
MOJI CATEGORY = 6
MOJI CATEGORY = 7
MOJI CATEGORY = 8
MOJI CATEGORY = 9
MOJI CATEGORY = 10

C R E F I L E B

Feature Vector File Creation for Binary Image

2値文字画像の特徴ベクトル（加重方向指数ヒストグラム）ファイルの作成

作成	鶴岡信治	1985年 10月
形式	コンプリート・プログラム	言語: FORTRAN サイズ: 160行

(1) 概要

2値文字画像(ETL-8B: 原データのサイズは横64×縦63)から特徴ベクトル（加重方向指数ヒストグラム）を抽出し、ファイルをディスクに作成する。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(CREFILEB)に用意されているのでSUBMITすればよい。

プログラム本体はEXAMPLE.FORT(CREFILEB)に格納されている。

作成ファイルはGO.FT11F001で定義しているので、データセット名(DSN)、及びファイルの大きさを設定する、スペース パラメータ (SPACE: 作成文字数によって異なる)は利用者が適宜、指定する。

EXAMPLE.FORT(CREFILEB)をTSSモードで実行することもできる。この処理はプリンタに多くの出力をしないので、TSSモードでの実行が簡単である。

この場合には、使用データセットとして、ETL-8Bを選択するものとすると、以下のように実行に先立ち、ALLOC コマンドを実行する必要がある。

また、出力データセットをDD名をFT11F001として、ALLOCとATTR コマンドを用いるか、ALLOC2 コマンドを用いて確保しておく必要がある。

例えば

```
ALLOC FI(ETLDA) DA(ETL8.BVOL01) SHR
ATTR SM RECFM(FB) LRECL(264) BLKSIZE(2640)
ALLC US(SM) F(FT11F001) DA(E8BHIST1) SP(3780,1) B REL NEW
FORT EXAMPLE.FORT(CREFILEB) GO(NOAE)
```

といった具合に。

(3) 使用しているサブルーチン名

F I L T M 8 , N O R M B 3 , S M O O T H , M S W A K U
B D F L 2 (S P I D E R より) , H I S T S S , W G T I N 4 , W E I G T 1

(4) 実行結果

出力は作成データセット上に行われる所以、プリントあるいはディスプレイ上には、以下のような簡単なメッセージだけである。

MOJI CATEGORY = 1

MOJI CATEGORY = 2

MOJI CATEGORY = 3

MOJI CATEGORY = 4

MOJI CATEGORY = 5

MOJI CATEGORY = 6

MOJI CATEGORY = 7

MOJI CATEGORY = 8

MOJI CATEGORY = 9

MOJI CATEGORY = 10

(5

) 参考文献

- 1) 原田, 鶴岡, 木村, 三宅：“加重方向指数ヒストグラムと疑似ベイズ識別法を用いた手書き漢字・ひらがな認識”，信学技報，PRL83-68(1984)

C R E F I L E H

Feature Vector File Creation for 16-level Image

多値文字画像の特徴ベクトル（加重方向指数ヒストグラム）ファイルの作成

作成	鶴岡信治	1985年 10月
形式	コンプリート・プログラム	言語: FORTRAN サイズ: 170行

(1) 概要

多値文字画像(ETL-8H)から特徴ベクトル（加重方向指数ヒストグラム）を抽出し、ファイルをディスクに作成する。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(CREFILEH)に用意されているのでSUBMITすればよい。

プログラム本体はEXAMPLE.FORT(CREFILEH)に格納されている。

作成ファイルはGO.FT11F001で定義しているので、データセット名(DSN)、及びファイルの大きさを設定する、スペース パラメータ (SPACE: 作成文字数によって異なる) は利用者が適宜、指定する。

EXAMPLE.FORT(CREFILEH)をTSSモードで実行することもできる。この処理はプリンタに多くの出力をしないので、TSSモードでの実行が簡単である。

この場合には、使用データセットとして、ETL-8H を選択するものとすると、以下のように実行に先立ち、ALLOC コマンドを実行する必要がある。

また、出力データセットをDD名をFT11F001として、ALLOCとATTR コマンドを用いるか、ALLOC2 コマンドを用いて確保しておく必要がある。

例えば

```
ALLOC FI(ETLDA) DA(ETL8.VOL01) SHR
ATTR SM RECFM(FB) LRECL(264) BLKSIZE(2640)
ALLC US(SM) F(FT11F001) DA(CREFILEH.DATA) SP(3780,1) B REL NEW
FORT EXAMPLE.FORT(CREFILEH) GO(NOAE)
```

といった具合に実行する。

(4) 実行結果

求めた加重方向指数が、ディスク上のファイルに作成されると共に、以下のようにプリンタ、或いはディスプレイ上に表示される。

(5) 参考文献

1)原田, 鶴岡, 木村, 三宅：“加重方向指数ヒストグラムと疑似ベイズ識別法を用いた手書き漢字・ひらがな認識”，信学技報，PRL83-68(1984)

(C R E F I L E H 実行結果)

KAGYU HOUKOU HISTGRAM (4-HOUKOU)					KAGYU HOUKOU HISTGRAM (4-HOUKOU)					KAGYU HOUKOU HISTGRAM (4-HOUKOU)					KAGYU HOUKOU HISTGRAM (4-HOUKOU)				
1.0	3.3	3.8	0.6		0.1	1.1	0.9	0.1		0.3	2.0	1.3	0.1		0.1	0.2	0.1	0.1	
1.3	4.0	7.2	2.3		0.5	2.6	2.9	0.3		0.4	4.0	2.3	1.2		0.1	0.5	0.9	1.2	
2.0	3.1	2.8	1.7		3.1	4.9	1.6	1.2		2.2	3.6	2.2	5.0		0.4	1.0	1.0	1.4	
2.1	1.6	2.8	1.6		1.2	1.5	2.5	2.3		1.3	1.4	1.2	1.3		0.5	0.5	0.4	0.0	
0.6	3.6	1.5	0.1		0.1	1.2	0.8	0.0		0.6	4.0	0.7	0.0		0.2	0.6	0.3	0.0	
4.9	5.8	5.6	1.7		0.7	2.7	2.1	0.1		1.1	5.0	1.8	0.8		0.3	0.8	1.5	1.6	
2.1	2.7	2.1	0.8		2.0	2.7	0.9	1.9		4.6	5.8	1.5	4.3		0.4	0.7	0.6	1.7	
1.8	1.7	0.7	0.5		1.0	1.0	1.5	2.7		2.8	2.1	1.0	1.8		1.0	0.5	0.1	0.0	
2.1	6.5	5.9	1.8		0.5	1.1	2.4	0.6		0.9	1.3	1.7	1.2		0.2	1.3	1.1	0.6	
3.7	8.4	8.3	3.6		1.9	1.4	1.8	1.4		3.8	4.7	4.1	6.1		0.6	2.1	2.0	1.5	
1.5	5.9	10.9	2.9		1.7	3.5	2.4	0.9		1.6	3.0	2.4	2.3		0.3	1.3	2.1	0.7	
3.1	5.9	4.9	4.1		1.3	3.3	1.5	0.4		0.8	0.3	0.2	0.4		0.1	0.9	2.7	1.5	
1.0	3.6	3.3	1.5		0.2	1.8	2.6	0.4		0.2	0.8	1.0	0.4		0.2	1.2	1.2	0.5	
4.6	11.0	10.5	5.1		1.2	2.4	2.6	0.8		3.7	4.5	1.9	6.6		1.0	2.7	2.8	2.6	
2.4	10.3	12.5	2.6		1.9	3.0	2.8	0.8		2.8	4.2	1.2	2.1		0.7	2.9	2.0	0.8	
3.6	5.3	5.5	2.4		2.3	3.6	1.2	0.3		0.4	0.4	0.2	0.2		0.5	1.4	3.0	0.9	

RECTEST0

Test Program (0)

辞書作成及び識別のためのテストプログラム (0)

作成 鶴岡信治 1985年 6月

形式 コンプリート・プログラム 言語:FORTRAN サイズ: 72行

1) 概要

16値文字画像（平仮名 E T L - 6 H : 原データのサイズは横64*縦63）の文字カテゴリー、文字ナンバーを読み込んで、加重方向指數ヒストグラム法の特徴ベクトル（方向指數）を求めて、ファイルに出力する。後のテストプログラムで使用する準備である。

2) 使用法

SMART. CNRL(RECTEST1), SMART. CNTL(RECTEST2), SMART. CNTL(RECTEST3)において辞書を作成するためのデータの作成プログラムとして使用する。

プログラム本体はEXAMPLE.FORT(TESTK0)に格納されている。

作成ファイルはGO.FT11F001で定義しているので、データセット名(DSN)、及びファイルの大きさを設定する、スペース パラメータ (SPACE: 作成文字数によって異なる) は利用者が適宜、指定する。

EXAMPLE.FORT(TESTK0)をTSSモードで実行することもできる。

この場合には、使用データセットとして、ETL-6H を選択するものとすると、以下のように実行に先立ち、ALLOC コマンドを実行する必要がある。

また、出力データセットをDD名をFT11F001として、ALLOCとATTR コマンドを用いるか、ALLOC2 コマンドを用いて確保しておく必要がある。

例えば

```
ALLOC FI(ETLDA) DA(ETL6.VOL01) SHR
ATTR SM RECFM(FB) LRECL(266) BLKSIZE(2660)
ALLC US(SM) F(FT11F001) DA(&&E6DATA) SP(10,10) T REL NEW
FORT EXAMPLE.FORT(TESTK0) GO(NOAE)
```

といった具合に実行する。

(3) 使用しているサブルーチン名

B I N A R Y, S M O O T H, N O R M B 3, M S W A K U, F I L T M 8,
B D F L 2, H I S T S S, W G T I N 4
H I S T 1, T H D S 2 (S P I D E Rより)

(4) 実行結果

求めた特徴ベクトルは、ファイルに出力されるが、実行過程がよく分かるように、
以下のようなメモが、プリンタ、あるいはディスプレイに出力される。

(R E C T E S T 0 実行結果)

MO	I	I	C A T E G O R Y	= 5	1
MO	I	I	C A T E G G O R Y	= 5	2
MO	I	I	C A T E G G O R Y	= 5	3
MO	I	I	C A T E G G O R Y	= 5	4
MO	I	I	C A T E G G O R Y	= 5	5
MO	I	I	C A T E G G O R Y	= 5	6
MO	I	I	C A T E G G O R Y	= 5	7
MO	I	I	C A T E G G O R Y	= 5	8
MO	I	I	C A T E G G O R Y	= 5	9
MO	I	I	C A T E G G O R Y	= 6	0
MO	I	I	C A T E G G O R Y	= 6	1
MO	I	I	C A T E G G O R Y	= 6	2
MO	I	I	C A T E G G O R Y	= 6	3
MO	I	I	C A T E G G O R Y	= 6	4
MO	I	I	C A T E G G O R Y	= 6	5
MO	I	I	C A T E G G O R Y	= 6	6
MO	I	I	C A T E G G O R Y	= 6	7
MO	I	I	C A T E G G O R Y	= 6	8
MO	I	I	C A T E G G O R Y	= 6	9
MO	I	I	C A T E G G O R Y	= 7	0
MO	I	I	C A T E G G O R Y	= 7	1
MO	I	I	C A T E G G O R Y	= 7	2
MO	I	I	C A T E G G O R Y	= 7	3
MO	I	I	C A T E G G O R Y	= 7	4
MO	I	I	C A T E G G O R Y	= 7	5
MO	I	I	C A T E G G O R Y	= 7	6
MO	I	I	C A T E G G O R Y	= 7	7
MO	I	I	C A T E G G O R Y	= 7	8
MO	I	I	C A T E G G O R Y	= 7	9
MO	I	I	C A T E G G O R Y	= 8	0
MO	I	I	C A T E G G O R Y	= 8	1
MO	I	I	C A T E G G O R Y	= 8	2
MO	I	I	C A T E G G O R Y	= 8	3
MO	I	I	C A T E G G O R Y	= 8	4
MO	I	I	C A T E G G O R Y	= 8	5
MO	I	I	C A T E G G O R Y	= 8	6
MO	I	I	C A T E G G O R Y	= 8	7
MO	I	I	C A T E G G O R Y	= 8	8

RECTEST1

Test Program (1)

辞書作成及び識別のためのテストプログラム (1)

作成	木村文隆	1985年 11月
形式	コンプリート・プログラム	言語:FORTRAN サイズ: 53行

(1) 概要

RECTEST0で求めた特徴ベクトルを用いて、統計量計算ルーチンMSTA T4とMSTAT5を使い、識別用の辞書を作成した後、それらを使って識別関数CD1, MQD2, SS1, MS1の値を計算し出力する。

実行に当たっての注意

プログラム中のLはクラス(字種)の番号、Mはクラス内通し番号である。FT03に割り当てるファイルは、クラス番号、クラス内通し番号及び特徴ベクトルの成分64個を項目とするレコードからなる順編成ファイルで、最初の100レコードがクラス1、次の100レコードがクラス2というように、全部で3800レコードを含んでいる。このファイルには、電総研手書き文字データベースETL6の中の数字0~9、特殊記号+, -及び英字A~Zを、100文字/字種、計3800文字、特徴抽出ルーチンで処理して得た特徴ベクトルが入っている。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(RECTEST1)に用意されているのでSUBMITすればよい。この場合には、RECTEST0で行われる特徴ベクトル抽出のステップまで含まれているので、前もってRECTEST0を実行する必要はない。プログラム本体はEXAMPLE.FORT(TESTK1)に格納されている。

EXAMPLE.FORT(TESTK1)をTSSモードで実行することもできる。
この場合には、既に求めてある特徴ベクトル使用するので、実行に先立ち、ALLOCコマンドを実行する必要がある。例えば

```
ALLOC FI (FT03F001) DA (&&E6DATA) SHR  
FORT EXAMPLE.FORT(TESTK1) GO (NOAE)
```

といった具合に実行する。

(3) 使用しているサブルーチン名

M S T A T 4 , M S T A T 5

(4) 実行結果

以下(次ページに亘る2ページ)に示すように、識別関数 C D 1 , M Q D 2 , S S 1 , M S 1 の値を計算し、出力する。

(R E C T E S T 1 実行結果)

CLASS	NO.	FUN.	CD1	22.2	54.0	51.5	52.3	50.1	46.8	62.6	31.2	47.3	54.1		
1	1	CD1		22.2	54.0	51.5	52.3	50.1	46.8	62.6	31.2	47.3	54.1		
		MQD2		-98.31673.1	299.1	666.2	527.9	200.3	714.5	396.7	594.0	408.3			
				-98.31673.1	299.1	666.2	527.9	200.3	714.5	396.7	594.0	408.3			
				-98.31673.1	299.1	666.2	527.9	200.3	714.5	396.7	594.0	408.3			
				-98.31673.1	299.1	666.2	527.9	200.3	714.5	396.7					
		SS1		0.9	0.6	0.8	0.7	0.7	0.8	0.6	0.9	0.8	0.7		
				0.9	0.6	0.8	0.7	0.7	0.8	0.6	0.9	0.8	0.7		
				0.9	0.6	0.8	0.7	0.7	0.8	0.6	0.9	0.8	0.7		
				0.9	0.6	0.8	0.7	0.7	0.8	0.6	0.9				
		MS1		163.3	119.8	135.7	128.6	118.9	140.7	118.9	156.3	120.5	142.8		
				163.3	119.8	135.7	128.6	118.9	140.7	118.9	156.3	120.5	142.8		
				163.3	119.8	135.7	128.6	118.9	140.7	118.9	156.3	120.5	142.8		
				163.3	119.8	135.7	128.6	118.9	140.7	118.9	156.3				
1	2	CD1		15.7	43.3	50.3	50.8	48.8	46.8	59.1	38.0	44.6	57.5		
		MQD2		15.7	43.3	50.3	50.8	48.8	46.8	59.1	38.0	44.6	57.5		
				15.7	43.3	50.3	50.8	48.8	46.8	59.1	38.0	44.6	57.5		
				15.7	43.3	50.3	50.8	48.8	46.8	59.1	38.0				
		SS1		-105.7	803.5	340.6	534.6	323.8	184.6	416.1	400.4	467.2	569.0		
				-105.7	803.5	340.6	534.6	323.8	184.6	416.1	400.4	467.2	569.0		
				-105.7	803.5	340.6	534.6	323.8	184.6	416.1	400.4	467.2	569.0		
				-105.7	803.5	340.6	534.6	323.8	184.6	416.1	400.4				
		MS1		1.0	0.7	0.7	0.7	0.7	0.8	0.7	0.8	0.8	0.6		
				1.0	0.7	0.7	0.7	0.7	0.8	0.7	0.8	0.8	0.6		
				1.0	0.7	0.7	0.7	0.7	0.8	0.7	0.8	0.8	0.6		
				1.0	0.7	0.7	0.7	0.7	0.8	0.7	0.8				
		MS1		121.3	94.8	95.6	98.1	92.4	101.0	95.1	106.7	92.8	92.5		
				121.3	94.8	95.6	98.1	92.4	101.0	95.1	106.7	92.8	92.5		
				121.3	94.8	95.6	98.1	92.4	101.0	95.1	106.7	92.8	92.5		
				121.3	94.8	95.6	98.1	92.4	101.0	95.1	106.7				
2	1	CD1		13.3	44.9	56.4	52.1	52.0	52.0	63.3	42.3	44.6	55.1		
		MQD2		13.3	44.9	56.4	52.1	52.0	52.0	63.3	42.3	44.6	55.1		
				13.3	44.9	56.4	52.1	52.0	52.0	63.3	42.3	44.6	55.1		
				13.3	44.9	56.4	52.1	52.0	52.0	63.3	42.3				
		SS1		-114.51381.3	582.2	751.0	383.3	401.3	386.8	469.3	724.1	839.2			
				-114.51381.3	582.2	751.0	383.3	401.3	386.8	469.3	724.1	839.2			
				-114.51381.3	582.2	751.0	383.3	401.3	386.8	469.3	724.1	839.2			
				-114.51381.3	582.2	751.0	383.3	401.3	386.8	469.3					
		MS1		1.0	0.6	0.7	0.7	0.7	0.7	0.6	0.8	0.7	0.7		
				1.0	0.6	0.7	0.7	0.7	0.7	0.6	0.8	0.7	0.7		
				1.0	0.6	0.7	0.7	0.7	0.7	0.6	0.8	0.7	0.7		
				1.0	0.6	0.7	0.7	0.7	0.7	0.6	0.8				
		MS1		144.8	94.7	108.5	116.1	113.3	107.1	115.5	126.1	100.7	109.9		
				144.8	94.7	108.5	116.1	113.3	107.1	115.5	126.1	100.7	109.9		
				144.8	94.7	108.5	116.1	113.3	107.1	115.5	126.1	100.7	109.9		

			144.8	94.7	108.5	116.1	113.3	107.1	115.5	126.1		
2	2	CD1	13.3	56.6	52.7	48.9	56.6	51.9	64.7	35.3	48.7	50.2
			13.3	56.6	52.7	48.9	56.6	51.9	64.7	35.3	48.7	50.2
			13.3	56.6	52.7	48.9	56.6	51.9	64.7	35.3	48.7	50.2
		MQD2	13.3	56.6	52.7	48.9	56.6	51.9	64.7	35.3		
			-120.61598.6	503.0	721.0	510.6	364.4	575.2	493.3	808.2	757.1	
			-120.61598.6	503.0	721.0	510.6	364.4	575.2	493.3	808.2	757.1	
			-120.61598.6	503.0	721.0	510.6	364.4	575.2	493.3	808.2	757.1	
			-120.61598.6	503.0	721.0	510.6	364.4	575.2	493.3			
		SS1	1.0	0.6	0.7	0.8	0.6	0.7	0.6	0.8	0.7	0.7
			1.0	0.6	0.7	0.8	0.6	0.7	0.6	0.8	0.7	0.7
			1.0	0.6	0.7	0.8	0.6	0.7	0.6	0.8	0.7	0.7
		MS1	1.0	0.6	0.7	0.8	0.6	0.7	0.6	0.8		
			157.5	105.4	120.4	126.6	119.8	121.5	116.7	137.1	113.7	119.7
			157.5	105.4	120.4	126.6	119.8	121.5	116.7	137.1	113.7	119.7
			157.5	105.4	120.4	126.6	119.8	121.5	116.7	137.1	113.7	119.7
			157.5	105.4	120.4	126.6	119.8	121.5	116.7	137.1	113.7	119.7
3	1	CD1	25.0	45.3	62.1	45.4	49.9	57.4	63.5	46.4	44.3	51.3
			25.0	45.3	62.1	45.4	49.9	57.4	63.5	46.4	44.3	51.3
			25.0	45.3	62.1	45.4	49.9	57.4	63.5	46.4	44.3	51.3
		MQD2	25.0	45.3	62.1	45.4	49.9	57.4	63.5	46.4		
			-50.01357.3	848.5	510.1	308.2	457.5	245.7	708.2	634.6	623.9	
			-50.01357.3	848.5	510.1	308.2	457.5	245.7	708.2	634.6	623.9	
			-50.01357.3	848.5	510.1	308.2	457.5	245.7	708.2	634.6	623.9	
			-50.01357.3	848.5	510.1	308.2	457.5	245.7	708.2			
		SS1	0.9	0.7	0.6	0.8	0.7	0.7	0.6	0.8	0.8	0.7
			0.9	0.7	0.6	0.8	0.7	0.7	0.6	0.8	0.8	0.7
			0.9	0.7	0.6	0.8	0.7	0.7	0.6	0.8	0.8	0.7
		MS1	0.9	0.7	0.6	0.8	0.7	0.7	0.6	0.8		
			150.3	102.3	116.8	133.5	127.8	114.9	131.2	131.0	121.0	125.1
			150.3	102.3	116.8	133.5	127.8	114.9	131.2	131.0	121.0	125.1
			150.3	102.3	116.8	133.5	127.8	114.9	131.2	131.0	121.0	125.1
			150.3	102.3	116.8	133.5	127.8	114.9	131.2	131.0	121.0	125.1
3	2	CD1	27.9	43.9	51.7	56.3	47.7	46.5	53.8	44.8	43.4	67.2
			27.9	43.9	51.7	56.3	47.7	46.5	53.8	44.8	43.4	67.2
			27.9	43.9	51.7	56.3	47.7	46.5	53.8	44.8	43.4	67.2
		MQD2	27.9	43.9	51.7	56.3	47.7	46.5	53.8	44.8		
			-64.0	504.2	530.9	674.8	399.8	312.7	361.6	681.1	544.4	852.1
			-64.0	504.2	530.9	674.8	399.8	312.7	361.6	681.1	544.4	852.1
			-64.0	504.2	530.9	674.8	399.8	312.7	361.6	681.1	544.4	852.1
			-64.0	504.2	530.9	674.8	399.8	312.7	361.6	681.1		
		SS1	0.9	0.7	0.7	0.7	0.8	0.8	0.7	0.7	0.8	0.6
			0.9	0.7	0.7	0.7	0.8	0.8	0.7	0.7	0.8	0.6
			0.9	0.7	0.7	0.7	0.8	0.8	0.7	0.7	0.8	0.6
		MS1	0.9	0.7	0.7	0.7	0.8	0.8	0.7	0.7		
			135.9	119.8	100.3	107.4	105.0	106.0	115.9	107.0	113.3	91.2
			135.9	119.8	100.3	107.4	105.0	106.0	115.9	107.0	113.3	91.2
			135.9	119.8	100.3	107.4	105.0	106.0	115.9	107.0	113.3	91.2
			135.9	119.8	100.3	107.4	105.0	106.0	115.9	107.0		

RECTEST2

Test Program (2)

辞書作成及び識別のためのテストプログラム (2)

作成	木村文隆	1985年 11月
形式	コンプリート・プログラム	言語:FORTRAN サイズ: 80行

(1) 概要

RECTEST0で求めた特徴ベクトルを用いて、統計量計算ルーチンMSTA T5を使って識別用の辞書を作成しファイルに出力する。

実行に当たっての注意

実際の文字認識実験では、実験の度に辞書を作成することはせず、一度辞書を作ったならば、それをファイルに書き込み、以後の実験では、ファイルから辞書を読み込んで使用する。このテストプログラム2は、FT10に割り当てられたファイルの中の計100字／字種を使い辞書を作成し、FT15に割り当てられたファイルに出力する辞書作成プログラムである。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(RECTEST2)に用意されているのでSUBMITすればよい。この場合には、RECTEST0で行われる特徴ベクトル抽出のステップまで含まれているので、前もってRECTEST0を実行する必要はない。プログラム本体はEXAMPLE.FORT(TESTK2)に格納されている。

EXAMPLE.FORT(TESTK2)をTSSモードで実行することもできる。

この場合には、既に求めてある特徴ベクトル使用するので、実行に先立ち、ALLOCコマンドを実行する必要がある。

また、出力データセットをDD名をFT15F001として、ALLOCとATTR コマンドを用いるか、ALLOC2 コマンドを用いて確保しておく必要がある。

例えば

```
ALLOC FI(FT10F001) DA(&&E6DATA) SHR
ATTR SM RECFM(FB) LRECL(16900) BLKSIZE(16900)
ALLC US(SM) F(FT15F001) DA(DICTIOE6.DATA) SP(10, 10) T REL NEW
```

FORT EXAMPLE. FORT(TESTK2) GO(NOAE)
といった具合に実行する。

(3) 使用しているサブルーチン名

M S T A T 5

(4) 実行結果

求めた辞書データは、ファイルに出力される。

プリンタあるいはディスプレイ上には、メモとして以下のような簡単な出力がされるだけである。

(R E C T E S T 2 実行結果)

```
L = 1 END
L = 2 END
L = 3 END
L = 4 END
L = 5 END
L = 6 END
L = 7 END
L = 8 END
L = 9 END
L = 1 0 END
L = 1 1 END
L = 1 2 END
L = 1 3 END
L = 1 4 END
L = 1 5 END
L = 1 6 END
L = 1 7 END
L = 1 8 END
L = 1 9 END
L = 2 0 END
L = 2 1 END
L = 2 2 END
L = 2 3 END
L = 2 4 END
L = 2 5 END
L = 2 6 END
L = 2 7 END
L = 2 8 END
L = 2 9 END
L = 3 0 END
L = 3 1 END
L = 3 2 END
L = 3 3 END
L = 3 4 END
L = 3 5 END
L = 3 6 END
L = 3 7 END
L = 3 8 END
```

RECTEST3

Test Program (3)

テストプログラム

作成	木村文隆	1985年 11月
形式	コンプリート・プログラム	言語:FORTRAN サイズ: 70行

(1) 概要

辞書を読み込んで、修正2次識別関数MQD2を使って識別実験を行い、誤読文字のリストと精読率をプリントする。

実行に当たっての注意

MCAMINとMCAMAXは、実験対象とする字種の開始番号と終了番号で、
1 <= MCAMIN <= MCAMAX <= 38 を満たすものである。MNOMINとMNOMAXは、実験対象とするクラス内通し番号である。

このテストプログラム3では、MCAMIN、MCAMAX、MNOMIN、
MNOMAXをREAD文で、キー入力するようにしてある。プログラムTESTK3中に
あるルーチンSORTSIは、自動コールライブラリ中のソーティングルーチンで、各字種に対する識別関数の値をソートし、最小値を求めるのに利用している。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(RECTEST3)に用意されているのでSUBMITすればよい。RECTEST3には、特徴抽出(TESTK0)、辞書作成(TESTK2)ステップまで含まれている

認識のプログラム本体はEXAMPLE.FORT(TESTK3)に格納されている。

EXAMPLE.FORT(TESTK0)、EXAMPLE.FORT(TESTK2)、に続いて、EXAMPLE.FORT(TESTK3)を TSSモードで実行することもできる。

この場合には、既に作成済みの辞書データ(FT15)とテスト文字用データを読み込む必要があるので、実行に先立ち、ALLOC コマンドを実行する必要がある。

また、出力データセットをDD名をFT15F001として、ALLOCとATTR コマンドを用いるか、ALLOC2 コマンドを用いて確保しておく必要がある。

例えば

ALLOC FI(FT10F001) DA(&&E6DATA) SHR

ALLOC FI(FT15F001) DA(&&E6DIC) SHR

FORT EXAMPLE FORT(TESTK3) GO(NOAE)

といった具合に実行する。

(3) 使用しているサブルーチン名

S O R T S I

(4) 実行結果

認識結果が以下のように出力される。

(R E C T E S T 3 実行結果)

KEYIN MCAMIN,MCAMAX,MNOMIN,MNOMAX					
CATNO=51 MNO=1	31	21	1	11	28
	-110.568	-110.568	-110.568	-110.568	111.672
CATNO=51 MNO=2	31	21	1	11	36
	-116.807	-116.807	-116.807	-116.807	136.879
CATNO=51 MNO=3	31	21	1	11	35
	-125.517	-125.517	-125.517	-125.517	174.278
CATNO=51 MNO=4	31	21	1	11	36
	-127.434	-127.434	-127.434	-127.434	257.130
CATNO=51 MNO=5	21	1	31	11	25
	-71.224	-71.224	-71.224	-71.224	192.534
CATNO=51 MNO=6	1	21	31	11	17
	-101.893	-101.893	-101.893	-101.893	231.846
CATNO=51 MNO=7	31	21	1	11	28
	-129.051	-129.051	-129.051	-129.051	77.970
CATNO=51 MNO=8	31	21	1	11	36
	-112.064	-112.064	-112.064	-112.064	198.075
CATNO=51 MNO=9	31	21	1	11	28
	-118.305	-118.305	-118.305	-118.305	30.277
CATNO=51 MNO=10	21	11	1	31	26
	-115.383	-115.383	-115.383	-115.383	343.503
CATNO=51 MNO=11	31	21	1	11	28
	-122.199	-122.199	-122.199	-122.199	103.940
CATNO=51 MNO=12	31	21	1	11	28
	-118.255	-118.255	-118.255	-118.255	46.918
CATNO=51 MNO=13	31	21	1	11	28
	-108.734	-108.734	-108.734	-108.734	275.431
CATNO=51 MNO=14	31	21	1	11	28
	-113.043	-113.043	-113.043	-113.043	259.712
CATNO=51 MNO=15	31	21	1	11	35
	-107.777	-107.777	-107.777	-107.777	174.209

(以下省略)

D I S P T S T

Test Program for GSPDSP

G S P D S P のサンプルプログラム

作成	竹下 鉄夫	1 9 8 6 年 5 月
形式	コンプリート・プログラム	言語: FORTRAN サイズ: 44行

(1) 概要

グラフィックディスプレイに文字パターンを出力するG S P D S Pの使用法を説明するテストプログラムである。

(2) 使用法

このテストプログラムは、グラフィック表示機能のある端末から、T S S モードで実行する。(バッチジョブとしては動作しないので、注意すること)

T S S モードには、次のような使用モードがあるが、どれを使用してもよい。

(a) P F D E D I T 内では ‘RUN FORT F I’ と入力する

(b) F S O E D I T 内では ‘RUN’ と入力する

(c) T S S コマンドモードでは ‘FORT データセット名’ と入力する

21、22行のコメントを付け変えることによりグラフィックディスプレイのタイプが変わる。

(3) 注意

(a) テストプログラムはメモリ領域を大量に使用するため、LOGON時にS I Z E パラメータを100M指定すること。

例 LOGON TSS Z49999A/PASS SIZE(100M)

(4) 使用しているサブルーチン名

I N T D S P, D S P G S P, E N D D S P

(5) 実行結果

グラフィック画面上に文字図形が、枠で囲って出力される。

XY8HB

Line Drawing of Binary Patterns Using XY-Plotter for ETL8

XYプロッターによる2値パターンの線画出力 (ETL8用)

作成	鶴岡信治	1987年 6月
形式	コンプリート・プログラム	言語:FORTRAN サイズ:70行

(1) 概要

データ文 (read(5,*)で読まれるデータ) で指定した文字の2値パターンをA4版1枚に横4文字、縦5文字、計20文字、XYプロッター (NLPまたはOPR) で出力する。 (ETL8用)

(2) 使用法

データ文は、出力したい文字を1行で1文字指定する。指定する項目は、文字データベースETL8の巻番号 (1-32, 1-4桁目), 巷ごとのセット番号 (1-5, 5-8桁目), 字種番号 (1-956, 9-12桁目) であり、それぞれ整数4桁で指定する。また、データ文の最後の行は0以下の数を1-4桁目に記入しておくこと。

NLPに出力するジョブ制御文のデータセットがSMART.CNTL(XY8HB)に用意されているので、SUBMITすればよい。

OPRに出力する場合には上記ジョブ制御文のパラメータの内、XYNLP=ON を削除した上で、

```
//GO.XYOUT DD DSN=A49999A. XY.DATA, UNIT=PUB,  
//           SPACE=(TRK,(5,5),RLSE), DISP=(NEW,CATLG)
```

を加えた上でSUBMITすればよい。

プログラム本体はEXAMPLE.FORT(XY8HB)に格納されている。

EXAMPLE.FORT(XY8HB)をTSSモードで実行することもできる。出力をOPRに出すものとすると、以下のようなコマンドを入力すればよい

```
ALLOC FI(ETLDA) DA(ETL8.VOL01) SHR  
ALLOC FI(XYOUT) DA(XY.DATA) NEW SP(1 1) T  
FORT EXAMPLE.FORT(XY8HB) GO(NOAE)
```

実行が終了した時点で、

XYPLOT XY. DATA OPR(プリンタID)
と入力すればプリンタIDで指定したO P Rに画像が出力される。

(3) 使用しているサブルーチン名

X I N T, X V I E W P, R E T L 8 H, F I L T M 8, B I N A R Y
S M O O T H, N O R M B 3, P L T B 4 5, X E N D

(4) データ文の例

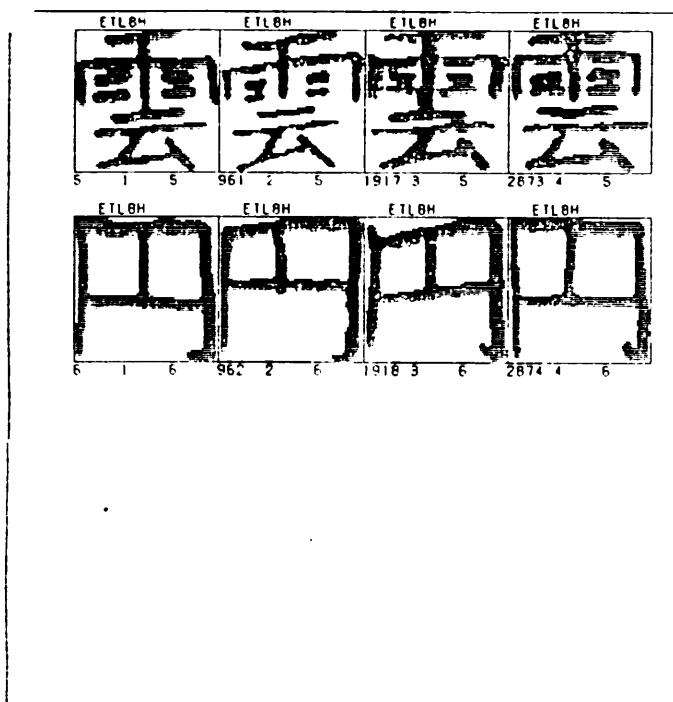
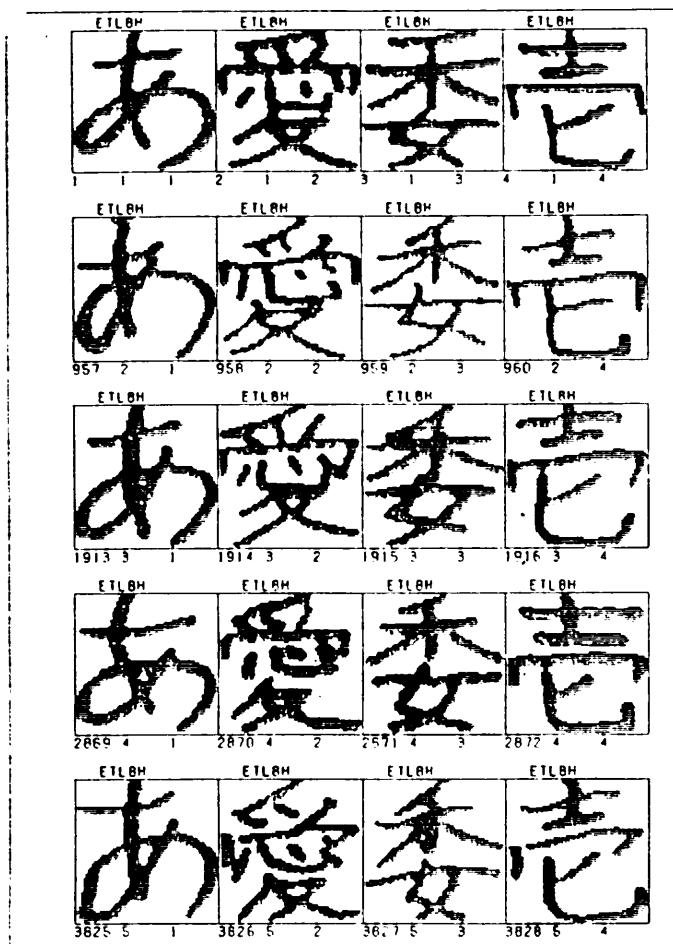
//GO. SYSIN DD *

```
1   1   1
2   1   2
3   1   3
4   1   4
1   1   2
.....
.....
.....
-1
//
```

(5) 実行結果

以下(次ページ)に示す通りである。

(X Y 8 H B 実行結果)



X Y 8 H I S T

Line Drawing of 4-Direction Index Histograms Using XY-Plotter for ETL8

X Y プロッターによる加重方向指数ヒストグラムの線画出力 (ETL8用)

作成	鶴岡信治	1987年 6月
形式	コンプリート・プログラム	言語: FORTRAN サイズ: 155行

(1) 概要

データ文 (read(5,*)で読まれるデータ) で指定した文字の加重方向指数ヒストグラム (4×4 の小領域の 4 方向) を A4 版 1 枚に横 4 文字, 縦 5 文字, 計 20 文字分, X Y プロッター (NLP または OPR) で出力する (ETL8用)

(2) 使用法

データ文は、出力したい文字を 1 行で 1 文字指定する。指定する項目は、文字データベース ETL8 の巻番号 (1-32, 1-4 行目), 巷ごとのセット番号 (1-5, 5-8 行目), 字種番号 (1-956, 9-12 行目) であり、それぞれ整数 4 行で指定する。また、データ文の最後の行は 0 以下の数を 1-4 行目に記入しておくこと。

NLP に出力するジョブ制御文のデータセットが SMART.CNTL(XY8HIST) に用意されているので、SUBMITすればよい。

OPR に出力する場合には上記ジョブ制御文のパラメータの内、XYNLP=ON を削除した上で、

```
//GO.XYOUT DD DSN=A49999A.XY.DATA, UNIT=PUB,  
//           SPACE=(TRK,(5,5),RLSE), DISP=(NEW,CATLG)
```

を加えた上で SUBMIT すればよい。

プログラム本体は EXAMPLE.FORT(XY8HIST) に格納されている。

EXAMPLE.FORT(XY8HIST) を TSS モードで実行することもできる。出力を OPR に出すものとすると、以下のようなコマンドを入力すればよい。

```
ALLOC FI(ETLDA) DA(ETL8.VOL01) SHR  
ALLOC FI(XYOUT) DA(XY.DATA) NEW SP(1 1) T  
FORT EXAMPLE.FORT(XY8HIST) GO(NOAE)
```

実行が終了した時点で、

XYPLOT XY. DATA OPR(プリンタID)
と入力すればプリンタIDで指定したOPRに画像が出力される。

(3) 使用しているサブルーチン名

XINT, XVIEWP, RETL8H, FILTM8, BINARY
SMOOTH, NORMMM3, MSWAKU, BDFL2 (SPIDERより)
HISTS, WGTIN4, WEIGT1, PLTH45, XEND

(4) データ文の例

GO. SYSIN DD *

```
1   1   1
2   1   2
3   1   3
4   1   4
1   1   2
.....
.....
.....
-1
```

(5) 実行結果

以下(次ページ)に示す通りである。

(X Y 8 H I S T 実行結果)

ETLBH	ETLBH	ETLBH	ETLBH
- L -	- - -	- - -	- - -
- L -	L - L	L - L	L - L
✓ ✓ -	✓ - -	✓ - -	L - L
- - -	- - -	- - -	- L -
1 1 1	2 2 1	3 3 1	4 4 1
ETLBH	ETLBH	ETLBH	ETLBH
- L -	- - -	- - -	- - -
- L -	L - w L	L - L	- - -
✓ ✓ -	✓ - x -	- - -	L - L
- - -	- - -	- - -	- L -
957 1 2	958 2 2	959 3 2	960 4 2
ETLBH	ETLBH	ETLBH	ETLBH
- L -	- - -	- - -	- - -
- L -	L - L	L - L	L - L
✓ ✓ -	✓ - -	✓ - -	L - L
- - -	- - -	- - -	- L -
1913 1 3	1914 2 3	1915 3 3	1916 4 3
ETLBH	ETLBH	ETLBH	ETLBH
- L -	- - -	- - -	- - -
- L -	L - L	L - L	L - L
✓ ✓ -	✓ - -	✓ - -	L - L
- - -	- - -	- - -	- L -
2869 1 4	2870 2 4	2871 3 4	2872 4 4
ETLBH	ETLBH	ETLBH	ETLBH
- L -	- - -	- - -	- - -
- L -	L - L	L - L	L - L
✓ ✓ -	✓ - -	✓ - -	L - L
- - -	- - -	- - -	- L -
3825 1 5	3826 2 5	3827 3 5	3828 4 5

ETLBH	ETLBH	ETLBH	ETLBH
- - -	- - -	- - -	- - -
L L L L	L L L	L L L	L L L
L - -	L - -	L - -	- - -
- - -	- - -	- - -	- - -
5 5 1	961 5 2	1917 5 3	2873 5 4
ETLBH	ETLBH	ETLBH	ETLBH
L - - L	L - - L	L - - L	L - - L
L L L	L L L	L L L	L L L
L L L	L L L	L L L	L L L
. . L	. . L	. . L	. . L
6 6 1	962 6 2	1918 6 3	2874 6 4

XY8HISTB

Line Drawing of Binary Patterns and 4-Direction Index Histograms
Using XY-Plotter for ETL8

XYプロッターによる2値パターンと加重方向指数ヒストグラムの線画出力（ETL8用） XY8HBとXY8HISTの機能を合わせたもの

作成	鶴岡信治	1987年 7月
形式	コンプリート・プログラム	言語：FORTRAN サイズ：155行

(1) 概要

データ文 (read(5,*)で読まれるデータ) で指定した文字の2値パターンと加重方向指数ヒストグラム (4×4 の小領域の4方向) をつづけて、XYプロッター (NLPまたはOPR) で出力する。A4版1枚に横4文字、縦5文字、計20文字分出力する (ETL8用)。

(2) 使用法

データ文は、出力したい文字を1行で1文字指定する。指定する項目は、文字データベースETL8の巻番号 (1-32, 1-4桁目), 卷ごとのセット番号 (1-5, 5-8桁目), 字種番号 (1-956, 9-12桁目) であり、それぞれ整数4桁で指定する。また、データ文の最後の行は0以下の数を1-4桁目に記入しておくこと。

NLPに出力するジョブ制御文のデータセットがSMART.CNTL(XY8HISTB)に用意されているのでSUBMITすればよい。

OPRに出力する場合には上記ジョブ制御文のパラメータの内、XYNLP=ON を削除した上で、

```
//GO. XYOUT DD DSN=A49999A. XY. DATA, UNIT=PUB,  
//           SPACE=(TRK,(5,5),RLSE), DISP=(NEW,CATLG)
```

を加えた上でSUBMITすればよい。

プログラム本体はEXAMPLE.FORT(XY8HISTB)に格納されている。

EXAMPLE.FORT(XY8HISTB)をTSSモードで実行することもできる。出力をOPRに出すものとすると、以下のようなコマンドを入力すればよい。

```
ALLOC FI(ETLDA) DA(ETL8.VOL01) SHR
```

```
ALLOC FI (XYOUT) DA (XY. DATA) NEW SP(1 1) T  
FORT EXAMPLE. FORT (XY8HISTB) GO (NOAE)
```

実行が終了した時点で、

```
XYPLOT XY. DATA OPR(プリンタID)
```

と入力すればプリンタIDで指定したOPRに画像が出力される。
れている。

(3) 使用しているサブルーチン名

```
XINT, XVIEWP, RETL8H, FILTM8, BINARY  
SMOOTH, NORMM3, MSWAKU, BDFL2 (SPIDERより)  
HISTSS, WGTIN4, WEIGT1, PLTH45, PLTB45  
PLT2TI, XEND
```

(4) データ文の例

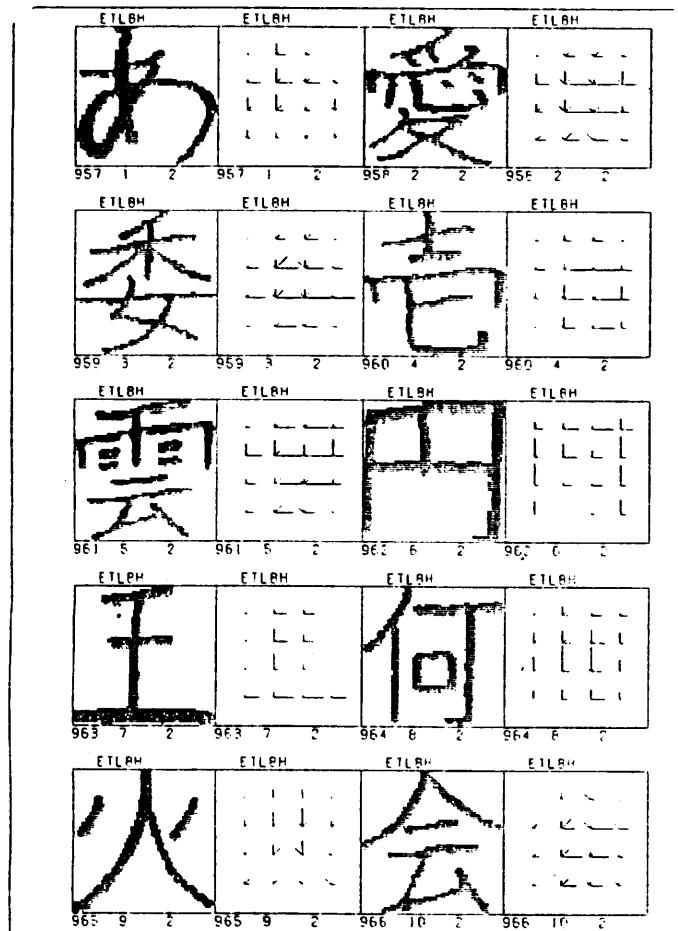
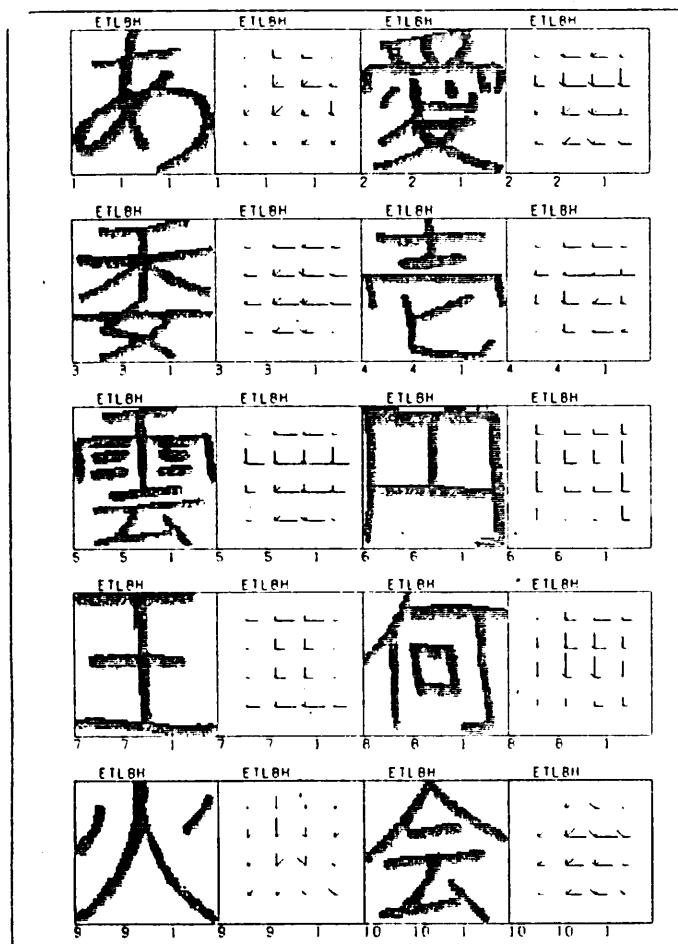
```
//GO. SYSIN DD *
```

```
1 1 1  
2 1 2  
3 1 3  
4 1 4  
1 1 2  
.....  
.....  
.....  
-1  
//
```

(5) 実行結果

以下（次ページ）に示す通りである。

(X Y 8 H I S T B 実行結果)



F A X P A R A S

Test Program for FXPPLP

F X P P L P のサンプルプログラム

作成	竹下 鉄夫	1 9 8 6 年 2 月
形式	コンプリート・プログラム	言語 : FORTRAN サイズ : 1 4 4 行

(1) 概要

ファクシミリ、ヴァーサテックプリンタ・プロッタに文字パターンを出力する際、最初にパラメータをセットするために使用する F X P P L P の使用法を説明するテストプログラムである。

会話型でパラメータを入力すると F X P P L C で入力したパラメータチェックを行う。

(2) 使用法

T S S モードで実行する。 (バッチジョブとしては動作しない)

- (a) P F D P F D E E D I T 内では ‘R U N F O R T F I ’ と入力する
- (b) F S O E D I T 内では ‘R U N ’ と入力する
- (c) T S S コマンドモードでは ‘F O R T データセット名’ と入力する

(3) 使用しているサブルーチン名

F X P P L P , F X P P L C

(4) 実行結果

次ページに示すように、ディスプレイ上に入力を促すプロンプトが出るので、必要なデータをキーボードから入力することにより、パラメータチェックをした結果が出力される。

(FAXPARAS 実行結果)

```
FORTRAN77 EX COMPILER ENTERED
END OF COMPIRATION,HIGHEST SEVERITY CODE=00
I1      = 64
J1      = 63
SIZE    = A4
I2      = 1728
J2      = 2500
FRAME   = YES
PLEASE INPUT KSCALE (INTEGER)
00380001 ?
2
PLEASE INPUT IWIDE (INTEGER)
00550001 ?
2
PLEASE INPUT AISSTEP (FORMAT F6.3)
00610001 ?
1.250
PLEASE INPUT AJSTEP (FORMAT F6.3)
00670001 ?
1.20
I1      = 64
J1      = 63
SIZE    = A4
I2      = 1728
J2      = 2500
AISSTEP= 1.2500000
AJSTEP= 1.1999998
FRAME   = YES
KSCALE= 2
TITLE  = I1= 64,J1= 63, SIZE=A4, IS= 1.25,JS= 1.20,
FRAME=YES,WIDE=2,SCALE=2
ALL PARAMETR OK ? 'Y' OR 'N' (FORMAT A1)
01010001 ?
Y
CALL FXPPLP !!!
```

(以下省略)

F A X T S T 1 S

Test Program for FXPPLP, FXPPLR & FXPPLO

F X P P L P, F X P P L R, F X P P L Oのサンプルプログラム（その1）

作成	竹下 鉄夫	1 9 8 6 年 2月
形式	コンプリート・プログラム	言語:FORTRAN サイズ: 1 2 0 行

(1) 概要

ファクシミリ、ヴァーサテックプリンタ・プロッタに文字パターンを出力するF X P P L P, F X P P L R, F X P P L Oの使用法を説明するテストプログラムである。

出力用紙サイズ‘A4’，ファクシミリ出力の例である。ヴァーサテックプリンタ・プロッタに出力したい場合は35行のCHARACTER文の変数FAXPPの初期値を‘FAX’から‘PPL’とすれば良い。

(2) 使用法

このテストプログラムは、TSSモードで実行する。

(バッチジョブとしては動作しないので、注意すること)

TSSモードには、次のような使用モードがあるが、どれを使用してもよい。

- (a) PFD PFDE EDIT内では‘RUN FORT FI’と入力する
- (b) FSO EDIT内では‘RUN’と入力する
- (c) TSS コマンドモードでは‘FORT データセット名’と入力する

(3) 注意

ヴァーサテックプリンタ・プロッタを使用する場合は

- (a) RUN前に次のコマンドをTSSコマンドモードで入力しておくこと。
IPP L

- (b) RUNの後で実際にヴァーサテックプリンタ・プロッタに出力依頼をするため、次のコマンドをTSSコマンドモードで入力すること。
R P P L E (ON)

(4) 使用しているサブルーチン名

F X P P L P, F X P P L C, F X P P L R, F X P P L O

F A X T S T 2 S

Test Program for FXPPLP, FXPPLR & FXPPLO

F X P P L P, F X P P L R, F X P P L O のサンプルプログラム（その2）

作成	竹下 鉄夫	1 9 8 6 年 2月
形式	コンプリート・プログラム	言語 : FORTRAN サイズ : 1 1 5 行

(1) 概要

ファクシミリ、ヴァーサテックプリンタ・プロッタに文字パターンを出力する F X P P L P, F X P P L R, F X P P L O の使用法を説明するテストプログラムである。

出力用紙サイズ ‘B 4’ , ファクシミリ出力の例である。ヴァーサテックプリンタ・プロッタに出力したい場合は3 4行のCHARACTER文の変数F A X P P L の初期値を ‘F A X’ から ‘P P L’ とすれば良い。

(2) 使用法

T S S モードで実行する。（バッチジョブとしては動作しない）

- (a) P F D E D I T 内では ‘R U N F O R T F I ’ と入力する
- (b) F S O E D I T 内では ‘R U N ’ と入力する
- (c) T S S コマンドモードでは ‘F O R T データセット名’ と入力する

(3) 注意

ヴァーサテックプリンタ・プロッタを使用する場合は

- (a) R U N 前に次のコマンドをT S S コマンドモードで入力しておくこと。

I P P L

- (b) R U N の後で実際にヴァーサテックプリンタ・プロッタに出力依頼をするため、次のコマンドをT S S コマンドモードで入力すること。

R P P L E (ON)

(4) 使用しているサブルーチン名

F X P P L P, F X P P L C, F X P P L R, F X P P L O

(5) 実行結果 省略

R D T E S T B I

File Read Test for Binary Image

特徴抽出ファイルの読み出しテスト

作成	竹下鉄夫	1985年 11月
形式	コンプリート・プログラム	言語:FORTRAN サイズ: 44行

(1) 概要

C R E A T E B I にて作成したファイルを読み出し、特徴抽出ベクトルを格納するファイルが正しく作成されたことを確認する。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(RDTESTBI)に用意されているのでSUBMITすればよい。プログラム本体はEXAMPLE.FORT(RDTESTBI)に格納されている。

EXAMPLE.FORT(RDTESTBI)をTSSモードで実行することもできる。

作成ファイルはGO.FT03F001で定義しているので、データセット名(DSN)は利用者が適宜、指定する。

特定の字種(変数:IRD MJ C)の指定した順序の文字(変数:IRD MJ N)の特徴ベクトルの内容を読み出すことが出来る。

(3) 使用しているサブルーチン名

無し。

(4) 実行結果

次ページに示す。

(R D T E S T B I 実行結果)

(R D T E S T H X 実行結果)

(R D T E S T K T 実行結果)

```

MOJI CATEGORY = 10
MOJI NUMBER = 100
    0.328   0.656   0.656   0.438   0.190   0.084   0.384
    0.438   0.546   0.326   0.218   0.000   0.024   0.414
    0.000   0.000   0.000   0.000   0.000   0.000   0.330
    0.000   0.000   0.000   0.000   0.000   0.000   0.350
    0.000   0.000   0.000   0.000   0.000   0.054   0.354
    0.000   0.438   0.656   0.656   0.656   0.796   0.438
    0.000   0.248   0.626   0.626   0.356   0.302   0.024

```

R D T E S T H X

File Read Test for 16-Level Image

特徴抽出ファイルの読み出しテスト

作成	竹下鉄夫	1985年 11月
形式	コンプリート・プログラム	言語:FORTRAN サイズ: 44行

(1) 概要

C R E A T E H X にて作成したファイルを読み出し、特徴抽出ベクトルを格納するファイルが正しく作成されたことを確認する。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(RDTESTHX)に用意されているのでSUBMITすればよい。プログラム本体はEXAMPLE.FORT(RDTESTHX)に格納されている。

EXAMPLE.FORT(RDTESTHX)をTSSモードで実行することもできる。

作成ファイルはG0.FT03F001で定義しているので、データセット名(DSN)は利用者が適宜、指定する。

特定の字種(変数: I R D M J C)の指定した順序の文字(変数: I R D M J N)の特徴ベクトルの内容を読み出すことが出来る。

(3) 使用しているサブルーチン名

無し。

(4) 実行結果

60ページに示す。

R D T E S T K T

File Read Test for KATAKANA Character

特徴抽出ファイルの読み出しテスト

作成 竹下鉄夫 1985年 11月

形式 コンプリート・プログラム 言語:FORTRAN サイズ: 45行

(1) 概要

C R E A T E K T にて作成したファイルを読み出し、特徴抽出ベクトルを格納するファイルが正しく作成されたことを確認する。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(RDTESTKT)に用意されているのでSUBMITすればよい。プログラム本体はEXAMPLE.FORT(RDTESTKT)に格納されている。

EXAMPLE.FORT(RDTESTKT)をTSSモードで実行することもできる。

作成ファイルはGO.FT03F001で定義しているので、データセット名(DSN)は利用者が適宜、指定する。

特定の字種(変数: I R D M J C)の指定した順序の文字(変数: I R D M J N)の特徴ベクトルの内容を読み出すことが出来る。

(3) 使用しているサブルーチン名

無し。

(4) 実行結果

60ページに示す。

T S T H I S T S

Test Program for Calculating Packed Image Data

パッキング画像データのヒストグラムを求めるテストプログラム

作成	竹下鉄夫	1985年 11月
形式	コンプリート・プログラム	言語:FORTRAN サイズ: 47行

(1) 概要

A4版大のFAX形式の画像データ（横1728×縦2500の標本点よりなる）をアンパックすることなく、画像の黒部分の数を数えて、ヒストグラムを求めて、その値をプリンタあるいはディスプレイ上に表示する。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(TSTHISTC)に用意されているのでSUBMITすればよい。プログラム本体はEXAMPLE.FORT(TSTHISTS)に格納されている。
EXAMPLE.FORT(TSTHISTS)をTSSモードで実行することもできる。

(3) 使用しているサブルーチン名

RWHIST, CLHIST。

(4) 実行結果

縦方向のヒストグラムが1728個、横方向のヒストグラムが2500個それぞれ求められるが、数が多すぎるので」、50番目毎の計算値が表示される。
(具体的な出力表示は省略する。)

T S T S O U K N

Test Program for Calculating Auto-correlation Coefficients

バックキング画像データのヒストグラムを求め、自己相関係数を計算する。

作成	竹下鉄夫	1985年 11月
形式	コンプリート・プログラム	言語:FORTRAN サイズ: 49行

(1) 概要

A4版大のFAX形式の画像データ（横1728×縦2500の標本点よりなる）をアンパックすることなく、画像の黒部分の数を数えて、列方向のヒストグラムを求め、その値を基にして自己相関係数を計算し、結果をプリンタあるいはディスプレイ上に表示する。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(TSTSOUKN)に用意されているのでSUBMITすればよい。プログラム本体はEXAMPLE.FORT(TSTSOUKN)に格納されている。
EXAMPLE.FORT(TSTSOUKN)をTSSモードで実行することもできる。

(3) 使用しているサブルーチン名

RWHIST, SOUKAN。

(4) 実行結果

列方向のヒストグラムを求めて、これをプリンタあるいはディスプレイ上に表示した後、自己相関係数を計算し、始めの1/6の数を5つつ毎に表示する。
(具体的な出力表示は省略する。)

LINESG

Test Program for LINESG Subroutine

LINESGサブルーチンのテストプログラム

作成	吉村ミツ	1987年	9月
形式	コンプリート・プログラム 言語:FORTRAN サイズ: 46行		

(1) 概要

文字の特徴点を表すX Y座標点列から、連結性(4, 8)にしたがって直線を発生させ、2値の線文字画像を得て、POUT16などでラインプリンターに出力する。

(2) 使用法

ジョブ制御文のデータセットがSMART.CNTL(LINESG)に用意されているのでSUBMITすればよい。プログラム本体はEXAMPLE.FORT(LINESG)に格納されている。

(3) 使用しているサブルーチン名

LINESG, POUT16

(4) 実行結果

次ページに示す。

1970-01-01 00:00:00

1970-01-01 00:00:00

1970-01-01 00:00:00

0 0 0 0 0 0 0
0 0 0 0 0 0 0

1970-01-01 00:00:00

1970-01-01 00:00:00

0 0 0 0 0 0 0
0 0 0 0 0 0 0

1970-01-01 00:00:00