

BASIC 图象处理 程序 150 例

[日] 塩野 充 著

王东生 王熙法 庄镇泉 译

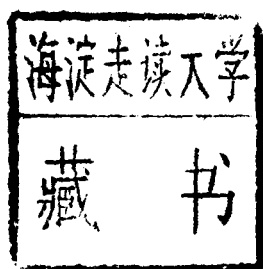
BASIC

中国科学技术大学出版社

TP312
SJE/1

BASIC 图象处理程序 150 例

[日] 森俊二 主编 塩野充 著
王东生 王煦法 庄镇泉 译



中国科学技术大学出版社

1992.12.196

PC-9800シリーズ

BASIC 画像処理プログラム150 選

C 塩 野 充 1988

昭和 63 年 6 月 25 日 第 1 版第 1 刷発行

監修者 森 俊二 著 者 塩野 充

BASIC 图象处理程序 150 例

[日] 塩 野 充 著

*

王东生 王煦法 庄镇泉 译

中国科学技术大学出版社出版

(安徽省合肥金寨路 96 号, 邮政编码 230026)

中国科学技术大学印刷厂印刷

安徽省新华书店发行

*

开本: 850×1168/32 印张: 13.25 字数 341 千

1992 年 2 月第 1 版 1992 年 2 月第 1 次印刷

印数: 1-3000 册

ISBN7-312-00302-8/TP·33

[皖]第 08 号 定价: 6.70 元

内 容 提 要

本书介绍了 150 个图象处理程序,并给出了源程序清单。所有程序都用 BASIC 语言写成,并附有简洁的注释。全书共 15 章,每章介绍图象处理的一个方面的内容及其相应的一组程序。其中第 1 至第 14 章包括图象处理中的灰度处理,特征提取,几何变换,数学变换,图象的分析、编码、解码和表示方法等方面的内容。第 15 章是与图象处理有直接关系的各种图象模式的产生方法。

本书系统地描述了各种图象处理方法和程序,特别适合于图象处理的初学者或非图象处理专业的有关人员。因此,本书可用作大专院校有关专业的数学参考书或实验教材,也可作为研究所,工矿企业等各行业有关人员的参考书或使用手册。

15265/02

译 者 的 话

在当前高度发达的信息时代,包括图象处理在内的视觉信息处理正越来越引起人们的重视。其中图象处理技术已日益普及,并在军事、航天、医学、工业生产、材料、办公室自动化等各个领域发挥着巨大的作用。

然而,目前已有的有关图象处理的专著和教科书,大多数都侧重于图象处理理论或硬件系统的介绍,很少涉及具体的图象处理程序。而市售的图象处理软件包,价格昂贵且通用性差,也不能作为图象处理的入门工具。因此,人们热切希望能有一本易懂而又实用的介绍微机图象处理程序设计的参考书,特别是对于初学者或非图象处理专业的人员,尤其如此。

日本塩野充先生所著《BASIC 图象处理程序 150 例》一书,是一本能满足上述要求的极有参考价值的好书。读者可以从中学习图象处理的基本理论以及相应的程序设计方法,也可以直接使用书中的程序上机运行,或者对书中给出的 150 个程序作进一步扩充和完善,组成适合自己需要的图象处理程序库。

原书中所有程序都是以 PC-9800 系列微机为基础的,我们已经将它移置到 IBM-PC 系列微机上,使用 5 $\frac{1}{4}$ 寸 1.2M 软磁盘 2 张。若读者需要使用,请参看本书附录。

原书中所有程序的注释都是日文的。为了便于读者理解原程序,我们把它们直接译成中文,并将其中重要的注释部分译成英文保留在磁盘中,以便读者既能对照阅读本书和磁盘中的源程序,又能直接运行磁盘中的程序。本书的其它阅读方法请参阅原作者写的“本书的阅读方法”。

译 者

1991 年 3 月

于合肥

主 编 的 话

图象识别技术不如 AI 中专家系统那么新颖,但其历史却不算短。借助于 19 世纪后半叶迅速发展起来的应用数学,图象处理技术取得了实质性的进展,其技术资源十分丰富。

当代计算机科学的进步令人瞩目,即使个人计算机也已具备了十多年前实验室中仅有的一台小型机的性能。比如,它能令人满意地进行十分费时的 Huffman 变换等。当然,这种计算取决于处理数据的维数,但使用具有实际意义的维数的数据是可行的。

根据这一情况,以电子综合研究所为中心研制了有关图象识别技术的子程序集 SPIDER,并被广泛应用于实际工作中。但是一般人员还不能方便地在个人机上使用该程序集,塩野充先生的这本关于图象处理的 BASIC 程序的著作包括了进行实际的图象处理研究所必需的各种基本子程序。该程序集程序简洁,注释适当,不仅大学师生,而且一般人员也可方便灵活地使用。因此,可望以此进一步促进图象处理研究的发展。

不仅如此,计算机的发展正在改变以图书为载体传播传统文化的作用,开拓了传播文化的新层次。人的大脑与古希腊时代相比并无差异,但用于思考的环境却发生了很大的变化。研究者之所以能进行更高层次的研究,归根到底是因为利用了这种称之为环境的基础。因此,对那些孜孜不倦地建立这些基础的前人们,我们将永世不忘,感谢不尽。

森 俊 二

1988 年 4 月

前 言

近年来,由于微处理器技术的飞速进步,个人计算机的性能也显著地提高。特别是最新 16 位个人机的性能几乎已等同于十多年前的大型机的性能。在这种形势下,个人计算机迅速普及,技术性计算除了涉及事务计算之外,还遍及社会中所有领域的各个角度,用途极为广泛,因而发展十分活跃。

就计算机的用途而言,即使大型机也是如此,最初都是进行技术计算和事务计算等符号层次的“数字计算”的机器。随着技术的进步,计算机不仅可处理枯燥无味的数字,而且可以处理文字、图形和图象等。因此,计算机的输出,从昔日那种仅仅罗列数字和文字的打印机时代,发展成具有在 CRT 显示器上进行彩色显示的软拷贝或是由彩色打印机输出硬拷贝等各种形式的输出手段。

因此,图象处理这一曾被认为只能用大型计算机才能完成的工作,现在用个人计算机也能完成得相当好,尤其是最新 16 位个人计算机的处理速度非常快,在实用的图象处理中它正日益得到充分的利用。图象处理具有极其广泛应用领域的处理技术,它不仅对电子信息学的技术人员或研究人员有用,而且在医学、气象、建筑、土木、物理、化学、地质艺术、服装艺术以及其它商业上的用途也是不言而喻的,甚至连历史和文学领域的研究人员也开始使用它。

图象处理目前如此普及的主要原因在于用个人计算机进行图象处理已成为现实。若仅限于大型计算机进行图象处理,则很难想象会有如此程度的普及。图象处理和一般的数值计算不一样,初学者想要编写图象处理程序也并非易事。尽管图象处理的入门书已大量问世,但大多数是学术性内容,而且是作为大专院校的教科书。当非专业人员打算在个人计算机上编制图象处理程序时,这些

书不能立即派上用处。即使这些书不是描述学术性内容,也大多数是面向行家的侧重于硬件描述的内容。因此,就象很多人即使不理解计算机工作原理也会使用个人计算机一样,非专业人员在未明确理解图象处理算法的情况下也能很快地运算图象处理技术,但这方面的入门书却很少。而且现成的图象处理软件包大多数价格很高,谁也不会随便地购买。由于它们是软件包,用户往往很难购买到自己所要求的程序,而不得不把不需要的程序也一齐买进。

本书为了适应初学者的需要,并避免发生上述情况,给出了用 BASIC 编写的 150 个基本图象处理程序的原始程序清单,并附简单的说明。读者只要由键盘照原样打入自己所要的程序就可立即进行图象处理。特别值得一提的是,这样做并不需要理解图象处理程序的设计思想或算法。当然,有理解地使用则更好。为此,在程序清单中有许多日文注释(本书已译成中文——译者)以说明处理步骤。

本书所给出的图象处理程序全部以 PC-9800 系列微机为对象。由于大多数程序都是用标准 BASIC 写的,所以对微机机种的依赖性较小,大部分程序可不加修改地在其它微机上使用。另外,考虑到有的读者想备齐本书中所有 150 个程序,故还备有软盘另外出售。

本书所收录的 BASIC 图象处理程序构成了名为 WIPER(Wide usage Image Processing programs for Education and Research)的图象处理程序库。所谓 WIPER,是意味着图象处理的目的之一就象汽车上的刮水刷一样“让人们易于觉察和理解”。

本书所登载的 150 个程序全部是作者精心编写而成。但因才疏学浅,错误难免,若蒙读者发现指出,将不胜感谢。若本书能对需要微机图象处理的读者有用,本人将深感荣幸。

最后,对在百忙中对拙著进行主编的森俊二先生深表谢意,并深深感谢在本书出版中给予不少帮助的奥姆公司。

塩野充

本书的阅读方法

本书从第 1 章“二值化”到第 15 章“模式产生”共由 15 章组成。

每章表示图象处理的一个重要方面。各章中的每一节对应一个图象处理程序。全书 150 节,共登载 150 个图象处理程序。每个程序由一个主程序和一个子程序组成。与 FORTRAN 不同,BASIC 的子程序功能较弱,在结构上不能作为一个独立的程序加以处理。因此,不得不以附属于主程序的形式来表示子程序。在各程序中的部分子程序的形式如:

```
'WIPER #ab—cd=====00...0   * □□□...□=====
* △△△
.....
.....
RETURN
'====="□□□...□"=====
```

即用 WIPER 标记的行是子程序的开始,程序的最后一行是子程序的结束。WIPER 后面的 #ab—cd 是子程序的标识号码,ab 是章号(01~15),cd 是节号(01~)。00...0 是该子程序的日文标题(已译成中文——译者),其后的 □□□...□ 是该程序(也包括主程序)在软盘中的文件名。最后一行也以 □□□...□ 的形式出现。第 2 行的 △△...△ 是子程序名(子程序入口标号名)。通常,□□□...□ 与 △△...△ 是一样的,但在部分使用图形画面的程序中,文件名则以小写字母相区别,即 □□□...□ 用小写字母,△△...△ 用大写字母(因标号名只用大写字母)。

文件名的规则可归纳为:

大写字母的文件名……仅使用文本画面的程序；

小写字母的文件名……文本画面和图形画面都使用的程序。

程序全部用 ASCII 码存贮，若不用 ASCII 码存贮，则不能与其它子程序合并。用 ASCII 码存贮文件的操作为：

SAVE“文件名”，A✓

即文件名后面只要用一个逗号和 A 就可以了。

实际使用本书中的程序时，子程序部分可照原样采用，主程序部分可根据自己的目的和用途作些修改。这里所给出的主程序，始终只是其后面子程序使用方法的一个例子。

本书中各节开头都有非常简单的程序说明。其中所使用的图象名 $f(i,j)$, $g(i,j)$, $r(i,j)$, $a(i,j)$, $b(i,j)$ 等，在程序中则分别成为大写的图象数组名 $F\%(I,J)$, $G\%(I,J)$, $R(I,J)$, $A(I,J)$, $B(I,J)$ 。其它参数在程序中的名称，每次都有说明。

图象数组是二维数组，因此，其行和列的定义有二种方法。第一种方法与数学中矩阵一样， $f(i,j)$ 中的 i 定义为行（从上往下第几行的单位）， j 定义为列（从左往右第几列的单位）。这种方法称为 M 型（MATRIX 型）。第二种方法与此相反， $f(i,j)$ 中的 i 定义为列， j 定义为行，这种方法称为 F 型（FORTRAN 型）。归纳一下则有：

M 型 $\rightarrow f(i \text{ 行}, j \text{ 列})$ ；

F 型 $\rightarrow f(i \text{ 列}, j \text{ 行})$ 。

本书中考虑到与数学中矩阵的对应，所有程序中都采用 M 型。

WIPER 的最大特点就是在程序中使用了很多为说明算法而用的日文注释行（已译成中文——译者）。因此，即使无特别说明，只要阅读程序清单，就可大致理解其内容。但是，程序中的注释行和各节的简单说明，在许多情况下还不能充分解释程序，此时都标注了参考文献，请读者适当参阅。

实际使用 WIPER 时，有几点值得注意。一是与 FORTRAN 的子程序不一样，使用 BASIC 时，没有“参数”的概念，因此，主程序中的变量名也是子程序中的变量名，它们是共用的（名称也一样）。

不象 FORTRAN 子程序那样,除用作参数传递信息外,主程序中与子程序中的变量即使名称相同,也毫无关系。因此,在把几个子程序合并用来进行复杂处理时,要特别注意这一点。

还有一点需要注意。在使用 NEC PC-9800 系列的 N88 日文 BASIC(86)时,不论是 BASIC 版还是 MS-DOS 版,所说明的一个数组的大小(数组元素数)都是有限制的,即

整数型数组……数组元素数在 32767 个以下(若取正方形画面,则最大取 181×181);

实数型数组……数组元素数在 16383 个以下(若取正方形画面,则最大取 127×127);

双精度实数型数组……数组元素数在 8191 个以下(若取正方形画面,则最大取 90×90)。

上述要求是 N88 日文 BASIC(86)中 DIM 语句的语法规则,与微机中 RAM 容量设置无关。因此,当取比上述要求更大的画面时,需要把画面分割成几个来进行处理,此时在处理分割开的部分画面的连接处时,需格外注意花点工夫。

本书中的 BASIC 程序,为了尽可能与 FORTRAN 保持对应关系(换句话说就是为使 FORTRAN 用户也容易适应),所有程序都有说明语句 DEFINT I-N,以与 FORTRAN 的 I-N 规则对应,即以字母 I,J,K,L,M,N 开头的变量名,全部被默认为整数型说明,而其它变量名则全部取实数型。但是在变量名后面附加%的变量是整数型,附加!的变量是实数型,附加#的变量是双精度型,附加\$的变量是字符型。这些都是明确的类型说明。在这种情况下,与 I-N 规则无关。

另外,由于数组说明全部用 OPTION BASE 1,使数组下标的下限为 1,这与 FORTRAN 一致(通常,BASIC 的数组下标下限为 0)。

程序中出现的数组,在所有子程序的开头处都用注释行加以说明。输入数组适合于在进入子程序之前存放输入数据,或者是用作比较大的作业区域。输出数组用作放置子程序的处理结果。输

入数组和输出数组必须在主程序中通过 DIM 语句进行数组说明。内部数组是仅在子程序中使用的比较小的作业区域,并在子程序中进行数组说明,因而不必在主程序中进行数组说明。诸如输入参数,输出参数等也全部用注释行加以说明。

目 次

译者的话	(I)
主编的话	(III)
前言	(V)
本书的阅读方法	(VI)

1 二值化

1.1 二值化(非零元素取1法)“BINAF0”	(1)
1.2 二值化(固定阈值法)“BINAF1”	(2)
1.3 二值化(双固定阈值法)“BINAF2”	(3)
1.4 二值化(各自象素阈值法)“BINAF3”	(5)
1.5 判断分析二值化法“HANBET”	(6)
1.6 P块法“PTILE”	()

2 灰度变换

2.1 剪裁(半阈值法)“CLIPPI”	(12)
2.2 负象生成“NEGA”	(13)
2.3 负数灰度值的正数化“NPLUS”	(14)
2.4 把实数灰度值整数化“NLEVEL”	(16)
2.5 灰度值实数化“NODO01”	(18)
2.6 锯齿波变换“NOKOGI”	(19)
2.7 灰度级变换“GRAYTR”	(21)
2.8 灰度对数变换“GRAYLG”	(22)
2.9 灰度等高线“TOUKOU”	(24)

2.10	灰度分布的规范化“NBUNPU”	(27)
2.11	直方图均衡“HISTEQ”	(29)
2.12	灰度直方图计算“NHIST1”	(32)
2.13	累加灰度直方图计算“NHIST2”	(33)
2.14	规范化灰度直方图表示“hist01”	(35)
2.15	整数灰度直方图表示“histst”	(37)

3 噪声消除

3.1	二值图象的黑白点噪声滤波“GOMA2C”	(40)
3.2	消除孤立黑象素点“KORITU”	(41)
3.3	3×3 均值滤波器“FILT33”	(43)
3.4	N×N 均值滤波器“FILTNN”	(46)
3.5	有选择的局部平均化“SELAVR”	(47)
3.6	N×N 中值滤波器“MEDIAN”	(51)
3.7	十字型中值滤波器“CMEDIA”	(54)
3.8	N×N 最频值滤波器“MODFIL”	(56)

4 微分运算

4.1	纵横方向微分运算“BIBUN1”	(59)
4.2	双方向一次微分运算“BIBUN2”	(61)
4.3	二次微分(1)“NIJIB1”	(62)
4.4	二次微分(2)“NIJIB2”	(64)
4.5	二次微分(3)“NIJIB3”	(65)
4.6	拉普拉斯算子运算“LAPLAC”	(68)
4.7	Kirsch 边缘检出“KIRSCH”	(70)
4.8	Prewitt 微分运算“PREWIT”	(73)
4.9	Sobel 微分运算“SOBEL”	(76)
4.10	Robert 微分运算“ROBERT”	(78)
4.11	Robinson 边缘检出“ROBINS”	(80)

4.12	Frei&chen 边缘检出“FRCHEN”	(83)
5 投影量计算		
5.1	垂直/水平投影量计算“TOUEI1”	(87)
5.2	对角/反对角方向投影量计算“TOUEI2”	(88)
5.3	垂直/水平方向投影表示“projel”	(91)
5.4	对角/反对角方向投影表示“proje2”	(93)
6 黑区域处理		
6.1	二值图形的区域标记“RLABEL”	(98)
6.2	二值图形的小区域消除“SMLDEL”	(101)
6.3	二值图形封闭区域大小的计算“RGNSIZ”	(104)
6.4	二值图形缩退(1)“SHRIN1”	(108)
6.5	二值图形缩退(2)“SHRIN2”	(111)
6.6	黑区域轮廓线抽取“RINKAK”	(114)
6.7	二值图形的边界线跟踪“KYOKAI”	(115)
7 特征抽取		
7.1	基于线图交叉数的特征点表示“TOKUCH”	(121)
7.2	曲折点检出“KADO”	(123)
7.3	二值图形的外接框检出“GAISSET”	(127)
7.4	二值图形的空穴数检出“HOLESU”	(128)
7.5	图形的欧拉数计算“EULER”	(131)
7.6	阈值法方向编码“HCODET”	(134)
7.7	最长方向法方向编码“HCODEL”	(140)
7.8	二值图形白区域的封闭率计算“TOJIRI”	(146)
7.9	图象的力矩计算“MOMENT”	(152)
7.10	图象的重心矩计算“CMOMEN”	(153)
7.11	图象的重心位置计算“JUSIN”	(155)

7.12	惯性主轴计算“SHUJIK”	(156)
7.13	二值图形封闭区域的复杂度计算“FUKUZA”	(158)
7.14	闭曲线的偏角微分函数计算“HENKAK”	(162)

8 图象之间的运算

8.1	图象的象素之间的算术运算“SANJUT”	(166)
8.2	二值图形的象素之间的逻辑运算“BLOGIC”	(168)
8.3	二值图形的海明距离计算“HMDIST”	(171)
8.4	图象的相似度计算“RUIJID”	(172)
8.5	周期卷积计算“CONV01”	(174)
8.6	非周期卷积计算“CONV02”	(176)
8.7	图象与常数四则运算“TEISUU”	(178)
8.8	图象数组的常数代入“DAINYU”	(180)
8.9	图象数组的部分复制“TENSHA”	(181)
8.10	图象矩阵转置“TENCHI”	(183)
8.11	图象的纵/横方向截面抽取“DANMEN”	(184)
8.12	图象数组的数据类型变换“TYPECV”	(186)

9 几何变换

9.1	图象放大(整数倍)“SZKAKU”	(188)
9.2	图象缩小(整数倍)“SZSHUK”	(190)
9.3	图象大小的实数倍变换(1)“SZCNV1”	(191)
9.4	图象大小的实数倍变换(2)“SZCNV2”	(195)
9.5	图象大小的实数倍变换(3)“SZCNV3”	(198)
9.6	图象的放大缩小“ZOOM”	(202)
9.7	图象的平移“HEIKOU”	(205)
9.8	图象的90度旋转“R90DEG”	(206)
9.9	最邻近法图象旋转“ROTAT1”	(209)
9.10	线性插值法图象旋转“ROTAT2”	(211)

9.11	3次插值法图象旋转“ROTAT3”	(214)
9.12	图象的倾斜变换“KEISHA”	(218)
9.13	图象的鼓形变形修正“TARUGA”	(220)
9.14	二值图形膨胀/收缩“BOUCHO”	(222)
9.15	二值图形的粗化处理(1)“FUTOM1”	(225)
9.16	二值图形的粗化处理(2)“FUTOM2”	(227)
 10 线条图形处理		
10.1	Hilditch 细化“HILDIT”	(229)
10.2	Deutch 细化“DEUTCH”	(233)
10.3	Rosenfeld 的 8 连接细化“ROSEN8”	(237)
10.4	Rosenfeld 的 4 连接细化“ROSEN4”	(241)
10.5	基于连接数的灰度图象的细化“GRTHIN”	(245)
10.6	线条图形的单纯化“SENBUN”	(248)
10.7	线条图形中的短枝消去“EDAJOK”	(250)
10.8	线条图形短缺处理“SHTGAP”	(253)
10.9	样条函数计算“SPLINE”	(258)
 11 数学变换		
11.1	二维快速傅里叶变换“XYFFT”	(262)
11.2	二维快速傅里叶反变换“XYIFFT”	(267)
11.3	二维 Walsh 变换“XYWALS”	(272)
 12 纹理分析		
12.1	空间自相关函数计算“ACORRE”	(278)
12.2	关于灰度参数的计算“NPARAM”	(281)
12.3	Hough 变换(1)“HOUGH1”	(284)
12.4	Hough 变换(2)“HOUGH2”	(286)
12.5	灰度共生矩阵计算“KYOUKI”	(289)