

计算机

图形学基础

JISUANJI TUXINGXUE JICHU

■ 陈锦昌 陈亮 主编

TP391.41

605

计算机图形学基础

陈锦昌 陈 亮 主编

华南理工大学出版社

·广州·

图书在版编目(CIP)数据

计算机图形学基础/陈锦昌,陈亮主编. —广州: 华南理工大学出版社, 2010.3
ISBN 978-7-5623-3259-6

I. ①计… II. ①陈…②陈… III. ①计算机图形学 IV. ①TP391.41

中国版本图书馆 CIP 数据核字 (2010) 第 027460 号

总发行: 华南理工大学出版社 (广州五山华南理工大学 17 号楼, 邮编 510640)

营销部电话: 020-87113487 87110964 87111048 (传真)

E-mail: scutc13@scut.edu.cn

<http://www.scutpress.com.cn>

责任编辑: 黄丽谊

印刷者: 广州市穗彩彩印厂

开本: 787mm × 1092mm 1/16 印张: 24.25 字数: 605 千

版次: 2010 年 3 月第 1 版 2010 年 3 月第 1 次印刷

印数: 1 ~ 2000 册

定价: 48.00 元 (含光盘)

版权所有 盗版必究

前 言

计算机图形学是计算机技术与传统图形学结合而成的一门学科。今天,随着计算机图形学在现代科学技术各个领域中的广泛应用,计算机图形学已成为计算机应用类课程中的一门必不可少的课程。目前,我国大多数高等院校都开设了计算机图形学课程,且大部分专业都把这门课程选定为必修课。

计算机图形学的教材是随着计算机硬件和计算机语言的发展而发展的。Visual Basic.NET 是在 Visual Basic 基础上发展起来的一种面向对象的程序设计语言,同时也提供了一个实用的图形界面软件开发环境。Visual Basic.NET 具有很强的图形图像处理功能,并广泛应用于图形设计、图像处理及多媒体技术中。近年来,不少学校已把 Visual Basic.NET 程序设计语言作为本科生计算机应用基础的入门语言。根据上述情况,参考教育部工程图学教学指导委员会对计算机图形学课程的基本要求,并结合多年来计算机图形学的教学实践经验,我们编写了本书。

本书将计算机图形学与 ~~计算机辅助设计~~ 相结合,系统地介绍了应用 Visual Basic.NET 程序设计语言的 ~~计算机绘图技术~~。本书共分十章。第 1 章介绍了计算机图形学和计算机辅助设计的发展历史;第 2 章介绍了应用 Visual Basic.NET 进行图形编程的基本方法;第 3 章介绍了直线、圆等基本图形的设计;第 4 章介绍了二维、三维图形变换、~~正投影变换~~、~~轴测投影变换~~、透视投影变换及程序设计方法;第 5 章介绍了平面曲线、Bézier 曲线及 B 样条曲线的程序设计方法;第 6 章介绍了空间常用曲面、Coons 曲面、Bézier 曲面及 B 样条曲面的程序设计方法;第 7 章介绍了位图图像文件及操作、位图图像处理和实现方法;第 8 章介绍了动画技术的实现方法;第 9 章介绍了一个小型交互式 CAD 系统的开发实例;第 10 章介绍了在 AutoCAD 软件中利用 VBA 进行二次开发的方法。

本书深入浅出、通俗易懂。各章节均含有大量的实例和应用程序,并配有大量插图,增强了本书的可读性。要学好计算机图形学课程,掌握应用 Visual Basic.NET 程序设计语言从事图形软件的开发,必须加强实践。本书每章后给出的习题,可供读者用以掌握和巩固所学的内容,从而使该书具有较好的实用性。

本书附有配套光盘,包括了书中的教学实例和习题解答,适合读者自学。

本书可作为高等工科院校非计算机专业本科生、专科生学习计算机图形学课程的教学用书,还可作为从事计算机图形学工作的工程技术人员参考书。

本书由陈锦昌、陈亮任主编。其中第 1 章、第 9 章由陈亮编写,第 3 章至第 6 章由陈锦昌、孙炜编写,第 2 章、第 8 章由李哲林编写,第 7 章由熊巍编写,第 10

章由张瑞秋编写。

本书在编写过程中参考了一些有关书籍,在此向有关作者表示衷心的感谢。书中存在的不足,恳请读者给予指正。

作 者

2009 年 11 月

目 录

第 1 章 计算机图形学概述	1
1.1 计算机图形学的发展概况	1
1.2 计算机图形学的主要应用领域	2
1.3 计算机图形学与计算机辅助设计(CAD)	3
1.4 计算机图形系统	4
习题 1	5
第 2 章 VB.NET 绘图基础	6
2.1 GDI+ 概述	6
2.2 绘图基础	10
2.3 绘图方法	28
习题 2	46
第 3 章 基本图形、图案设计	48
3.1 基本图形设计	48
3.2 圆弧连接程序设计	64
习题 3	93
第 4 章 图形变换及图形设计	97
4.1 二维图形变换	97
4.2 三维图形变换	127
4.3 正投影变换	137
4.4 轴侧投影变换	143
4.5 透视投影变换	149
4.6 凸平面立体隐藏线的消除	157
习题 4	168
第 5 章 曲线的程序设计	170
5.1 平面曲线	170
5.2 三次样条曲线	184
5.3 Bézier 曲线	188
5.4 B 样条曲线	191
习题 5	193
第 6 章 曲面程序设计	194
6.1 常见曲面	194
6.2 Coons 曲面	222
6.3 Bézier 曲面	227
6.4 B 样条曲面	231

习题 6	235
第 7 章 VB.NET 图像处理	236
7.1 位图图像文件及操作	236
7.2 位图图像的处理技术	246
7.3 位图图像的合成	263
习题 7	264
第 8 章 VB.NET 动画技术	266
8.1 动画技术基础	266
8.2 VB.NET 动画技术的实现方法	266
8.3 VB.NET 动画综合实例	273
8.4 综合实例源代码	281
习题 8	306
第 9 章 用 VB.NET 开发交互式 CAD 系统	309
9.1 交互式 CAD 系统的总体设计	309
9.2 创建基本图元类	313
9.3 创建交互绘图类	320
9.4 图元的选择与删除	328
9.5 图元几何变换	335
9.6 添加绘图辅助工具	342
习题 9	344
第 10 章 VBA 程序设计	345
10.1 VBA 简介	345
10.2 VBA 语言基础	350
10.3 VBA 程序设计举例	356
10.4 模块、过程及窗体的调用	361
10.5 读写数据文件	367
10.6 扩展数据和扩展记录数据	370
10.7 AutoCAD 与 Excel 的通讯	375
习题 10	378
参考文献	379

第 1 章 计算机图形学概述

计算机图形学是研究通过计算机将数据转换为图形,并在专用显示设备上显示的原理、方法和技术的学科。目前,计算机图形学已广泛地应用于工程技术与社会生活的各个领域,如机械、电子、建筑、船舶制造、航空航天、交通运输、文化教育、娱乐影视,等等。可以说,以计算机图形学为核心的 CAD/CAM 技术在工业上的应用和普及程度,已经成为了一个国家现代化水平的重要标志之一,并且对科学技术的发展和进步产生了深远的影响。

1.1 计算机图形学的发展概况

计算机图形学的发展始于 20 世纪 50 年代初。1950 年美国麻省理工学院成功研制出用阴极射线管(CRT)来作为简单图形输出设备的显示器。1958 年美国 Calcomp 公司将联机的数字记录仪发展成滚筒式绘图仪;GerBer 公司把数控机床发展成为平板式绘图机。1962 年在麻省理工学院林肯实验室,第一台光笔交互式图形显示器研制成功,这是 Ivan E. Sutherland 以博士论文的形式完成的研究课题,他在论文中首次使用了计算机图形学(Computer Graphics)这个术语,从而被公认为计算机图形学的创始人之一。1964 年,麻省理工学院的孔斯(Steven Coons)教授提出了被后人称为超限插值的曲面造型新思想,即通过插值四条任意的边界曲线来构造曲面,这一曲面被工业界和学术界称为 Coons 曲面。同一时期,法国雷诺汽车公司的工程师 Pierre Bézier 发展了一套被后人称为 Bézier 曲线、Bézier 曲面的理论,并成功地用于几何外形设计,开发了用于汽车外形设计的 UNISURF 系统。Coons 的方法和 Bézier 的方法是 CAGD 领域的开创性工作,两人也被称为现代计算机辅助几何设计技术的奠基人。

与此同时,在美国工业界,研制交互式图形显示器的工作也在开展,其中以 IBM 公司最具有代表性。20 世纪 60 年代,IBM 公司推出了 IBM2250 显示器,这是 IBM 计算机正式提供给工业界使用的第一代刷新式随机扫描图形终端。随后,洛克希德飞机公司基于 IBM2250 开发了 CADAM(Computer-Graphics Augmented Design And Manufacturing,计算机图形增强设计与制造)绘图加工系统,并于 1974 年起向外界转让,成为 IBM 主机上应用最广的 CAD/CAM 软件。

从 20 世纪 70 年代到 80 年代,随着计算机技术的进步,各种个人计算机、CAD 工作站逐渐发展起来。Apple、IBM PC、Apollo、SUN 等工程工作站就是在这—时期问世和发展起来的,这些产品普遍采用了光栅扫描显示的方法。同时,联网的分布式工作站也逐渐取代了分时式的主机-图形终端结构。另外,计算机图形学理论也有了快速的发展。20 世纪 70 年代,真实感图形学和实体造型技术相继产生,并出现了英国剑桥大学 CAD 小组的 Build 系统和美国罗彻斯特大学的 PADL-1 系统等实体造型系统。20 世纪 80 年代,相继出现了光线跟踪算法和辐射度算法,使真实感图形的显示算法日益成熟。进入 21 世纪以来,随着计算机运算能力的逐渐提高、图形处理速度的不断加快以及互联网技术的飞速发展,计算机图形学的各个研究方向都将得到充分发展。现在的计算机图形学已被广泛应用于计算机辅助设计

与制造(CAD/CAM)、科学计算可视化、动画和影视娱乐等各个领域。

我国的计算机图形学研究工作开始于 20 世纪 70 年代,从 80 年代开始有了较快的发展。目前,我国与发达国家在计算机图形学领域的差距正逐步缩小,新的图形学理论和算法正逐渐被提出,我国自主研发的计算机图形系统也被广泛应用。可以预见,在不远的将来,我国的计算机图形学的研究和应用水平将会大幅度提高,计算机图形学领域的研究成果也将越来越被广泛地应用于我国工业生产和社会生活的各个领域之中。

1.2 计算机图形学的主要应用领域

近年来,随着对计算机图形学原理的不断研究和计算机技术的不断发展,计算机图形学已经深入到人们生产、生活、教学和科研等众多领域之中。目前主要的应用领域有:

1. 计算机辅助设计与计算机辅助制造(CAD/CAM)

这是一个计算机图形学应用最为广泛、活跃及成熟的领域。计算机图形学在机械设计、建筑设计和电子产品设计等方面都发挥了巨大的作用。从简单的日常用品到复杂的汽车、轮船、飞机;从普通的住宅到复杂的场馆、会所、写字楼;从印刷电路板到大规模集成电路,计算机图形学将人们从传统的、繁重的手工绘图和设计中解放出来,不仅提高了工作效率,还大大提高了设计精度。另一方面,计算机图形学还可以直接将设计成果进行后续工艺的加工处理。应用计算机图形学建立起来的机器视觉系统,可以自动识别零件的毛坯,并将其输送给数控加工中心按照设计要求进行加工,加工后的零件又可以进行自动装配。例如,利用计算机图形学,美国波音公司在世界上第一次实现了波音 777 飞机的无图纸、无模型化的设计和生产。

2. 科学计算可视化

科学研究中往往需要处理和分析大量的数据,而通过计算机图形学可以将科学计算过程中的大量数据转换为几何图形及图像信息在屏幕上直观地显示出来,以帮助人们进行科学的理解和分析。目前,科学计算可视化已经广泛应用于医学、流体力学、有限元分析、气象分析等领域。例如,通过远程并行计算机资源,用体绘制技术实现的 CT 扫描三维数据场动态显示,为远程医疗手术提供了可能;利用分布式虚拟风洞,可以在虚拟环境中实现三维流场中物体周围气流的模拟和分析;采用超级计算机建立起来的交互分布环境可以对各种复杂的气象数据进行图形化的分析和处理,为各种复杂天气做出准确的预报。

3. 地理信息系统(GIS)

地理信息系统是建立在地理图形基础上的信息管理系统,目前已经在许多国家得到了广泛的应用。在由图形技术、数据库技术和管理信息相结合的地理信息系统中,图形起着核心和控制的作用。利用计算机图形生成技术可以绘制地理图、地形图、矿藏分布图、海洋地理图、人口分布图等各类图形,生成这些图形的数据来源于管理信息,但同时又可以生成统计图形和分析图形,为信息管理提供决策支持。

4. 虚拟现实及计算机动画艺术

以计算机图形学为基础,结合图像处理、计算机视觉、计算机辅助设计及交互技术发展起来的虚拟现实及计算机动画艺术近年来成为各国研究的热点。采用虚拟现实技术,可以方便地进行人物和自然景物的高度仿真,这种技术已经广泛地应用在游戏设计、电子商务、

战场模拟、影视制作、城市建设等方面。而计算机动画艺术则是在虚拟现实的基础上融入了艺术的成分。采用计算机设计和制作的艺术作品被广泛地应用在影视特技和广告设计中,其丰富的色彩、逼真的特效和低廉的成本使传统的动画设计方法逐渐被淘汰。

计算机图形学的应用远远不止上述几个方面,其在许多方面都有着很好的应用,如人机交互图形化接口设计、GPS、服装设计、计算机辅助教学等。近年来,随着个人计算机图形功能的增强和高性能图形工作站的出现,特别是图形生成和显示技术的发展,计算机图形学的应用范围正在不断扩大。

1.3 计算机图形学与计算机辅助设计(CAD)

作为计算机图形学应用的最重要的一个分支,计算机辅助设计是随着计算机图形学的发展而发展起来的。20世纪60年代以前,由于人们无法有效地描述三维模型几何数据间的拓扑关系,无法建立起有效的数学模型来描述自由曲线、曲面,这时的CAD技术仅仅是简单的计算机二维绘图技术,只能表示形体的基本几何信息。直到Bézier算法的提出,才使得人们使用计算机处理曲线、曲面变为可能,同时也使得法国达索飞机制造公司的开发者们能在二维绘图系统CADAM的基础上,开发出以表面模型为特点的自由曲面建模方法,推出了三维曲面造型系统CATIA。CATIA的推出,标志着CAD技术摆脱了单纯模仿工程图纸三视图的方式,已经能够用计算机完整描述产品零件的重要信息,从而为后期的CAD技术的发展打下了坚实的基础。UG, CV, SDRC等公司在此之后纷纷推出了各自的CAD系统。

表面模型只能表示形体的表面信息,对质量、重心、惯性矩等的描述则无能为力。20世纪70年代末期,出现了更为精确的实体造型算法,SDRC公司于1979年率先发布了世界上第一个完全基于实体造型技术的大型CAD/CAE软件I-DEAS。实体造型技术能够精确表示零件的全部属性,给设计带来了惊人的方便,它主导着CAD技术的发展方向。

进入20世纪80年代中期,CV公司提出了一种比传统“无约束自由实体造型”更新颖的算法——参数化实体造型方法。这是一种十分新颖的思想,其特点是基于特征、全尺寸约束、全数据相关和尺寸驱动的设计与修改。但是CV公司最终否决了参数化技术方案。参数化技术的策划者们集体离开了CV公司,另外成立了一个参数化技术公司,开始开发名为Pro/E(Pro/Engineer)的参数化设计软件。早期的Pro/E软件虽然性能很低,只能完成简单的工作,但它参数化技术的思想却几乎成为后来CAD业界的标准。

进入20世纪90年代后,参数化设计思想已经被广泛采用,而这一时期的计算机图形学技术也有了飞速发展,各种新的算法、新的理论层出不穷。SDRC公司提出了一种比参数化技术更为先进的实体造型技术——变量化实体造型技术,并于1993年推出全新体系结构的I-DEAS Master series软件。

20世纪90年代以后,随着个人计算机硬件设备的发展,Windows操作系统的日益完善和计算机图形学的不断进步,以Windows为平台的CAD软件快速发展,各种CAD软件不断涌现,版本不断更新。在这一时期,基于更优化图形学算法和软硬件平台的中小型CAD软件得到了迅速的发展,其以较低的系统配置、简单的操作界面、快速实用的系统功能和简单易学等优势迅速在各个行业中普及。一些大公司也纷纷将自己的CAD软件从原有的图形工作站中脱离出来,使其能够在个人计算机上顺利运行。

另一方面,由于计算机图形学、网络技术等的不断进步和新的发展要求,新一代的 CAD 软件逐渐向着智能化、集成化、网络化、并行化的方向发展。

1.4 计算机图形系统

计算机图形系统一般包括软件系统和硬件系统两部分。其中,软件系统是计算机图形系统的核心,硬件系统为软件系统的正常运行提供了基础保障和运行环境。此外,任何功能强大的计算机图形系统都只是一个辅助工具,系统的运行离不开系统使用人员的创造性思维活动。因此,使用计算机绘图系统的技术人员也属于系统组成的一部分。将软件、硬件及使用人员这三者有效地融合在一起,是发挥计算机图形系统强大功能的前提。

1.4.1 硬件系统

计算机图形系统的硬件主要由主机、输入设备(如键盘、鼠标、扫描仪等)、输出设备(如显示器、绘图仪、打印机等)、信息存储设备(主要指外部存储设备,如硬盘、软盘、光盘、U 盘等)、网络设备以及多媒体设备等组成。计算机图形系统的基本硬件构成如图 1-1 所示。

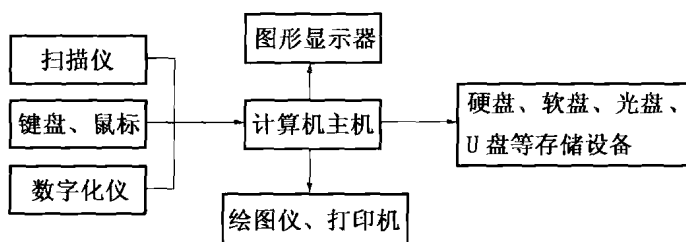


图 1-1 计算机图形系统的基本硬件组成

1. 计算机主机

主机由中央处理器(CPU)和内存储器(简称为内存)等组成,是整个计算机绘图系统的核心。CPU 的性能、内存的容量和速度以及显示芯片的性能都是影响计算机图形系统的重要指标。

2. 外存储器

外存储器简称为外存。虽然内存可以直接和运算器、控制器交换信息,存取速度快,但内存成本较高,且其容量受到 CPU 直接寻址能力的限制。外存作为内存的后援,可以将大量的程序、数据库、图形文件存放在外存中,待需要时再调入内存进行处理,同时也方便携带,便于文件管理。外存储器通常包括硬盘、软盘、光盘、U 盘等。

3. 图形输入设备

图形输入设备在计算机绘图硬件系统中占有重要的地位,操作者不仅要使用输入设备对图形进行输入、绘制及修改,同时还需要对输入的图形进行图形变换(如缩放、平移、旋转)等操作。目前,计算机绘图系统主要的输入设备有键盘、鼠标、扫描仪、数字化仪等。

4. 图形输出设备

图形显示器是计算机绘图系统中最为重要的硬件设备之一,主要用于图形图像的显示和人机交互操作,是一种交互式的图形显示设备。常用的图形输出设备包括图形显示器、绘

图仪、打印机等。

1.4.2 软件系统

计算机软件系统是指控制计算机运行,并使计算机发挥最大功效的各种程序、数据及文档的集合。在计算机图形系统中,软件配置水平决定着整个计算机图形系统的性能优劣。可以说,硬件是计算机图形系统的物质基础,软件是计算机图形系统的核心。从计算机图形系统的发展趋势来看,软件占据着愈来愈重要的地位,目前,系统配置中的软件成本已经超过了硬件。

计算机软件一般分为系统软件和应用软件。系统软件主要用于计算机的管理、维护、控制、运行,以及计算机程序的编译、装载和运行。目前常用的操作系统软件有 Windows 2000/NT/XP/2003/Vista、UNIX、Linux 等。应用软件是在系统软件的基础上,针对某一专门应用领域而开发的软件。

计算机辅助设计软件是计算机图形系统软件中最为重要的一个分支,经过长期的发展,计算机辅助设计软件已经有了众多成熟的版本。目前常用的软件有 UG、Pro/E、SolidWorks、Inventor、AutoCAD 等,国产软件有 CAXA 电子图版、圆方 BtoCAD、中望 CAD、浩辰 CAD 等。此外,还有一些专业技术人员开发的适合自身专业领域的个性化 CAD 软件,这类软件一般有两种:一种是完全运行于独立的图形平台,如 UG、Pro/E、SolidWorks、AutoCAD 等;另一种是在现有的图形平台上进行的二次开发,这类软件既借助于原来图形平台的强大图形处理能力,又结合自身专业领域的特殊要求,开发成本低,使用效率高。大型的计算机辅助设计软件都提供了丰富的二次开发接口,如 AutoCAD 就支持 AutoLisp、ARX、VBA 等多种语言的二次开发。

无论是哪一种计算机辅助设计软件,都是建立在对计算机图形学理论深入研究的基础上的,并随着计算机图形学理论的发展不断进步和完善。

习 题 1

1-1 计算机图形学与计算机辅助设计的发展经历了哪些重要阶段,并在哪些方面取得了突破性成果?

1-2 计算机图形系统由哪几部分组成,各部分分别起什么作用?

1-3 列举出你所熟悉的两到三种计算机辅助设计软件,了解该软件的发展历史、技术特点和应用范围。

第 2 章 VB.NET 绘图基础

2.1 GDI+ 概述

GDI+ (Graphics Device Interface Plus) 是 Windows 操作系统用来执行绘画及其他相关图形操作的一套子系统,是由 .Net Framework 中的 System.Drawing 命名空间提供的一组通过 C++ 类实现的应用程序编程接口,主要负责显示屏幕和打印设备输出的有关信息,它使开发人员可以利用 Windows 内置的图形功能轻松地创建图形应用程序。在 Visual Studio.NET 中 Microsoft 解决了 GDI 中的许多问题,GDI+ 对以前的 Windows 版本中的 GDI 进行了优化,并添加了许多新的功能。

作为图形设备接口的 GDI+ 使得应用开发人员在输出屏幕和打印机信息时无需考虑具体显示设备的细节,只需调用 GDI+ 库输出的类的一些方法即可完成图形操作,真正的绘图工作由这些方法交给特定的设备驱动程序来完成,GDI+ 使得图形硬件和应用程序相互隔离,从而使开发人员编写与设备无关的应用程序变得非常容易。本书将以 VB.Net 中的 GDI+ 的技术来实现各种绘图功能。

2.1.1 GDI+ 绘图的流程

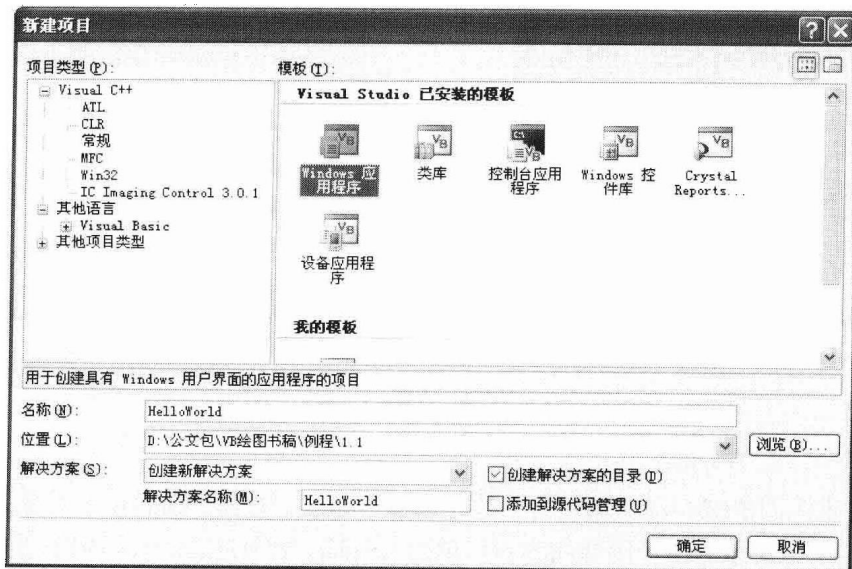
- (1) 创建 Graphics 对象;
- (2) 创建画笔和画刷;
- (3) 使用 Graphics 对象的绘图方法绘制图形、输出文字等;
- (4) 释放 Graphics 对象和绘图对象所占用的系统资源。

在 VB.NET 中开发 GDI+ 图形程序,首先必须创建一个 VB.NET 下的 Windows 应用程序。打开 Visual Studio.NET 后,在菜单中选择“文件/新建/项目”,弹出如图 2-1a 所示对话框;在对话框中的“项目类型”中选择“Visual Basic”,并在“模板”中选择“Windows 应用程序”;在“名称”文本框中输入“HelloWorld”作为应用程序名称,并单击“确定”,将出现如图 2-1b 所示的开发界面,即可开始设计界面和编写代码了。

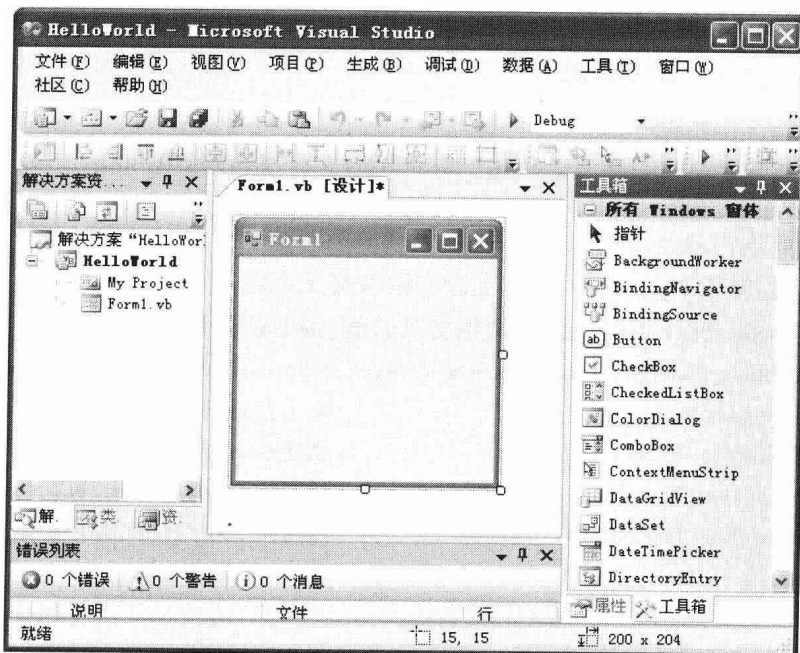
例 2-1 在主窗体 Form1 上显示“Hello, World!”。

解 在主窗体 Form1 上显示“Hello, World!”,编写的代码为

```
Private Sub Form1_Paint(ByVal sender As Object, ByVal e As System.Windows.Forms.PaintEventArgs)
Handles Me.Paint
    Dim g As Graphics = e.Graphics           '取得 Graphics 对象
    Dim b As SolidBrush = New SolidBrush(Color.Blue) '创建画刷对象
    Dim f As Font = New Font("宋体", 15, FontStyle.Bold) '创建字体对象
    g.DrawString("Hello, World!", f, b, 30, 60) '绘制文字
    f.Dispose() '释放字体对象
    b.Dispose() '释放画刷对象
```



(a)新建项目界面



(b)开发界面

图 2-1 创建一个 VB.NET 应用程序

```
g.Dispose()
```

'释放 Graphics 对象

```
End Sub
```

运行结果如图 2-2 所示。

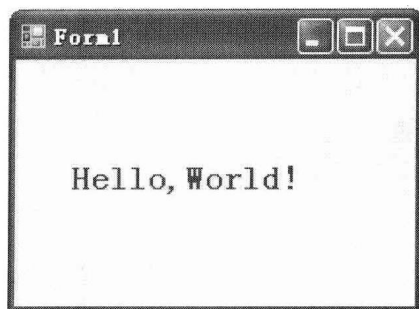


图 2-2 用 Graphics 对象输出文字

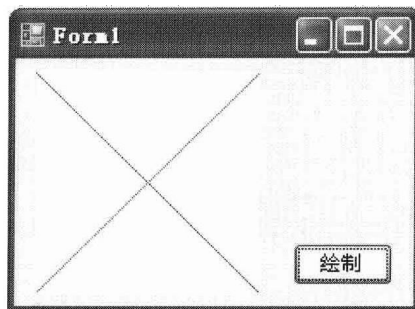


图 2-3 在 PictureBox 上绘制直线

2.1.2 创建 Graphics 对象

在 VB.NET 中用 GDI+ 创建绘图,必须先创建 Graphics 对象。Graphics 对象表示 GDI+ 的绘图表面,是用来绘制图形图像的容器。创建 Graphics 对象后,才可以使用 GDI+ 的画笔、画刷等结合 Color、Font 等对象进行绘制线条形状、填充区域、显示文本图像等操作。

VB.NET 中的窗体和所有具有 Text 属性的控件都可以作为绘制图形的表面。创建 Graphics 对象有以下 3 种方法。

1. 使用 CreateGraphics 方法创建

通过使用某控件或窗体的 CreateGraphics 方法来灵活获取对 Graphics 对象的引用,该对象表示该控件或窗体的绘图表面。这是一种常见的创建方法,其格式为

```
Dim 对象名 As Graphics
```

```
对象名 = 窗体名(或控件名).CreateGraphics
```

例 2-2 实现在 PictureBox1 控件上绘制两条交叉直线。

解 要在 PictureBox1 控件上绘制两条交叉直线,编写的代码为

```
Private Sub Draw_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Draw.Click
    Dim g As Graphics                '定义 Graphics 对象
    g = PictureBox1.CreateGraphics    '创建 Graphics 对象
    g.DrawLine(Pens.Blue, 0, 0, 130, 130) '绘制(0,0)到(130,130)的直线
    g.DrawLine(Pens.Blue, 0, 130, 130, 0) '绘制(0,130)到(130,0)的直线
    g.Dispose()                      '释放 Graphics 对象
End Sub
```

单击“绘制”按钮后可以绘制图形。运行结果如图 2-3 所示。

2. 利用 PaintEventArgs 参数传递 Graphics 对象

通过窗体或控件的 Paint 事件可以直接完成图形绘制,在编写 Paint 事件处理程序时,利用参数 PaintEventArgs 提供的窗体图形对象。格式为

```
Dim 对象名 As Graphics
```

```
对象名 = e.Graphics
```

在例 2-1 中采用的就是这种方式,通过调用窗体的 Paint 事件函数的 PaintEventArgs 参数来调用 Graphics 对象。

3. 从 Image 对象创建

创建 Graphics 对象还可以由 Image 对象或者 Image 类派生的任何对象来创建。调用 Graphics.FromImage, 提供要从其中创建 Graphics 对象的 Image 变量的名称, 如下面代码所示:

```
Dim myBitmap as New Bitmap("C:\myPic.bmp")
Dim g as Graphics = Graphics.FromImage(myBitmap)
```

例 2-3 实现从图像创建 Graphics 对象, 并在图像上绘制水印, 即增加文字和圆形。

解 从图像创建 Graphics 对象, 并在图像上绘制水印, 编写的代码为

```
Private Sub Draw_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Draw.Click
    Dim myBitmap As Bitmap = New Bitmap("d:\VB.Net\flower.bmp")
    Dim g As Graphics = Graphics.FromImage(myBitmap)
    g.DrawEllipse(Pens.White, 10, 10, 120, 120)
    g.DrawString("版权所有!", New Font("隶书", 8), Brushes.LightBlue, 50, 100)
    PictureBox1.Image = myBitmap
    PictureBox1.Refresh()
    PictureBox1.Image.Save("d:\VB.Net\flower1.bmp")
    g.Dispose()
End Sub
```

运行结果如图 2-4a 所示, 其中图 2-4b 为原图像, 图 2-4c 为加水印后保存的新图像。



(a) 运行界面



(b) 原图像



(c) 新图像

图 2-4 用 Image 对象创建 Graphics 对象

2.1.3 创建画笔和画刷

在获得一个 Graphics 对象之后, 接下来就要决定在表面上如何进行绘制。可能需要以下对象中的一个或几个: 画笔、画刷。在 GDI+ 中, Pen 类和 Brush 类分别表示画笔和画刷。在需要绘制直线、矩形和曲线的边界时使用画笔, 在需要填充图形对象时使用画刷。

Pen 类的构造函数使用颜色和宽度作为参数。以下代码将创建一个宽度为 2 个像素的红色画笔和一个宽度为 1 个像素的蓝色画笔。

```
Dim redPen As Pen = New Pen(Color.Red, 2)
Dim bluePen As Pen = Pens.Blue
```

Brush 类的功能是通过它的多个派生类 SolidBrush、HatchBrush 等来实现的。以下代码将

创建一个黑色的实心画刷。

```
Dim blackBrush As SolidBrush = New SolidBrush(Color.Black)
```

2.1.4 使用 Graphics 对象

在创建 Graphics 对象、画笔或画刷之后,就可以绘制直线、边框、曲线或图像了。Graphics 类为绘制和填充图形、曲线或图像提供了相应的方法。例如,DrawEllipse 将使用特定的画笔来绘制椭圆的边界,而 FillEllipse 将使用一种填充颜色来绘制椭圆。绘制方法使用画笔作为参数,而填充方法则使用画刷作为参数。本章 2.3 节将详细讨论绘制和填充方法。

2.1.5 释放 Graphics 对象

在 Graphics 对象使用完毕后,必须将其释放。在 VB.NET 中提供了一个 Dispose 方法,此方法可以用来释放 Graphics 对象、画笔对象等所占用的全部资源。

例如,创建 g 为 Graphics 对象,使用完后用 g.Dispose()来释放。

2.2 绘图基础

2.2.1 坐标系统

2.2.1.1 基础知识

GDI+ 绘图技术中,最基本的是坐标系统。GDI+ 默认的坐标系统与数学中的坐标系统并不一样,如果要在 GDI+ 中使用新的坐标系统,则需要在默认坐标系的基础上进行坐标的变换,例如旋转、平移等。GDI+ 中使用三种坐标系统来表示位置信息:世界坐标系、页面坐标系、设备坐标系。在缺省情况下,三种坐标系的原点都是点(0,0),这个点位于工作区的左上角,x 轴正方向水平向右,y 轴正方向垂直向下(图 2-5)。应用 GDI+ 在屏幕上绘制图形之前,坐标先要经过两次坐标位置信息的变换:第一次变换将世界坐标转换为页面坐标,第二次变换将页面坐标转换为设备坐标。



图 2-5 默认的坐标系

世界坐标系是在 GDI+ 绘图时传递给绘图方法的坐标系,就是我们日常工作的坐标系,Graphics 类提供了 TranslateTransform、RotateTransform 和 ScaleTransform 方法实现坐标系的平移、旋转和缩放。

页面坐标系是指绘图表面(如窗体或控件)使用的坐标系,通过不同度量单位来对应真实世界的尺度,Graphics 类提供了用于操作页面变换的 PageUnit 和 PageScale 属性,可进行页面坐标度量单位信息的获取和设置,如表 2-1 所示。

设备坐标系是指在进行绘制的显示设备(如显示器或打印机)上所使用的坐标系。在显示器中使用像素(pixel)作为度量单位,在打印机中使用点数(point)作为度量单位。Graphics 类提供了两个只读属性:DpiX 和 DpiY,用于检查显示设备每英寸的水平点数和垂直点数。