

Delphi

数字图像 处理及高级应用

刘 骏 主 编



科学出版社

www.sciencep.com

内 容 简 介

本书详细介绍了利用 Delphi 进行图像处理的技术,常用的图像格式,以及 Delphi 图像处理的常用方法 Scanline。本书共 8 章,内容包括图像的基本概念、图像的点运算、图像的几何变换、图像的颜色系统、图像的增强、图像代数与分隔、图像的特效、图像处理综合实例,前面 7 章比较详细地介绍了图像处理的内容,同时提供了非常详细的程序代码,第 8 章是编者自己创作或者平时收集的一些经典的例子。本书提供了丰富的源代码,并提供了详细的注释,为读者的学习提供方便。

本书可以作为高等院校有关老师的教学参考书或高年级学生及研究生的自学用书,也可作为软件开发人员的参考书。

图书在版编目(CIP)数据

Delphi 数字图像处理及高级应用/刘骏主编. —北京:科学出版社, 2003

ISBN 7-03-012163-5

I.D... II.刘... III.软件工具—程序设计 IV.TP311.56

中国版本图书馆 CIP 数据核字(2003)第 080353 号

策划编辑:李 娜/责任编辑:韩 洁

责任印制:吕春珉/封面设计:飞天创意

科学出版社 出版

北京东黄城根北街16号

邮政编码:100717

<http://www.sciencep.com>

双青印刷厂 印刷

科学出版社发行 各地新华书店经销

*

2003 年 9 月第 一 版 开本:787×1092 1/16

2003 年 9 月第一次印刷 印张:22 3/4

印数:1—4 000 字数:522 000

定价:40.00 元(含光盘)

(如有印装质量问题,我社负责调换〈环伟〉)

前 言

图像处理是一门理论与实践紧密结合的学科,一直以来受到人们的重视。Visual C++ (以下简称 VC)作为图像处理的常用工具,虽然功能强大,但由于其落后的 MFC 架构,使得很多初级程序员望而却步;Delphi 作为 Windows 环境下面面向对象快速开发工具 (RAD),以其高速的开发效率深得程序员的喜爱,它的编译器的编译速度是 VC 的几倍,并且具有良好的封装性。

Delphi 除了在数据库方面有很强的优势以外,在图像处理上也有着非常强大的功能,Delphi 进行图像处理的速度并不亚于 VC,同时大大提高了开发效率,只需要少量代码就能实现很多复杂的功能。但是目前专门介绍 Delphi 数字图像处理的书还很少,以前的 Delphi 书籍在讲到图像这一节基本上都是一笔带过,没有做系统介绍,本书的写作正是基于这一点,编者把平时工作中遇到的和平时收集的一系列图像处理方面的问题做了一个系统的整理,而成此书。随着图像、多媒体技术的发展以及人们对开发效率的追求,使用 Delphi 来进行图像处理无疑是最好的选择。

同其他数字图像处理图书相比,本书具有如下显著特点:

- 1) 详细介绍了各种常用、经典图像处理理论。
- 2) 详细介绍小波压缩算法以及实现原理,同时介绍了 JPEG2000 图像处理技术。
- 3) 完善的代码支持,本书光盘提供了各实例的源代码;为了节省篇幅,书中只给出了核心的代码和详细的注释。
- 4) 本书介绍了近 30 个各种图像应用实例,包括实际工程应用的具体实例及其源代码,读者可以直接引用。
- 5) 语言简洁易懂,读者很容易明白。

本书包括 8 章,分别为:

第 1 章为图像的基本概念,详细介绍了最新的 JPEG2000 图像格式以及各种其他常用的文件格式,同时介绍各种基础理论。

第 2 章为图像的点运算,对图像的灰度、对比度、灰度直方图、着色、反色等效果做了详细的介绍,并对图像的反走样做了全面的说明,对图像的 Gamma 校正和特大位图的创建方法也有深入说明。

第 3 章是图像的几何变换,除了介绍常见的几何变换以外还增加了图像的扭曲,波浪和远视图效果。

第 4 章为图像的颜色系统,不仅介绍了颜色空间的基本概念,还给出了比较重要的颜色空间的转换原理和代码,如 RGB 空间到 CMYK 空间的相互转换,RGB 空间和 HSL 空间之间的相互转换等。

第 5 章是图像的增强,介绍了灰度直方图的拉伸、图像的伪彩色增强等常见处理的原理。

第 6 章是图像的代数与分割,主要介绍了数学形态学在图像处理中的应用,以及经

典的边缘检测方法。

第 7 章为图像特效，图像的特效包括了一些经典的动态效果和静态效果。

第 8 章是应用篇，给出一些非常好的实用的例子。本书从理论到实践（程序实现）介绍的都非常详细，相信 Delphi 的初中级程序员很快就会掌握。

本书提供了丰富的源代码，并提供了详细的解释，为读者的学习提供方便。

本书可以作为高等院校有关老师的教学参考书或高年级学生或者研究生的自学用书，也可作为软件开发人员的参考书。

本书由刘骏主编，参与写作的人员有王华、汪洁、刘海英、白露、王占全、庞演、顾强、李平、孙阳、宁海洋、吴阳、张曙等。同时还要感谢一些网友，如活跃在大富翁论坛上的 huazai 网友、卷起千堆雪 tyn 网友，以及其他的一些不知名的热心的网友的帮助。

由于时间仓促和作者的水平有限，书中错误和不妥之处在所难免，敬请读者批评指正。如果有任何问题，可以登录 www.2wintech.net 询问，作者会在一周之内做出答复。

作 者

2003 年 6 月

目 录

第 1 章 图像的基础知识	1
1.1 图像的基本概念	1
1.1.1 图像的应用	1
1.1.2 位映像图像和向量图像	2
1.1.3 图像处理中的分辨率	5
1.2 三基色原理和图像的输入	7
1.2.1 可见光谱与光度学参量	7
1.2.2 三基色原理	8
1.2.3 视觉生理和心理规律	8
1.3 图像的几种常见的格式	10
1.3.1 BMP 格式	11
1.3.2 GIF 格式	14
1.3.3 JPEG 格式	17
1.3.4 PCX 格式	18
1.3.5 TIF 格式	18
1.3.6 JPEG2000 格式	18
1.4 图像格式转换器实例	22
1.4.1 程序功能	22
1.4.2 程序实现	22
1.5 图像浏览器实例	28
1.5.1 程序功能	28
1.5.2 程序实现	28
1.6 Delphi 图像处理中 Scanline 的用法	34
1.6.1 pf8bit 的位图	35
1.6.2 pf24bit 的位图	35
1.6.3 pf32bit 的位图	36
1.6.4 pf8bit 向 pf24bit 转换	37
小结	39
第 2 章 图像的点运算	40
2.1 图像灰度处理	40
2.1.1 实现原理	40
2.1.2 程序实现	40
2.2 图像的灰度直方图	43
2.2.1 实现原理	43
2.2.2 程序实现	44

2.3	图像的二值化	46
2.3.1	实现原理	46
2.3.2	程序实现	48
2.4	图像亮度处理	49
2.4.1	实现原理	49
2.4.2	程序实现	49
2.5	图像对比度处理	51
2.5.1	实现原理	51
2.5.2	程序实现	51
2.6	饱和度调节	53
2.6.1	实现原理	53
2.6.2	程序实现	53
2.7	图像着色	54
2.7.1	实现原理	54
2.7.2	程序实现	55
2.8	图像反色	56
2.8.1	实现原理	56
2.8.2	程序实现	58
2.9	图像曝光	63
2.9.1	实现原理	63
2.9.2	程序实现	63
2.10	Gamma 校正	64
2.10.1	实现原理	64
2.10.2	程序实现	65
2.11	迷人的万花筒	66
2.11.1	实现原理	66
2.11.2	程序实现	66
2.12	位图的反走样	68
2.12.1	实现原理	68
2.12.2	程序实现	69
2.13	位图的与、或操作	73
2.13.1	实现原理	73
2.13.2	程序实现	74
2.14	创建大型位图以及统计位图颜色	76
2.14.1	实现原理	76
2.14.2	程序实现	77
2.15	位图的噪声调节	79
2.15.1	实现原理	79
2.15.2	程序实现	80

小结	81
第3章 图像的几何变换	82
3.1 图像的平移	82
3.1.1 实现原理	82
3.1.2 程序实现	82
3.2 图像的缩放	84
3.2.1 实现原理	84
3.2.2 程序实现	84
3.3 图像的旋转	87
3.3.1 90度旋转	87
3.3.2 任意角旋转	91
3.4 图像的镜像	95
3.4.1 实现原理	95
3.4.2 程序实现	95
3.5 图像扭曲	99
3.5.1 实现原理	99
3.5.2 程序实现	99
3.6 图像的波浪效果	103
3.6.1 实现原理	103
3.6.2 程序实现	103
3.7 远视图	105
3.7.1 实现原理	105
3.7.2 程序实现	105
3.8 裁剪和合并	108
3.8.1 实现原理	108
3.8.2 程序实现	108
小结	109
第4章 图像的颜色系统	110
4.1 颜色的基本概念	110
4.2 颜色空间简介	111
4.2.1 RGB 颜色空间	111
4.2.2 YIQ 颜色空间	111
4.2.3 YUV 颜色模型	112
4.2.4 HSV 颜色模型	113
4.2.5 CMYK 模型	114
4.2.6 CIE-XYZ 颜色空间	115
4.2.7 Lab 颜色空间	116
4.3 颜色空间的转换	117
4.3.1 RGB 颜色空间和 HSL 颜色空间相互转换	117

4.3.2	RGB 颜色空间和 CMYK 颜色空间的相互转换.....	120
4.3.3	RGB 和 HSV 颜色空间互换.....	121
4.4	亮度/饱和度调整.....	123
4.4.1	实现原理.....	123
4.4.2	程序实现.....	123
4.5	通道与模式.....	126
4.5.1	实现原理.....	126
4.5.2	程序实现.....	126
4.6	RGB 颜色调整.....	127
4.6.1	实现原理.....	127
4.6.2	程序实现.....	127
4.7	特殊色彩的实现.....	129
4.7.1	实现原理.....	129
4.7.2	程序实现.....	129
4.8	颜色量化与减色.....	154
4.8.1	实现原理.....	154
4.8.2	程序实现.....	154
4.9	颜色混合.....	160
4.9.1	实现原理.....	160
4.9.2	程序实现.....	161
小结		163
第 5 章	图像的增强.....	164
5.1	图像增强概述.....	164
5.2	灰度线性变换.....	164
5.2.1	实现原理.....	164
5.2.2	程序实现.....	165
5.3	灰度非线性变换.....	166
5.3.1	实现原理.....	166
5.3.2	程序实现.....	167
5.4	灰度直方图拉伸.....	169
5.4.1	实现原理.....	169
5.4.2	程序实现.....	170
5.5	图像锐化与图像平滑.....	172
5.5.1	图像的锐化.....	172
5.5.2	图像平滑.....	175
5.6	伪彩色增强.....	178
5.6.1	实现原理.....	178
5.6.2	程序实现.....	178
5.7	中值滤波.....	180

5.7.1 实现原理.....	180
5.7.2 程序实现.....	180
小结.....	183
第6章 图像代数与图像分割.....	184
6.1 图像的腐蚀.....	184
6.1.1 实现原理.....	184
6.1.2 程序实现.....	188
6.2 图像的膨胀.....	190
6.2.1 实现原理.....	190
6.2.2 程序实现.....	191
6.3 图像的结构开和结构闭.....	192
6.3.1 实现原理.....	192
6.3.2 程序实现.....	195
6.4 图像的细化.....	196
6.4.1 实现原理.....	196
6.4.2 程序实现.....	197
6.5 图像的边缘检测.....	200
6.5.1 实现原理.....	200
6.5.2 程序实现.....	203
6.6 图像的 Hough 变换.....	207
6.6.1 实现原理.....	207
6.6.2 程序实现.....	208
6.7 图像的轮廓提取.....	212
6.7.1 实现原理.....	212
6.7.2 程序实现.....	213
6.8 图像的识别和模板匹配.....	214
6.8.1 投影法.....	215
6.8.2 差影法.....	217
6.8.3 模板匹配.....	221
小结.....	222
第7章 图像的特效处理.....	223
7.1 图像的滑入和卷帘显示效果.....	223
7.1.1 图像的卷帘显示效果.....	223
7.1.2 图像的滑入效果.....	226
7.2 图像的淡入淡出效果.....	228
7.3 扩散效果.....	230
7.3.1 实现原理.....	230
7.3.2 程序实现.....	230
7.4 百叶窗效果和马赛克效果.....	233

7.4.1	百叶窗效果.....	233
7.4.2	马赛克效果.....	234
7.5	交错效果.....	236
7.5.1	垂直交错效果.....	236
7.5.2	水平交错效果.....	237
7.6	浮雕效果.....	239
7.6.1	灰色浮雕.....	239
7.6.2	彩色浮雕.....	240
7.7	图像的中心渐出和渐入效果.....	242
7.7.1	图像的中心渐出效果.....	242
7.7.2	图像的中心渐入效果.....	243
7.8	图像的雨滴效果和积木效果.....	244
7.8.1	图像的雨滴效果.....	244
7.8.2	图像的积木效果.....	245
	小结.....	246
第 8 章	综合实例.....	247
8.1	利用 Delphi 实现桌面更换.....	247
8.1.1	实现原理.....	247
8.1.2	程序实现.....	247
8.2	图片文件的加密解密.....	248
8.2.1	位图文件的加密解密.....	248
8.2.2	JPG 文件加密解密.....	252
8.3	自定义光标的实现.....	256
8.4	基于 Delphi 的图像漫游.....	257
8.4.1	实现原理.....	257
8.4.2	程序实现.....	257
8.5	用 Delphi 实现屏幕图像捕捉.....	259
8.6	图片存取到流以及从流中复原.....	260
8.6.1	实现原理.....	260
8.6.2	程序实现.....	260
8.7	Delphi 图像处理在纺织检测中的应用.....	262
8.7.1	Hough 变换进行边缘检测.....	263
8.7.2	纱线参数测量.....	264
8.7.3	织物表面粗糙度比较.....	264
8.7.4	织物疵点检测.....	265
8.8	Photoshop 中流动蚂蚁线的实现.....	266
8.8.1	矩形流动蚂蚁线.....	266
8.8.2	椭圆流动蚂蚁线.....	267
8.8.3	任意位置流动蚂蚁线.....	269

8.9 用 Delphi 读取 JPEG 文件的缩览图.....	270
8.9.1 实现原理.....	270
8.9.2 程序实现.....	271
8.10 Delphi 数据压缩/解压缩处理.....	272
8.10.1 实现原理.....	272
8.10.2 程序实现.....	273
8.11 特大位图的快速显示.....	274
8.11.1 实现原理.....	274
8.11.2 程序实现.....	275
8.12 Photoshop 中的喷枪实现.....	277
8.12.1 实现原理.....	277
8.12.2 程序实现.....	278
8.13 颜色填充.....	279
8.13.1 种子填充.....	279
8.13.2 路径填充.....	285
8.14 位图与组件.....	286
8.14.1 Combobox 中添加位图.....	286
8.14.2 ListBox 中显示位图.....	287
8.14.3 RichEdit 中内容显示为位图.....	288
8.15 颜色拾取器.....	290
8.15.1 实现原理.....	290
8.15.2 程序实现.....	290
8.16 位图的打印.....	292
8.16.1 实现原理.....	292
8.16.2 程序实现.....	292
8.17 Delphi 图像处理在交通中的应用——车牌识别.....	294
8.18 位图文件信息写到文本文件以及恢复.....	295
8.19 放大镜.....	298
8.19.1 实现原理.....	298
8.19.2 程序实现.....	299
8.20 调色板创建及应用.....	304
8.20.1 创建调色板.....	304
8.20.2 调色板应用.....	305
8.21 图像的局域网传输.....	307
8.21.1 实现原理.....	307
8.21.2 程序实现.....	308
8.22 图像纵横比率最佳调节.....	318
8.22.1 实现原理.....	318
8.22.2 程序实现.....	320

8.23	JPEG 格式图片错误信息显示.....	322
8.24	JPG 图片存取到数据库	329
8.24.1	实现原理.....	329
8.24.2	程序实现.....	330
8.25	基于小波变换的 JPEG2000 压缩实现	331
8.25.1	实现原理.....	331
8.25.2	程序实现.....	333
8.26	傅里叶变换.....	345
8.26.1	实现原理.....	345
8.26.2	程序实现.....	348
参考文献		351

第1章 图像的基础知识

本章给读者阐述了一些图形图像的基本概念,介绍一些图像的基本格式和图像的获取方法,并通过两个实例给大家展示了如何用 Delphi 来进行图像处理的编程。

1.1 图像的基本概念

1.1.1 图像的应用

传统的电脑只能处理文字、数字,最多是简单的图形。近年来,随着电脑硬件技术的飞速发展和更新,使得计算机处理图形图像的能力大大增强。以前要用大型图形工作站来运行的图形应用软件,或是特殊文件格式的生成及对图形所做的各种复杂的处理和转换,如今,很普通的家用电脑就完全可以胜任,使用 Photoshop、CorelDraw、3DS MAX 或是别的软件就可做出精美的图片或是逼真的三维物体图。

在当今信息社会,以多媒体为代表的信息技术和信息产业的发展和应用对人类社会产生的影响和作用愈来愈明显,愈来愈重要。多媒体的发展和应用,极大地推动了诸工业的相互渗透和飞速发展,逐步改变了整个人类社会的工作结构和生活方式。

所谓多媒体,即多种信息媒介,通常包括以下几种:文本、图形、影像、声音、视频、动画。可以看出,多媒体的应用在很大程度上依赖于丰富多彩的图形和图像,也就是说,图形图像技术的飞速发展也是必然趋势,掌握图形图像处理技术对现代人来说也是很必要的。

计算机图形学是研究用计算机生成、处理和显示图形的一门科学。为了生成图形,首先要有原始数据或数学模型(如工程人员构思的草图、地形航测数据、飞机模型数据、总体方案模型等),这些数字化的输入信息经过计算机处理后变成图形输出。图形从原始数据生成图像数据经过了一系列变换过程,每个变换过程都可能产生不同于输入数据的输出数据,这些数据需要按一定的结构进行组织,形成一系列描述图形数据的文件,这类文件通常被称为图形文件(也称为图形图像文件),而图像文件是描述图像数据的文件,它是图形文件的一种特例。

在图形生成过程中有多种类型的数据,如模型数据、场景数据和图像数据等,因此,图形文件所描述的图形层次就不一样,这也是产生多种图形文件的一个重要原因。另一方面,在同一个描述层上,由于每种图形软件包使用自己的格式保存图形数据,随着图形应用软件包的不断增多,图形文件的格式也会越来越多,虽然国际标准化组织(ISO)为解决图形信息的共享问题,建立了一系列图形文件标准(如 CGM),但是这些标准较难得到广大用户和厂商的支持,从而形成了目前这种多种图形文件共存的局面。

图形文件有以下特点:①数据量大。由于现在数据获取手段日趋先进,可以得到的

数据越来越复杂,数据量也越来越大。②**结构性强**。数据在本质上分为数字化的和模拟的两种。模拟信息可以转换为数字信息。数字系统中的最基本单位是位(Bit),其他结构单位都以位为基础。在较低层次上可以是“构造块”(如浮点数、整数和字符);在较高层次上可以是记录(如Pascal中)或结构(如C语言中),而图形文件就是由特定的结构或记录组成的。每种图形文件都按自己的方式组织图形信息,由于图形文件包含的数据量大,所以很多图形文件都使用压缩算法来压缩图形数据。

1.1.2 位映像图像和向量图像

图像可粗分为两大类:位映像图像和向量图像。基于计算机的位映像图像是对电视图像的数字化的,它易于描述真实景物,真实世界中的景物可以用扫描仪生成图像文件并在计算机上显示。而向量图像易于表达艺术家设计的图形。这两者在表达方式上不同。为简单起见,可把位映像图像看成是点矩阵(简称点阵)。对于单色位映像图像或打印机输出的图像而言,矩阵中的每个点要么为1要么为0(1代表黑,0代表白或相反)。在图形学中,把矩阵中的点称为像素(Pixel)。

位映像图像根据彩色数分为以下四类:单色图像、具有4~16种彩色的图像、具有32~256色的图像和256色以上的图像。也可把这四类图像称为单色图像、低彩色分辨率图像、中等彩色分辨率图像和高彩色分辨率图像。

在讨论位映像图像的彩色时,通常用保存彩色信息所需的位数来定义彩色数。把单色图像称为是1位图像,这是因为图像中的每个像素仅需1位信息;把16色图像称为是4位彩色图像,这是因为图像中的每个像素需4位信息;要表示16种不同的彩色,像素必须由4位组成,由于4色图像和8色图像不太常用,所以一般也就用不到“2位彩色图像”和“3位彩色图像”。在PC机上,另一种常见的图像是256色图像,也称8位彩色图像。256色图像有照片效果,比较真实。另外一种具有全彩色照片表达能力的图像为24位彩色图像。由于彩色的种类很多,每个像素需24位,使得彩色图像所需的存储空间很大。

在计算机里,可视信息是以一个大的比特阵列的形式存放的,每个比特对应一个微小的电子门,门可以打开,也可以关闭(事实上,半导体门的两个状态分别对应一个高电平和一个低电平,从软件的角度看,只有两个状态,通常称之为1态和0态)。图像上的每一个点对应计算机存储器内的一个或多个比特,以这种方式存储或显示的图像叫位图图像,或简单地称之为位图。通过改变计算机缓冲区各位的状态,可以控制显示的内容。显示硬件解释显示缓冲区的内容,从而在显示器屏幕上显示图像。。

接下来将介绍彩色图形编程的各种细节,但目前,还必须掌握基于调色板的显示方式的基本原理。当使用各种不同的显示模式时,软件把一个颜色编号放在相对于像素的计算机内存。在双色模式中,颜色编号只能取两个值:0或者1,通常0代表黑色,1代表白色(如果所用的显示器使用的是有颜色的荧光粉,则可能是淡黄色或绿色)。由于每个像素的颜色仅依赖于一个信息位,因此,这种颜色也叫“1比特”颜色。

对于更复杂的颜色,要经过两步才能真正显示屏幕上每个像素的颜色。首先,软件把颜色编号放在计算机内存。在16色模式中,颜色的编号可以是0~15间的任一个值,由于存储16种不同的颜色需要4个信息位,所以16色模式叫“4比特”模式。同样,

在 256 色模式中, 每个像素颜色编号的取值可高达 255, 要存储像素的颜色需要 8 个信息位。为了确定每个颜色编号所对应的真实颜色, 显示硬件要参考调色板的颜色值。调色板是一组独立于存储各个像素颜色编号存储区的视频存储区。调色板中的颜色值指定了屏幕上像素的红、绿、蓝三个基色的混合比例, 屏幕上的每个像素对应一个颜色号。不同的像素的颜色对应不同的调色板颜色值。

存储调色板上每种颜色所需的准确位数取决于显示硬件, 例如, 在 EGA 调色板上的每种颜色值有 6 个比特, 2 比特用于红色, 2 比特用于绿色, 2 比特用于蓝色。颜色在经过图像处理软件的数字化处理之后, 转变成了数字的形态, 即由一个一个的位组成, 位中存储颜色的情况如下:

1 位: 2 种颜色。

2 位: 4 种颜色。

4 位: 16 种颜色。

8 位: 256 种颜色。

16 位: 65536 种颜色。

24 位: 1677 万种颜色。

32 位: 1677 万种颜色和 256 级灰度值。

36 位: 687 亿种颜色和 4096 级灰度值。

通常所称的标准 VGA 显示模式是 8 位显示模式, 即在该模式下能显示 256 种颜色; 而高彩色 (Hi Color) 显示是 16 位显示模式, 能显示 65536 种颜色, 也称 64K 色; 还有一种真彩色 (True Color) 显示模式是 24 位显示模式, 能显示 1677 万种颜色, 也称 16M 色, 这是现在一般 PC 机所能达到的最高颜色显示模式, 在该模式下看到的真彩色图像的色彩已和高清晰度照片没什么差别了。

在图像文件的存储格式中也是以位来存储颜色的。由于图像文件的存储格式非常多, 这里仅以 Truevision 公司设计的 32 位 TGA 文件格式为例进行说明, 在该种格式文件中, 32 位被分为两部分, 其中 24 位是颜色部分, 另外 8 位是 ALPHA 值部分, 记录着 256 级灰度, 用以加强真彩色的质量。

计算机屏幕上的每一个像素对应内存中的一个数值, 显示硬件解释该数值, 以产生实际的色点。屏幕上像素的点数及颜色值决定了显示的解析度。屏幕上水平方向的像素个数叫水平解析度, 每一列上像素的个数叫垂直解析度, 给定时间内在屏幕上能够同时显示的颜色数叫颜色解析度。尽管从技术上来讲, 解析度既指尺寸解析度又指颜色解析度, 但通常所指的是水平和垂直方向的解析度, 例如, 虽然从技术上讲, 颜色数是解析度的一部分, 像“每一种视频适配器都有最大的解析度和最多的颜色数”这样是不准确的短语。

从支持 720×438 的双色模式的大力神图形适配器, 到支持 1024×768 的 256 色或更高模式的 Super VGA 卡, 每一种视频适配器都有所支持的最大解析度及颜色数。大多数的图形硬件都支持几种不同的显示模式, 从而能够为某一应用程序在速度、解析度和颜色数之间找到一种最佳的平衡。

随着图形硬件种类的不断增多, 记住不同图形卡 and 不同模式下的解析度和颜色数并不是一件容易的事, 各种不同的和 PC 兼容的图形卡所支持的显示模式是不尽相同的,

好的显示卡会支持很高的分辨率。解析度高于 VGA 的卡通常划归于界限还不明确的 Super VGA 类（或简称为 SVGA 和 SVG）。一些权威机构以及大多数的 PC 杂志，坚持把 SVGA 专用于 800×600 的模式，而用 Super VGA、SVGA 或 “beyond SuperVGA” 指 1024×768 或更高的解析度模式。

由于 Super VGA 的范围很广，分类也不明确，很多用户难以找到支持自己特有的 SVGA 的软件，而程序员则更难写出支持大量 SVGA 卡的软件。幸运的是，在 20 世纪 80 年代后期，成立了视频标准联合会，以设计急需的 Super VGA 标准。1989 年，该显示硬件和图形软件联合会推出了主要基于 800×600 的标准，但许多工业界的领导人士提出批评，认为这一标准在出台之前就已过时。1990 年，VESA 推出了一个全面的标准，以此作为回应，该标准包括了上至 1280×1024 的 256 色模式。VESA 标准包含一个编程 Super VGA 的软件接口，通过一个特殊的驱动程序，现行卡可以支持这种界面，而不需要改变其硬件结构。可以找到支持几乎所有 Super VGA 卡的通用软件，不管这种软件是旧的，还是新的，Super VGA 解析度的 VESA 标准模式号都是一致的。

请注意，目前几乎所有的 Super VGA 视频卡都能模仿传统的计算机图形适配器（CGA），增加图形适配器（EGA）和视频图形阵列（Video Graphics Array, VGA）的低解析度模式，某些 Super VGA 卡也能模仿大力神图形适配器（Hercules Graphics Adapter，也叫做 HGA 或单色图形），低性能的大力神标准在非常便宜的 PC 中的应用也还可见。

图形一般指用计算机绘制(draw)的图，如直线、圆、圆弧、矩形、任意曲线和图表等；图像则指由输入设备捕捉实际场景画面产生的数字图像。数字图像通常有位图和向量图形两种表示形式。

位图图像（Bit-mapped-graphics），以记录屏幕上图像的每一个黑白或彩色的像素来反映图像。每一个像素有特定的位置和颜色值。位图适用于具有复杂色彩、明度多变、虚实丰富的图像，例如照片、绘画等。使用位图格式的绘画程序叫做位图绘画程序，例如 Adobe Photoshop。位图以与屏幕相对应的存储位来记忆和处理图像，把图形作为点的集合，这是绘画程序应用的典型文件格式。位图图像依赖于解析度，放大和以高清晰度打印时，容易出现锯齿状的边缘。像素的多少决定文件的大小和图像细节的丰富程度。

位图图像由数字阵列信息组成，用以描述图像中各像素点的强度与颜色。位图适合于表现含有大量细节（如明暗变化、场景复杂和多种颜色等）的画面，并可直接、快速地在屏幕上显示出来。位图占用的存储空间较大，一般需要进行数据压缩。为了便于位图的存储和交流，产生了种类繁多的文件格式，常见有 PCX、BMP、DLB、PIC、GIF、TGA 和 TIFF 等。

向量图形（Vector Graphics）的特点是，绘画程序中物体定位、形体构造建立在以数学方式记录构件（图形元素）的几何性质上，例如直线、曲线、圆形、方形的形状和大小，它不记录像素的数量，在任何解析度下输出都同样清晰。例如 Adobe Illustrator 就是使用这种格式的软件。向量格式更适合于以线条物体定位为主的绘制，通常用于计算机辅助设计（CAD）和工艺美术设计、插图等。使用物体定位绘画程序可以把特定物体作为一组，单独改变线条的长度，可放大或缩小原形，并可进行移动和重叠操作。但是在屏幕上显示的时候，由于监视器的特点，向量图也是以像素方式来显示的。

向量图形是用一组指令集合来描述图形的内容，这些指令用来描述构成该图形的所

有直线、圆、圆弧、矩形、曲线等图的位置、维数和形状。在屏幕上显示向量图形要有专门软件将描述图形的指令转换成在屏幕上显示的形状和颜色。用于产生和编辑向量图形的程序通常称为 Draw 程序。这种程序可以产生和操作向量图形的各个成分,并对向量图形进行移动、缩放、旋转和扭曲等操作;使用向量图形的一个很大的优点就是容易进行这类操作。但是,用向量图形格式表示复杂图像(如人物或风景照片)的开销太大,因此向量图形主要用于表示线框型的图画、工程制图、美术字等。绝大多多数 CAD 和 3D 造型软件使用向量图形作为基本的图形存储格式。

向量图的优点也就在于它在任何解析度下输出都同样清晰。通常情况下好像位图文件的色彩更饱满一些,但经过放大后它就会显示出色点,而向量图经过放大后,清晰度不会产生太大变化,这一点是向量图像和点阵图像的主要区别,所以为了保持放大后的效果,很多点阵图像通常是先转化成向量图像,再进行放大缩小的处理。

1.1.3 图像处理中的分辨率

在前面的介绍中,曾提到过几种不同的分辨率,初次进行数字图像的处理时,分辨率(Resolution)这个概念经常令人感到混乱。在这里谈一下图像处理中常见的也是最重要的几种分辨率类型:屏幕分辨率、设备分辨率、网屏分辨率以及图像分辨率。

1. 屏幕分辨率

屏幕分辨率就是用户在屏幕上观察图像时所感受到的分辨率。一般屏幕分辨率是由计算机的显示卡所决定的。例如标准的 VGA 显示卡的分辨率是 640×480 ,即宽 640 点(像素),高 480 点(像素)。至于较高级的显示卡,通常可以支持 800×600 或是 1024×768 以上。

2. 位分辨率

位分辨率(Bit Resolution)又称位深,是用来衡量每个像素储存信息的位数。这种分辨率决定了每次在屏幕上可显示多少种颜色。一般常见的有 8 位、24 位或 32 位颜色。

3. 设备分辨率

设备分辨率(Device Resolution)又称输出分辨率,指的是各类输出设备每英寸上可产生的点数,如显示器、喷墨打印机、激光打印机、热式打印机、绘图仪分辨率。这种分辨率通过 dpi(Dot Per Inch)这个单位来衡量。一般来讲,PC 显示器的设备分辨率在 $60 \sim 120$ dpi 之间,而打印机的设备分辨率则在 $180 \sim 720$ dpi 之间,数值越高,效果越好。

4. 网屏分辨率

网屏分辨率(Screen Resolution)又称网屏频率,指的是打印灰度级图像或分色所用的网屏上每英寸的点数。这种分辨率通过每英寸的行数(epi)来标定。

5. 图像分辨率

图像分辨率(Image Resolution)指的是图像中储存的信息量,这种分辨率又有多种衡量法,典型的是以每英寸的像素数(ppi)来衡量。图像分辨率和图像尺寸一起决定文