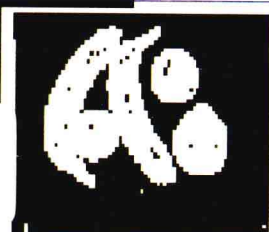




高等学校应用型特色规划教材

数字图像处理

王润辉 主 编
黄彩云 杨红云 刘 迪 副主编



免费赠送电子课件

- ◎ 系统地讲述了数字图像技术的基础知识、基本理论和基本方法，书中原理推导详细，给致力于步入该领域的人员打开了方便之门。
- ◎ 包含大量的实例分析和相关的MATLAB代码，有利于学生理解和掌握数字图像相关技术。



清华大学出版社

高等学校应用型特色规划教材

数字图像处理

王润辉 主 编

黄彩云 杨红云 刘 迪 副主编



清华大学出版社
北 京

内 容 简 介

本书系统地讲述了数字图像技术的基础知识、基本理论和基本方法。重点突出、实用性强,力求从初学者角度分析原理和相关技术,书中原理推导详细,给致力于步入该领域的人员打开了方便之门。本书主要内容包括:概论、数字图像处理基础、图像变换、图像增强、图像压缩编码、图像分割、图像描述及图像复原。

本书既可作为高校电子工程、计算机、自动化、通信工程等相关专业的高年级本科教材,也可作为相关领域的研究生教材。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

数字图像处理/王润辉主编;黄彩云,杨红云,刘迪副主编. --北京:清华大学出版社,2013

(高等学校应用型特色规划教材)

ISBN 978-7-302-31168-3

I. ①数… II. ①王… ②黄… ③杨… ④刘… III. ①数字图像处理—高等学校—教材 IV. ①TN911.73

中国版本图书馆 CIP 数据核字(2012)第 317532 号

责任编辑:李春明

封面设计:杨玉兰

责任校对:王 晖

责任印制:王静怡

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社总机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载: <http://www.tup.com.cn>, 010-62791865

印 刷 者:北京市人民文学印刷厂

装 订 者:三河市李旗庄少明印装厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:14.25 彩 插:1 字 数:342 千字

版 次:2013 年 4 月第 1 版

印 次:2013 年 4 月第 1 次印刷

印 数:1~3000

定 价:26.00 元

产品编号:039267-01

前 言

数字图像技术是 20 世纪 60 年代末发展起来的新兴学科。近年来, 由于大规模集成电路和计算机技术的发展以及诸多领域的需求, 数字图像技术已经逐步涉及人类生活和社会发展的各个方面, 并显示出广阔的应用前景, 例如航空航天、生物医学工程、工业、军事、公安、通信等领域。就发展方向而言, 数字图像技术正在向实时性、智能化、普及化、网络化方向发展; 就技术方法而言, 除了传统方法外, 将小波、模糊、神经网络、遗传算法、分形等智能信息处理技术运用于数字图像技术中, 使其更具活力。

为了更好地培养数字图像技术人才, 本书在内容上力求系统翔实、通俗易懂。本书共 8 章, 内容包括概论、数字图像处理基础、图像变换、图像增强、图像压缩编码、图像分割、图像描述及图像复原。

本书具有如下特色。

- (1) 内容系统。本书系统地讲述了数字图像技术的基础知识、基本理论和基本方法。
- (2) 重点突出。本书侧重原理分析和与原理相关的实例分析。
- (3) 实用性强。本书包含大量的实例分析和相关的 MATLAB 代码, 更能够加深学生理解和掌握数字图像相关技术。
- (4) 通俗易懂。本书力求从初学者角度分析原理和相关技术, 原理推导详细, 给致力于步入该领域的人员打开了方便之门。

本书第 1 和 8 章由东北电力大学王润辉编写, 第 2 和 7 章由东北电力大学刘迪编写, 第 3 和 5 章由湖南涉外经济学院黄彩云编写, 第 4 和 6 章由江西农业大学杨红云编写。全书由东北电力大学王润辉统稿。

本书在编写过程中参考了大量专业书籍, 对本书的完成起到了一定的指导意义, 特向这些作者致以崇高的敬意和感谢。

鉴于作者水平有限, 书中难免存在不妥和疏漏之处, 敬请广大读者批评指正。

编 者

目 录

第 1 章 概论..... 1	第 3 章 图像变换..... 41
1.1 数字图像技术的发展..... 1	3.1 图像的空域变换..... 41
1.2 图像及其表示..... 1	3.1.1 图像的代数变换..... 41
1.2.1 图像和图像的类别..... 2	3.1.2 图像的几何变换..... 42
1.2.2 数字图像及其表示法..... 2	3.2 图像的正交变换..... 46
1.2.3 图像与图形..... 6	3.2.1 傅里叶变换..... 46
1.3 图像的显示..... 6	3.2.2 离散余弦变换..... 51
1.4 图像的存储..... 10	3.2.3 沃尔什-哈达玛变换 (W-H 变换)..... 53
1.4.1 图像文件..... 10	3.2.4 K-L (Karhunen-Loeve)变换..... 56
1.4.2 内存图像..... 17	3.2.5 离散小波变换..... 58
1.5 图像像素间关系..... 19	3.3 各种变换方法的比较..... 65
1.6 图像技术及应用..... 20	小结..... 65
1.7 MATLAB 和图像处理工具箱简介..... 21	习题..... 66
小结..... 22	
习题..... 22	第 4 章 图像增强..... 67
第 2 章 数字图像处理基础..... 24	4.1 图像增强技术概述..... 67
2.1 概述..... 24	4.1.1 图像增强处理的目的..... 67
2.2 数字图像处理系统..... 25	4.1.2 图像增强方法的分类..... 68
2.2.1 图像采集模块..... 25	4.2 时域图像增强..... 68
2.2.2 图像显示模块..... 26	4.2.1 图像灰度增强..... 68
2.2.3 图像存储模块..... 26	4.2.2 图像去噪处理..... 78
2.2.4 图像传输模块..... 27	4.3 频域图像增强..... 83
2.2.5 图像处理模块..... 27	4.3.1 频域低通滤波方法..... 84
2.2.6 采样及量化..... 29	4.3.2 频域高通滤波方法..... 89
2.3 图像与视觉之间的关系..... 31	4.3.3 同态滤波..... 92
2.3.1 颜色基础..... 31	小结..... 94
2.3.2 颜色模型..... 32	习题..... 94
2.3.3 人眼与视觉信息..... 37	
2.3.4 图像质量评价方法..... 38	第 5 章 图像压缩编码..... 97
小结..... 40	5.1 概述..... 97
习题..... 40	5.1.1 数据压缩的基本概念..... 97



5.1.2 图像压缩编码的必要性.....	98	6.4.2 梯度算子和 Roberts 算子	138
5.1.3 图像压缩编码的可行性.....	99	6.4.3 Sobel 算子和 Prewitt 算子	139
5.1.4 图像压缩方法的分类.....	100	6.4.4 Laplacian 算子.....	141
5.2 统计编码.....	102	6.4.5 Canny 算子.....	142
5.2.1 几个基本概念.....	102	6.5 区域分割.....	143
5.2.2 霍夫曼编码.....	104	6.5.1 区域增长	143
5.2.3 算术编码.....	106	6.5.2 区域分裂与合并	145
5.2.4 游程编码.....	108	6.6 Hough 变换.....	148
5.2.5 LZW 编码.....	110	6.6.1 基本思想	148
5.3 预测编码.....	110	6.6.2 Hough 变换的实现步骤.....	149
5.3.1 预测编码简介.....	110	6.6.3 Hough 变换计算.....	150
5.3.2 线性预测编码(DPCM 编码) 算法.....	111	小结.....	154
5.3.3 帧间预测编码.....	113	习题.....	154
5.4 变换编码.....	115	第 7 章 图像描述	156
5.4.1 变换编码的基本原理.....	115	7.1 图像目标表达.....	156
5.4.2 变换编码中的几个问题.....	115	7.1.1 边界表达	156
5.4.3 DCT 变换编码.....	116	7.1.2 区域表达	160
5.5 数字图像压缩国际标准.....	117	7.2 边界描述子.....	161
5.5.1 静止图像压缩标准 JPEG	117	7.2.1 简单的边界描述子	161
5.5.2 新一代静止图像压缩标准 JPEG2000	122	7.2.2 形状数	162
5.5.3 视频压缩标准.....	123	7.2.3 傅里叶描述子	163
小结.....	127	7.3 区域描述子.....	165
习题.....	127	7.3.1 简单的区域描述子	165
第 6 章 图像分割	129	7.3.2 拓扑描述	166
6.1 图像分割的定义.....	129	7.3.3 形状描述	167
6.2 图像分割的基本方法.....	130	7.4 形态学描述.....	169
6.3 阈值分割.....	131	7.4.1 腐蚀	170
6.3.1 基本原理.....	131	7.4.2 膨胀	174
6.3.2 双峰阈值法.....	131	7.4.3 细化	176
6.3.3 最佳阈值法.....	133	7.5 主分量描述.....	176
6.3.4 迭代阈值法.....	134	小结.....	179
6.3.5 最大方差阈值法.....	136	习题.....	179
6.4 边缘检测.....	137	第 8 章 图像复原	181
6.4.1 边缘检测原理.....	137	8.1 图像退化与复原概述.....	181
		8.2 图像的退化模型.....	182

8.2.1	连续系统退化模型分析.....	183
8.2.2	离散系统退化模型分析.....	186
8.3	非约束复原.....	190
8.3.1	非约束复原的基本原理.....	190
8.3.2	逆滤波.....	191
8.4	约束复原.....	193
8.4.1	约束复原的基本原理.....	193
8.4.2	维纳滤波.....	194
8.4.3	约束最小平方滤波.....	197
8.5	非线性复原方法.....	199

8.5.1	最大后验复原	199
8.5.2	最大熵复原	200
8.5.3	投影复原	201
8.6	其他图像复原技术.....	202
8.6.1	几何畸变校正	202
8.6.2	盲目图像复原	204
8.6.3	人机交互式复原	205
	小结.....	207
	习题.....	207
	附录.....	208

第 1 章 概 论

【教学目标】

通过本章学习，使学生了解图像的基本概念、图像与图形的区别与联系、连续图像与数字图像的区别与联系，掌握图像的表示方法、图像的显示原理和图像的存储方法，了解数字图像技术的主要应用领域以及国内外数字图像技术的发展现状。

本章从图像的基本概念出发，介绍图像与图形的区别与联系、图像与数字图像的区别与联系，给学生建立起初步的图像印象；重点介绍图像的表示方法、图像的显示原理和图像的存储方法；数字图像技术的主要应用领域和国内外数字图像技术发展现状。通过以上方面的学习，使学生建立起完整的数字图像技术概念，为后续章节的学习打下坚实基础。

1.1 数字图像技术的发展

20 世纪初，图像技术受到了极大的限制，直到 20 世纪 20 年代，纽约至伦敦海底电缆传输一幅数字化的新闻图片，传递时间从一个多星期减少到 3 个小时，图像技术才有了些起色。20 世纪 50 年代，随着计算机技术的发展，数字图像技术才真正地引起人们的兴趣。20 世纪 60 年代中期，美国喷气推进实验室利用数字图像技术对太空船发回的月球照片进行处理，将数字图像技术应用到太空研究中。20 世纪 60 年代末和 70 年代初，发明了计算机断层技术(CT)，数字图像技术开始应用于医学领域。随后几年，数字图像技术继续用于空间研究计划并成为一门新兴的学科。20 世纪 80 年代，各种硬件的发展使人们不仅能够处理二维图像，而且开始进入三维图像领域。20 世纪 90 年代，数字图像技术已经逐步涉及人类生活和社会发展的各个方面，如航空航天、生物医学工程、工业、军事、公安、通信等。到目前为止，就发展方向而言，数字图像技术已向实时性、智能化、普及化、网络化和低成本的方向发展；就技术方法而言，主要将小波、模糊、神经网络、遗传算法、分形等智能信息处理技术运用于数字图像中，使得其更具活力。总之，数字图像技术在 40 多年的时间里，迅速发展成一门独立的学科。随着计算机技术和半导体工业的发展以及各种实际应用的需求，可以预料，数字图像技术的发展必将更加迅猛。

1.2 图像及其表示

理解图像概念和图像的表示法是学习图像技术的基础，只有对数字图像数值表示法有充分认识后，才能对其进行各种操作，才能充分利用数字图像技术。

1.2.1 图像和图像的分类

自然界的图像无论是在亮度、色彩，还是在空间分布上都是连续的。图像概念可从下列两个方面解释：“图”是物体投射或反射光的分布，“像”是人的视觉系统接收图后在大脑中形成的印象或反映。人类通过眼、耳、鼻、舌、体接收信息来感知世界，其中约有75%的信息是通过视觉系统获取的。

图像的种类较多，分类也较为复杂，按照所占的空间维数来分，可分为二维图像、三维图像以及多维图像；按照人眼通常所能感受的成像光源波长范围来分，可分为可见光图像和不可见光图像；按照图像是否随时间变化来分，可分为静止图像和运动图像；按照图像的色彩来分，可分为无彩色图像和彩色图像；按照图像的色彩空间和位置空间的取值范围来分，可分为连续图像和抽样图像。一幅静止的二维连续图像可用 $f(x,y)$ 来表示，该连续图像是由无数个像素(光点)组成的，其中 x 、 y 代表像素所在的二维空间位置， f 代表该像素的灰度、颜色等， x 、 y 、 f 的值都是实数，由于它们是由无数像素和无数级灰度或颜色组成的，因此不能直接用计算机来处理，必须通过数字化设备对它们进行抽样变成数字图像后才能进行计算机处理。

1.2.2 数字图像及其表示法

数字图像是利用各种数字成像设备(如数码相机、扫描仪等)直接从客观景物、图片等连续图像得到的，从原理上也可以看成是从连续图像经抽样得来的。数字图像也用 $f(x,y)$ 来表示， x 、 y 、 f 的值是整数，并且是有限数值，因此可以利用计算机来处理。数字图像通用表示法如图 1.1 所示。

$$f = \begin{bmatrix} f_{(1,1)} & f_{(1,2)} & \cdots & f_{(1,N)} \\ f_{(2,1)} & f_{(2,2)} & \cdots & f_{(2,N)} \\ \vdots & \vdots & \vdots & \vdots \\ f_{(M,1)} & f_{(M,2)} & \cdots & f_{(M,N)} \end{bmatrix}$$

图 1.1 数字图像通用表示法

无彩色图像可分成二值图像(只有黑白两种亮度等级)和灰度图像(有多种亮度等级)。二值图像如图 1.2 所示。每个像素的亮度用一个数值来表示，通常用数值 0 和 1 表示，一般 0 表示黑、1 表示白，二值子图像(图中小方框围起的部分)的矩阵表示法如图 1.3 所示。



图 1.2 黑白二值图像

1	0	1	0	0	0
0	0	0	1	1	1
1	1	0	1	0	1
0	0	0	0	0	0
0	0	0	0	0	0
1	0	0	1	1	1

图 1.3 6×6 个像素的二值子图像矩阵表示法

【例 1.1】图 1.2 为二值图像，以 tu1.jpg 为文件名存于 C 盘 matlab 目录下，其子图像为图中小方框围起的部分，图 1.3 为子图像矩阵表示法。用 MATLAB 编程显示子图像数值。

例 1.1 的 MATLAB 代码如下：

```
Clear
I=imread('c:\matlab\tu1.jpg'); %输入文件
y=I(341:346,471:476) %取子图像
```

命令窗口下运行结果如下：

```
1 0 1 0 0 0
0 0 0 1 1 1
1 1 0 1 0 1
0 0 0 0 0 0
0 0 0 0 0 0
1 0 0 1 1 1
```

灰度图像的每个像素灰度都是介于黑和白之间的一个灰度颜色，可以有不同灰度级图像。例如，8 级灰度图像，如图 1.4 所示。每个像素的亮度也用一个数值来表示，通常数值范围是 0~7，一般 0 表示黑、7 表示白，而 1~6 表示从黑到白之间的过渡色。8 级灰度子图像的矩阵表示法如图 1.5 所示。

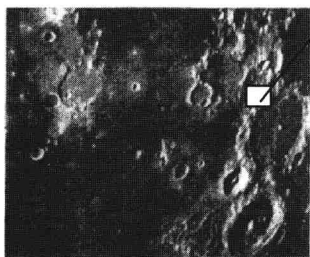


图 1.4 8 级灰度图像

0	1	2	2	7	6
3	5	5	1	0	0
3	4	6	7	3	2
0	1	2	2	5	6
3	5	3	1	0	0
3	5	2	4	3	2

图 1.5 6×6 个像素 8 级灰度子图像矩阵表示法

【例 1.2】图 1.4 为 8 级灰度图像，以 tu2.jpg 为文件名存于 C 盘 matlab 目录下，其子图像为图中小方框围起的部分，图 1.5 为子图像矩阵表示法。用 MATLAB 编程显示子图像数值。

例 1.2 的 MATLAB 代码如下：

```
Clear
I=imread('c:\matlab\tu2.jpg'); %输入文件
y=I(141:146,411:416) %取子图像
```

命令窗口下运行结果如下：

```
0 1 2 2 7 6
3 5 5 1 0 0
3 4 6 7 3 2
0 1 2 2 5 6
3 5 3 1 0 0
3 5 2 4 3 2
```

对于 256 级灰度图像而言，通常数值范围是 0~255，即可以用一个二进制字节来表示，

一般 0 表示黑、255 表示白，而数字 1~254 表示从黑到白之间的过渡色。256 级灰度图像如图 1.6 所示，256 级灰度子图像的矩阵表示法如图 1.7 所示。



图 1.6 256 级灰度图像

128	37	54	0	255	97
78	1	17	244	132	3
99	77	89	234	232	8
34	66	123	122	11	233
125	99	65	24	222	55
12	19	88	212	221	3

图 1.7 6×6 个像素 256 级灰度子图像矩阵表示法

【例 1.3】 图 1.6 为 256 级灰度图像，以 tu3.jpg 为文件名存于 C 盘 matlab 目录下，其子图像为小方框围起的部分，图 1.7 为子图像矩阵表示法。用 MATLAB 编程显示子图像数值。

例 1.3 的 MATLAB 代码如下：

```
Clear
I=imread('c:\matlab\tu3.jpg');    %输入文件
y=I(541:546,361:366)            %取子图像
```

命令窗口下运行结果如下：

```
128    37    54     0   255    97
 78     1    17   244   132     3
 99    77    89   234   232     8
 34    66   123   122    11   233
125    99    65    24   222    55
 12    19    88   212   221     3
```

彩色图像如图 1.8 所示。计算机屏幕上显示的彩色图像是由彩色像素构成的，每个彩色像素含 R、G、B 三种成分，其中 R 表示红色，G 表示绿色，B 表示蓝色。RGB 又是由不同的亮级来描述的。随着彩色像素的 R、G、B 分量值的不同，像素会呈现出不同的颜色，图像会呈现出彩色信息。彩色子图像的矩阵表示法如图 1.9 所示。

【例 1.4】 图 1.8 所示为彩色图像(可参看前面彩色插图)，以 tu4.jpg 为文件名存于 C 盘 matlab 目录下，其子图像为图中小方框围起的部分，图 1.9 为子图像矩阵表示法。用 MATLAB 编程显示子图像数值。

例 1.4 的 MATLAB 代码如下：

```
Clear
[x,map]=imread('c:\matlab\tu4.jpg');    %输入文件
imshow(I)                                %显示文件
y=x(241:243,90:92)                       %取子图像
R=x(241:243,90:92,1)
255  240  240
255   0   80
255   0    0
G=x(241:243,90:92,2)
```

```

0   160   80
255 255 160
0   255   0
B=x(241:243,90:92,3)
0   80   160
0   0   240
255 255 255

```



图 1.8 彩色图像

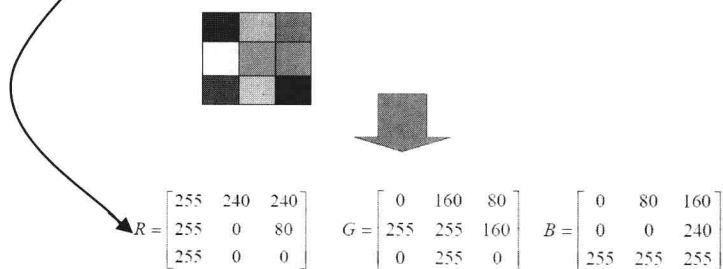


图 1.9 3×3 彩色子图像的矩阵表示法

【例 1.5】 图 1.6 为 256 级灰度图像，以 tu3.jpg 为文件名存于 C 盘 matlab 目录下，用 MATLAB 编程将其转换为二值图像。

例 1.5 的 MATLAB 代码如下：

```

Clear
I=imread('c:\matlab\tu3.jpg');           %输入文件
J=im2bw(I,0.5);                           %转换为二值图像，第二个参数在 0~1 间
imshow(J)                                  %显示文件
imwrite(J,'c:\matlab\tu5.jpg','jpg');      %文件存盘

```

命令窗口下运行结果如图 1.10 所示。



图 1.10 由图 1.6 转换成的二值图像

【例 1.6】图 1.8 为彩色图像,以 tu4.jpg 为文件名存于 C 盘 matlab 目录下,用 MATLAB 编程将其转换为二值图像。

例 1.6 的 MATLAB 代码如下。

```
Clear
[x,map]=imread('c:\matlab\tu4.jpg'); %输入文件
J= im2bw(x,map,0.5); %转换为二值图像
imshow(J) %显示文件
imwrite(J,'c:\matlab\tu6.jpg','jpg'); %文件存盘
```

命令窗口下运行结果如图 1.11 所示。

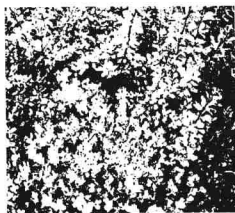


图 1.11 由图 1.8 转换成的二值图像

1.2.3 图像与图形

图形和图像是两个容易混淆的概念,其实图形与图像是两个不同的对象。图形是指由外部轮廓线条构成的矢量图,即由计算机绘制的直线、圆、矩形、曲线、图表等,如图 1.12 所示;而图像是由扫描仪、摄像机等输入设备捕捉实际的画面产生的,是由像素点阵构成的位图。图形是用一组指令集合(编程)来产生的内容,如描述构成该图的各种图元位置维数、形状等,描述对象可任意缩放,不会失真;而图像是用数字来描述像素点的强度和颜色,描述信息文件存储量较大,所描述对象在缩放过程中会损失细节或产生锯齿。图形描述的是轮廓,不是很复杂,其色彩不是很丰富的对象,如几何图形、工程图纸、CAD、3D 造型软件等;而图像表现的是含有大量细节(如明暗变化、场景复杂、轮廓色彩丰富)的对象,如照片、绘图等,通过图像软件可进行复杂图像的处理,以得到更清晰的图像或产生特殊效果。图形的产生是主观的,图像的产生则是客观的。

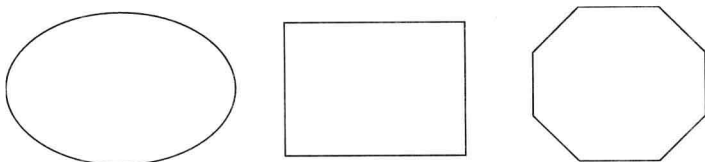


图 1.12 几种几何图形

1.3 图像的显示

显示器屏幕是由许许多多光点构成的,显示图像时一个光点对应图像的一个像素。显

示时采用扫描的方法,普通 CRT 显示器如图 1.13 所示,电子枪每次从左到右扫描一行,为每个像素着色,然后从上到下扫描若干行,就扫过了一屏。为了防止闪屏,每秒要扫 25 屏以上。我国标准为每秒 50 屏,每屏显示的像素多少称为屏幕分辨率,屏幕分辨率为 640×480 ,意思是说每行是由 640 个像素组成的,一共有 480 行,共显示 640×480 个像素。

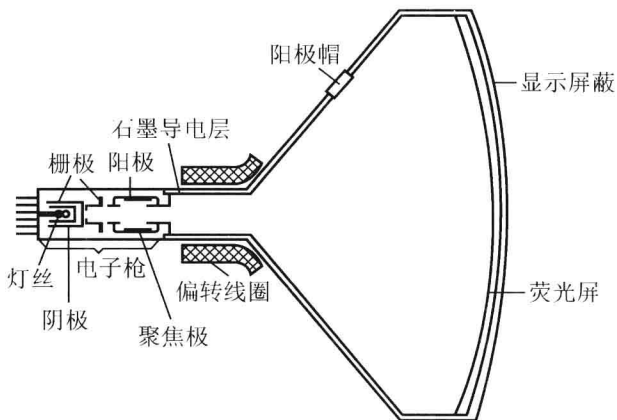


图 1.13 单枪 CRT 显示器

图 1.13 所示显示器称为位映像设备,而所谓位映像就是指一个二维的像素矩阵。位图就是采用位映像方法显示和存储的图像。显示器显示图像是工作在图形方式下,黑白二值图像每个像素有两种灰度选择,对应显示光点的亮与不亮。显示缓冲区(显示存储器)一字节对应 8 个像素,像素值为 1 或 0(一般亮为 1,不亮为 0),分别由电子束打(或不打)到屏幕相应位置来控制。灰度图像的每个像素都是介于黑和白之间的一个灰度颜色,可以有不同灰度级图像。对于 256 级灰度图像而言,每个像素有 256 种灰度选择,显示缓冲区一字节对应 1 个像素的灰度值,像素值为 $0 \sim 255$ 、一般亮为 255,不亮为 0。其他值对应其他灰度颜色,分别由电子束打到屏幕相应位置的强弱来控制。彩色图像由光的三基色 R、G、B 组合而成,自然界中的所有颜色都可以由红、绿、蓝(R、G、B)组合而成。普通 CRT 显示器一个彩色像素是由红、绿、蓝三个光点组成的,一般分别由红、绿、蓝三个电子枪打到屏幕对应发光点形成。三枪 CRT 显示器如图 1.14 所示。

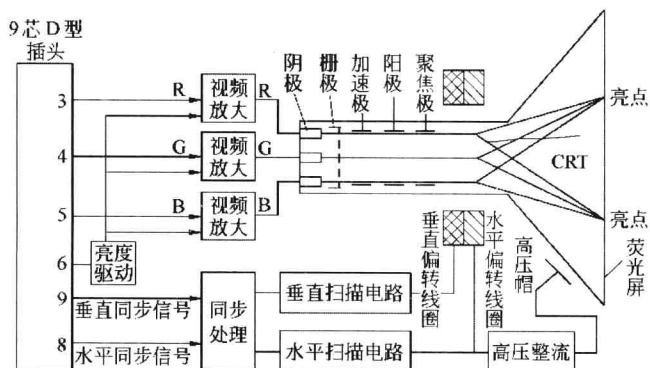


图 1.14 三枪 CRT 显示器



红颜色的深红、浅红到不含红色成分之间有 256 个等级, 0 级表示不含红色成分, 255 级表示深红, 一般由红色电子枪打到屏幕对应发光点的强弱来控制。同样, 绿色和蓝色也被分成 256 个等级。根据红、绿、蓝各种不同的组合可以表示出 $256 \times 256 \times 256$ (约 1670 万) 种颜色, 这么多种颜色对于人眼来说已经足够丰富了。表 1.1 列出了一些常见的颜色组合。

表 1.1 常见颜色的 RGB 组合值

颜 色	R	G	B
红	255	0	0
蓝	0	0	255
绿	0	255	0
黄	255	255	0
紫	255	0	255
青	0	255	255
白	255	255	255
黑	0	0	0
灰	128	128	128

对于真彩图像, 显示缓冲区 3(24 位真彩)或 4(32 位真彩)字节对应 1 个像素, 红色、绿色和蓝色各对应一字节值, 第 4 字节对应像素透明度。真彩色图并不是说一幅图包含了所有的颜色, 而是说它具有显示所有颜色的能力, 即最多可以包含所有的颜色。在 Windows 下 DIB 位图, RGB 颜色阵列存储的格式为 BGR。也就是说, 对应 24 位的 DIB 位图像素数据格式是蓝色 B 值、绿色 G 值、红色 R 值; 对应 32 位的 DIB 位图像素数据格式是蓝色 B 值、绿色 G 值、红色 R 值, 透明通道 A 值。透明通道也称 Alpha 通道, 该值是像素点的透明属性, 取值为 0(全透明)~255(不透明)。对于 24 位的图像来说, 因为没有 Alpha 通道, 所以整个图像都不透明。有些图像文件像素数据格式为 RGB, 有些则为 BGR, 在编程对图像文件处理时应注意。

真彩图像虽为目前的主流图像, 但在实际应用中仍然有一些其他彩色图像在使用, 这些图像的图像数据一般不是像素值, 需要查颜色表才能找到真正的像素值。颜色表又称调色板或颜色查找表。引入颜色表的目的是为早期图像节省显示缓冲区空间。例如, 有一个长宽各为 240 个像素, 颜色数为 16 色的彩色图像, 每一个像素都用 R、G、B 三个分量表示。因为每个分量有 256 个级别, 要用 8 位(bit), 即 1 字节来表示, 所以每个像素需要用 3 字节。整个图像要用 $240 \times 240 \times 3$, 约 168kB 显示缓冲区空间。如果用颜色表的方法, 就能节省显示缓冲区空间, 因为是一个 16 色图像, 也就是说这幅图中最多只有 16 种颜色, 可以用一个 16 行 24 位的表, 表中的每一行记录一种颜色的 R、G、B 值。这样当表示一个像素的颜色时, 只需要指出该颜色是在表中第几行, 即该颜色在表中的索引值即可。举个例子, 如果表的第 0 行为 255、0、0(红色), 那么当某个像素为红色时, 该像素值只需为 0 即可。在这种情况下, 16 种状态可以用 4 位二进制数表示, 所以一个像素要用 0.5 字节。整个图像要用 $240 \times 240 \times 0.5$, 约 28kB, 再加上表占用的字节为 $3 \times 16 = 48$ 字节, 整个占用的字节数约为前面的 1/6。这张 R、G、B 表, 就是人们常说的调色板(Palette), 又称颜色查找表或颜色表。Windows 位图、图像文件格式(如 PCX、TIF、GIF)等都用到调色板技术, 所

以很好地掌握调色板的概念是十分有用的。

在 MATLAB 中图像显示使用的是图像显示函数 `imshow()`、`montage()`、`image()` 和 `imagesec()`，其中最为常用的是 `imshow()` 函数。`imshow()` 函数的格式如下。

`imshow(I,n)`: 将图像 `I` 显示成 `n` 级灰度图像。

`imshow(I,[low high])`: 以指定的灰度范围显示灰度图像，即限制显示图像的灰度范围。

`imshow(BW)`: 使用调色板显示二值图像。例如 `imshow(BW,[1 0 0,0 0 1])`，其中数值 0 显示为红色，数值 1 显示为蓝色。

`imshow(X,map)`: 显示索引图像。

`imshow(RGB)`: 显示真彩图像。

`imshow filename`: 在未用 `imread()` 函数先输入图像的情况下，直接显示图像文件。

【例 1.7】图 1.6 为 256 级图像，以 `tu3.jpg` 为文件名存于 C 盘 `matlab` 目录下，用 MATLAB 编程实现。

- (1) 将其以 8 级灰度图像显示；
- (2) 将其以 100~200 灰度范围显示；
- (3) 将其转换为二值图像，并以数值 0 显示为红色，数值 1 显示为蓝色；
- (4) 在未用 `imread()` 函数先输入图像的情况下，直接显示图像文件。

例 1.7 的 MATLAB 代码如下：

```
Clear
I=imread('c:\matlab\tu3.jpg');    %输入文件
imshow(I,8)                        %显示文件
imshow(I,[100 200])               %显示文件
J=im2bw(I)                         %显示文件
imshow(J,[1 0 0,0 0 1])           %显示文件
```

命令窗口下运行结果如图 1.15 所示(可参看前面彩色插图)：其中图 1.15(a)为以 8 级灰度显示的图像；图 1.15(b)为以 100~200 灰度范围显示的图像；图 1.15(c)为转换成的二值图像，数值 0 显示为红色，数值 1 显示为蓝色；图 1.15(d)为在未用 `imread()` 函数先输入图像的情况下，直接显示图像。

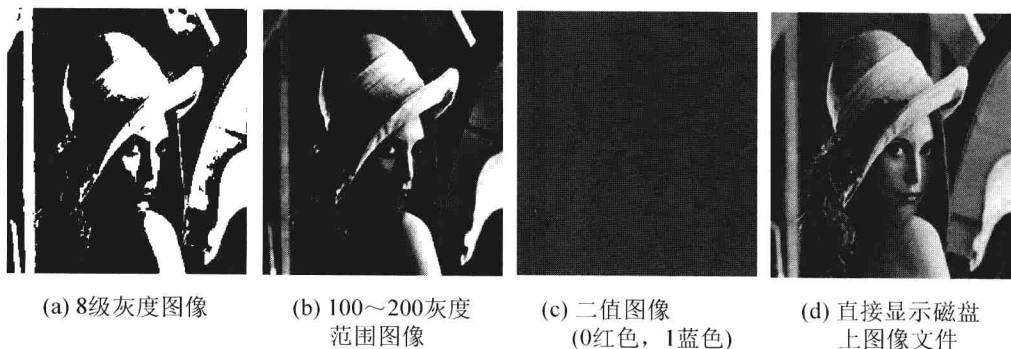


图 1.15 命令窗口下运行的结果图

1.4 图像的存储

一幅图像可以以文件的形式永久存放在外存中，要对图像进行处理必须将图像文件调入内存，要想显示图像需将内存中的图像传到显示存储器中。当然，图像从外存到内存、从内存传到显存，其原理上不是简单地传送，而是需要一定的软件技术。目前，流行的图像编程语言主要有 C++ 和 MATLAB 两种。C++ 编程工具——VC++ 编程较为复杂，需要编程人员了解图像文件格式、图像文件如何调入内存等较全面的知识；MATLAB 编程较简单，因为它提供了将一些常用的格式图像文件直接读入内存并将图像像素数据赋给二维矩阵变量的函数；C++ 编程工具——VC++.NET 也能将一些常用的不同格式图像文件直接读入内存。本书虽采用 MATLAB 编程，但为了适用更广泛的读者，本节仍较详细地剖析了几种图像文件格式及内存图像，以便不同编程人员理解相关原理。

1.4.1 图像文件

一幅图像在外存中是以文件形式永久存储的，图像文件具有多种格式。为了便于将不同的格式图像文件读入内存，同时将内存中的图像以文件的形式永久存在外存中，必须了解图像文件的格式，即图像文件的数据构成。目前已有几十种图像文件格式，它们是由不同的公司、厂商等制定的。例如，BMP、GIF、PCX、JPEG 等格式文件，分别是由微软、CompuServe 公司、Zsoft 公司、软件开发联合会制定的。

1. BMP 格式文件

BMP 文件是 Windows 采用的图形文件格式。在 Windows 环境下运行的所有图像处理软件都支持 BMP 图像文件格式。Windows 系统内部各图像绘制操作都是以 BMP 为基础的。Windows 3.0 以前的 BMP 图像文件格式与显示设备有关，因此把这种 BMP 图像文件格式称为设备相关位图(DDB)。此后出现了与显示设备无关的 BMP 图像文件格式，把这种 BMP 图像文件格式称为设备无关位图(DIB)，使用设备无关位图 DIB 的目的是为了让 Windows 能够在任何类型的显示设备上显示图像。BMP 格式文件包含位图文件头、位图信息头、彩色查找表(真彩图像无此项)和图像数据。

1) 位图文件头

BMP 文件也称为位图，位图文件头包含文件类型、文件大小、存放位置等信息。用 BITMAPFILEHEADER 结构可定义为

```
typedef struct tagBITMAPFILEHEADER {
    WORD bfType;           //文件标识，2字节的内容用来识别位图的类型
                           //该值必须是 0x424D，也就是字符'BM'
    DWORD bfSize;          //用字节表示整个文件的大小
    UINT bfReserved1;      //保留，必须设置为 0
    UINT bfReserved2;      //保留，必须设置为 0
    DWORD bfOffBits;       //从文件头开始到实际的图像数据之间的字节偏移量
} BITMAPFILEHEADER;
```