

第三章 交互技术

交互式绘图是指设计者在操作计算机系统绘图时，人与计算机之间可进行信息交换，从而完成复杂的绘图任务。利用一般高级语言绘制图形，为了在屏幕上显示图形，首先需要编制相应的绘图程序，一个绘图程序画出一个图形，需要修改时，必须先修改程序或根据要求再重新编写程序来实现。计算机绘图软件要求通过人机交互式方式进行图形绘制、修改，交互技术计算机绘图软件使用更方便、直观。目前几乎所有绘图应用软件都为人机交互式方法，如 Autodesk 公司出品的 AutoCAD、PhotoShop、3DMAX 等等。交互式

绘图已成为编制计算机绘图软件必不可少的特点。Visual C++方便的创建界面功能使编制交互式绘图程序变得相当简单。

交互式绘图系统中应具用数据结构的一些数据文件，这些文件中保存着构造图形的几何信息和拓扑信息、属性信息、以及一些非几何数据信息等。图形应用程序是系统中的核心部分，它从应用数据/模型中取得所需要的几何数据及属性信息，按照要求对数据进行变换处理，生成图形，并在图形输出设备上（屏幕、打印机）输出图形，图形的生成、修改、编辑等操作用户可利用人机交互绘图软件、图形输入设备（如键盘、光笔、鼠标等）进行控制输入，绘制需要的图形。

实现软件的交互技术首先应理解以下问题：如何设计一个好的用户接

口、为什么要定义逻辑输入设备、交互式绘图技术有哪些？

3.1 用户接口设计

用户接口决定用户与计算机如何进行信息交换的技术。

用户接口包括用户通过什么途径与图形系统进行联系，通过什么手段来操作系统的功能实现等。用户接口最重要的就是高效率和对用户的友好性。

用户接口一般需要处理用户模型、屏幕的有效利用、用户的反馈、设计的一致性原则、回退和出错处理、联机帮助等。

1. 用户模型

用户模型

(User Mode) 是用户接口设计的基础，它提供给用户有关他所处理的对象以及作用于这些对象的处理过程的一个概念性模型。

2. 显示屏幕的有效利用为使屏幕得到有效利用软件设计中应考虑如下几个问题:

(1) 信息显示的布局合理性。

(2) 充分而又正确地使用图符 (应用图符 (application icons)、控制图符 (control icons))

(3) 恰当地使用各种表示方法进行选择性信息显示。3. 反馈反馈就是动态地显示系统运行中所发生的一些变化, 以便更有效地进行交互作用 4. 设计的一致性原则一致性原则是指在设计系统的各个环节时, 应遵从统一

的、简单的规则，保证不出现例外和特殊的情况。还应考虑按照用户认为最正常、最合乎逻辑的方式来设计，减少用户记忆量。 **回退和出错处理**

回退（undo）机制包括取消机制、确认机制等。对可能导致错误的一些动作进行预测，设计好的诊断程序，提供出错消息等。**约束机制：**动作与对象相一致

6. 联机帮助

为用户提供**联机帮助(On-Line Help)**措施，能在用户操作过程中的任何时刻提供请求帮助。适应不同的用户，提供多种方法使软件能适应不同熟练程度的用户。

3.2 逻辑输入设备与输入处理

3.2.1 逻辑输入设备

为了使图形软件包独立于具体的硬件设施，图形输入命令常不涉及具体的输入设备，而只涉及该命令所需的数据。根据图形输入信息的不同性质，PHIGS 和 GKS 将各种图形输入设备从逻辑上分为六种：1. **定位设备**典型方法是定位屏幕光标，把屏幕上的光标移到所需要的位置上，然后按下鼠标，同志计算机取出该处的坐标值，再在此位置构件图形或进行其它操作。定位设备有鼠标器、操纵杆、跟踪球、空间球、数字化仪的触笔或手动光标等。交互时绘图时，一般需要定位，定位就是提供当前绘图的所在位置的

屏幕坐标。在绘制图形时，用户构造图形或对图形进行修改时都需要频繁地定位。

2. **笔划设备** 笔划设备的输入等于多次调用定位设备，产生一系列的坐标值，根据产生的坐标值可产生多边形和曲线等

3. **定值设备** 定值设备常用来输入各种参数和数据。

4. **字符串设备** 字符串设备即进行字符串输入

5. **选择设备** 选择设备用来选择菜单选项、属性选项和用于构图的对象形状等。

6. **拾取设备** 用拾取技术拾取一个图形对象。处理好设备的拾取技术常利用以下方法：在图形对象生成时就对每一个对象确定其拾取优先级、采用依次对拾取图形设立标志的办法、• 找距离最近的对象优先拾取。

3.2.2 输入模式输入模式即如何管理、控制多种输入设备进行工作。常用的输入模式有请求 (**request**)、采样 (**sample**)、事件 (**event**) 及其组合形式等几种。

- 1. 请求方式 (request mode)** 输入设备在应用程序的控制下工作, 程序在输入请求发出后一直被置于等待状态直到数据输入。
- 2. 取样方式 (sample mode)** 此时, 应用程序和输入设备同时工作, 当输入设备工作时, 存储输入数据, 并不断地更新当前数据, 当程序要求输入时, 程序采用当前数据值。
- 3. 事件方式 (event mode)** 每次用户对输入设备的一次操作以及形成的数据叫做一个**事件(Event)**。

思想: 一般一个事件发生时, 往往来不及进行处理, 于是, 就要把事件

按先后次序排成队列，以便先进先出，即先到的事件进入排队，先被取出进行处理。当某设备被置成事件方式，程序和设备同时工作。

4. 输入方式的组合使用一个应用程序同时可在几种输入模式方式下应用几个不同的输入设备来进行工作。

3.3 交互式绘图技术

交互技术指使用输入设备进行输入的技术。下面介绍一些常用的基本交互绘图技术，这些技术可作为设计应用系统用户接口的基本要素包括：定位技术、橡皮条技术、拖拽技术、菜单技术、定值技术、拾取技术、网格与吸附技术

定位技术

定位操作是图形输入和图形操作中常用的输入操作之一。定位有直接定位和间接定位两种方式。直接定位是指使用定位设备直接在屏幕上指定一个点的位置；间接定位是指通过定位设备的运动控制屏幕上的映射光标来进行定位。

2. 约束

约束即在图形绘制过程中对图形的方向、对齐方式等进行规定和校准。

3. 拖曳技术

拖曳技术是将形体在空间移动的过程动态地、连续地表示出来，直到用户满意的结果为止。这种技术常用于部件的装配、模拟现实生活中的实际过程。

4. 橡皮筋技术

橡皮筋技术即针对输入要求，动态地、连续地将输入过程表现出来，直到产生用户满意的输入结果为止。

5. 定值技术

定值技术在交互过程中应用很多，而且是必不可少的。用户经常需要输入一个数值，指定一个数量，完成这种任务需要确定精度（单位），需要的设备是键盘或电位计。定值技术有两种：键入数值、改变电位计阻值产生要求的数量。

6. 菜单技术 菜单是一种很重要的交互技术。它可用于指定命令、确定操作对象或选定属性等多中选一的场合。使用菜单可较好地改善应用系统的用户接口的友善性。包括以下几个方面：菜单的层次结构、菜单的

表示、菜单的显示控制、菜单的选择。**7. 拾取技术** 在图形系统交互作用的许多操作中，常常要在一个分层的对象结构或虽不分层但很复杂的对象结构中拾取一个基本对象（如最底层的对象或一个简单的部分）或一些基本对象的集合（如非最底层的对象），然后对其施加某种操作。拾取一个基本的对象可以通过以下一些方法来实现：指定名称法、特征点法、外接矩形法、分类法、直接法。**8. 网格与吸附** 网格化是帮助绘制整齐、精确图形的一种技术。网格化一般用在用户坐标系统中，按从用户坐标系统的窗口到屏幕视口的变换映射到屏幕上去。网格一般是规则的，且覆盖整个显示区。应用程序将定位器坐标舍入到最近的网格交叉点上去，从而使绘制的图形规

¹² CAD 教育网制作www.cadedu.com

整、精确。有时要从已有的某线段上的点或它的顶点开始绘制另一条线段或其他图形，直接使用定位设备来定位很难保证其重合性。吸附技术则可克服上述困难。

3.4 交互技术程序设计案例

一、程序设计功能说明

演示用 Visual C++编写图形程序时基本交互绘图技术的实现：、定位技术，文本输入、拖曳技术等。如图 3-1 所示为程序运行时的主界面，用鼠标选择不同的菜单及下拉菜单实现：鼠标画线（橡皮条技术）、利用拖曳技术

绘制矩形、文本输入以及鼠标在客户区移动同时在状态栏动态显示鼠标经过各点的坐标值等功能，如图 3-2 所示。



图 3-1

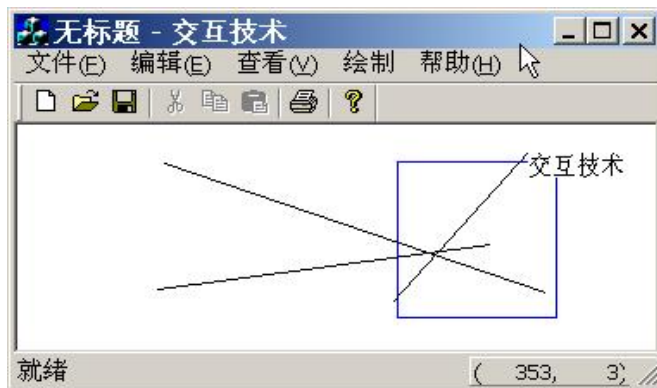


图 3-2

二、程序设计步骤

1. 创建工程名称为“交互技术”单文档应用程序框架（详细创建过程请参见第一章“基本图形的生成”应用程序）

2. 编辑菜单资源

根据 3.1 表中定义编辑菜单资源, 设计如图 3-1 所示的菜单项。

表 3.1 菜单资源表

菜单标题	菜单项标题	标示符 ID
基本图形变换	直线	ID_DRAW_LINE
	矩形	ID_DRAW_RECT
	标注文本	ID_DRAW_TEXT

3. 添加消息处理函数

利用 ClassWizard (建立类向导) 根据表 3.2 为应用程序添加与菜单项相关的消息处理函数建立如下的消息映射函数。

表 3.2 菜单项的消息处理函数

菜单项 ID	消息	消息处理函数
ID_DRAW_LINE	CONMMAN	OnDRAWLINE
ID_DRAW_RECT	CONMMAN	OnDRAWRECT
ID_DRAW_TEXT	CONMMAN	OnDRAW_TEXT
D_MIRROR_X	CONMMAN	OnIDMIRRORX
ID_MIRROR_Y	CONMMAN	OnIDMIRRORY
ID_MIRROR_O	CONMMAN	OnIDMIRRORO
ID_SCALINGXY	CONMMAN	OnIDSCALINGXY
ID_ROTATIONXY	CONMMAN	OnIDROTATIONXY

4. 在几何图形变换 View.h、几何图形变换 View.cpp 添加完成各个菜单消息及鼠标响应等处理函数，实现既定功能，

```
// 几何图形变换 View.cpp : implementation of the CMyView class
```

```
CMyView::CMyView()
```

```
{
```

程序代码见纸书.

```
}
```

说明:

1. 以上仅列出几何图形变换 View.h、几何图形变换 View.cpp 的全部代

码，其它文件代码全部为系统自动建立，无需改正，为节省篇幅此处省略，详见光盘。代码中黑体部分为手工输入，其它代码为系统生成。

2. 进行文本标注时，在点击功能菜单项之前应先在客户区中选中一点作为文本标注的起始位置。

3. 捕捉所有的鼠标输入介绍

在用鼠标交互绘制一个图形元素时，在已经开始绘制的情况下(如绘制一条直线，在暗中了直线的起点并在屏幕上拖动时)，不希望系统还进行其他操作（如打开菜单执行其它命令）避免造成系统流程和变量初始化等方面的错误。VC++提供两个成员函数来实现以上的功能需要。

SetCapture 函数运行后, 时所有的后续鼠标输入被送至当前的 CWnd 对象, 这样就把所有的鼠标输入采集到, 以防系统进行其它程序。ReleaseCapture 与 SetCapture 函数相反, 用于释放这种捕捉。这样在开始绘图时, 运行 SetCapture 函数, 在图形元素交互运行完成后, 再运行 ReleaseCapture 函数。

4. 在屏幕上拖动图形

在用鼠标交互绘制图形元素时, 为了直观的看到所绘制的图形, 一般采用拖动图形的方式。此时一般图形元素只剩最后一个控制点, 鼠标拖动改变此控制点形成图形元素的具体形状。产生拖动效果, 例如在交互绘制直线

时，在确定了起点后，鼠标移动，拖动一条“橡皮”线移动。

实现拖图形的功能首先要用 CDC 类的函数 `SetRop2` 设置绘制。当把绘制模式设置成 `R2_NOT` 时，因为两次绘制统一图形元素屏幕不便，所以在用鼠标交互绘制图形元素时，在只剩一个控制点时，当鼠标移动时，首先把上一个移动点所决定的图形元素绘制一次（擦除上一个图形元素），再把当前鼠标点决定的图形元素绘制一次，并记录下鼠标点作为下一个移动点时的上一个点，这样当鼠标移动时就实现了拖图形的效果。在编制拖动图形元素的程序时，要注意对第一次移动操作进行处理，因为第一次移动图形元素时，没有所谓“上一个”图形元素要擦除。

3.5 课后练习

1. 一个交互绘图系统的交互任务是什么？
2. 设计交互式绘图系统的主要原则？
3. 编程实现橡皮条绘制直线和圆。
4. 编程实现交互拾取一条直线。