国内首本基于ActionScript讲解图形图像算法及游戏特效制作的图书

Game Design and Develop
游戏设计与开发

点睛 ActionScript 3.0 游戏互动编程

资深ActionScript程序员多年实战经验总结
梳理大型网页游戏编程要点，快速掌握艺术与编程的精髓

游志德 著

中国工信出版集团
人民邮电出版社
POSTS & TELECOM PRESS

点睛 ActionScript 3.0 游戏互动编程

游志德 著

人民邮电出版社
北京

图书在版编目（CIP）数据

点睛 : ActionScript3.0游戏互动编程 / 游志德著
. -- 北京 : 人民邮电出版社, 2015.9
ISBN 978-7-115-39428-6

Ⅰ. ①点… Ⅱ. ①游… Ⅲ. ①动画制作软件 Ⅳ.
①TP391.41

中国版本图书馆CIP数据核字(2015)第160952号

内 容 提 要

ActionScript 3.0 是一种强大的面向对象编程语言，它还是一种适合快速构建效果丰富的互联网应用程序的语言，它所构建的应用程序已经成为 Web 体验的重要部分。

本书分 3 个部分，第一部分介绍可视化软件在图像处理方面的一些特性；第二部分介绍各种图像处理技术的工作原理；第三部分结合书中所讲的知识点，帮助读者构建各种有趣的应用、游戏或者交互动画效果。

本书属于进阶类的指南，适合有一定 ActionScript 3.0 编程基础的读者。通过阅读本书，读者将对艺术编程有更加深入细致的了解和体验。

安徽大学图书馆
藏书
www.lib.ahu.edu.cn

◆ 著　　游志德
责任编辑　陈冀康
责任印制　张佳莹　焦志炜

◆ 人民邮电出版社出版发行　　北京市丰台区成寿寺路 11 号
邮编　100164　　电子邮件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
北京中新伟业印刷有限公司印刷

◆ 开本：787×1092　1/16
印张：42.75　　彩插：4
字数：1009 千字　　2015 年 9 月第 1 版
印数：1 – 2 000 册　　2015 年 9 月北京第 1 次印刷

定价：99.00 元

读者服务热线：(010)81055410　印装质量热线：(010)81055316
反盗版热线：(010)81055315

序

我自小就酷爱手工制作，也对各种古怪的科学问题很感兴趣，所以在拥有电脑之前，房间里总是堆满各种自认为很有“技术含量”的手工艺品。它们都跟爱因斯坦小时候做的那两张板凳一样，难看得让人无法直视，也没几个人看得懂这其中所蕴含的“价值”。购买电脑之后，我的爱好一直没变，所以很快就迷上了可以制作和拼接简单动画的 PowerPoint。

随着学习的逐步深入，我发现 PPT 开始不能满足自己日益增长的制作需求，于是开始尝试转到更为专业的 Flash 软件中。但受设计和审美水平所限，我的作品依然丑陋不堪，为此我选择一头扎入 ActionScript 的海洋中。在学习了最基本的 gotoAndPlay 等语句后，我发现 ActionScript 非常适合制作各种交互程序，从网站到游戏乃至视频互动，Flash 似乎都无所不能。这点从 Macromedia 的第一件作品——迪士尼公司形象展示网站也能看得出来。于是我铁了心要把 Flash 和 ActionScript 学好，制作出真正拿得出手的作品来。

Flash 自诞生至今已经走过了 20 多个春秋，发展方向也逐步趋于多元化，这让越来越多的人认识 Flash 并加入到这一行列，从这点上说确实是一件好事。但与此同时，这让 Flash 在自己原来所擅长的领域——交互设计方面开始停滞不前，在同类产品中呈现出“逆水行舟，不进则退”的尴尬局面，并且在市场的牵制和开发者的呼声下，不得不尝试在网页游戏和移动端等方面寻求出路，却屡屡碰壁。回过头看下 Flash 的老本行，在新生的 HTML5 面前，它的优势似乎也不再像以前那么明显了。

但无论如何，Flash/ActionScript 还是在一次次的市场冲击下成功生存了下来。我们也很庆幸地看到，Flash 从未放弃互动程序和交互设计等传统领域，也一直坚持在这方面推出 Starling、增强版粒子系统、异步位图解码、svg 解析、矢量数据实时读取、Kuler 社区等振奋人心的新功能，使 Flash 互动程序和游戏等应用继续占有很高的市场份额。

所以我也一直坚守 Flash 和 ActionScript 的阵地，并为了实现自己制作出优秀作品的梦想，而一直努力提高自己在编程和设计方面的水平。然而，理工细胞较丰富的我，前者的进步速度远高于后者，导致我至今都未能做出一款完整的、赏心悦目的作品。大家总喜欢用“蛮漂亮”、“还不错”、“挺好的”这类比“一般”稍好一点的词汇对我的作品进行评价。

当然了，现在是团队合作的时代，只要大家齐心协力，发挥各自的专长，就一定能制作出创意策划、设计技术俱佳的互动程序产品。但作为一名程序员，工作久了以后，我便逐渐产生厌烦的情绪——3 年来，我都在别人的安排下做事，虽然产品是做出来了，开发人员里也有我的名单，但我总觉得缺少了一些什么……没错，缺少的是自己的思想和情感，留给大家的，只剩一个没有灵魂的空壳。ActionScript 似乎已沦落为程序员们赚钱的一个工具，它已经失去了 Flash 本来的意义。

此外，国内的游戏质量也在呈现直线下滑的趋势，玩家们几乎都吐槽美术和策划，认为是

这两方面的水平拉低了整款游戏的档次。但站在一名开发人员的角度去审视这一问题，我认为程序员也有责任。审美水平、制作态度以及对美术人员的理解，很大程度上都受自身美术修养的影响。

某天，QQ 群里有位网友问我们如何用程序给文字实现描边，群友们很快就给出了答案。没错，文字描边对于 ActionScript 来说不是一个难题，业界已经有比较成熟的解决方案。但就在那一瞬间，我的思维突然变得发散了：描边是不是只能有这一个方法？它的算法本质是什么？如果转到别的程序语言上，还能实现吗？描边的边缘略显粗糙了，有办法修饰它的细节吗？描边的样式过于单一，是否可以制作出更多更丰富的样式呢？

围绕着这些问题，我以 Photoshop 的效果为目标，以 ActionScript 3.0 为基本工具，开始了图像处理方面的深入研究，力求让做出来的作品源于 Photoshop、高于 Photoshop，让科技和艺术来一场美丽的邂逅！

部分研究成果发布到 QQ 群以后，网友们对我的作品都表示出高度的认可，也终于用“很漂亮”一词来给我评价了，并且表示很期待我发布研究成果的那一天。最最关键的是，这引起了他们的共鸣——原来，跟我同行的朋友也都渴望着有自己拿得出手的作品，能够在艺术方面也有所造诣，而做美术的朋友们也被 ActionScript 的交互效果所深深震撼。与此同时，网友们的强烈反响再次勾起了我小时候的教师梦，虽然现在我已经基本定型，想当上教师的难度很大，但我还是希望把自己的作品分享给朋友，把自己的研究成果以图书的形式传授给大家。

不得不说，这本书的诞生是一个非常艰难的过程。我从 2012 年年初就开始规划本书的创作，到 2013 年年底书稿结构成型，期间多个始料未及的技术难题阻挡着我前进的道路。有时为了不到 50 页的一章，我要花费将近半年的时间去查阅各种资料，请教各路高手，在确保每个实现步骤都准确无误之后，才敢写进书稿里，否则对不起自己，更对不起购买本书的每位读者。

在此非常感谢网名为“博图轩”的编辑，他给我传授了很多宝贵的经验，并且为我提供了丰富的读者资源，让更多有需要的朋友读到这本书。可以说，正是他，一位优秀的伯乐编辑，让这本凝聚了我多年心血的书捧在你们的手上。

感谢为本书劳心劳力的出版社编辑和工作人员，是你们成就了本书的诞生！

感谢在出版方面和 IT 领域有着丰富经验的前辈朋友们，你们对我的指导让我顺利攻克各种难题，在创作的过程中少走了很多弯路！

感谢一路以来支持和关注我的网友们，是你们多次让我重拾自信，坚持完成本书的创作！

感谢 20 多年来为我无偿付出的父母，你们对我的悉心照顾和无微不至的关怀让我可以把更多的精力投入到写书的工作上！

受笔者水平所限，书中错误和疏漏之处在所难免，恳请广大读者批评指正。

此为试读，需要完整PDF请访问：www.ertongbook.com

前言

本书的读者对象

追溯至1946年，计算机在第三次工业革命的推动下得以诞生并发展，从此地球上多了一类人，他们的世界很简单，每天不是和0接触，就是跟1来往。他们低调做人、高调做事，他们务实为民、不求名利，他们就是可爱可人、可歌可泣的IT工作者——程序员！

都说程序员不善表达，没错，离开0和1，程序员就几乎不会说话了。与此同时，程序员也恰恰是最善于表达的人，他们只用0和1这两个数字，就可以准确无误地将用户的需求传达给电脑并使其正确运行，其语言之简练，其算法之精准，其逻辑之严谨，其结果之准确，试问除了程序员，还有谁可以做得到？因此，与其说程序员不善表达，还不如说是外行人没读懂程序员的世界。

对于程序员来说，255是个神圣的数字，是一个8位二进制字节可以表示的最大整数，它代表了饱满，象征着完美，是程序员至高无上的追求。曾经有不知多少个夜晚，255这个数字让我辗转反侧，彻夜难眠。但每当别人问起我最近在忙啥，我都会愣住，为255翻滚的思绪总是难以言表。因为我心里很明白，在他们眼里，255只是个再普通不过的数字而已，真要说给他们听，不是换来朋友的不解就是招来旁人的嘲讽。

然而，让我感到欣慰的是，随着数字图像技术的日益发展，代码已经不再是程序员的专利，255也逐渐走上艺术的舞台为人所知。它既可以描绘漫天飘雪的北国风光，也可以抒发樱花绽放的浪漫情怀，既能让你驰骋在一马平川的绿野平原，也能使你陶醉于宁静深邃的碧海蓝天……所有这些梦境般的画面，都有着一个共同点：至少有一种颜色的光会100%地反射到我们的眼球。而在计算机图像领域，用于表示100%的，正是这个代表完美的数字——255！

就这一点而言，无论是对美术一窍不通的程序员，还是对程序涉足尚浅的设计师，乃至与这两个行业都绝缘的同胞们，都应拥有“艺术家”这一称号，因为不懈地追求完美，正是艺术家生活态度的重要体现，而本书更是艺术与科学完美结合的产物。

因此，本书面向的读者群比较广泛，只要您热爱生活，或者喜欢艺术，抑或拥有一颗追求完美的心，就一定可以在本书中找到适合自己的内容。而且本书的重点在于讲解艺术编程的思想，无论是站在艺术的角度来诠释，还是从科学的层面来解读，本书都经得起时间的考验和世俗的漂染。它不会因为手游市场的异军突起而备受冷落，不会因为HTML5的横空出世而迷失自我，更不会因为软件的更新换代而销声匿迹。它就像一杯绿茶，香久益清、味久益醇，每个时代都赋予它不同的气息，弥散于地球的每一个角落，给这个浮躁的社会开辟一片又一片的净土。

可见，艺术是一个很广泛的话题，我不可能通过一本书就涉及到艺术编程的方方面面，因此，先选定一个合适的领域，再围绕相关的技术点逐一展开进行讨论，对我和大家而言，思路都会更加清晰。所以本书会从交互程序中较为关键的技术点出发，讲述自己对艺术编程的一些不太深入的理解，以求抛砖引玉，让更多的读者加入到该领域，共同探讨程序的精妙与艺术的真谛所在。

本书主要面向传统 FlashPlayer（即 Stage3D 以前的版本）进行开发，同时会介绍 FlashPlayer11 的部分新功能对艺术编程的影响及其应用领域。大家不必担心旧版的功能可能会被淘汰，本书的重点是艺术编程的思想，在艺术的字典里，只有对经典的诠释，没有对过时的遗弃。

虽然交互程序种类繁多，形式也不尽相同，但无论是网站、游戏，还是视频应用、在线软件，其界面中的大部分美术元素都可以通过导入美术设计师制作好的素材，然后使用程序控制其显示隐藏、播放暂停等，来实现产品需求中的交互效果。然而，并非所有的美术素材都适合直接导入到程序中，尤其在中国这片沉淀了五千多年汉字文化的土地上，将数以千计、笔划复杂的字符全部嵌入其中，对于网速尚未发达的中国来说，将是一个相当致命的举措。

大多数情况下，开发人员都通过调用操作系统自带的字体（设备字体）来显示界面中的文本内容。然而，这仅有的几种字体在大多数情况下都过于古板，与绚丽精致的网站界面、活泼生动的游戏场景往往不太搭调；而且在目前的电脑操作系统中，设备字体往往会出现各种较为刺眼的锯齿，从而对交互产品的美观性造成不同程度的破坏。

因此，调用到设备字体以后，我们一般都需要对其进行一些美化处理，比如消除锯齿、描边、变形等，以提高设备字体与整个产品的融合度。本书就以设备字体的处理为中心，和大家一起探讨图像艺术编程的点点滴滴。

在 Windows 系统里，比较常用的有宋体、黑体、楷体、隶书等，宋体过于单薄；而楷体和隶书艺术感稍强；相比之下，字型浑厚有力且没有太多修饰的黑体就比较合适了，但是黑体也有一些不足，比如边缘过于生硬，整体感略显呆板。几经纠结，我决定使用稍加修饰过的、Windows 7 年代才开始在中国普及的微软雅黑设备字体来进行图形处理和特效制作。

言归正传，本书属于中高级编程书籍，所以我会假设大家已经具备一定的 ActionScript3 编程基础，掌握了 ActionScript3 的基本语法并熟悉常用的 API，不会就这些方面再进行详细而深入的讲解。所以，对 ActionScript3 尚没有太多了解的设计师朋友或者之前使用其他程序语言的开发者们，在阅读本书之前，我建议您先购买一到两本讲解基础的书以便随时查阅。

我个人比较推荐的几本书有：《ActionScript 3.0 Cookbook》、《ActionScript 3.0 权威指南》、《Flash ActionScript 3.0 从入门到精通》、《ActionScript 3.0 溢彩编程》（该书非常适合设计师阅读）、《ActionScript 3.0 图像处理基础教程》。

此外，在开始阅读本书之前，我建议大家先配置好开发的环境，以确保书中的代码在您的计算机上可以正常编译与运行，对开发环境配置不太熟悉的朋友可以参阅本书的附录。

一切准备就绪，艺术编程之旅正式启航！

本书内容及编排

全书共分为三部分。

第一部分介绍色彩的相关知识及 Flash、Photoshop 等可视化软件在图形图像处理方面的一

些特性，包括常用的色彩模式、Photoshop 图层效果、ActionScript 滤镜、混合模式等知识，并指出这些特性在不同软件下的异同。

第二部分以应用数学为思想，以 BitmapData、像素为工具，深入剖析各种图像处理技术的工作原理，让大家可以深入到算法的层面上认识图像处理，在 ActionScript 乃至其他支持像素级编程的语言中实现与 Photoshop 等专业图像软件相媲美的图形效果。

第三部分将充分发挥程序在交互、随机等方面的优势，运用以上学过的知识制作各种有趣的应用、游戏或者交互动画效果，让大家的程序作品以艺术的形式走进千家万户，让与您互动的每一位观众都为之感染。

本书的几个约定

在本书中，大家会看到不同样式的文字，它们有各自的含义，下面就本书用到的几种样式列举如下。

正文内容：与使用 TextField.appendText() 方法相比，使用 += 将文本追加到 TextField 要慢许多倍。

代码片段如下所示。

代码清单 0-0

```
package{
    import flash.display.*;
    import flash.text.*;
    public class TextFieldSelectionTest extends Sprite{
        private var _bold_btn:Sprite;                    //加粗按钮
        private var _input_txt:TextField;                //输入文本框
        private var _bold_format:TextFormat;             //粗体文本格式
        public function TextFieldSelectionTest(){
            init();
        }
    }
}
```

受篇幅所限，代码中的 import 部分可能会被省略，大家可以按书中的代码正确输入完以后，通过软件的自动补全功能把 import 部分补上。此外，部分 asdoc 注释也将简化为普通注释，大家可以打开配套源码中的源代码自行查阅方法中的 asdoc 注释。

小提示　文件夹（包括完整路径）最好都用英文来命名，中文名在 Flash Builder 的某些版本里会报出“导出的项目不支持双字节字符”的警告。

开发环境的配置

本书主要用到的制作工具有 Flash CS6/5.5（也可称作 Flash IDE）、Flash Builder

4.6/FlashDevelop 4.0.0 RTM 和 Photoshop CS5（版本都不会太高），如无特别说明，书中在提到 PS 的时候，指的就是 Photoshop CS5。大家可以使用这些软件的更高版本。在开始本书的学习之前，请大家先安装好上述软件并熟悉它们的使用方法和基本操作。

另外，笔者还用到了下列辅助软件：Microsoft Office Excel/WPS 表格和 ColorPicker。

为了让大家的艺术之旅不受外界环境的干扰，我建议大家创建一个专用的目录，里面包含 FlashProject、FBProject 和 PSProject 3 个子文件夹，此外再新建一个名为“PublicClasses”的目录，各种常用的工具类均可添加到里面，以便让多个项目重用。

接下来大家要在代码编写软件（Flash CS6/5.5 和 Flash Builder4.6）的首选参数里指定公共库的文件夹路径。

配置 Flash CS6/5.5 的环境

打开 Flash CS6/5.5，选择菜单项【编辑】|【首选参数】，在弹出的对话框中选择“ActionScript”系列。

单击【ActionScript 3.0 设置】按钮，在弹出的对话框中把 PublicClasses 的目录添加到源路径列表中，如图 0-1 所示。

图 0-1　添加源路径

配置 Flash Builder 4.6 的环境

打开 Flash Builder 4.6，选择菜单项【窗口】|【首选项】，在弹出的对话框里定位到“常规”-“工作空间”-“链接的资源”，按照图 0-2 所示的对话框添加一个路径变量。

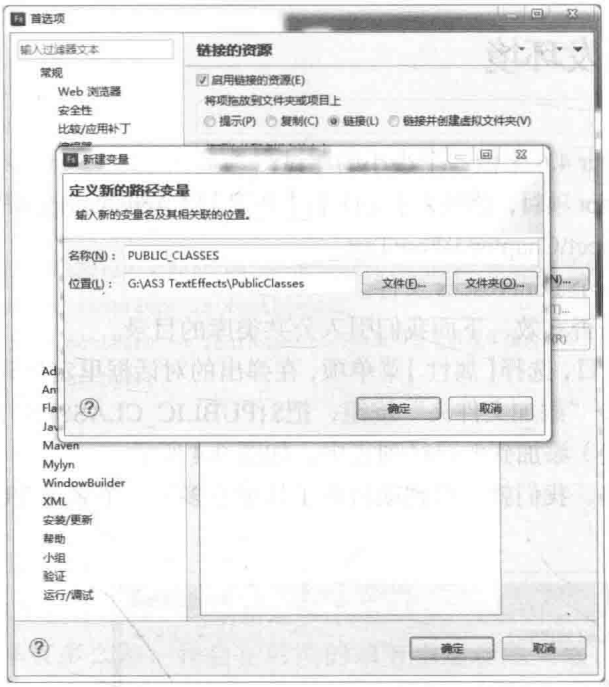


图 0-2　Flash Builder 的路径变量

配置 Photoshop CS5 的环境

作为一名“开发者”，我没有太深入地去学习这款设计软件，所以用到的功能大都比较简单，只要确保软件能正常启动，可以打开图片，新建 psd，就好了。

Photoshop 默认会在系统目录创建虚拟内存，有需要的读者可以自行选择菜单项【编辑】|【首选项】|【性能】打开如图 0-3 的窗口进行设置。

图 0-3　Photoshop 虚拟内存（暂存盘）的设置

测试配置好的开发环境

我们用 Flash Builder 4.6 来做一个小小的测试。

新建一个 ActionScript 项目，路径为【文件】|【新建】|【ActionScript 项目】，项目名为 FontTest，文件夹定位在“FBProject\Chapter 1\FontTest”。

软件会自动生成一个 FontTest.as 文件。

为了验证公共库是否有效，下面我们引入公共类库的目录。

右击“FontTest”项目，选择【属性】菜单项，在弹出的对话框里定位到“资源”-“ActionScript 构建路径”系列，单击“添加文件夹”按钮，把${PUBLIC_CLASSES}目录（在链接的资源里已经指定好了具体位置）添加到源路径列表中，如图 0-4 所示。

单击“确定”按钮，我们就会看到项目的子目录下多了一个名为 PUBLIC_CLASSES 的文件夹，如图 0-5 所示。

图 0-4　将 PUBLIC_CLASSES 添加到项目的源路径列表

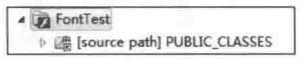


图 0-5　引用源路径后的项目结构

我们在该公共库下添加一个名为 FontUtil 的工具类，右击${PUBLIC_CLASSES}目录，选择【新建】|【ActionScript 类】菜单项，在弹出的对话框里，输入类名称“FontUtil”，包路径一栏输入 com.gemei.utils，然后单击【确定】按钮。

我们把 FontUtil.as 的代码修改如下所示。

此为试读，需要完整PDF请访问：www.ertongbook.com

代码清单 0-1

```
package com.gemei.utils{
    public class FontUtil{
        /*判断当前系统是否已经安装了指定名称的字体
          @param fontName 要查找的字体名称*/
        public static function hasFont(fontName:String):Boolean{
            var fontList:Array = Font.enumerateFonts(true);//获取当前已安装字体的列表
            for each(var item:Font in fontList){
                if(item.fontName == fontName){ //如果列表里出现字体名称相匹配的项
                    return true;
                }
            }
            return false;
        }
    }
```

我给 FontUtil 工具类创建了一个判断指定字体是否已经安装到系统的方法，大家暂时不用纠结这当中的运行效率有多么差，我会在后续的章节中逐步地优化它。下面我们先试着在 FontTest 项目里使用它。

FontTest.as 的代码调整如下。

代码清单 0-2

```
package{
    public class FontTest extends Sprite{
        public function FontTest(){
            var fontName:String = "微软雅黑";
            if(FontUtil.hasFont(fontName)){
                trace("您的系统已经安装了" + fontName + "字体");
            }else    {
                trace("您的系统尚未安装" + fontName + "字体");
            }
        }
    }
}
```

至此，全部代码均已书写完成，按 F11 键欣赏一下自己的劳动成果吧！

因为这段代码没有创建任何显示对象，所以浏览器将是一片空白。不用担心，控制台已经把运行结果告诉大家了。把浏览器最小化，大家将看到 Flash Builder 的控制台打印出“您的系统已经安装了微软雅黑字体”或者“您的系统尚未安装微软雅黑字体”这样的一句话，如图 0-6 所示。

图 0-6　测试第一个项目

如果大家电脑上的效果跟作者预期的一致，没出任何意外，那么恭喜您，Flash Builder 的开发环境已经配置成功！

安装微软雅黑字体

本书大部分实例都将使用微软雅黑的设备字体来进行创作，所以，如果 FontTest 在大家电脑上的运行结果是“微软雅黑尚未安装”，那么请您把配套源码中的“微软雅黑.ttf”复制到控制面板的字体文件夹下以完成微软雅黑字体的安装，或者复制到您的素材目录，以快捷方式的形式进行安装（Win7/8 在右键菜单中选择该操作项，WinXP 用粘贴快捷方式的方法）。

欢迎加入本书读者 QQ 群：371259465。

基于Flash简单滤镜的水晶字体(第3章)

基于Flash简单滤镜的塑料字体(第3章)

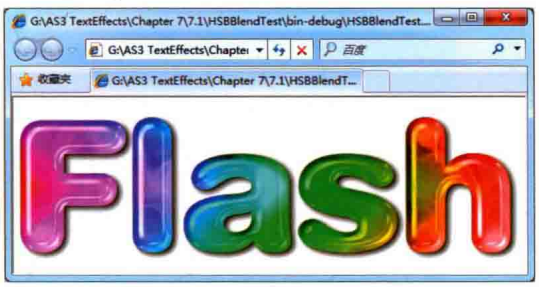
基于Flash简单滤镜的巧克力字体(第3章)

基于Flash简单滤镜的云彩纹理字体(第3章)

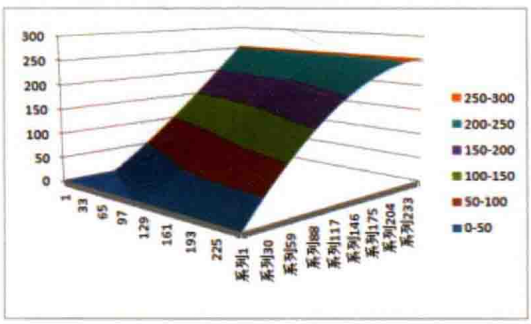
基于Flash简单滤镜的大理石字体(第3章)

基于Flash复杂滤镜的彩虹糖字体(第4章)

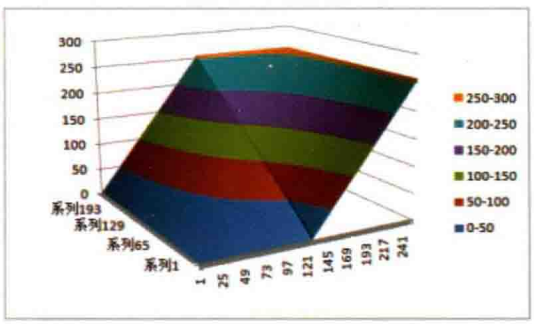
多图层混合模拟较复杂的Photoshop图层效果(第5章)



基于HSB混合的彩色琥珀字体(第6章)

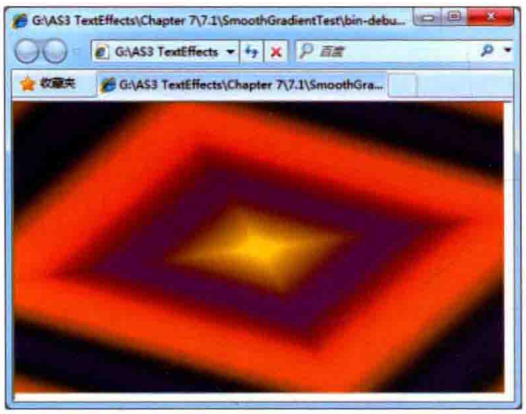


用三维图表分析混合模式的作用原理(第6章)

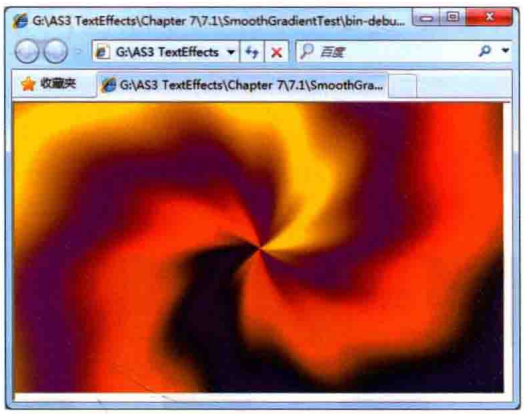


用三维图表分析混合模式的作用原理(第6章)

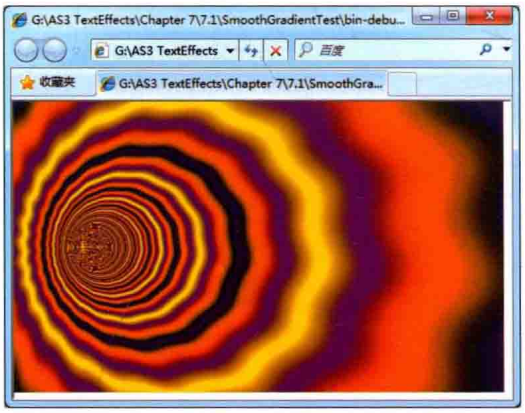
用混合模式查找边缘(第6章)



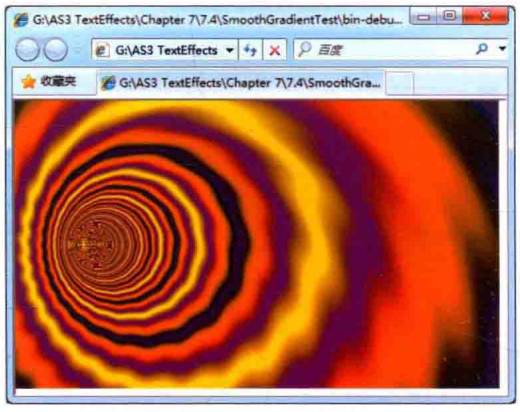
用ActionScript模拟Photoshop的菱形渐变(第8章)



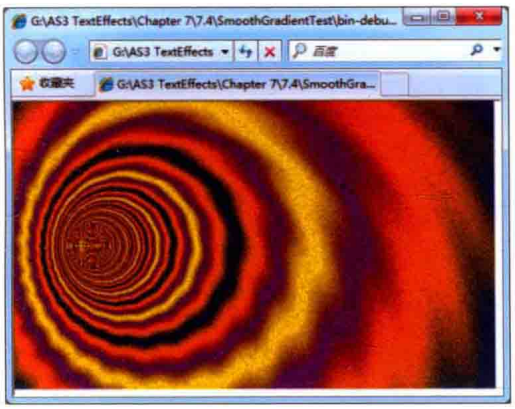
基于Photoshop渐变算法的抖动螺旋渐变(第8章)



模拟焦点偏移的抖动放射渐变(第8章)



风火轮放射渐变效果(第8章)

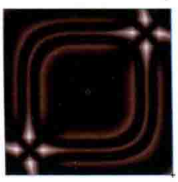


加入杂色后的抖动渐变(第8章)

用ActionScript模拟Photoshop的投影等高线，制作荧光灯字体(第10章)

用ActionScript模拟Photoshop的发光等高线，制作草莓蛋糕字体(第11章)

用ActionScript模拟Photoshop的发光等高线，制作椰蓉蛋糕字体(第11章)



用ActionScript模拟Photoshop的光泽等高线(第12章)

用ActionScript模拟Photoshop的光泽等高线(第12章)

用ActionScript模拟Photoshop的斜面等高线(第13章)

用ActionScript模拟Photoshop的斜面等高线(第13章)

基于自定义图层算法制作的Q版字体(第14章)

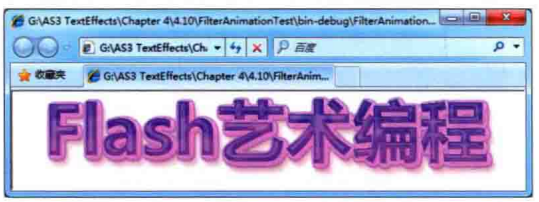
基于自定义图层算法制作的水晶高光字体(第14章)

基于自定义图层算法制作的不均匀描边效果(第14章)

应用了两次外侧渐变发光后的效果(第15章)

内外侧渐变发光联合打造的塑料质感字(第15章)

制作基于Photoshop算法的彩字生成器(第16章)



等高线循环流动的效果(第15章)

基于Photoshop斜面等高线的蓝光金属字体(第17章)

基于双投影的水滴字体(第17章)

基于Photoshop斜面等高线的蓝光金属字体(第17章)

基于Photoshop斜面和投影等高线的彩虹糖字体(第17章)

基于自定义图层样式的高光字体(第17章)

基于HSB混合的琥珀字体(第17章)

此为试读，需要完整PDF请访问：www.ertongbook.com