

5.1 案例 1 旋转的风车

【要求】

分别制作“风车叶子”元件和“风车”元件,实现旋转的风车动画效果,最终效果如图 5-1 所示。



图 5-1 旋转的风车效果

【知识点】

图层、时间轴、钢笔工具、颜料桶工具、变形面板、滤镜、补间动画、元件

【操作步骤】

1. 新建 An 文档

(1) 打开 Animate(以下简称 An)应用程序,选择“文件”|“新建”,打开“新建文档”对话框,如图 5-2 所示,平台类型选择 Action Script 3.0,角色动画标准文档默认的宽为 640px,高为 480px,帧速率为 30fps,背景颜色为白色。单击“创建”按钮,创建了一个 An 新文档,进入 An 主界面。

(2) 选择“文件”|“保存”,保存到合适位置,文件名为“旋转的风车”。此时文件全称为

“旋转的风车.flc”。选择“窗口”|“工作区”|“传统”，这时 An 界面左边是工具箱，右边是“属性”面板，上面是“时间轴”面板，中下方是舞台。

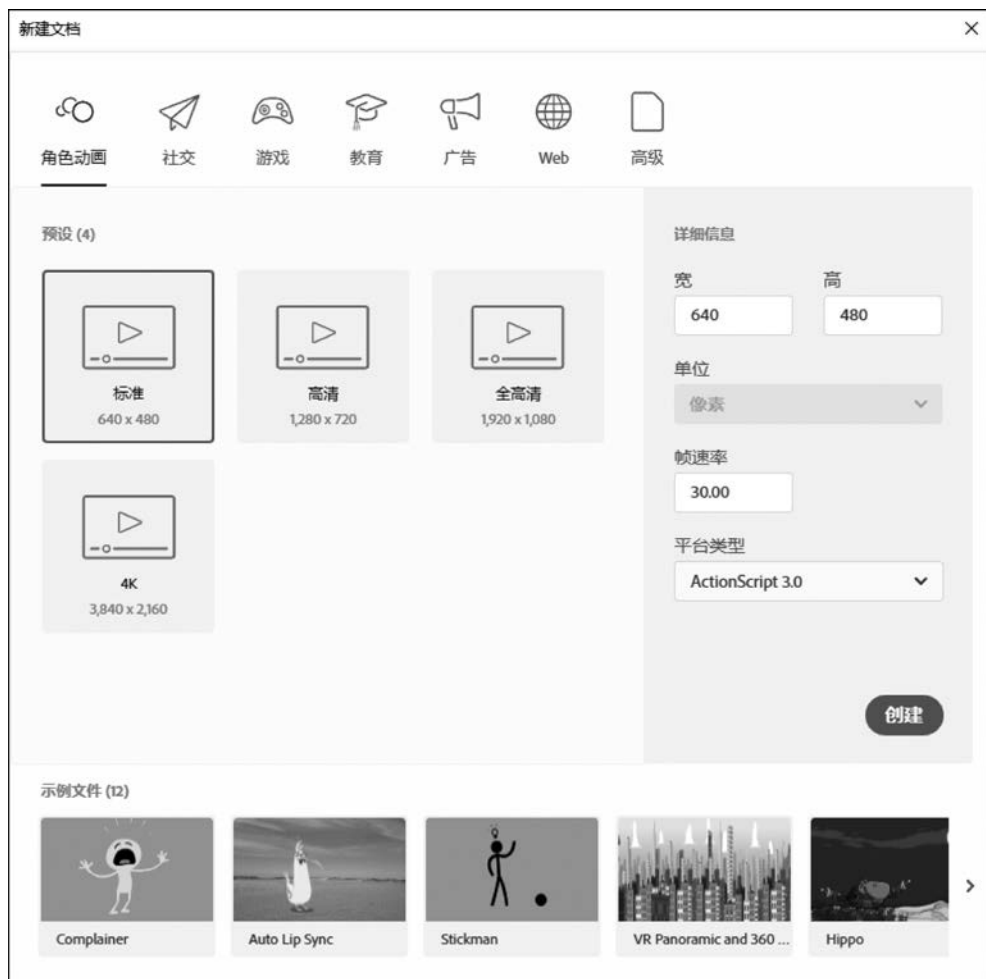


图 5-2 An 新建文档

2. 新建“风车叶子”元件

(1) 选择“插入”|“新建元件”，弹出“创建新元件”对话框，如图 5-3 所示，名称输入“风车叶子”，类型选择“图形”，单击“确定”按钮。



图 5-3 “创建新元件”对话框

(2) 进入编辑元件界面,中间有一个“+”号。选择工具箱“钢笔工具”,单击“+”号,再按顺时针方向单击其他点,建立如图 5-4(a)所示梯形,按 Esc 键结束钢笔绘制。

(3) 重新选择“钢笔工具”,单击对角线两个点画上连线如图 5-4(b)所示,选择工具箱“选择工具”,光标移向对角线时,光标显示,此时向右上角拖动鼠标,对角线变成了曲线,如图 5-4(c)所示。

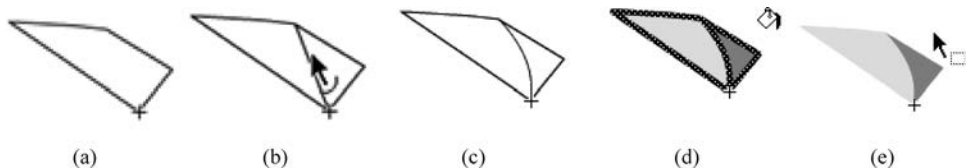


图 5-4 新建“风车叶子”元件

(4) 使用“选择工具”拖动选中整个图形,选择“修改”|“分离”(可以使用 Ctrl+B 组合键),将图形分离。如果分离命令为灰色,则表示已经分离,不用操作。单击图形其他位置,不要选中图形。

(5) 使用工具箱“颜料桶工具”,在属性窗口工具中设置不同填充颜色(右边部分填充色选择颜色 #00CC00,左边部分填充色选择颜色 #FFFF00)进行填充,如图 5-4(d)所示。如果没法填充,一般是由于图形没有完全封闭。



图 5-5 “风车叶子”元件在库面板中

(6) 使用“选择工具”单击一根连线,按 Delete 键删除,将其他线都删除,如图 5-4(e)所示。

(7) 单击舞台左上角向左箭头返回“场景 1”,结束“风车叶子”元件编辑。选择“窗口”|“库”,打开库(如果菜单中显示 ,表示已经打开),在“库”面板中单击名称为“风车叶子”的元件,在上方会预览显示其内容,如图 5-5 所示。

3. 制作“风车”元件

(1) 拖动“库”面板中的“风车叶子”元件到舞台,如图 5-6(a)所示,这样就在舞台上创建了一个实例。选择工具箱“任意变形工具”,如图 5-6(b)所示,图形显示 8 个控点;移动中间的注册点(空心圆点)到十一点,如图 5-6(c)所示。

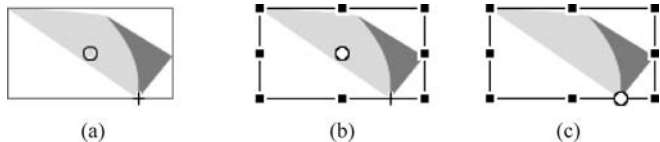


图 5-6 拖动“风车叶子”元件

(2) 选择“窗口”|“变形”,打开“变形”面板,如图 5-7 所示,旋转处输入“90”,再单击“重置选区和变形”按钮 3 次。

(3) 此时复制并旋转生成了共 4 个风车叶子,如图 5-8(a)所示。拖动选中所有叶子,如图 5-8(b)所示。



图 5-7 变形旋转

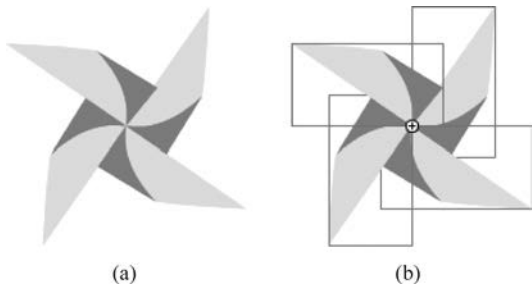


图 5-8 复制旋转

(4) 选择“修改”|“转换为元件”，弹出“转换为元件”对话框，名称输入“风车”，类型选择“影片剪辑”，如图 5-9(a)所示，单击“确定”按钮。此时风车如图 5-9(b)所示。



图 5-9 “转换为元件”对话框和风车

4. 滤镜效果

(1) 单击舞台中“风车”实例，再单击其“属性”面板“滤镜”右边的“添加滤镜”按钮，在出现的菜单中选择“投影”，设置滤镜属性投影效果，如图 5-10 所示，模糊 X: 8px，模糊 Y: 8px，强度: 50%，其他默认。

(2) 添加模糊滤镜，设置模糊效果(模糊 X: 2px，模糊 Y: 2px)。

5. 动画处理

(1) 使用时间轴“新建图层”按钮，分别新建“中心”层、“杆”层。将原来“图层_1”改名为“风车”。单击“中心”层第 1 帧，选择工具箱“椭圆工具”，笔触为无，填充色为黄色，在风车中心位置画一个圆。单击“杆”层第 1 帧，选择工具箱“矩形工具”，笔触为无，填充色为 #00CC00，从风车中心往下画一个矩形。

(2) 移动各层，使层的顺序(从上到下: 中心、风车、杆)如图 5-11 所示。

(3) 右击“风车”图层第 1 帧，在弹出的快捷菜单中选择“创建补间动画”。单击“风车”



图 5-10 设置滤镜属性

图层第 1~30 帧中的任意帧,在“属性”面板中,设置“补间动画”属性,旋转:顺时针,计数:1 次,如图 5-12 所示。



图 5-11 调整各层顺序



图 5-12 设置“补间动画”属性

(4) 右击“中心”层第 30 帧,在弹出的快捷菜单中选择“插入帧”。单击“杆”层第 30 帧,按 F5 键插入帧。此时时间轴如图 5-13 所示。

(5) 新建一图层,移至最上层,在舞台上加上自己的姓名和学号,以后每个案例需要同样操作。

(6) 单击舞台空白位置,在“属性”面板中,可设置文档属性宽和高,使其与风车大小适

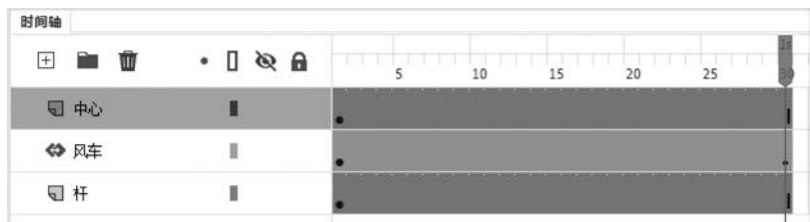


图 5-13 时间轴

当匹配。当前帧设置为第 1 帧,按住 Ctrl 键并单击各图层,可将各图层都选中,移动风车等内容到文档中间合适位置。

(7) 选择“文件”|“保存”,保存文件为“旋转的风车.flc”。按 Ctrl+Enter 组合键测试影片,测试后自动生成“旋转的风车.swf”。

(8) 选择“文件”|“导出”|“导出动画 GIF”,在弹出的对话框中选择“保存”按钮,其他默认,将导出文件命名为“旋转的风车.gif”。

5.2 案例 2 动态书写文字

【要求】

输入“宁大”文字,加上毛笔元件,制作动画“动态书写文字”,使毛笔随着书写笔画运动,有动态写字的感觉,最终效果如图 5-14 所示。



图 5-14 动态书写文字效果图

【知识点】

文本工具、选择工具、库文件导入、分离、逐帧动画、变形、时间轴、图层和元件

【操作步骤】

1. 输入文字

(1) 新建一个 500×300px 的“动态书写文字”An 文档,帧速率设置为 6fps,舞台背景颜色设置为淡黄色(#FFFFCC)。在“时间轴”面板中选择“图层_1”名称,改名为“文字”。

(2) 选择工具箱“文本工具”**T**,在属性窗口中,将字符系列设置为“华文行楷”,字符大小设置为 200 点,文本颜色任意,单击舞台输入“宁大”。

(3) 用“选择工具”选中输入的文字,选择“窗口”|“对齐”,打开“对齐”面板,选中“与舞台对齐”选项,单击“垂直居中分布”按钮  和“水平居中分布”按钮 ,使文字居于舞台中间,如图 5-15 所示。

(4) 单击“文字”图层,选中文字,执行“修改”|“分离”(Ctrl+B 组合键)两次(第一次分离是将“宁大”两个字分开,第二次分离是将文本打散转换为图形),分离后文字应有小网格点,这样才能对其进行擦除操作,如果擦除后立刻恢复原状,表示没有分离完成。

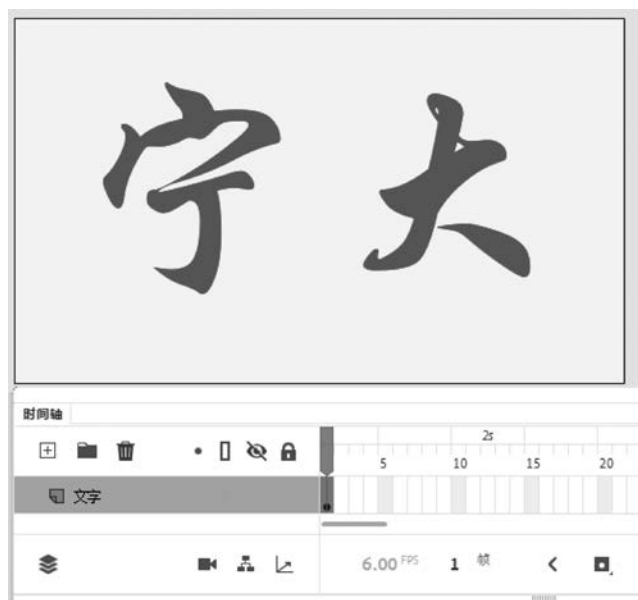


图 5-15 输入文字

2. 插入关键帧,文字擦除处理

(1) 单击“文字”图层第 1 帧,选择“插入”|“时间轴”|“关键帧”(F6 键),使用工具箱“橡皮擦工具”,将文字按照笔画相反的顺序,倒退着将文字擦除。如图 5-16 所示,“大”字已经被擦除了一部分,“文字”图层已经插入了第 2 帧。

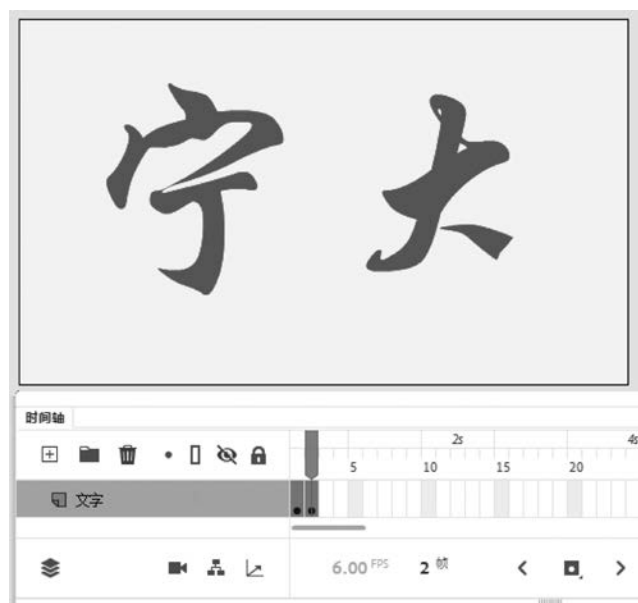


图 5-16 文字擦除开始

(2) 按快捷键 F6 插入第 3 帧,然后再擦除一部分文字,倒退着将文字擦除。擦除时注意重复的笔画应该先保留(先保留“大”字一横完整,到下一笔画再擦除),如图 5-17 所示。这样反复,每擦一次按 F6 键一次,每次擦去多少决定写字的快慢,为了使动画效果流畅自然,可根据文本笔画数及复杂程度平均分配帧数。笔者一共使用了约 30 个关键帧(不同人操作可能不同),把所有的文字部分擦完。

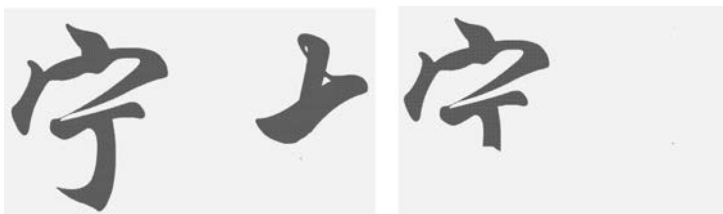


图 5-17 文字擦除过程

(3) 在“文字”图层中,右击第 1 帧,在弹出的快捷菜单中选择“所有帧”,选中所有关键帧。选择“修改”|“时间轴”|“翻转帧”,或者右击选择“翻转帧”,将“文字”图层顺序完全颠倒过来。翻转后,第 1 帧(如果这一帧没有任何内容,则删除)就只剩最后一点的一部分了,如图 5-18 所示。此时测试影片已经有文字动态效果了。单击图层上的锁定按钮  将“文字”图层锁定,以免误操作。

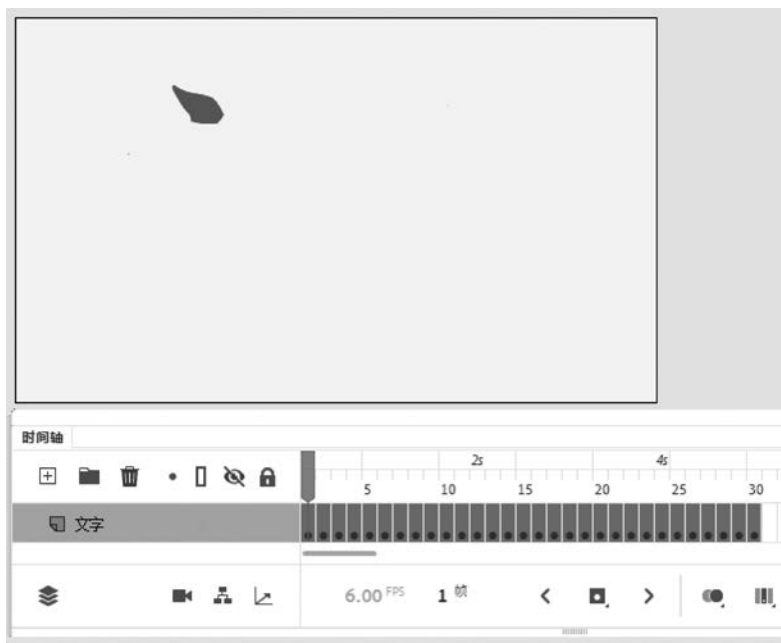


图 5-18 “翻转帧”后效果

3. 添加毛笔

(1) 新添加图层,改名为“毛笔”。选择“文件”|“导入”|“打开外部库”,打开素材“毛笔元件.flc”,会弹出“库-毛笔元件.flc”外部库窗口,该窗口中已有“毛笔”元件,如图 5-19 所示。

(2) 单击“毛笔”图层第 1 帧,拖动毛笔元件到舞台中,此时毛笔元件已经被复制到了自

已库中,将外部库窗口关闭。单击“任意变形工具”,缩放、旋转毛笔,使毛笔变形到合适大小和形状,拖动“毛笔”图形到文字起始位置,如图 5-20 所示。



图 5-19 库-毛笔元件

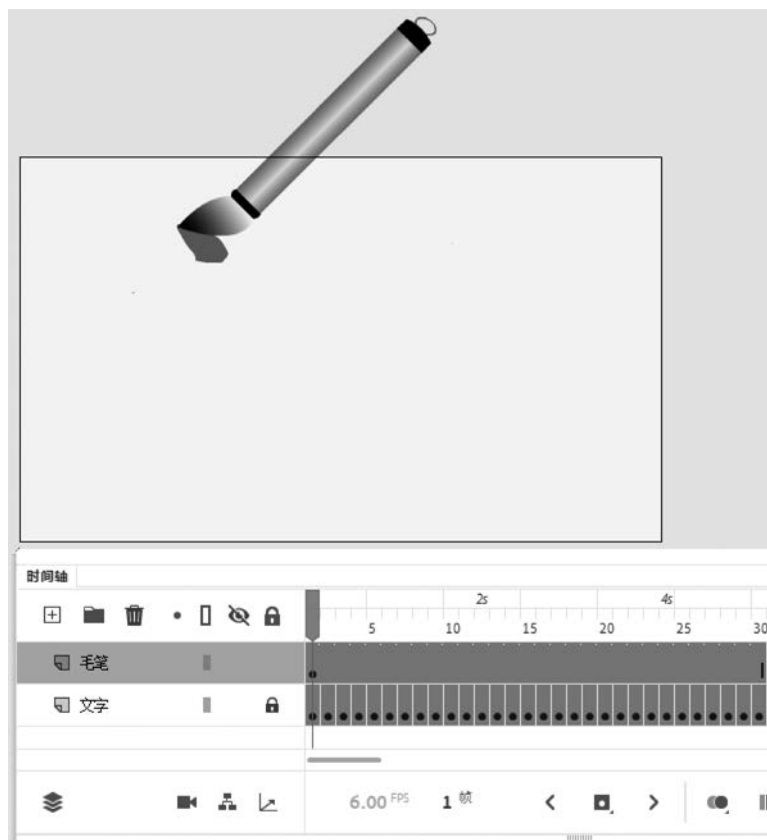


图 5-20 拖动毛笔到文字起始位置

(3) 单击“毛笔”图层第 2 帧,按 F6 插入关键帧,此时在“毛笔”图层中也插入了与“文字”图层相同个数的关键帧,拖动“毛笔”图形到当前已写笔画的最后位置,如图 5-21 所示。

(4) 这样反复操作,单击“毛笔”图层其他关键帧,用“选择工具”移动毛笔,使毛笔始终

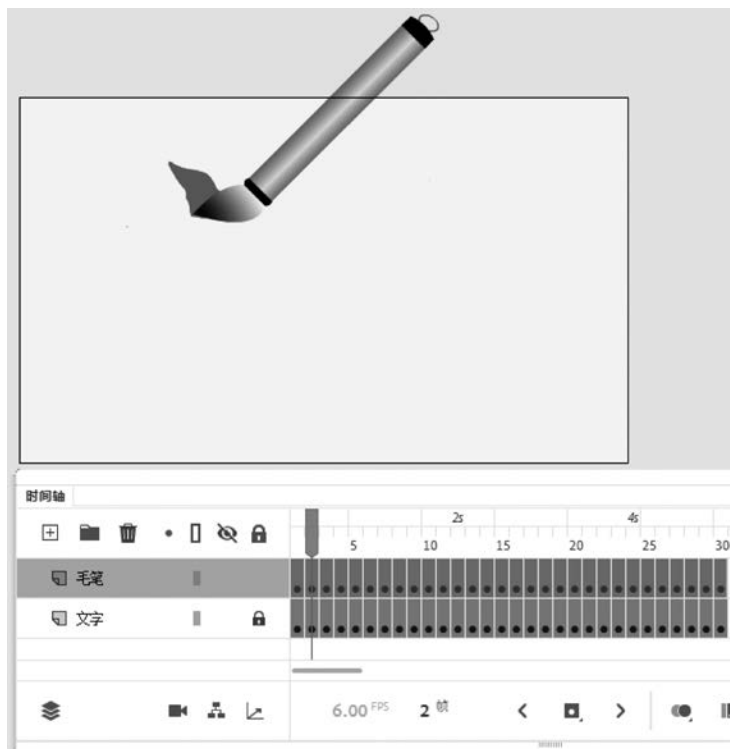


图 5-21 将毛笔拖放到已写笔画的最后位置

随着笔画最后的位置走,如图 5-22 所示。

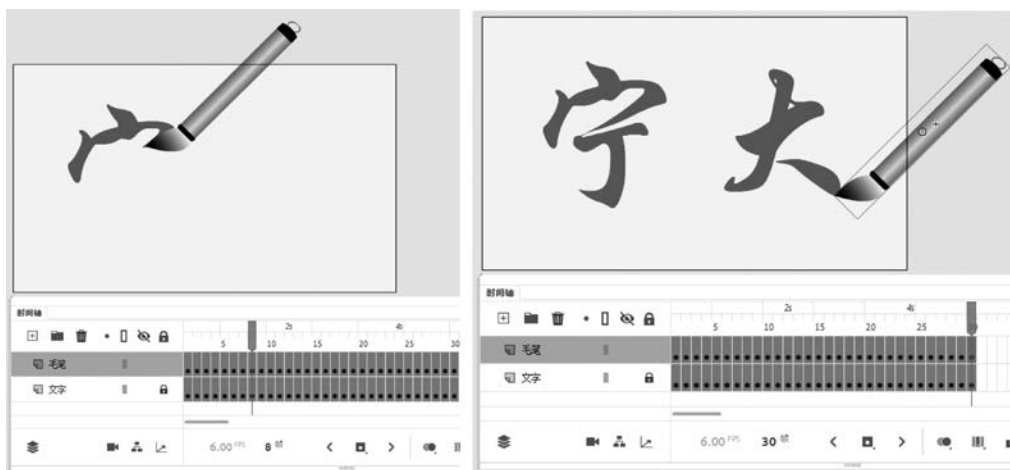


图 5-22 毛笔始终随着笔画最后的位置走

(5) 按 Ctrl+Enter 组合键测试影片,保存影片为“动态书写文字.flc”。

5.3 案例 3 图片遮罩

【要求】

通过一幅静态图片,运用各种工具制作图片遮罩效果,遮罩窗口从五角星形状变为圆

形,再变成“宁波大学”文字窗口,最终效果如图 5-23 所示。



图 5-23 图片遮罩效果

【知识点】

多角星形工具、椭圆工具、遮罩动画、补间形状动画、文本工具

【操作步骤】

1. 插入形状和文字

(1) 新建一个“图片遮罩效果”An 文档,大小设置为 $550 \times 400\text{px}$,帧速率设置为 30fps。选择“文件”|“导入”|“导入到舞台”,选择图片“宁波大学.jpg”导入到舞台,通过属性窗口更改图片大小与舞台大小一致。选择“窗口”|“对齐”,单击“左对齐”按钮和“顶对齐”按钮,将图片正好覆盖舞台。

(2) 在“时间轴”面板中选择“图层_1”第 60 帧,右击,在弹出的快捷菜单中选择“插入关键帧”。



图 5-24 星形设置

(3) 新建“图层_2”,单击其第 1 帧,选择工具箱“多角星形工具”,单击“属性”面板|“工具选项”|“样式”|“星形”,如图 5-24 所示。

(4) 拖动鼠标在舞台上画出一个五角星,将其设置在舞台中央,如图 5-25(a)所示。在“时间轴”面板中选择“图层_2”第 20 帧,右击,在弹出的快捷菜单中选择“插入空白关键帧”,选择工具箱“椭圆工具”,在舞台上画一个正圆,将其设置在舞台中央,如图 5-25(b)所示。

(5) 在“时间轴”面板中选择“图层_2”第 40 帧,右击,在弹出的快捷菜单中选择“插入空白关键帧”,选择工具箱“文本工具”,在属性窗口中设置字符系列为“华文琥珀”,字符大小为 120 点。单击舞台输入“宁波大学”文字,使用“对齐”面板,将其设置在舞台中央,如图 5-26 所示。

2. 创建遮罩动画和补间形状动画

(1) 如果“图层_2”第 60 帧没有插入任何帧,则选择它,右击,在弹出的快捷菜单中选择“插入帧”,如果已经存在普通帧,则不需要插入。

(2) 右击“图层_2”图层,在弹出的快捷菜单中选择“遮罩层”。

(3) 右击“图层_2”中的第 1 帧,在弹出的快捷菜单中选择“创建补间形状”。此时,时间轴如图 5-27 所示。

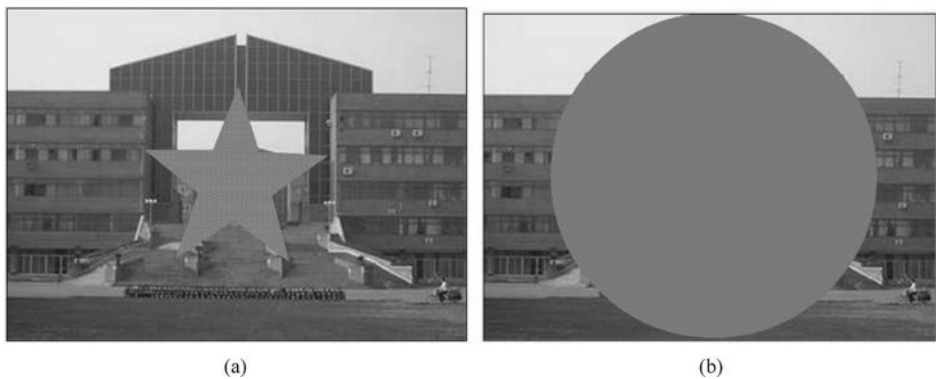


图 5-25 画五角星和圆



图 5-26 输入文字

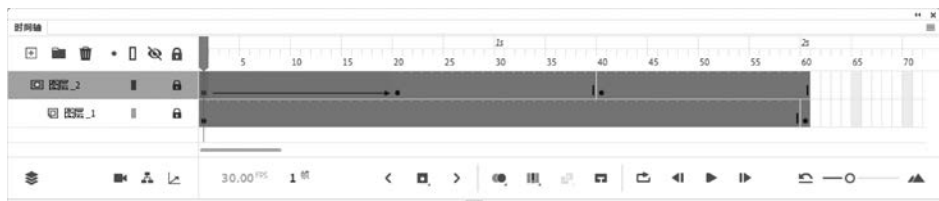


图 5-27 时间轴

(4) 分别单击“图层 2”第 1、10、20、40 帧,可以观察到如图 5-28 所示的结果。从第 1 帧到第 20 帧是形状逐渐变化的过程,有形状遮罩效果和文字遮罩效果等。

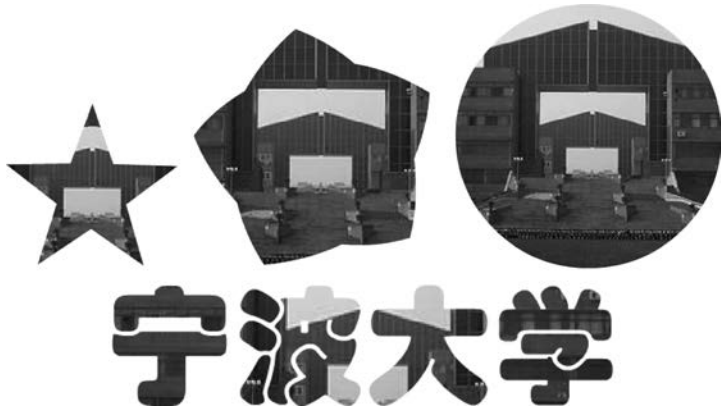


图 5-28 各形状遮罩效果

(5) 按 Ctrl+Enter 组合键测试影片,从五角星遮罩效果到圆形遮罩效果是一个形状渐变过程和文字遮罩效果。保存影片为“图片遮罩效果.fl”,并使用文件导出 GIF 动画文件,再观察动画效果。

5.4 案例 4 放大镜

【要求】

通过两张内容一样、大小不同的图片文件,制作“放大镜”阅读效果,最终效果如图 5-29 所示。



图 5-29 放大镜效果图

【知识点】

椭圆工具、帧复制、遮罩动画、传统补间动画

【操作步骤】

1. 导入图片和初画放大镜

(1) 新建 640×480px 大小的 An 文档“放大镜.fl”,选择“文件”|“导入”|“导入到舞台”,选择图片文件“古文书法图.jpg”和“放大图.jpg”一起导入舞台。右击图片,在弹出的快捷菜单中选择“分散到图层”,两图片分别在两个图层,修改图层_1 名为“放大镜镜片”,另两图层名去掉“.jpg”,如图 5-30 所示。

(2) 移动“放大图”图层到“古文书法图”上方。隐藏“放大图”图层,使用“对齐”面板将“古文书法图”图层中的对象垂直、水平对齐(相对于舞台对齐)。

(3) 右击“放大镜镜片”图层,在弹出的快捷菜单中选择“锁定其他图层”,光标定位到时间轴第 1 帧,选择“椭圆工具”,设置笔触颜色为黑色,笔触大小为 3,填充颜色为红色,按住 Shift 键在图片文字上方绘制一个正圆。单击选择工具,双击选中正圆全部(包括笔触和圆),“属性”面板中设置宽度和高度均为 150px。

(4) 使用“选择工具”将正圆移动到舞台右上合适位置(顶部与舞台靠齐,最后两列字在正圆差不多中间位置);使用“线条工具”在圆的一侧绘制线条作为放大镜手柄,上部分笔触为 3,下部分笔触为 7,笔触颜色为黑色,如图 5-31 所示。如果此时处于“绘制对象”模式,则切切换回场景 1。

2. 处理放大镜和制作传统补间动画

(1) 单击“古文书法图”图层时间轴第 20 帧,按 F5 键插入普通帧,分别单击“放大图”和“放大镜镜片”图层时间轴第 20 帧,按 F6 键插入关键帧。单击选择“放大镜镜片”图层时间轴第 20 帧,此时镜片和镜框都处于选中状态,用键盘方向键 ↓ 移动放大镜到舞台下方,如



图 5-30 导入图片并分散到图层



图 5-31 初画放大镜

图 5-32 所示。

(2) 右击“放大镜片”图层，在弹出的快捷菜单中选择“复制图层”，修改新图层名为“镜框”。

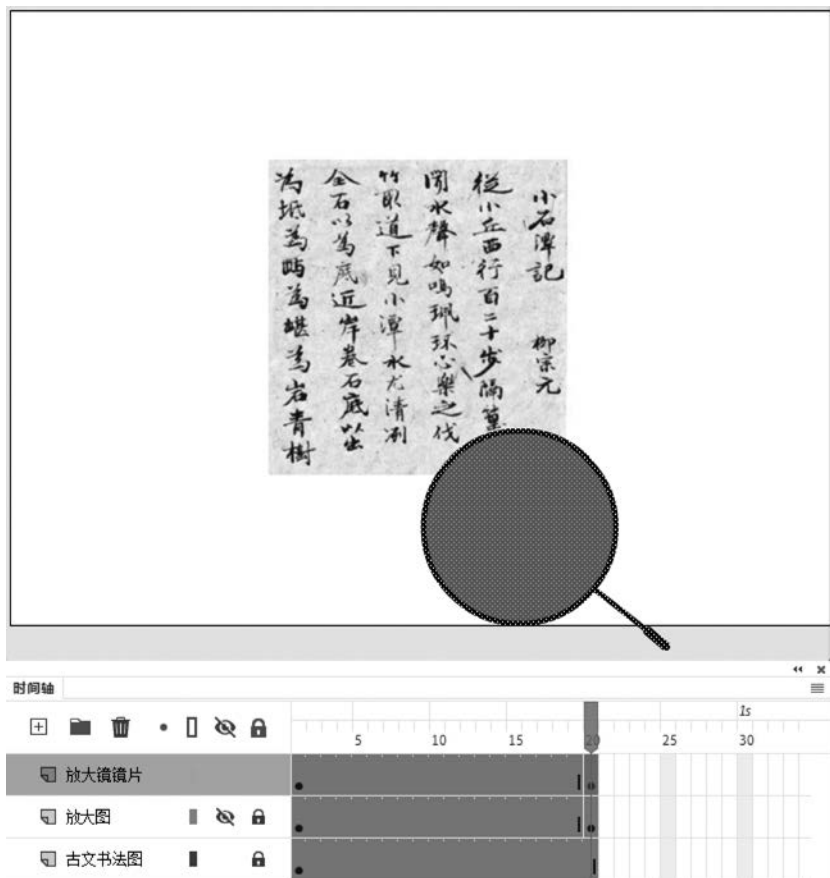


图 5-32 移动放大镜

(3) 锁定并隐藏“放大镜镜片”图层以外的其他图层,使用“选择工具”、删除键将第 1 帧与第 20 帧中的镜框及把手(所有黑色线条)清除,只剩下红圆镜片。

(4) 锁定并隐藏“镜框”图层以外的其他图层,使用“选择工具”、删除键将第 1 帧与第 20 帧镜片(所有红色填充)清除。

(5) 分别选择前 3 个图层,右击第 1~20 帧中的任意帧,在弹出的快速菜单中选择“创建传统补间”命令。

3. 制作遮罩动画和调整放大图位置

(1) 显示并锁定所有图层,右击“放大镜镜片”图层,在弹出的快捷菜单中选择“遮罩层”,第 1 帧显示效果如图 5-33 所示,此时遮罩显示的内容和放大镜所在位置不一致,接下来需要调整“放大图”图层中的图片位置。

(2) 隐藏“放大镜镜片”图层和“放大图”图层,播放头放在第 1 帧上,注意观察放大镜圆心所在的位置,如图 5-34 所示,调整放大图位置以此为基准。显示并解锁“放大图”图层。用键盘方向键移动放大图,使放大镜的中心点与刚才基准点类似,如图 5-35 所示。

(3) 同样的方法,调整“放大图”图层第 20 帧图片对象的位置。播放头放在第 20 帧上,注意观察放大镜中心所在的位置,如图 5-36 所示。显示并解锁“放大图”图层。用键盘方向键移动放大图,使放大镜的中心点与刚才基准点类似,如图 5-37 所示。



图 5-33 遮罩初始效果



图 5-34 第 1 帧原图与放大镜位置

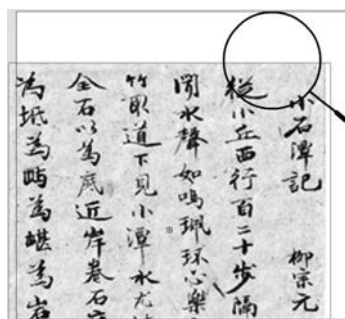


图 5-35 第 1 帧放大图位置

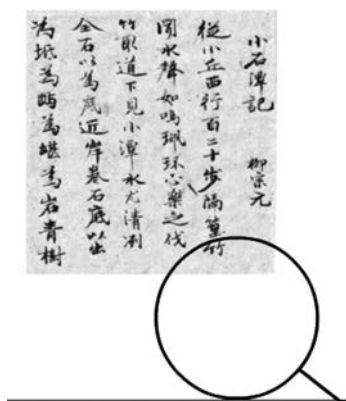


图 5-36 第 20 帧原图与放大镜位置

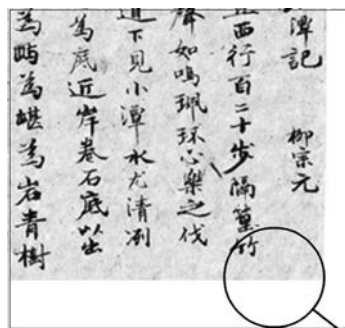


图 5-37 第 20 帧放大图位置

(4) 此时“时间轴”面板如图 5-38 所示,按 Ctrl+Enter 组合键可以观测影片播放效果,调低帧速率到 10fps 再观察效果。到目前为止右边两列放大效果设置完毕。



图 5-38 最后两列设置完成时间轴

4. 处理左边 1、2 列和 3、4 列放大效果

(1) 锁定所有图层。在“镜框”图层中,右击第 1 帧,在弹出的快捷菜单中选择“复制帧”;分别右击第 21、41 帧,在弹出的快捷菜单中选择“粘贴帧”。右击第 20 帧,在弹出的快捷菜单中选择“复制帧”;分别右击第 40、60 帧,在弹出的快捷菜单中选择“粘贴帧”。

(2) 同上方法处理“放大镜镜片”图层和“放大图”图层。“古文书法图”图层第 60 帧插入普通帧。

(3) 隐藏“放大图”图层,显示并解锁“放大镜镜片”图层和“镜框”图层,按 Ctrl 键同时选中两图层的第 21 帧,按键盘方向键←,同时移动放大镜镜片和镜框,使镜片中心点大概在第 3、4 列中间位置,如图 5-39 所示。同样方法处理两图层的第 40 帧,如图 5-40 所示。

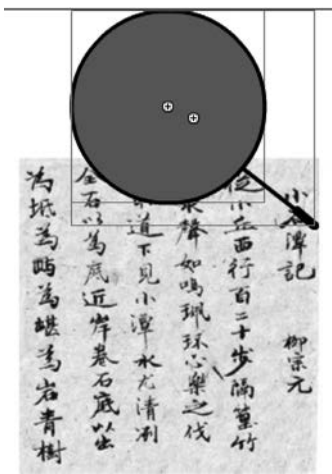


图 5-39 第 21 帧放大镜位置

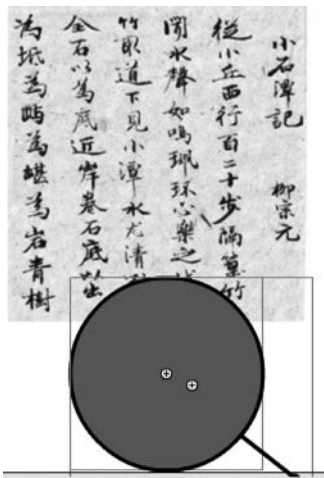


图 5-40 第 40 帧放大镜位置

(4) 隐藏“放大镜镜片”图层和“放大图”图层,播放头放在第 21 帧上,注意观察放大镜中心所在的位置,如图 5-41 所示,调整放大图位置以此为基准。显示并解锁“放大图”图层。用键盘方向键水平移动放大图,使放大镜的中心点与刚才基准点类似,如图 5-42 所示。同上方法处理第 40 帧。



图 5-41 第 21 帧原图与放大镜位置

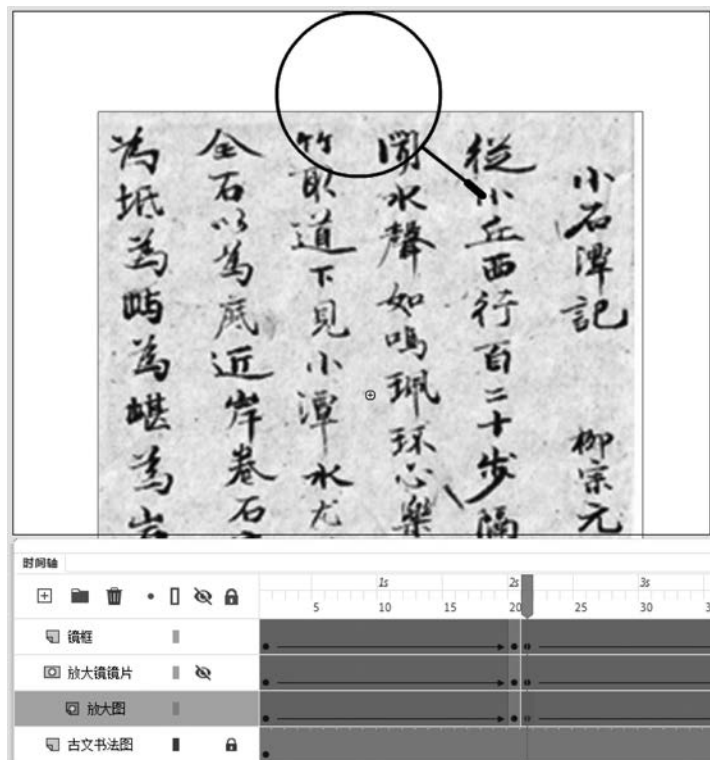


图 5-42 第 21 帧放大图位置

(5) 参照上面(3)、(4)步骤处理第 1、2 列(最左边两列)文字放大镜效果。测试并保存文档为“放大镜.flas”。

5.5 案例 5 红星闪闪

【要求】

制作“红星闪闪”动画：先制作五角红星，再制作闪光，最终效果如图 5-43 所示。

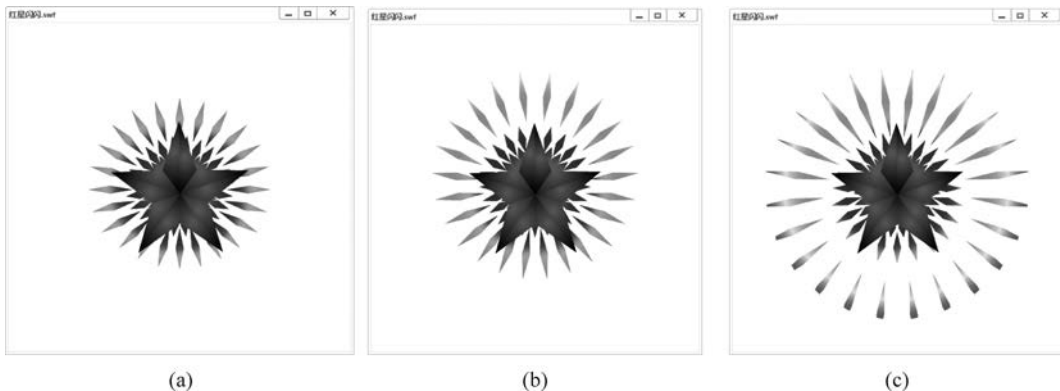


图 5-43 红星闪闪效果图

【知识点】

多角星形工具、颜料桶工具、变形、矩形工具、重置选区和变形、“样本”面板、“变形”面板、“对齐”面板、传统补间动画、遮罩层动画

【操作步骤】

1. 制作五角红星

(1) 新建一个 An 文档，大小设置为 $500 \times 500\text{px}$ ，帧速率设置为 12fps，背景颜色设置为白色。

(2) 单击“多角星形工具”，设置笔触颜色任意，填充为无，工具选项中样式设置为“星形”，边数为 5，在舞台中画一个水平五角星。用“选择工具”拖动全选五角星，设置其属性，宽和高均为 200px ，如图 5-44(a)所示。利用“对齐”面板将五角星水平垂直居中于舞台。

(3) 利用“线条工具”绘制多条线条，将五角星内部用线连接起来，如图 5-44(b)所示。

(4) 利用“颜料桶工具”分别给五角星各个区域填充颜色(如果不能分别填充，则需要将五角星分离)，填充色选择“样本”面板中左下角中默认颜色的第三个颜色(红色渐变)，如图 5-44(c)所示。

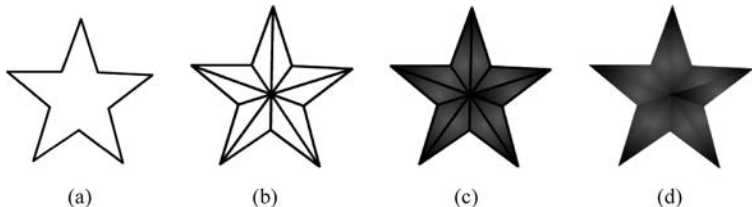


图 5-44 五角星制作

(5) 利用“选择工具”和删除键清除五角星所有线条,如图 5-44(d)所示。锁定并隐藏“图层_1”图层。

2. 制作闪光

(1) 新建“图层_2”图层,选择工具箱“矩形工具”,填充色为“样本”面板中左下角中默认颜色的最后一个颜色(彩色渐变),笔触为无。在舞台靠左中位置画一很细的矩形,选中矩形后,单击“任意变形工具”,此时如图 5-45(a)所示,将中间的注册点(空心圆点)拖动到右下角一点,如图 5-45(b)所示。



图 5-45 制作细长矩形闪光条

(2) 打开“变形”面板,设置旋转角度为 15° ,单击“重置选区和变形”按钮多次,复制完成细长矩形闪光条圆形排列,如图 5-46(a)所示。

(3) 拖动鼠标选中所有矩形,利用“对齐”面板将其水平、垂直居中于舞台(如果居中后发现细长矩形乱置,则在居中前可将其转换为元件)。

(4) 新建“图层_3”图层,右击“图层_2”第 1 帧,在弹出的快捷菜单中选择“复制帧”,右击“图层_3”第 1 帧,在弹出的快捷菜单中选择“粘贴帧”,选择“修改”|“变形”|“水平翻转”,此时效果如图 5-46(b)所示。

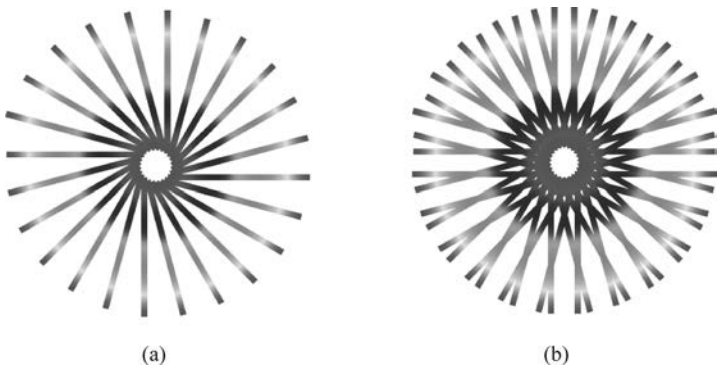


图 5-46 制作闪光

(5) 单击选择“图层_2”第 40 帧,按 F6 键插入一个关键帧;右击“图层_2”第 1 帧,在弹出的快捷菜单中选择“创建传统补间”,在其属性窗口中,将补间旋转设置为“逆时针 $\times 1$ ”。单击选择“图层_3”第 40 帧,按 F6 键插入一关键帧;右击“图层_3”第 1 帧,在弹出的快捷菜单中选择“创建传统补间”,在属性窗口中,将补间旋转设置为“顺时针 $\times 1$ ”。

(6) 右击“图层_1”第 40 帧,在弹出的快捷菜单中选择“插入帧”。拖动“图层_1”到最上层,并取消隐藏,显示“图层_1”图层。右击“图层_3”,在弹出的快捷菜单中选择“遮罩层”,此时时间轴如图 5-47 所示。

(7) 保存影片文件为“红星闪闪.flc”,按 Ctrl+Enter 组合键测试影片效果,因为每个同学制作的细长矩形宽度不一致,所以效果也会有所不同。

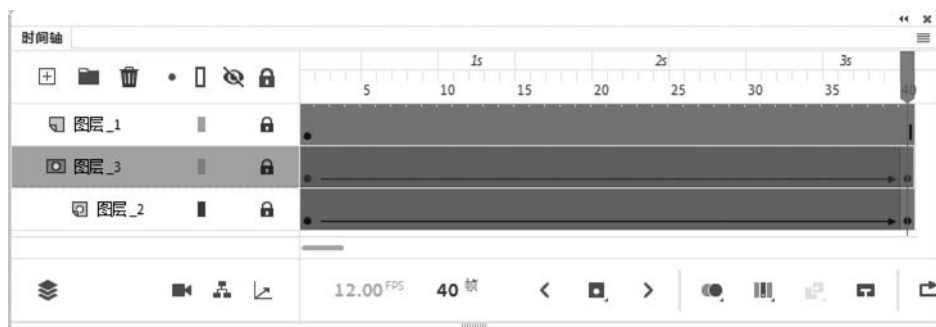


图 5-47 时间轴

5.6 案例 6 月球绕地球转

【要求】

已有“太空 1.jpg”“太空 2.jpg”“地球地图.jpg”“月球.png”图片文件,如图 5-48 所示。

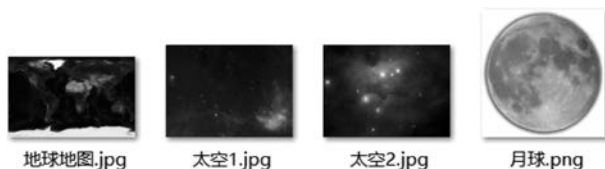


图 5-48 “月球绕地球转”素材

现要求制作月球绕地球转的动画,效果如图 5-49 所示,太空背景有渐变效果,地球可以自转,月球绕着地球转。



图 5-49 月球绕地球转效果

【知识点】

Alpha、补间动画、遮罩动画、引导动画

【操作步骤】

1. 制作太空渐变背景

(1) 新建 An 文档“月球绕地球转.fl a”,大小设置为 $550 \times 400\text{px}$,帧速率设置为 30fps。将图层_1 重命名为“太空渐变背景”,光标定位到时间轴第 1 帧,选择“文件”|“导入”|“导入

到舞台”,选择图片文件“太空 1.jpg”,出现“此文件看起来是图像序列的组成部分。是否导入序列中的所有图像”,选择“否”按钮,将其导入舞台,在图片对象“属性”面板中设置位置 X 和 Y 均为 0。

(2) 右击时间轴第 1 帧,在弹出的快捷菜单中选择“创建补间动画”,弹出“将所选的内容转换为元件以进行补间”,单击“确定”按钮。

(3) 单击时间轴第 30 帧,按 F6 键插入关键帧。单击舞台中“太空 1”图片,在“属性”面板中设置色彩效果样式 Alpha 值为 50%。

(4) 右击时间轴第 31 帧,在弹出的快捷菜单中选择“插入空白关键帧”,参考上面步骤,第 31 帧导入“太空 2”图片,在图片“属性”面板中设置位置 X 和 Y 均为 0。

(5) 右击第 31 帧创建补间动画,光标指向第 31 帧后,待出现双向箭头时,拖动延长到第 60 帧。在第 60 帧处按 F6 键插入关键帧,单击“太空 2”图片设置色彩效果样式,将 Alpha 值设置为 50%。锁定“太空渐变背景”图层。

2. 创建引导层等图层

(1) 新建图层,将其命名为“月球”,第 1 帧导入“月球.png”图片,利用“变形”面板将图片缩小为原来的 15%,移动月球到舞台左下角。

(2) 右击“月球”图层,在弹出的快捷菜单中选择“添加传统运动引导层”,上方即创建了一个“引导层:月球”图层。

(3) 新建两个图层:“地球地图”和“遮罩层”图层,此时时间轴和图片第 1 帧效果如图 5-50 所示。

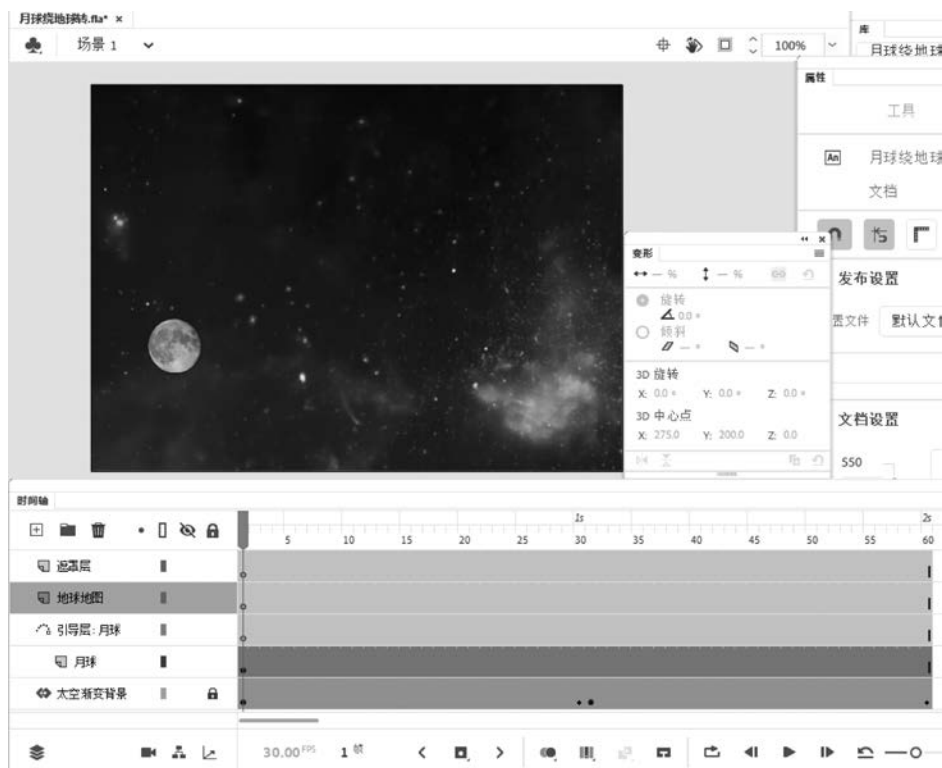


图 5-50 插入月球、新建图层后效果

3. 实现月球运动

(1) 单击“遮罩层”图层第 1 帧,在工具箱中选择“椭圆工具”,笔触为无,填充为红色,结合 Shift 键,在舞台上画一正圆,在“属性”面板中设置宽和高为 150px,使用“对齐”面板“水平中齐”和“垂直中齐”将正圆位于舞台正中央。

(2) 单击“引导层:月球”图层第 1 帧,在工具箱中选择“椭圆工具”,笔触颜色为红色,笔触大小为 3,填充为无,在舞台上画一比“遮罩”图层正圆大的椭圆,选择“任意变形工具”调整大小和位置,如图 5-51 所示。

(3) 选择“橡皮擦工具”在椭圆左下角月球附近擦除一段,使椭圆留出一个缺口,如图 5-52 所示。

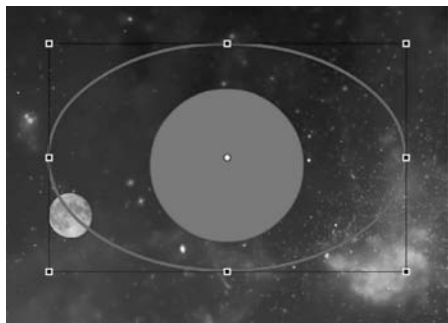


图 5-51 椭圆引导线

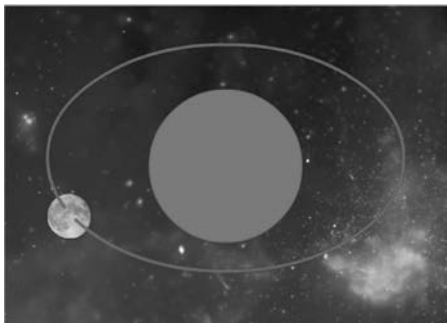


图 5-52 有缺口的椭圆引导线

(4) 在“月球”图层第 60 帧处插入关键帧。单击“月球”图层第 1 帧,用“选择工具”拖动月球到椭圆缺口的下端,使中心点与椭圆引导线正好贴合,如图 5-53 所示。单击“月球”图层第 60 帧,拖动月球到椭圆缺口的上端,使中心点与椭圆引导线正好贴合,如图 5-54 所示。

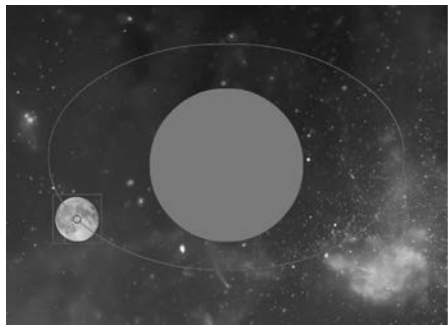


图 5-53 月球与引导线下端对准

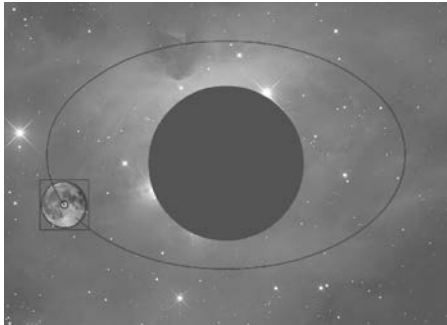


图 5-54 月球与引导线上端对准

(5) 右击“月球”图层第 1 帧,在弹出的快捷菜单中选择“创建传统补间”,此时测试影片,月球已经可以绕着中间正圆运动了。锁定“月球”图层和“引导层:月球”图层。

4. 地球自转效果

(1) 单击“地球地图”图层第 1 帧,导入“地球地图.jpg”图片,在“属性”面板中,调整高度为 150px,宽度不变。移动地球地图覆盖“遮罩”图层红色正圆,使其右端与“遮罩”图层正圆右端靠齐,如图 5-55 所示。

(2) 右击“地球地图”图层第 1 帧,在弹出的快捷菜单中选择“创建补间动画”,弹出“将

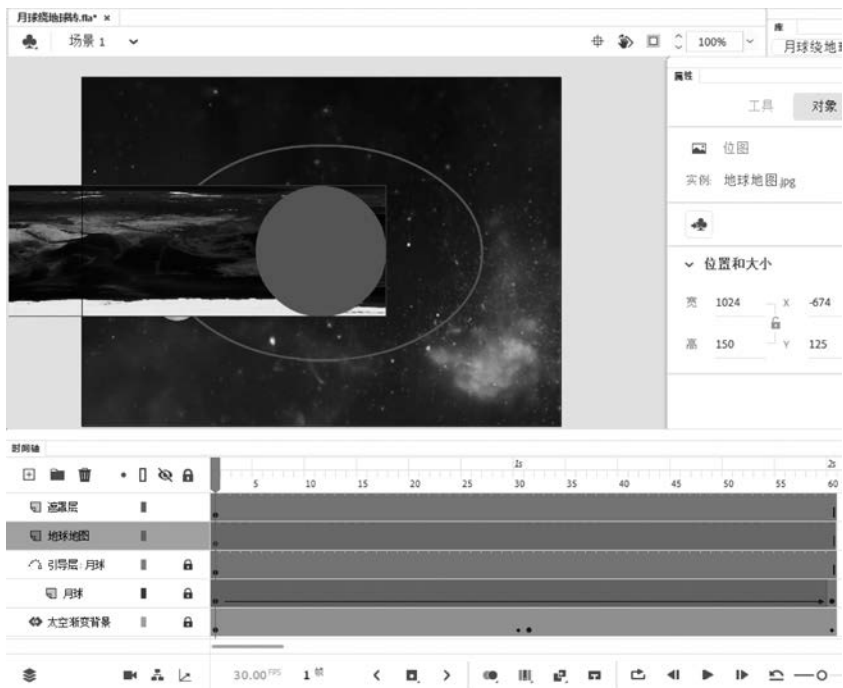


图 5-55 第 1 帧地图与正圆右侧对准

所选的内容转换为元件以进行补间”，单击“确定”按钮。右击“地球地图”图层第 60 帧，在弹出的快捷菜单中选择“插入关键帧”|“位置”，插入属性关键帧。使用键盘→方向键移动地球地图直至左端与正圆对齐，如图 5-56 所示。

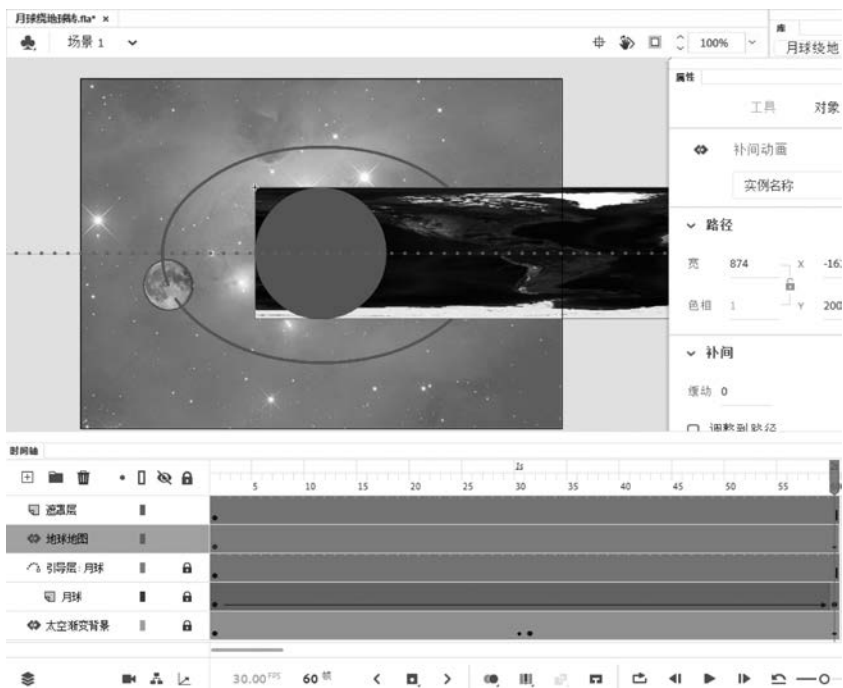


图 5-56 第 60 帧地图与正圆左侧对准

(3) 右击“遮罩层”图层,在弹出的快捷菜单中选择“遮罩层”,隐藏引导层,定位“时间轴”面板的播放头到第 31 帧,显示的舞台效果和时间轴如图 5-57 所示。按 Ctrl+Enter 组合键测试影片,保存文档。

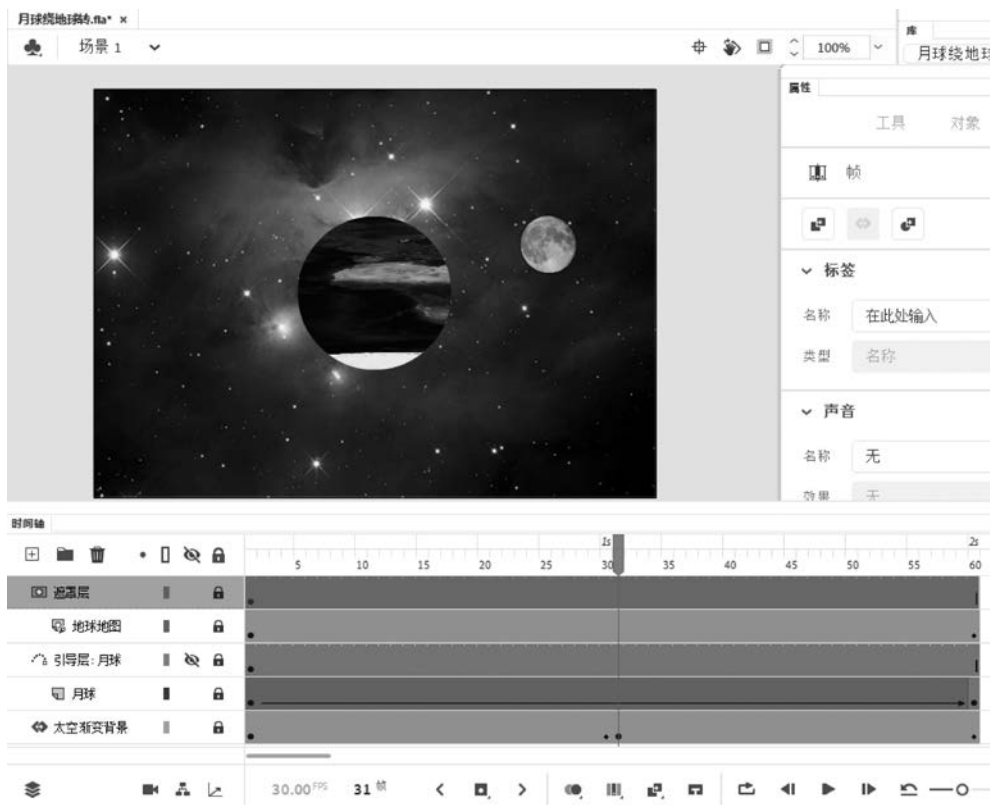


图 5-57 第 31 帧效果

5.7 案例 7 兔子跑步

【要求】 已有“兔子 1. png”~“兔子 8. png”和“跑道.jpg”图片文件,如图 5-58 所示。

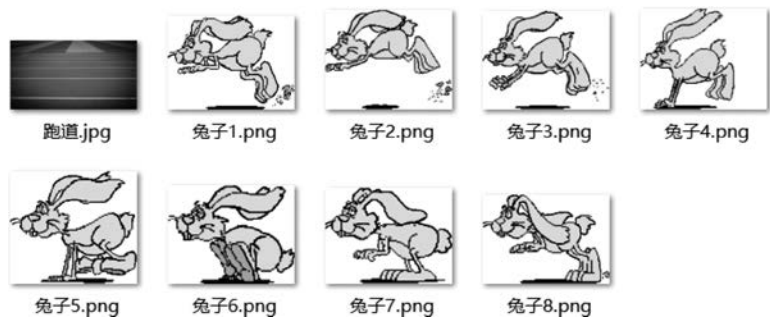


图 5-58 “兔子跑步”素材

现要求制作“兔子跑步”动画:将“兔子 1. png”~“兔子 8. png”制作成“兔子奔跑”元件,使用各种动画类型制作兔子从右跑到左,再从上跑到下,然后从左跑向右的效果,最终效果

如图 5-59 所示。



图 5-59 “兔子跑步”效果图

【知识点】

垂直翻转、元件、钢笔工具、补间动画、传统补间动画、引导动画

【操作步骤】

1. 制作“兔子奔跑”元件

(1) 新建一个 An 文档,舞台宽高设置为 $640 \times 480\text{px}$,帧速率设置为 24fps。执行菜单“文件”|“导入”|“导入到舞台”,打开“导入”对话框,选择“兔子奔跑”所有素材文件。

(2) 单击“打开”按钮后,9 个图片文件都以左上角对齐方式重叠地列于舞台中,并且全部已经选中,光标移到舞台外单击鼠标,取消所有选择。拖动“跑道”图片到旁边,使“跑道”图片和其他图片分开一段距离。

(3) 拖动鼠标选中所有兔子图片(不要移动兔子图片),右击,在弹出的快捷菜单中选择“转换为元件”,弹出“转换为元件”对话框,设置名称为“兔子奔跑”,类型为“影片剪辑”,单击“确定”按钮。删除舞台中选中的兔子图片。

(4) 拖动“跑道”图片到舞台左上角对齐,在“属性”面板中设置图片宽高为舞台大小 $640 \times 480\text{px}$,设置位置 X 和 Y 均为 0,使图片恰好覆盖舞台。

(5) 右击“库”面板中的“兔子奔跑”元件,在弹出的快捷菜单中选择“编辑”,进入元件编辑界面,此时所有兔子图还是选中状态的,右击,在弹出的快捷菜单中选择“分散到图层”,删除“图层_1”图层。

(6) 单击“时间轴”中“兔子 2. png”图层的第 1 帧,光标指向第 1 帧,按住鼠标左键,当光标图标显示为矩形框时,拖动第 1 帧到第 2 帧;同样操作,将“兔子 3. png”图层的第 1 帧拖动到第 3 帧;将“兔子 4. png”图层的第 1 帧拖动到第 4 帧;将“兔子 5. png”图层的第 1 帧拖动到第 5 帧;将“兔子 6. png”图层的第 1 帧拖动到第 6 帧;将“兔子 7. png”图层的第 1 帧拖动到第 7 帧;将“兔子 8. png”图层的第 1 帧拖动到第 8 帧。此时,“兔子奔跑”元件制作完毕,如图 5-60 所示。

2. 创建补间动画

(1) 返回“场景 1”舞台。新建“图层_2”图层,拖动“兔子奔跑”元件到舞台右上边适当位置,并用变形工具将其缩小到适当大小。单击“图层_1”第 90 帧,按 F5 键插入帧。右击“图层_2”第 1 帧,在出现的快捷菜单中选择“创建补间动画”,光标指向第 24 帧,拖动到第 40 帧,将补间动画延长到 40 帧。此时当前帧应该是第 40 帧,拖动兔子移动到舞台最左边,如图 5-61 所示,自动在舞台上生成了运动轨迹。

(2) 单击“图层_2”第 41 帧,按 F6 键插入关键帧。右击兔子,在弹出的快捷菜单中选择

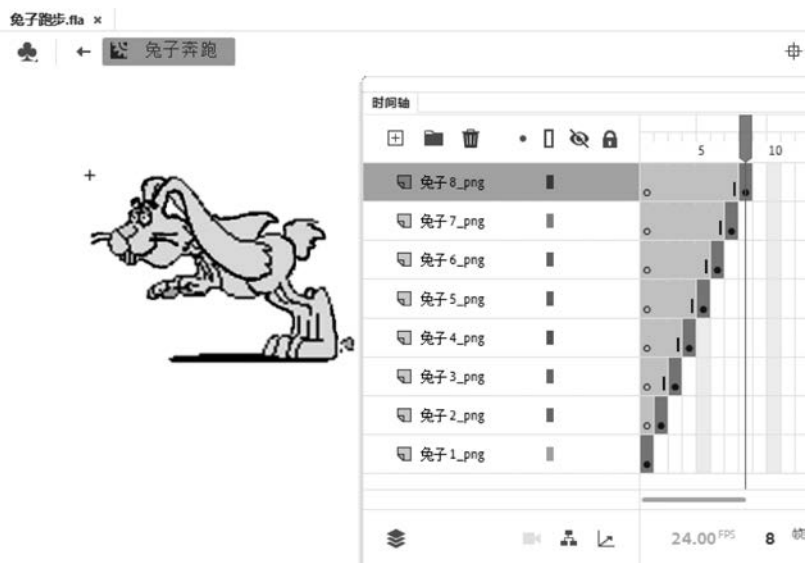


图 5-60 “兔子奔跑”元件制作

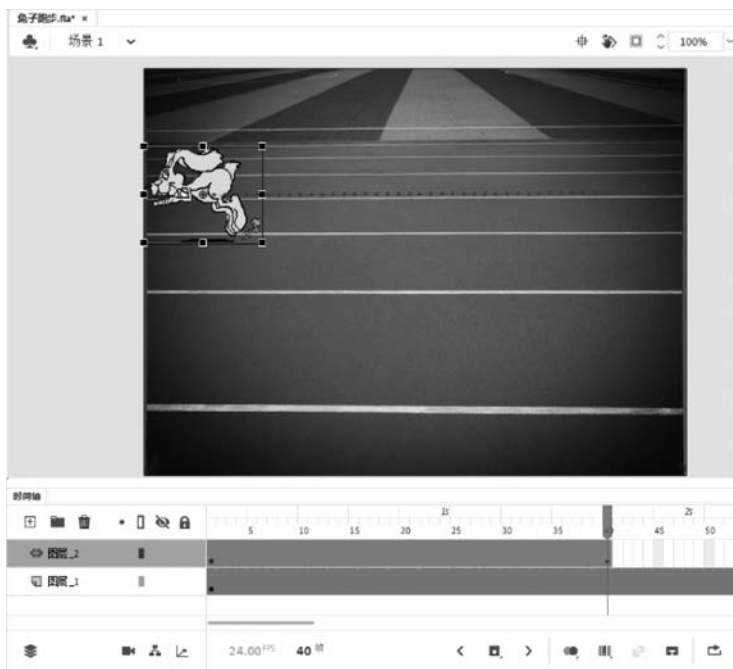


图 5-61 创建补间动画

“变形”|“逆时针旋转 90 度”，设置完成后，如图 5-62 所示，兔子也跟着旋转了。

(3) 单击“图层_2”第 50 帧，按 F6 键插入关键帧。往下拖动兔子，使兔子处于两根白线之间离我们最近的跑道中(接下来返回舞台右边的操作也可以使用类似的方法完成，下面将使用另一种方法来完成)。

3. 创建引导动画

(1) 新建“图层_3”图层；单击“图层_2”第 50 帧，右击舞台中的兔子，在出现的快捷菜单

中选择“复制”；锁定“图层_2”和“图层_1”，在“图层_3”第 51 帧中插入空白关键帧，右击舞台，在出现的快捷菜单中选择“粘贴到当前位置”。

(2) 单击“图层_3”第 51 帧，选择“修改”|“变形”|“垂直翻转”，再选择“修改”|“变形”|“顺时针旋转 90 度”，会感觉到兔子从左向右跑步了。单击“图层_3”第 90 帧，按 F6 键插入关键帧，右击“图层_3”第 51~90 任意一帧，在出现的快捷菜单中选择“创建传统补间”。

(3) 右击“图层_3”图层，在弹出的快捷菜单中选择“添加传统运动引导层”。在“图层_3”上方就出现了“引导层：图层_3”图层。

(4) 在“引导层：图层_3”第 51 帧插入空白关键帧，选择“钢笔工具”，单击兔子图片中心一点，再在跑道右边单击一点，即画了一条直线，这条直线就是引导线，如图 5-63 所示。



图 5-62 逆时针旋转 90°

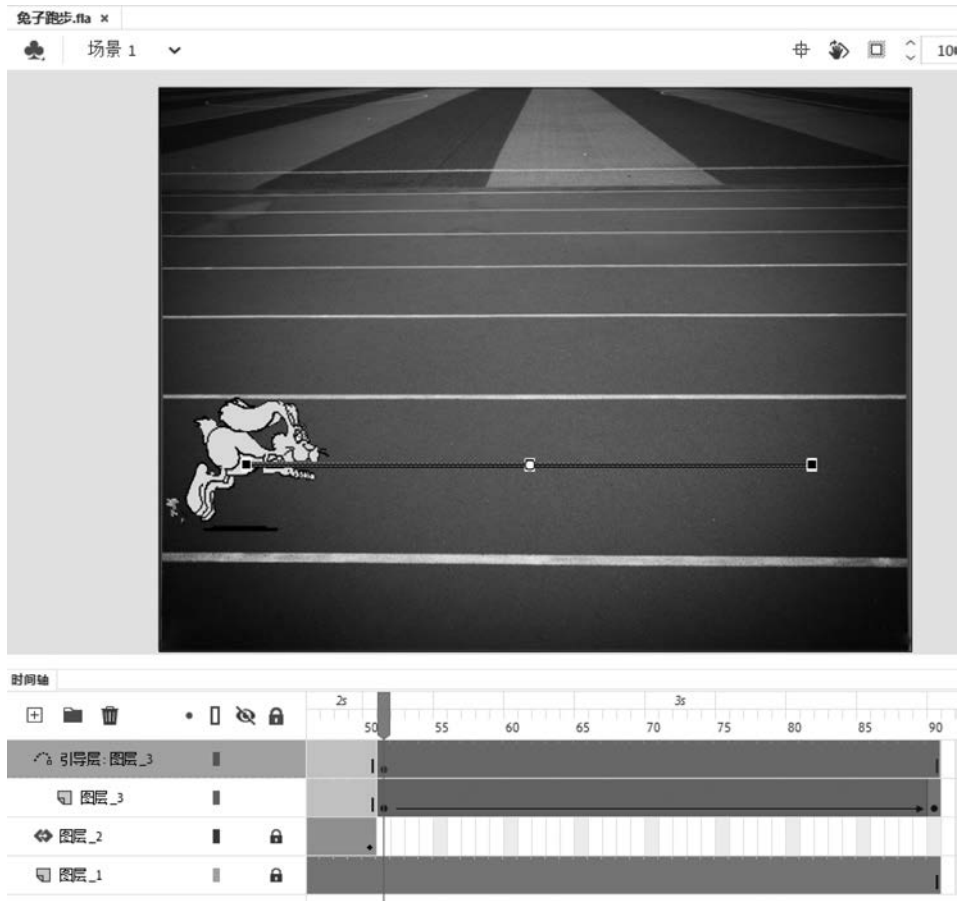


图 5-63 制作引导层

(5) 单击“选择工具”，单击“图层_3”第 51 帧，拖动兔子图片正好从引导线左端开始，如

图 5-64(a)所示;单击“图层_3”第 90 帧,拖动兔子图片正好在引导线右端结束,如图 5-64(b)所示。

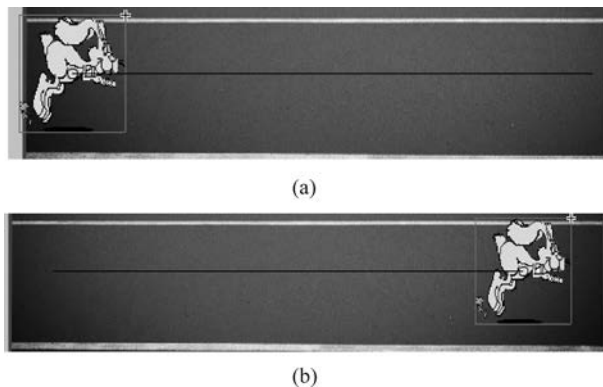


图 5-64 引导动画

(6) 按 Ctrl+Enter 组合键测试影片,此时兔子在跑道上跑的是三段直线。保存影片为“兔子跑步.flas”。

(7) 解锁所有图层,单击时间轴上不同的补间区域,用“选择工具”调整运动路径使原来的直线变为曲线,如图 5-65 所示,再测试影片,观察效果,另存影片为“兔子跑步(弯线).flas”。

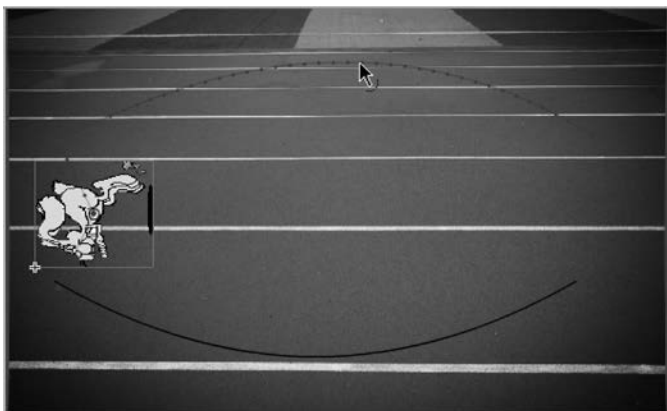


图 5-65 运动路径调整为曲线

5.8 案例 8 移动的球

【要求】

制作“移动的球”动画:先画三个椭圆,然后画三个球,实现球在各自的椭圆轨道中移动的效果,最终效果如图 5-66 所示。

【知识点】

椭圆工具、投影效果、发光效果、色彩效果、元件、引导动画

【操作步骤】

(1) 创建 An 新文档,将“图层_1”改名为“背景 1”,用“椭圆工具”绘制一个无填充色、笔

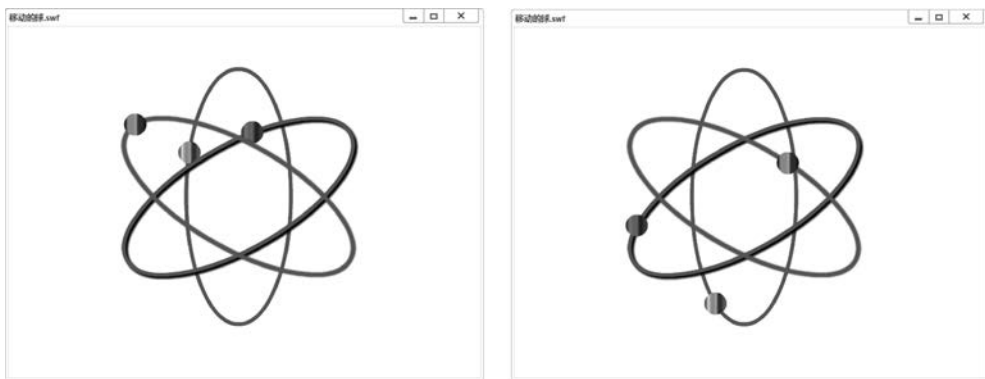


图 5-66 移动的球效果

触颜色为红色、笔触为 5 的长椭圆,使用“对齐”面板使椭圆居出于舞台。复制“背景 1”图层,改名为“引导层 1”。选中“背景 1”图层,新建一个“球 1”图层,在该层绘制一个无笔触色的正圆,填充为任意渐变色。

(2) 右击“引导层 1”图层,在弹出的快捷菜单中选择“引导层”,此时引导层 1 为  引导层 1,拖动“球 1”图层到“引导层 1”图层下方,使引导层 1 变成  引导层 1,表示引导设置成功。

(3) 右击“引导层 1”图层,在弹出的快捷菜单中选择“隐藏其他图层”,只显示“引导层 1”图层,用橡皮擦在椭圆上拖动擦去一小部分。只显示“球 1”图层,选中并右击球,在弹出的快捷菜单中选择“转换为元件”,转换成类型为“影片剪辑”的“球”元件。只显示“背景 1”图层,选中并右击椭圆,在弹出的快捷菜单中选择“转换为元件”,转换成类型为“影片剪辑”的“椭圆”元件。转换成元件的目的是可以设置滤镜等效果。

(4) 按 Ctrl 键并单击一起选中“引导层 1”“球 1”“背景 1”图层,右击选中图层,在弹出的快捷菜单中选择“复制图层”;将新复制的图层分别命名为“引导层 2”“球 2”“背景 2”。再复制“引导层 2”“球 2”“背景 2”,将新复制的图层分别命名为“引导层 3”“球 3”“背景 3”。

(5) 只显示“背景 2”图层,选中椭圆,用“变形”面板使其旋转 60°,同样设置“引导层 2”图层。只显示“背景 3”图层,选中椭圆,用“变形”面板使其旋转 120°,同样设置“引导层 3”图层。

(6) 只显示“球 1”和“引导层 1”图层,单击“球 1”第 50 帧,按 F6 键;单击“引导层 1”第 50 帧,按 F5 键;单击“背景 1”第 50 帧,按 F5 键。右击“球 1”层第 1~50 任意帧,选择“创建传统补间”。单击“球 1”第 1 帧,移动球到椭圆缺口的一端;单击“球 1”第 50 帧,移动球 1 到椭圆缺口的另一端。至此,球 1 能沿着椭圆顺利移动了。

(7) 只显示“背景 2”图层,用“选择工具”选中第 1 帧椭圆,在“属性”面板中设置“投影”滤镜效果。只显示“背景 3”图层,设置“发光”滤镜,具体自己设置。

(8) 设置“球 2”“球 3”色彩效果的色调、高级等样式,具体自己调整,几个球的颜色最好能很好区分。

(9) 参考第(6)步,使“球 2”“球 3”也能沿着椭圆移动。设计完时间轴和其中一帧效果如图 5-67 所示。

(10) 保存影片为“移动的球.flas”,测试影片效果。

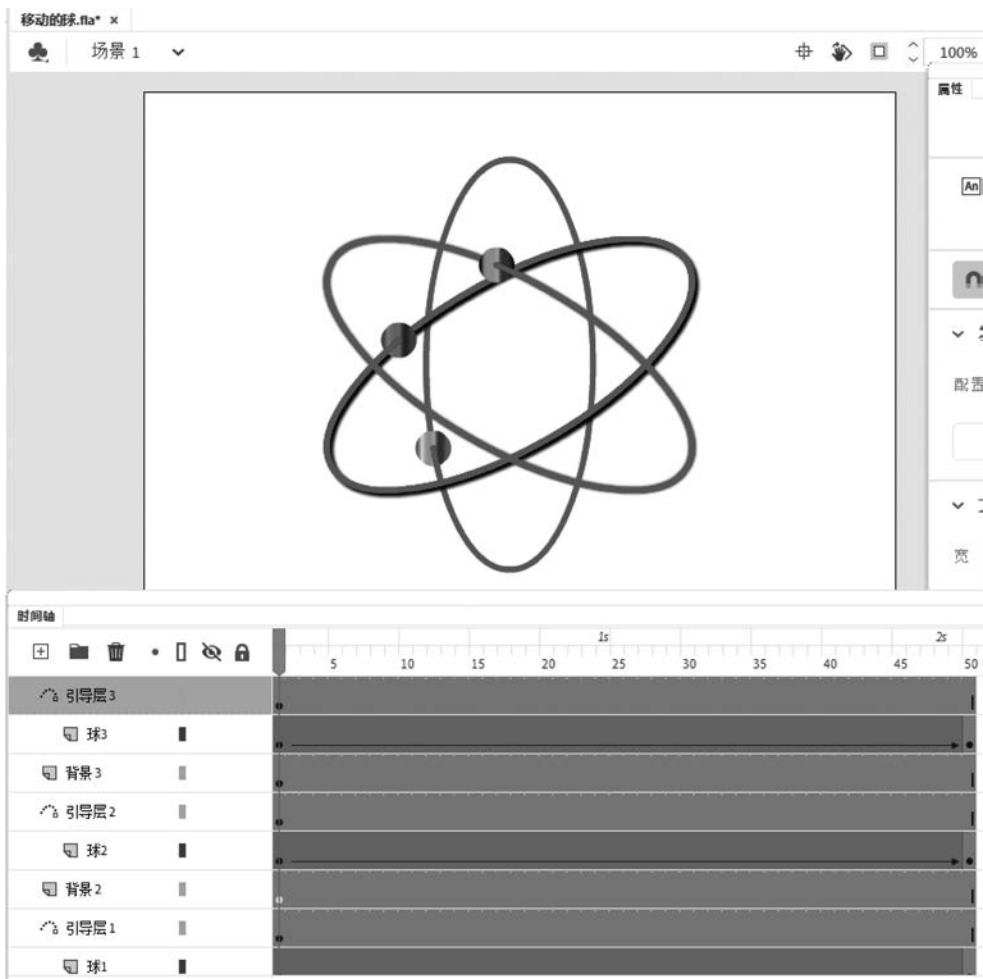


图 5-67 移动的球设计

5.9 案例 9 文字特效

【要求】

制作“文字特效”动画：环形文字“信息科学与工程学院欢迎你”，其中“信息科学”“与工程学院”“欢迎你”三组文字各使用一种动画；除了“欢迎你”的其他文字依次出现；文字先填上多彩色，再将多彩色去除显示；最终效果如图 5-68 所示。

【知识点】

分散到图层、重置选区和变形、文本工具、复制动画、预设动画

【操作步骤】

(1) 新建一个 An 文档，单击“文本工具”，设置“华文彩云、70 点”，在舞台上部中部位置输入一个“信”。单击“任意变形工具”，拖动文字中心注册点到舞台中央位置附近。

(2) 打开“变形”面板，单击“重置选区和变形”1 次，此时看起来没什么反应；“变形”面板中设置旋转角度为 30°后按 Enter 键，舞台上会出现另一个旋转之后的字，再单击“重置选



图 5-68 文字特效效果图

区和变形”10次。单击“文本工具”，光标选中要修改的文字，然后修改文字，原来所有字都是“信”，改成“信息科学与工程学院欢迎你”，如图 5-69 所示，用“选择工具”拖动选中所有文字，利用键盘方向键将文字圆尽量调整到舞台中央位置。

(3) 复制图层“图层_1”后，锁定“图层_1”图层。单击“图层_1_复制”图层，所有文字选中，右击选中的文字，在弹出的快捷菜单中选择“分散到图层”，删除“图层_1_复制”图层。按 Ctrl+A 组合键，选中所有文字，按 Ctrl+B 组合键分离文字。

(4) 使用“颜料桶工具”，在“属性”面板中设置填充色为多彩色，将文字填充成适合的颜色，如图 5-70 所示。



图 5-69 圆形文字



图 5-70 文字填充后

(5) 单击“信”图层，选择“窗口”|“动画预设”，在弹出的“动画预设”面板中选择“默认预设”中的“2D 放大”，如图 5-71 所示，单击“应用”按钮。右击“信”图层已创建的任意帧，在弹出的快捷菜单中选择“复制动画”；分别右击“息”“科”“学”图层第 1 帧，在弹出的快捷菜单中选择“粘贴动画”。这样“信”“息”“科”“学”文字都应用了“2D 放大”动画。

(6) 将“与”“工”“程”“学”“院”文字都应用“脉搏”动画。将“欢”“迎”“你”文字都应用“3D 螺旋”动画。

(7) 单击“息”图层，将光标指向图层第 1 帧，拖动到第 2 帧，这样这一层动画从第 2 帧开始了。单击“科”图层，光标指向图层第 1 帧，拖动鼠标到第 3 帧，这样这一



图 5-71 动画预设选择

层动画从第 3 帧开始了,同样处理“学”“与”“工”“程”“学”“院”图层,使得各图层动画可依次出现。

(8) 分别选中所有图层第 60 帧,按 F6 键。移动“图层_1”第 1 帧到第 60 帧,此时该图层第 1 帧为空白帧。单击第 70 帧,按 F6 键。时间轴设计如图 5-72 所示。

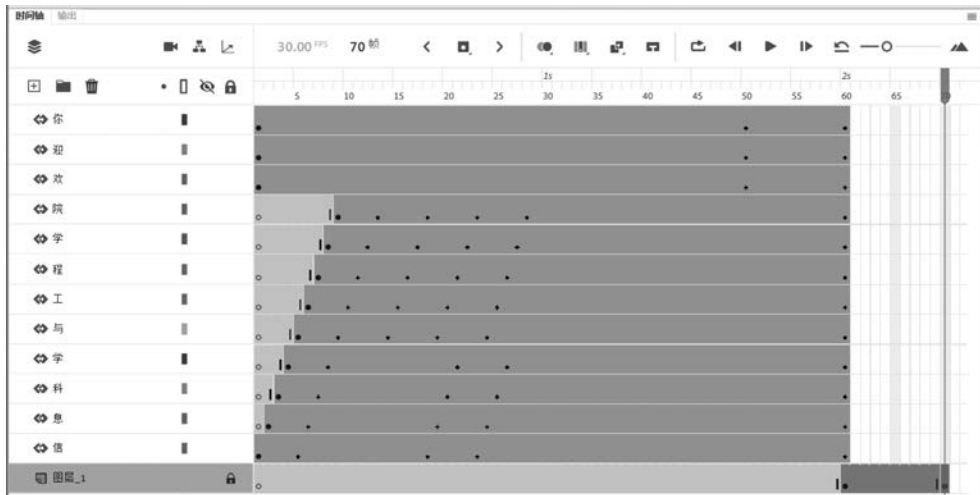


图 5-72 文字特效时间轴设计

(9) 保存影片为“文字特效.flc”,测试影片效果。

5.10 案例 10 拼图游戏

【要求】

An 时间轴上的帧指针默认是按顺序一帧一帧地往前走,也就是按序进行播放。如果需要改变帧指针的播放顺序,就必须在关键帧上添加必要的脚本代码,这种脚本称为帧动作。

要求制作“拼图游戏”动画:从网上搜索下载一幅风景图,分割成 9 份子图片,散落在外,然后将其拼成原图样。效果说明如下。

(1) 拼图初始状态,如图 5-73 所示,右上角显示图片原图,可用来拼图参照。

(2) 拼图过程状态,如图 5-74 所示,左上角区域为拼图区域,可拖动子图片到此区域,如果拼的位置正确,则留在拼图区域,不正确则返回原位置。

(3) 拼图成功状态,如图 5-75 表示,出现“拼图成功!”字样。

(4) 单击“再来”按钮,恢复到拼图初始状态,可重新开始拼图。

【知识点】

矩形工具、转换为元件、影片剪辑元件、按钮元件、ActionScript 脚本语言高级动画

【操作步骤】

1. Photoshop 中分割图片

(1) 网上搜索下载一幅风景图,用 Photoshop 打开该图,使用裁剪工具,工具选项宽度设置为 600px、高度设置为 450px,裁剪图片,存储为“风景原图.jpg”。新建垂直参考线

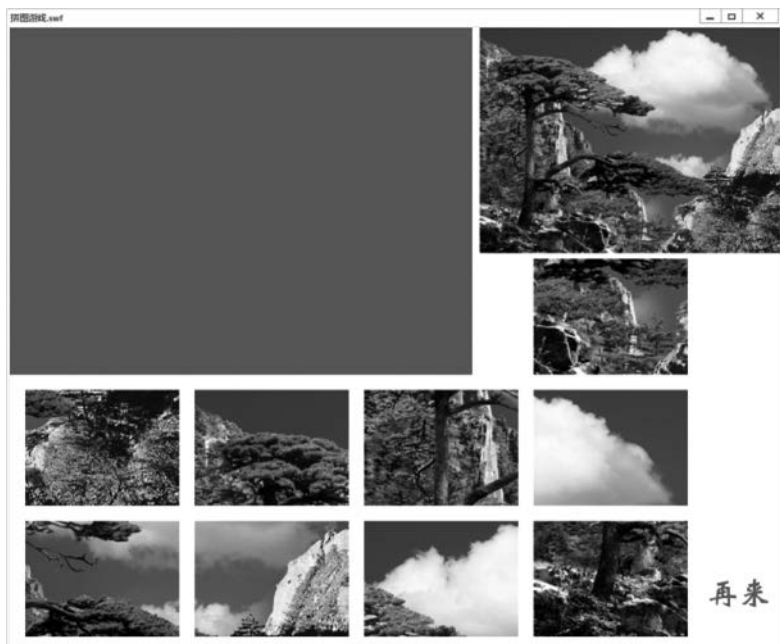


图 5-73 拼图初始状态



图 5-74 拼图过程中

200px、400px,水平参考线 150px、300px。

(2) 打开 Photoshop 工具箱的“切片工具”,在工具选项中单击“基于参考线的切片”按钮,将图片切成 9 份,如图 5-76 所示。

(3) 选择“文件”|“导出”|“存储为 Web 所用格式”,弹出“存储为 Web 所用格式”对话框,默认设置,单击“存储”按钮,弹出“将优化结果存储为”对话框,指定图片保存位置,将文件命名为 p.gif,单击“保存”按钮。在图片保存文件夹中增加了 images 子文件夹,打开后,



图 5-75 拼图成功效果



图 5-76 切片效果

里面已经保存有 9 幅分割完成的图片,如图 5-77 所示。

(4) 不保存“风景原图.jpg”图片,退出 Photoshop。

2. 创建拼图区域

(1) 打开 An 应用程序,新建 ActionScript 3.0 文档,文档大小设置为 $1000 \times 800\text{px}$,保存为“拼图游戏.flr”。

(2) 将“风景原图.jpg”和 images 子文件夹中的 9 幅图片导入库中。

(3) 选择工具箱“矩形工具”,在“属性”面板“矩形工具”下方选中“对象绘制模式”选项



图 5-77 分割后图片

在舞台上画一无笔触任意填充色的矩形,在“属性”面板中设置宽为 200px、高为 150px。右击矩形,在弹出的快捷菜单中选择“转换为元件”,弹出“转换为元件”对话框,名称为“j”,类型为“影片剪辑”,单击“确定”按钮。

(4) 先复制矩形 2 个,用“对齐”面板对齐,使其并列显示在左上角;选中第一排 3 个矩形,复制到第 2、3 排,移动调整矩形使其形成规整的拼图区域。

(5) 分别单击各矩形,在“属性”面板中设置实例名称分别为 j1、j2、j3(第一排)、j4、j5、j6(第 2 排)、j7、j8、j9(第 3 排)。请注意实例名称中字母均为小写,以下同。

3. 其他界面设计

(1) 拖动库中的“风景原图.jpg”图片到舞台右上角,使用“变形”面板将图片等比例缩小至原来的 65%。

(2) 拖动库中的 p_01.gif~p_09.gif 到舞台合适位置,“风景原图”下方一个为 p_08.gif(位置 X: 680, Y: 300); 拼图区域下方一排为: p_09.gif(X: 20, Y: 470)、p_01.gif(X: 240, Y: 470)、p_04.gif(X: 460, Y: 470)、p_03.gif(X: 680, Y: 470); 舞台最后一排为: p_05.gif(X: 20, Y: 640)、p_06.gif(X: 240, Y: 640)、p_02.gif(X: 460, Y: 640)、p_07.gif(X: 680, Y: 640)。此时设计界面如图 5-78 所示。

(3) 将子图片 p_01.gif 转化为影片剪辑元件 p1,并在“属性”面板中设置实例名称为 p1。其他子图片也转换成相应影片剪辑元件 p2~p9,并在“属性”面板中设置实例名称分别为 p2~p9。

(4) 在舞台右边空隙处输入“拼图成功!”文字,转化为影片剪辑元件,并将实例名称改为 finishtext。在舞台右下角,新建并插入按钮元件,元件中输入文字“再来”,修改按钮元件实例名称为 againbutton。

4. 加入代码

(1) 新建一图层,命名为 AS。按 F9 键或者右击第 1 帧,在弹出的快捷菜单中选择“动作”,进入代码编辑状态。输入以下代码并调试测试。



图 5-78 拼图游戏设计界面 1

```

finishtext.visible = false; //拼图成功标记
var f1,f2,f3,f4,f5,f6,f7,f8,f9:Boolean = false ; //各子图片拼成功标记
p1.addEventListener(MouseEvent.CLICK,ClickToDrag1); //侦听 p1 中鼠标按下事件并处理
function ClickToDrag1(event:MouseEvent):void
{p1.startDrag();} //p1 保持可拖动状态
stage.addEventListener(MouseEvent.CLICK,ReleaseToDrop1); //侦听鼠标释放事件并处理
function ReleaseToDrop1(event:MouseEvent):void
{if(p1.hitTestObject(j1)) //检测两个对象是否发生碰撞
    { p1.x = j1.x; p1.y = j1.y; f1 = true; //p1 放置到矩形 j1 位置,并标记
      if(f1 && f2 && f3 && f4 && f5 && f6 && f7 && f8 && f9 )
        {finishtext.visible = true;} //如果全部放置完成,则显示
      }else {p1.x = 240;p1.y = 470;} // 没碰撞到,则拖动的对象回到原位置
    p1.stopDrag();} // p1 停止拖动
p2.addEventListener(MouseEvent.CLICK,ClickToDrag2); //侦听 p2 中鼠标按下事件并处理
function ClickToDrag2(event:MouseEvent):void
{ p2.startDrag();}
stage.addEventListener(MouseEvent.CLICK,ReleaseToDrop2);
function ReleaseToDrop2(event:MouseEvent):void
{ if(p2.hitTestObject(j2))
    { p2.x = j2.x; p2.y = j2.y; f2 = true;
      if(f1 && f2 && f3 && f4 && f5 && f6 && f7 && f8 && f9 )
        {finishtext.visible = true;}
      }else {p2.x = 460; p2.y = 640; }
    p2.stopDrag();}
p3.addEventListener(MouseEvent.CLICK,ClickToDrag3); //侦听 p3 中鼠标按下事件并处理
function ClickToDrag3(event:MouseEvent):void
{ p3.startDrag();}
stage.addEventListener(MouseEvent.CLICK,ReleaseToDrop3);
function ReleaseToDrop3(event:MouseEvent):void
{ if(p3.hitTestObject(j3))
    { p3.x = j3.x; p3.y = j3.y; f3 = true;

```

```

        if(f1 && f2 && f3 && f4 && f5 && f6 && f7 && f8 && f9 )
        { finishtext.visible = true;}
        } else      {p3.x = 680; p3.y = 470; }
        p3.stopDrag(); }
p4.addEventListener(MouseEvent.CLICK, ClickToDrag4);    //侦听 p4 中鼠标按下事件并处理
function ClickToDrag4(event:MouseEvent):void
{    p4.startDrag();}
stage.addEventListener(MouseEvent.CLICK, ReleaseToDrop4);
function ReleaseToDrop4(event:MouseEvent):void
{    if(p4.hitTestObject(j4))
    {    p4.x = j4.x; p4.y = j4.y; f4 = true;
        if(f1 && f2 && f3 && f4 && f5 && f6 && f7 && f8 && f9 )
        {finishtext.visible = true;}
        } else      {p4.x = 460; p4.y = 470; }
        p4.stopDrag();}
p5.addEventListener(MouseEvent.CLICK, ClickToDrag5);    //侦听 p5 中鼠标按下事件并处理
function ClickToDrag5(event:MouseEvent):void
{    p5.startDrag();}
stage.addEventListener(MouseEvent.CLICK, ReleaseToDrop5);
function ReleaseToDrop5(event:MouseEvent):void
{    if(p5.hitTestObject(j5))
    {    p5.x = j5.x; p5.y = j5.y; f5 = true;
        if(f1 && f2 && f3 && f4 && f5 && f6 && f7 && f8 && f9 )
        {finishtext.visible = true;}
        } else      {p5.x = 20; p5.y = 640; }
        p5.stopDrag();}
p6.addEventListener(MouseEvent.CLICK, ClickToDrag6);    //侦听 p6 中鼠标按下事件并处理
function ClickToDrag6(event:MouseEvent):void
{    p6.startDrag();}
stage.addEventListener(MouseEvent.CLICK, ReleaseToDrop6);
function ReleaseToDrop6(event:MouseEvent):void
{    if(p6.hitTestObject(j6))
    {    p6.x = j6.x; p6.y = j6.y; f6 = true;
        if(f1 && f2 && f3 && f4 && f5 && f6 && f7 && f8 && f9 )
        {finishtext.visible = true;}
        } else {p6.x = 240; p6.y = 640; }
        p6.stopDrag();}
p7.addEventListener(MouseEvent.CLICK, ClickToDrag7);    //侦听 p7 中鼠标按下事件并处理
function ClickToDrag7(event:MouseEvent):void
{    p7.startDrag();}
stage.addEventListener(MouseEvent.CLICK, ReleaseToDrop7);
function ReleaseToDrop7(event:MouseEvent):void
{    if(p7.hitTestObject(j7))
    {    p7.x = j7.x; p7.y = j7.y; f7 = true;
        if(f1 && f2 && f3 && f4 && f5 && f6 && f7 && f8 && f9 )
        {finishtext.visible = true;}
        }else      {p7.x = 680; p7.y = 640 }
        p7.stopDrag();}
p8.addEventListener(MouseEvent.CLICK, ClickToDrag8);    //侦听 p8 中鼠标按下事件并处理
function ClickToDrag8(event:MouseEvent):void
{    p8.startDrag();}
stage.addEventListener(MouseEvent.CLICK, ReleaseToDrop8);
function ReleaseToDrop8(event:MouseEvent):void
{    if(p8.hitTestObject(j8))

```



```
{
    p8.x = j8.x; p8.y = j8.y; f8 = true;
    if(f1 && f2 && f3 && f4 && f5 && f6 && f7 && f8 && f9 )
    {finishtext.visible = true;}
    } else {p8.x = 680; p8.y = 300; }
    p8.stopDrag();}

p9.addEventListener(MouseEvent.CLICK,ClickToDrag9);    //侦听 p9 中鼠标按下事件并处理
function ClickToDrag9(event:MouseEvent):void
{
    p9.startDrag();}
stage.addEventListener(MouseEvent.CLICK, ReleaseToDrop9);
function ReleaseToDrop9(event:MouseEvent):void
{
    if(p9.hitTestObject(j9))
    {
        p9.x = j9.x; p9.y = j9.y; f9 = true;
        if(f1 && f2 && f3 && f4 && f5 && f6 && f7 && f8 && f9 )
        {finishtext.visible = true;}
        } else {p9.x = 20; p9.y = 470; }
        p9.stopDrag();}

againbutton.addEventListener(MouseEvent.CLICK, MouseClickHandler);
//“再来”按钮侦听单击事件,将所有子图片复原,并将标记都设置成原始状态.
function MouseClickHandler(event:MouseEvent):void
{ play();p8.x = 680;p8.y = 300; p9.x = 20; p9.y = 470; p1.x = 240;p1.y = 470; p4.x = 460;p4.y
= 470;
p3.x = 680;p3.y = 470; p5.x = 20;p5.y = 640; p6.x = 240;p6.y = 640; p2.x = 460;p2.y = 640; p7.x =
680;p7.y = 640;
    finishtext.visible = false;
    f1 = false,f2 = false,f3 = false,f4 = false,f5 = false,f6 = false,f7 = false,f8 = false,f9 =
false ;}
```

(2) 此时 An 设计窗口如图 5-79 所示,“库”面板中有设计界面中所有用到的素材。测试并保存影片。



图 5-79 拼图游戏设计界面 2

5.11 拓展操作题

1. 已有小鸟飞翔的 7 个静态图片,如图 5-80 所示,创建“小鸟飞翔”动画。如何将小鸟飞翔动作减慢些?请分别使用改变帧速率和总帧数方法实现。

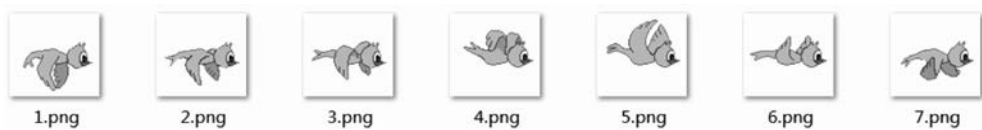


图 5-80 小鸟飞翔

2. 现有“操场航拍.jpg”和“跑步人.gif”两素材文件,要求制作跑步人顺着操场跑步,操场中间显示从右向左滚动的文字“宁波大学欢迎你”,效果如图 5-81 和图 5-82 所示。



图 5-81 操场跑步效果 1

提示:

- (1) 跑步人跑步部分用影片剪辑元件来完成。
- (2) 跑步人绕操场跑步用引导动画完成。
- (3) 中间“宁波大学欢迎你”文字用遮罩动画完成。

3. 按以下要求制作动画:

- (1) 设置电影舞台的大小为 $300 \times 300\text{px}$,背景色为淡黄色(颜色值为 #FFFFCC)。
- (2) 整个动画共占 30 帧;在舞台正中央绘制一个等边三角形 ABC;要求:①将等边三角形所在图层命名为“图形”层;②等边三角形的边长为 200px ,底边 BC 水平,边线的颜色为蓝色、线宽为 2px ,类型为实线,填充类型为零。
- (3) 制作一个画出等边三角形底边上的高 AD 的变形动画;要求:①高单独占一层,名称为“高”层;②高的颜色为红色、线宽为 2px ,类型为实线;③高由长度为 1 像素的线段逐



图 5-82 操场跑步效果 2

渐伸长得到,并以 A 点为起点。

(4) 标注字母 ABCD; 要求: ①所有标注字母单独占一层,并命名为“文字”层; ②标注字母字体为隶书、颜色为红色、字号为 20、位置适当。

4. 自创一个 An 案例: 可网上搜索原材料,再利用形状补间动画、引导动画、遮罩动画、影片剪辑元件等知识点合成最后效果。