

アルゴリズム・データベースに基づく 画像処理用言語 HILLS について

鈴木 秀智 鳥脇 純一郎
(豊橋技術科学大学)

1. まえがき

近年の画像処理の急速な発展に伴い、数多くの有用な処理手法が蓄積されてきたが、それらの多くは、専門外の研究者等が容易に利用できるように整備されてはいないため、各種の画像処理応用分野で十分に活用されているとはいえない。この点を解決するために、画像処理用の言語やパッケージが多数開発、試作されているが⁽⁴⁾、それらの大部分は、(1). 特定の用途や専用システム向きのもので、画像データの入出力、ファイルの管理、会話型システム、等に重点をおくもの (VICAR⁽¹⁾等)、または、(2). 通常の高級言語 (FORTRAN等) と同レベルのもので、特に画像データ (2次元配列データ) の基本的処理の記述を便利にしたもの (PAXII⁽²⁾, PIXAL⁽³⁾等) である。前者は汎用性に欠け、後者は、少し複雑な処理手順に対しては、利用者がプログラム化しなくてはならない、という欠点がある。後者に対しては、最近、画像処理用アルゴリズムデータベース (Image Processing Algorithm Database - IPAD) SLIP⁽⁴⁾, SPIDER⁽⁵⁾が開発されて、各処理のプログラミングの手間を省けるようになってきている。しかし、両者は、共に300個を超えるFORTRANサブルーチンの集合であり、しかも、サブルーチン名が処理機能を完全には反映していないため、応用分野の研究者が、必要な機能を有するサブルーチンを選択したり、その引数を正しく設定するのにかなりの時間と手間を要するのが実情であり、このことが、IPADの普及の障害の1つになっている。

そこで、このような問題を解決するために、筆者らは、IPADを基礎におく画像処理用言語 HILLS (High Level Language for SLIP), 及び、これを実現する HILLSシステムを開発することにした。HILLSは、基本的には、適当なレベルの処理単位をその機能を表すキーワード (kw) 系列で表現することにより、一連の画像処理手順を記述するという、極めて簡単な言語である。本文では、これらの概要と実験結果について述べる。

2. HILLSシステムの概要

HILLSの詳細な仕様を述べる前に、この言語を処理するHILLSシステムの機能及び構成について述べる。

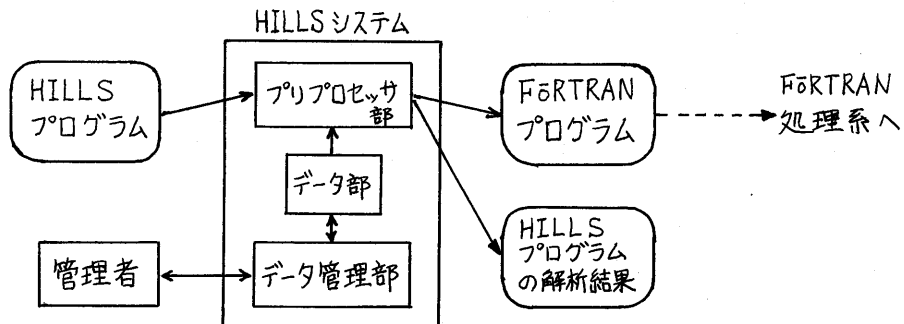


図1. HILLSシステムの概要

HILLSシステムは、HILLSで記述された処理を実行するFORTRANプログラムを生成するシステムである。HILLSの1つの文は、原則として、IPAD (SLIPまたはSPIDER)の1個のサブルーチンに対応し、HILLSシステムは、このサブルーチンと呼ぶCALL文を生成する。

図1. に示したように、HILLSシステムは、プリプロセッサ部、データ管理部、及び、データ部から成る。データ部は、HILLSで記述されたプログラムをFORTRANプログラムに変換するときに必要な諸情報を格納する部分である。データ管理部は、このデータ部のデータを管理するためのプログラムであり、各項目の追加、削除、及び、変更を行う機能をもつ。プリプロセッサ部は、データ部のデータを使って、HILLSプログラムをFORTRANプログラムに変換するためのプログラムである。このプログラムの処理結果として、略記部分をデフォルト機能で補正した完全なHILLSプログラム、エラーメッセージ、及び、FORTRANプログラムを出力する。この出力結果をFORTRAN処理系に引き渡すことにより、目的とする画像処理が実行される。

3. HILLSの仕様

本節では、言語HILLSの仕様について述べる。

HILLSプログラムは、いくつかのHILLS文とFORTRAN文とから成る。すなわち、FORTRAN文の混在を許している。HILLSプログラムの書式を表1. に示す。

表1. HILLSプログラムの構文則 (BNFによる)

<p><HILLSプログラム> ::= { <宣言文> } { <FORTRAN文関数定義文> } { <FORTRAN DATA文> } { 実行文 } <終了のHILLS文></p> <p><宣言文> ::= <HILLS宣言文> <FORTRAN宣言文></p> <p><HILLS宣言文> ::= : <u>TINT</u>, <整数型変数列>; : <u>REAL</u>, <実数型変数列>;</p> <p><整数型変数列> ::= <整数型変数> { , <整数型変数> }</p> <p><実数型変数列> ::= <実数型変数> { , <実数型変数> }</p> <p><実行文> ::= <HILLS実行文> <FORTRAN実行文></p> <p><HILLS実行文> ::= <サブルーチン直接指定文> <画像演算文> <一般処理指定></p> <p><サブルーチン直接指定文> ::= : <u>NAME</u>, <サブルーチン名> <条件指定部>;</p> <p><サブルーチン名> ::= 登録されたIPAD中にあるサブルーチン名</p> <p><画像演算文> ::= : <u>ARITH</u>, <画像演算式>;</p> <p><画像演算式> ::= <画像間2項点演算式> <画像定数間演算式></p> <p><画像間2項点演算式> ::= <画像配列名> = <画像配列名> <演算子@> <画像配列名> <画像配列名> = <u>.NOT.</u> <画像配列名></p> <p><画像定数間演算式> ::= <画像配列名> = <値> <演算子@> <画像配列名> <画像配列名> = <画像配列名> <演算子@> <値></p> <p><値> ::= <定数> <スカラー変数名></p> <p><演算子@> ::= + - * /</p> <p><演算子@> ::= <演算子@> <u>.MAX.</u> <u>.MIN.</u> <u>.OR.</u> <u>.AND.</u></p> <p><演算子> ::= <演算子@> <u>.NOT.</u></p> <p><一般処理指定文> ::= : <機能指定部> <条件指定部>;</p> <p><機能指定部> ::= <第1KW@> { , <機能指定用KW@> }</p>	<p>— 次項へ —</p>
--	----------------

<機能指定用KW@> ::= 各処理機能を示すためにあらかじめ用意されたKW
 <第1KW@> ::= ASSIGN | CON-COMP | DIST | FILTER | GENERATE | GEOM | INPUT |
L-OP | MOVE | ORTH | OUTPUT | QUANTIZE | RUN-LENG | SHAPE |
STAT | THIN | THRESHOL | TIME | TYPE CON V

<条件指定部> ::= { , <条件指定用KW> = <値> }

<条件指定用KW> ::= サブルーチンの各引数と対応したKW

<終了のHILLS文> ::= : END ;

<第1KW@> ::= TINT | TREAL | NAME | ARITH | END

<第1KW> ::= <第1KW@> | <第1KW@>

<機能指定用KW> ::= <第1KW> | <機能指定用KW@>

(注意)。変数の宣言は、宣言文以外に、HILLS実行文の<変数>、<画像配列名>の部分に書くことにより行うことができる。

・ 注釈行は、

<注釈行> ::= <HILLS注釈文> | <FORTRANコメント行>

<HILLS注釈文> ::= :: <コメント> ;

であり、終了のHILLS文より前ならばどこにでも書ける。

HILLSの実行文は、いくつかのKWの系列から成り、機能指定部と条件指定部に分けられる。前者は、その文の処理の機能(IPAD中の対応するサブルーチン名)を決定し、後者は、その処理の諸条件(サブルーチンの実引数)を設定する。実行文の例は図5を参照のこと。1つのHILLS文では、機能指定部は最大8個のKWから成り、条件指定部は各サブルーチンの引数の個数に対応した分だけの項目から成る。ただし、実際には、機能指定部、条件指定部の多くの部分が省略可能である。

4. HILLSシステムの構成

本節では、2.のHILLSシステムの各部についてもう少し詳しく述べる。

4.1 データ部 これは、プリプロセッサがHILLSプログラムを処理する際に必要となる情報が格納されている部分であり、大別すると、キーワード系列-サブルーチン名対応表(KW-S表)、及び、引数情報表(ARG表)から成る。

(1). KW-S表 KW系列が木構造で表現され、葉の部分に対応するサブルーチン名があり、これが同時にそのパスの終端であることを示す。一例を図2.に示す。この木構造データを実現するのに図3.に示すようなリスト表現法を用いた。リストの1要素(セル)は図3(b)のように、KW用フィールドとポインタ用フィールドから成る。パケットは、1つのパケットの範囲を示すポインタから成るヘッダ部と、いくつかのセルから成る。これを用いて図2.のデータの一部を表現したのが図3(a)である。あるKWを親とするKWの集合が1つのパケットになり、その親のKWのセルのポインタがこのパケットをさす(例えば、THINは、ITERATIV, SKEL-ETON, TRACE, L-MAX, SPECIFICから成るパケットをさす)。ただし、実際には、KWとサブルーチン名は、あらかじめ各々の一覧表(KW表, S表)をつくっておき、KW-S表のKWやサブルーチン名をさす部分には、各々の表へのポインタ値を入れるようにして、記録密度を高めている。

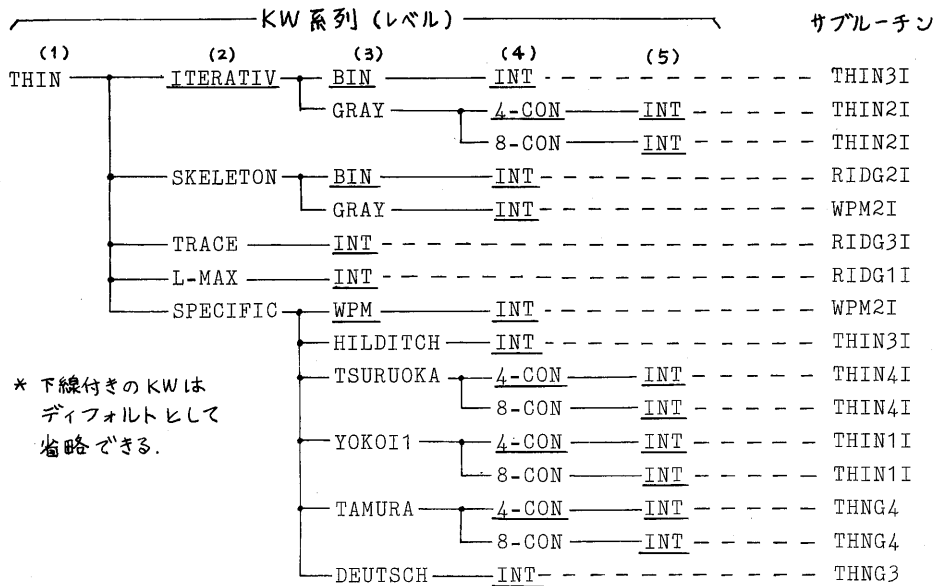


図2. KW系列とサブルーチンの対応例 (細線化処理)

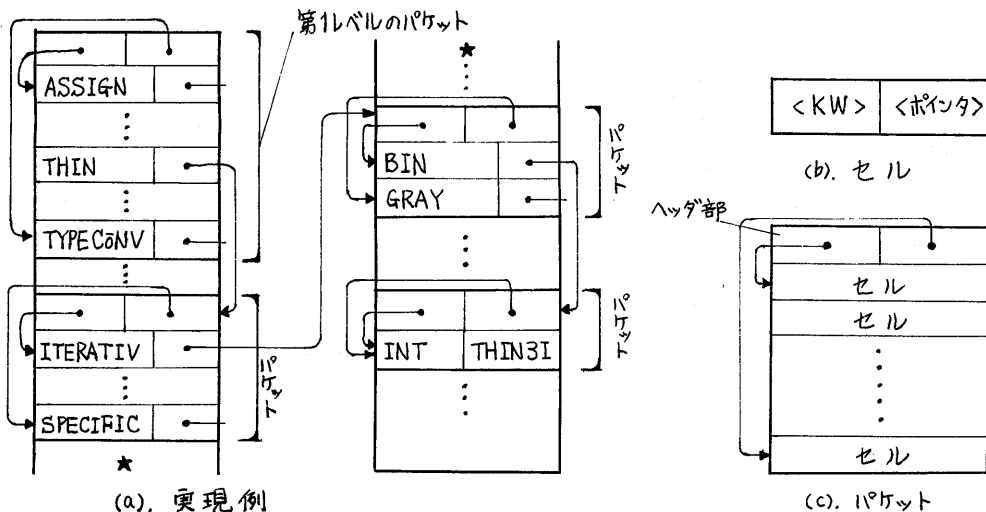


図3. KW-S表の表現方法

(2). ARG表 これは、HILLS システムに登録された各サブルーチンに対し、各引数ごとに、対応する条件指定用KW、型、次元、入出力関係、配列の大きさ、及び、各種のデフォルト値、等を登録したものを一つの表にまとめたものである(図4. 参照)。

図2. の一番上の行のKW系列(または図5(a))に対する、HILLSシステムによる、これらのデータの検索手順を図4. に示す。まず、KW-S表中でKW表中のTHINをさすセルを第1レベルのパケットで探し、そのセルのポインタにより、次のレベルのパケットへ行き、ITERATIVに対応するセルを探し、……、そして最後に、INTに対応するセルのポインタがS表をさすので、ここが終端となる。

る。S表のさされた部分を見ると、サブルーチン名及びARG表へのポインタがあるので、これらを使って、FORTRANのCALL文が生成できる。

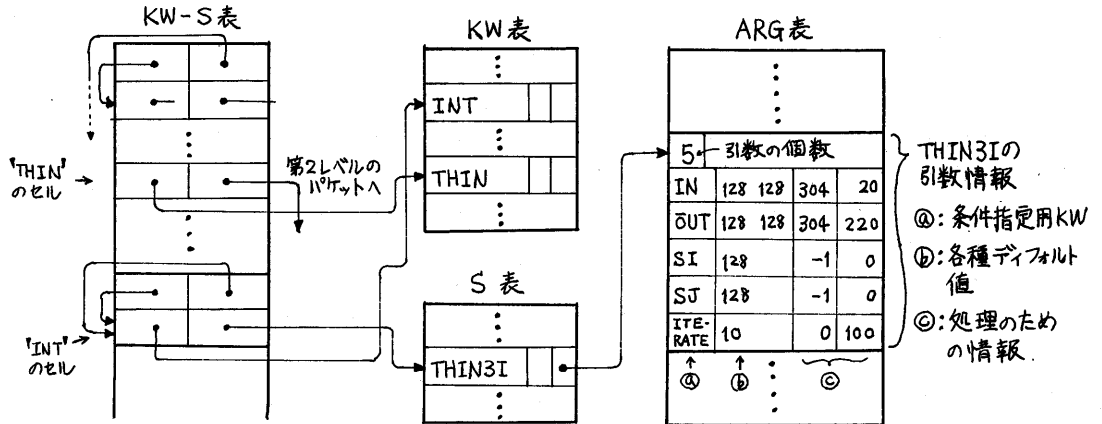


図4. 検索手順の例

4.2 データ管理部 これは、4.1のデータを管理するための部分である。図3のようなパケット方式ではデータの変更ができないので、ここでは、一般的なりスト表現を使っている。すなわち、1セル中に、自己のKW、第何のポインタ、子(長男)へのポインタを含むような表現方法である。このシステムを起動すると、データを上に述べた形式のリスト表現に変換して格納し、KW-S表への新しいリストの追加、あるリストの削除、KWの変更、KW系列の変更、サブルーチンの変更、デフォルトKWの変更、等を行うことができる。

4.3 プリプロセッサ部 これは、4.1のデータを使って、HILLSプログラムをFORTRANプログラムに変換する部分である。原則として、HILLSの1つの文ごとに、正当性の解析とFORTRAN文への変換を行っていく。ただし、宣言文については、宣言された変数をその都度登録しておき、最後にまとめてFORTRANの宣言文を生成するようにしている。この変換の一例を図5.に示す。

```

: THINNING, ITERATIVE, BINARY, INTEGER,
  IN=IA(128,128), OUT=IB(128,128), SI=128, SJ=128, ITERATE=10 ;
(a). HILLS実行文 (下線部は省略可. 連鎖型処理の場合は IN,OUTも省略可)

CALL THIN3I(IA,IB,128,128,10)
  
```

(b). 対応するFORTRAN文

図5. HILLS実行文の変換の一例

5. HILLSの特徴

4. のデータ及びプリプロセッサによって、言語HILLSに次に示すような特徴をもたせることができた。

(1). すぐれたアルゴリズムデータベース(A④)を基礎におくことにより、簡単な

KW 系列の指定で高度な画像処理機能を実現できる。

(2). デフォルト機能 HILLS プログラム中で、KW や変数宣言などを省略した場合、適切なKWの補足や変数の宣言等を自動的に行う機能である。

(3). HILLS文の自由度 HILLS 実行文中の第1 KW 以外のものは任意の順序で書くことができる。また、KW 自体の記述にも冗長性がある(例えば、THIN は THINNING と書いてもよい)。

(4). 連鎖型処理の自動結合 前段の処理の出力が次の段の処理の入力になるような処理系列を連鎖型処理という。この場合、この出力と入力との結合が可能なきには、プログラム上で陽に指定しなくても、自動的に配列をつかって、処理系列を結合することができる。

(5). サブルーチン名の直接指定 利用すべきサブルーチン名がわかっている場合、サブルーチン名で処理を指定することができる(表1.のサブルーチン直接指定文)。

(6). 画像演算の数式表現 画像間及び画像定数間の点演算⁽⁶⁾に対しては、KW 系列ではなくて、よりわかりやすい数式表現を許している(表1.の画像演算文)。

(7). FORTRAN文の混在 HILLS プログラム中でFORTRAN文を併用できる。これにより、HILLSに制御文を設ける手数を省いた。

(8). 汎用性、融通性 システムがすべてFORTRANで書かれているので、移植性、汎用性がある。また、IPADの変更や増強には、KW-S表等のデータ部の変更によって容易に対処でき、プリプロセッサ部は、データ部を変更することにより、異なった分野のADにも適用できる。

6. HILLSシステムの実現と実験

HILLSシステムのプリプロセッサ部とデータ管理部は、移植性を考慮してFORTRANで書かれていて、両部とも約1900ステップである。KW-S表、ARG表等のデータ部は、約49kBである。現在登録しているサブルーチンは、SLIPのものが180個、SPIDERのものが39個、その他1個の総計220個である。機能指定用KWは約170種類登録されている。

このHILLSシステムを使って、エッジ検出を行うHILLSプログラムを対応するFORTRANプログラムに変換するという実験を行った。このプログラムの手順を図6.に示す。

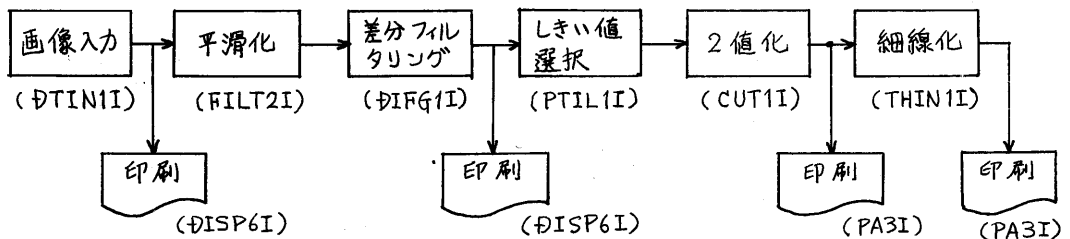


図6. 実験に用いた処理手順(エッジ検出) ()内は使用されるサブルーチン名

この手順を記述したHILLSプログラムが図7.(a)であり、これをHILLSシステムで変換して得られたFORTRANプログラムが図7.(b)である。処理時間は約15秒であ

る（使用した電子計算機はMELCOM COSMO 700-II）。ただし、変換を行う前にKW-S表、ARG表等を主記憶に読み込むようになっていて、上述の約75秒のうち約10秒はこのための時間である。

```

:: EDGE DETECTION ( 1981 9 18 ) ;
: TINT MT1(10), MT2(10), MT3(10), MT4(10) ;
  DATA MT1 / 17H** INPUT IMAGE **, 5*1H /
  DATA MT2 / 34H** AFTER PERFORMING DIF. FILTER **, 1H /
  DATA MT3 / 35H** AFTER PERFORMING THRESHOLDING **, 1H /
  DATA MT4 / 21H** RESULTING IMAGE **, 4*1H /
: INPUT
  OUT=IA(256,128), START-I=1, START-J=128, DATA=2 ;
  : OUTPUT, PICTURE, HALFTONE TITLE=MT1 ;
: FILTER, SMOOTH
  IN=IA, OUT=IB(256,128), REG-I=5, REG-J=5, GAIN=0.04 ;
: FILTER, DIFFERENCE, BASIC1
  OUT=IA, DI=0, DJ=3 ;
:: MAKING POINT, IA(I,J)<0, TO 0 ;
  DO 10 J=1,128
    DO 10 I=1,256
      IF( IA(I,J) .LT. 0 ) IA(I,J) = 0
10 CONTINUE
  : OUTPUT, PICTURE, HALFTONE TITLE=MT2 ;
: THRESHOLD, SET-TH, P-TILE IN=IA, RATE=0.75, TH=ICUT ;
: THRESHOLD, BINARIZE
  IN=IA, OUT=IB, TH=ICUT ;
  : OUTPUT, PICTURE, LITERAL, BINARY TITLE=MT3 ;
: THIN, SPECIFIC, YOKOII INOUT=IB ;
  : OUTPUT, PICTURE, LITERAL, BINARY IN=IB, TITLE=MT4 ;
: END ;

```

(a). HILLS プログラム

```

INTEGER MT1(10),MT2(10),MT3(10),MT4(10),IA(256,128),SYSV01,
#IB(256,128),ICUT,SYSV02(500),SYSV03
DATA MT1 / 17H** INPUT IMAGE **, 5*1H /
DATA MT2 / 34H** AFTER PERFORMING DIF. FILTER **, 1H /
DATA MT3 / 35H** AFTER PERFORMING THRESHOLDING **, 1H /
DATA MT4 / 21H** RESULTING IMAGE **, 4*1H /
CALL DTIN1I(IA,256,128,1,128,2,0,SYSV01)
CALL DISP6I(0,0,IA,256,128,1,256,1,128,7,MT1,1)
CALL FILT2I(IA,IB,256,128,5,5,0.04)
CALL DIFGLI(IB,IA,256,128,0,3)
DO 10 J=1,128
  DO 10 I=1,256
    IF( IA(I,J) .LT. 0 ) IA(I,J) = 0
10 CONTINUE
CALL DISP6I(0,0,IA,256,128,1,256,1,128,7,MT2,1)
CALL PTIL1I(IA,256,128,1,256,1,128,0,SYSV02,500,1.0,0.75,ICUT,
#SYSV03)
CALL CUT1I(IA,IB,256,128,ICUT,4HOVER)
CALL PA3I(0,0,IB,256,128,1H.,1H@,1,MT3,1)
CALL THIN1I(IB,256,128,4)
CALL PA3I(0,0,IB,256,128,1H.,1H@,1,MT4,1)
STOP
END

```

(b). (a.)に対応するFORTRAN プログラム

図7. 変換例（エッジ検出（図6.））

7. むすび

本文では、新たに開発した画像処理用言語HILLSとその処理システム（HILLSシステム）について述べた。

HILLSの使用経験はまだ極めて少ないが、直観的に理解しやすいKWによる記述、HILLSのきつ高度な機能（デフォルト機能等）、及び、基礎となるADの豊富な機能によって、処理手順の記述が比較的簡単になり、処理手順の把握も容易になると考えられる。このことから、画像処理の初心者やIPADにあまり詳しくない研究者が、HILLSを通して、IPADのすぐれた処理能力等を有効的に活用できるようになる、というこゝろが期待される。

今後の課題としては、(1).連鎖型処理、変数の多重使用、変数の型の自動変換などの機能の強化、(2).KWの画一化、(3).IPAD中の複数のサブルーチンを必要とするような処理系列を1つのHILLS文で記述、生成する機能、(4).会話型システムへの拡張、(5).実行の段階まで一貫して行えるようなシステムの開発、等が考えられる。これらの課題を解決することにより、KW系列や処理系列を、処理対象や目的に応じて、ある程度自動的に生成できるような知的なシステムへの拡張も可能になると考えられる。

<謝辞> システム作成に協力頂いた豊橋技術科学大学4年生の松下忠司君、及び、御討論頂いた名古屋大学工学部本多・福村研究室パターン認識グループの皆様へ感謝します。なお、本研究の一部は、文部省科研費、厚生省がん研究助成金（55-17）に依る。

<参考文献>

- (1). K. Preston, Jr. : "Image manipulative languages-a preliminary survey", in E. S. Gelsema and L. N. Kanal (eds.): Pattern Recognition in Practice, pp.5-20, North-Holland (1980)
- (2). E. Johnston : "The PAXII, picture processing system", in B. S. Lipkin and A. Rosenfeld (eds.) : Picture Processing And Psychopictorics, pp.427-512, Academic Press, N. Y. (1970)
- (3). S. Levialdi, M. Isoldi and G. Uccella : "Programming in PIXAL", Proc. the Workshop on Picture Data Description And Management, pp.74-79 (Aug. 1980)
- (4). 鳥脇, 福村 : "画像処理サブルーチンライブラリSLIPについて", 情報処理学会論文誌, 22, 4, pp. 353-359 (July 1981)
- (5). "SPIDER User's Manual", 電子技術総合研究所 (1980)
- (6). 横井, 鳥脇, 福村 : "標本化図形の演算系の代数的構造とその応用(I) - 定式化および基礎的諸演算について", 電子通信学会論文誌 60D, 6, pp. 411-418 (June 1977)