|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **研发生产中心** | **文档编号** |  | **版本** |  | **密级** |  |
| **项目名称** |  | | | | |
| **项目来源** |  | | | | |

*QR-RD-022(Ver1.2)*

**图片转视频（转场特效）**

**的特效文件制作及说明**

(内部资料 请勿外传)

|  |  |  |  |
| --- | --- | --- | --- |
| **编 写：** |  | **日 期：** |  |
| **检 查：** |  | **日 期：** |  |
| **审 核：** |  | **日 期：** |  |
| **批 准：** |  | **日 期：** |  |

**北京正奇联讯科技有限公司**

**版权所有 不得复制**

**文档变更记录**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **变更（+/-）说明** | **作者** | **版本号** | **日期** | **批准** |
| 1 | 建立 | 牛成磊 | 0.1 | 2014/9/5 |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
| 5 |  |  |  |  |  |
| 6 |  |  |  |  |  |
| 7 |  |  |  |  |  |
| 8 |  |  |  |  |  |

目 录

[1. 引言 4](#_Toc388356991)

[1.1 编写目的和范围 4](#_Toc388356992)

[1.2 术语表 4](#_Toc388356993)

[1.3 项目说明 4](#_Toc388356994)

[2. 功能和使用说明 4](#_Toc388356995)

[2.1 输入检测配置规范 5](#_Toc388356996)

[2.2 输出检测结果规范 5](#_Toc388356997)

[2.3 日志规范 6](#_Toc388356998)

[2.4 程序使用方式 6](#_Toc388356999)

[3. 部署说明 7](#_Toc388357000)

[3.1 硬件环境 7](#_Toc388357001)

[3.2 软件环境 7](#_Toc388357002)

[3.3 安装配置步骤 7](#_Toc388357003)

[4. 测试说明 8](#_Toc388357004)

[4.1 正确性验证 8](#_Toc388357005)

[4.2 鲁棒性测试 8](#_Toc388357006)

[4.3 准确度测试 8](#_Toc388357007)

[4.4 效率测试 9](#_Toc388357008)

[5. 备注 9](#_Toc388357009)

# 引言

## 编写目的和范围

方便以后特效的制作，以及说明转场特效原理。

## 术语表

|  |  |  |
| --- | --- | --- |
| **序号** | **术语或缩略语** | **说明性定义** |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |

## 项目说明

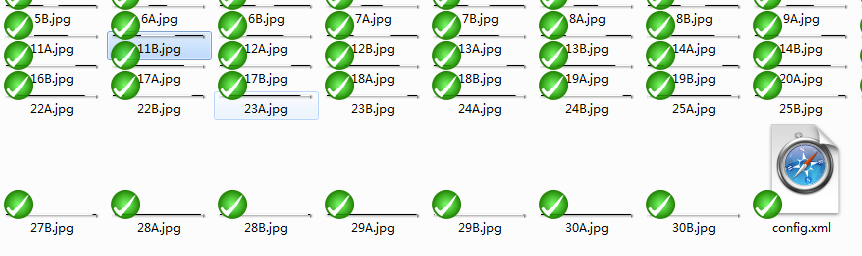
本软件主要功能是由图片生成视频，并加入转场特效，特效支持常见几种，有：淡入淡出、马赛克、形状、推进、擦除、随机样条等，图片和特效播放时间可控。

项目封装为可执行程序，输入和输出采用进程参数和标准XML文件方式。

# 特效原理

转场特效由两部分组成：Alpha通道和形状变换，大多数特效只需要两者之一便可实现，有些特效需要二者结合才能实现。从场景A切换到场景B，通过以上两种变化即可计算出中间帧，从而完成转场功能。

Alpha通道又称透明度，其值范围为0-255，因此非常适合用图像存储，因此特效文件使用图片存储Alpha通道信息。每个特效由至少4张图片描述Alpha变化，A起始/A结束/B起始/B结束；同样，也可以由多幅图像描述，文件名使用0A.jpg/1A.jpg…表示。为了制作方便和减少存储，Alpha通道可以有4种方式，有全局一致、行一致、列一致和全局非一致，如果整个图像的Alpha变化一致，则符合全局一致，可以用一个像素点表示；如果图像的每一行Alpha变化一致则符合行一致，可用一行像素表示，以此类推；对于行一致、列一致和全局非一致情况，请制作512×1、1×512、512×512大小图像，系统会自动插值以适应不同分辨率图像。



形状变换分为透视变换、仿射变换、平移变换。（这里需要一些线性代数基础知识或是计算机图形学），这些变换均可以通过一个3×3矩阵表示。

由于矩阵数据量小，因此其以文本方式存储在特效配置文件中，一个特效至少包含4个变换矩阵，A起始/A结束/B起始/B结束，加入需要加入中间过程，则过程位置以0-1之间的浮点数表示（如0.5说明在特效中间时间处的变换矩阵）

## 特效文件描述

<effect name="EraseOut" alphanum="31" duration="1.5" front="A" background="0">

<script pos="0">

<A>1,0,0,0,1,0,0,0,1</A> // 用于图像比例为1:1的

<B>1,0,0,0,1,0,0,0,1</B>

<A43>1,0,0,0,1,0,0,0,1</A43> // 用于图像比例为4:3的

<B43>1,0,0,0,1,0,0,0,1</B43>

<A169>1,0,0,0,1,0,0,0,1</A169> // 用于图像比例为16:9的

<B169>1,0,0,0,1,0,0,0,1</B169>

</script>

<script pos="1">

<A>1,0,0,0,1,0,0,0,1</A>

<B>1,0,0,0,1,0,0,0,1</B>

<A43>1,0,0,0,1,0,0,0,1</A43>

<B43>1,0,0,0,1,0,0,0,1</B43>

<A169>1,0,0,0,1,0,0,0,1</A169>

<B169>1,0,0,0,1,0,0,0,1</B169>

</script>

</effect>

# 特效制作

## 特效制作步骤

特效有简单复杂之分，大概可通过变换的复杂度和透明度变化的复杂度来衡量，由于这两种制作方式不同，所以步骤有些不同。

变换简单、透明度变化复杂的特效制作步骤：

第一步：先确定变换矩阵

第二步：用ps或matlab工具制作Alpha通道变化图片，如果变化复杂可先用视频编辑工具编辑黑白图片成视频，根据需要调节透明度变化，然后用matlab程序保存输出成特效Alpha文件。

变换复杂、透明度变化简单的特效制作步骤：

第一步：根据自己需求，画出每个阶段的位置草稿图

第二步：根据草稿图定坐标，把坐标写出到文本文件

第三步：运用工具计算【坐标文件】的变换矩阵R，把得出的9个数编辑到特效配置文件config.xml中，如有必要可通过matlab程序验证动画效果。

## 还提供很人性化的工具哦亲

为了方便特效制作，提供很人性化的工具，有3个比较有用：

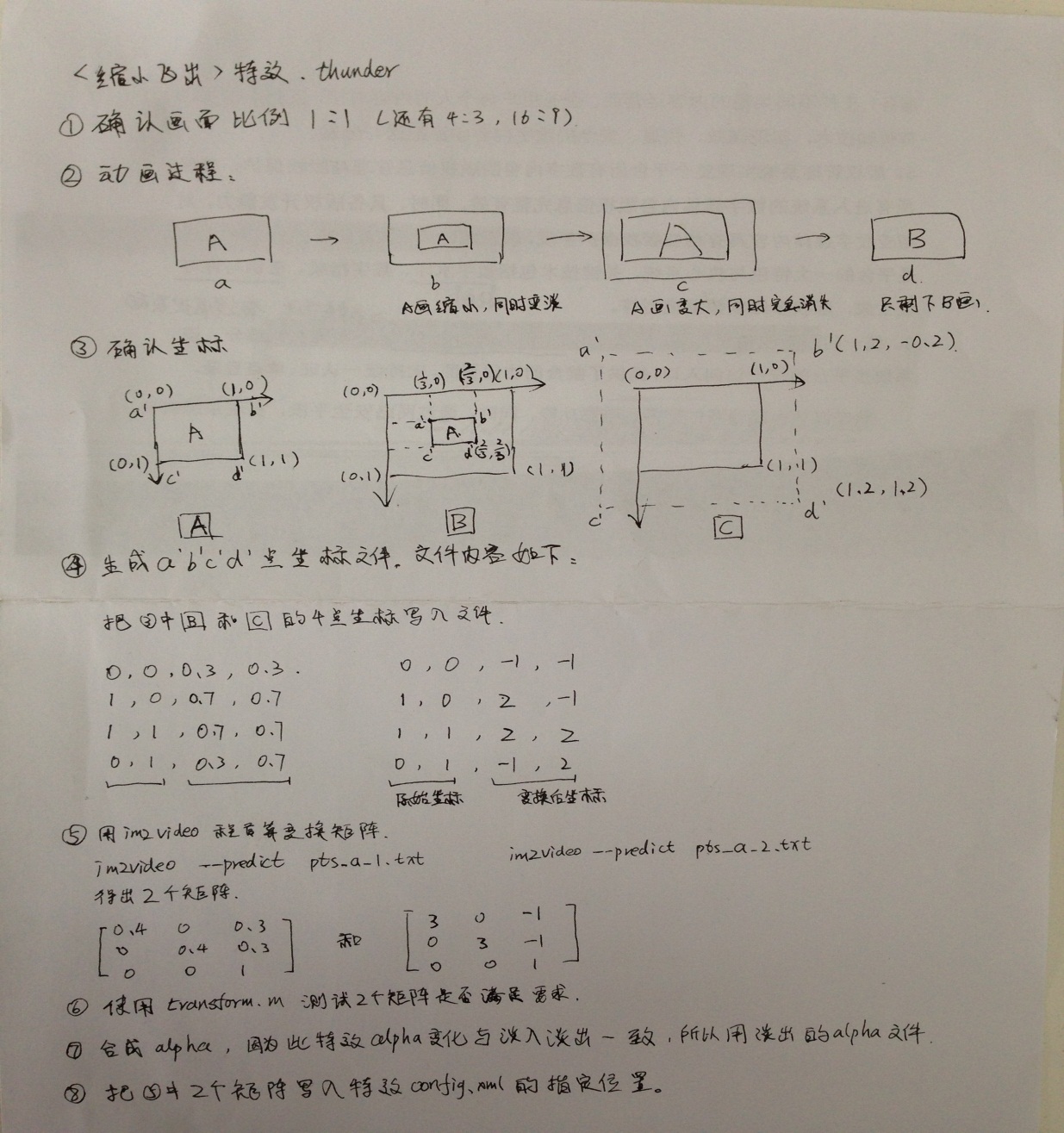
视频输出到图片matlab程序：用于从alpha视频中保存alpha图片到文件。imgen.m文件。

alpha计算matlab程序：用于计算alpha变化，可自定义公式。logis\_curve.m文件。

变换矩阵计算c程序：用于计算坐标变换的3×3矩阵。（im2video --help查看）

动画演示matlab程序：用于验证动画是否满足需求，与输出保持一致。transform.m和animation.m文件。

举例：（制作一个缩小飞出特效thunder）



特效文件为（矩阵信息见下面红色部分）：

<?xml version="1.0" encoding="UTF-8" standalone="no" ?>

<render version="2.0">

<effect name="Thunder" alphanum="30" duration="1.5" front="A" background="0">

<script pos="0">

<A>1,0,0,0,1,0,0,0,1</A>

<B>1,0,0,0,1,0,0,0,1</B>

<A43>1,0,0,0,1,0,0,0,1</A43>

<B43>1,0,0,0,1,0,0,0,1</B43>

<A169>1,0,0,0,1,0,0,0,1</A169>

<B169>1,0,0,0,1,0,0,0,1</B169>

</script>

<script pos="0.5">

<A>0.4,0,0.3,0,0.4,0.3,0,0,1</A>

<B>1,0,0,0,1,0,0,0,1</B>

<A43>0.4,0,0.3,0,0.4,0.3,0,0,1</A43>

<B43>1,0,0,0,1,0,0,0,1</B43>

<A169>0.4,0,0.3,0,0.4,0.3,0,0,1</A169>

<B169>1,0,0,0,1,0,0,0,1</B169>

</script>

<script pos="1">

<A>3,0,-1,0,3,-1,0,0,1</A>

<B>1,0,0,0,1,0,0,0,1</B>

<A43>3,0,-1,0,3,-1,0,0,1</A43>

<B43>1,0,0,0,1,0,0,0,1</B43>

<A169>3,0,-1,0,3,-1,0,0,1</A169>

<B169>1,0,0,0,1,0,0,0,1</B169>

</script>

</effect>

</render>

# 备注