

# 一种图象变形映射算法的研究

何国辉

(五邑大学计算机工程系, 广东 529020)

**摘要** 探讨了图象变形的原理, 提出了一种将矩形区域的二维图象向任意的二维曲线轮廓区域映射的算法, 并给出了利用位图格式文件经编程处理获得的一些结果。实例证明, 这种算法是行之有效的。

**关键词** 图象变形, 图象处理, 映射

## 1 问题的提出

图象变形技术在影视制作和商业广告制作方面的应用非常广泛, 例如: 1987年 ILM (Industrial Light and Magic) 公司的道格·西姆斯 (Dong Smythe) Ron Howard 制作的神话故事片《Willow》中, 就是通过变形技术, 实现了影片中的一系列变换: 手持魔棒的 Willow 凭借着魔棒的威力, 一下将山羊变成了一只驼鸟, 之后又变成了孔雀、海龟……, 最后变成了一个女巫。又如 PDI (Pacific Data Images) 公司的 Exxno 在一个商业广告片中, 又是利用变形技术, 将一辆飞驰的小汽车变成了一只奔跑的猛虎。这些处理给人们留下了十分深刻的印象, 也给现实生活增添了不少的乐趣。

上述变形的处理过程, 大多数只涉及到从一个具有矩形域的初始图象, 变化到另外一个具有相同区域的目的图象的算法。但是, 如果要做到将一幅具有矩形域的二维图象映射到一个非矩形域, 甚至是一个三维的曲面上, 以便产生一种扭曲、拉伸等特殊效果就不行了。因此, 本文提出了一种实现上述变形的映射算法。

## 2 图象映射算法的基本思路

根据一幅二维的矩形图象映射到目标区域的不同, 可以将其分为二维映射和三维映射 2 种算法。如

果目标区域是在一个二维平面上, 那么这个算法就是二维的, 否则称它是三维的。

无论是二维算法还是三维算法, 都可以假设给定的初始图象的特征矩阵为  $S$ , 目标图象的特征矩阵为  $T$ 。  $S$  和  $T$  分别表示为:

$$S = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1q} \\ a_{21} & a_{22} & \cdots & a_{2q} \\ \cdots & \cdots & \cdots & \cdots \\ a_{p1} & b_{p2} & \cdots & b_{pq} \end{bmatrix}$$

$$T = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ b_{m1} & b_{m2} & \cdots & b_{mn} \end{bmatrix}$$

对于二维算法如图 1 所示, 为了存贮的方便, 可以将目标图象的区域用一个矩形域表示, 图象的映射区  $Q$  就被包络在这一个矩形域中, 即:  $Q \subset T$ 。实际上,  $S$  向  $Q$  映射的结果就是  $T$  中某些元素  $b_{ij}$  的集合。

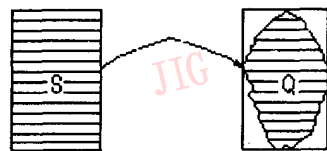


图 1 图象  $S$  向任意区域  $Q$  的映射

由于  $S$  和  $Q$  的大小形状不一致, 因此, 在图象的映射过程中, 必然会产生挤压或拉伸现象, 在受挤

压处,我们可以用受挤压的各点象素的平均值来表示;而在受拉伸处,则采用插值方法,将象素点空白的位置补上相应的结果;对于  $T$  中  $Q$  以外的部分用屏幕的背景色填充。

图2中在几何位置上, $S$ 中的第  $i$  行对应着  $Q$  中的第  $i'$  行,如果  $\overline{AB} < \overline{A'B'}$ ,则在宽度方向  $Q$  受拉伸影响;如果  $\overline{AB} = \overline{A'B'}$ ,则  $Q$  不受影响,即  $Q$  中的第  $i$  行结果与  $S$  中第  $i$  行相同;如果  $\overline{AB} > \overline{A'B'}$ ,则在宽度方向,  $Q$  受挤压影响。

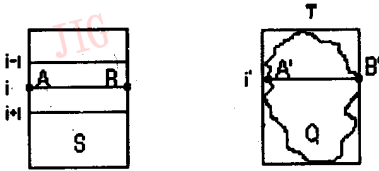


图2 图象  $S$  中的某行元素映射到  $Q$  中的对应关系

由于还要考虑在高度方向  $Q$  的挤压和拉伸情况,因此,通过比较  $\overline{AB}$  与  $\overline{A'B'}$  的大小后,再采用插值或取平均值的办法所获得的值,并不能完全确信是  $Q$  中第  $i'$  行的真正结果,而是还要根据  $S$  中的  $i-1$  和  $i+1$  行映射到  $Q$  中的区域情况来判断。如果  $S$  中第  $i-1$  行映射到  $Q$  中是小于  $i'$  的,且  $S$  中第  $i+1$  行映射到  $Q$  中是大于  $i'$  的,则  $Q$  中  $i'$  的结果不变。否则,  $i'$  的结果受挤压影响,其最后结果还要根据其它小于  $i$  行与大于  $i$  行的数行映射结果来确定。

如果  $S$  中第  $i-1$  行映射到  $Q$  中是第  $i''$  行,且  $i'' < i' - 1$ ,则在  $i''$  到  $i'$  行之间由插值方法求得。其它情况依此类推。

### 3 图象映射算法的处理方法

在实现上述算法的处理过程中,可以分两步走。第一步是将  $S$  中的元素映射到  $T$  中的全部区域,不考虑  $Q$  区域的影响。因此,此时得到的结果是一幅在高度和宽度方向分别产生均匀的拉伸或者挤压的图象。第二步是将所得到的结果再映射到  $T$  中的部分区域  $Q$ ,从而获得最终结果。

#### 3.1 原始图象 $S$ 向目标图象 $T$ 的全区域映射

为了获得目标区  $T$  的全区域图象映射结果,考虑到特征矩阵  $S$  和  $T$  中的  $p \neq m$  和  $q \neq n$ ,因此,在具体的处理过程还有一些技巧,即:

(1) 首先将  $S$  映射到特征矩阵  $T'$

(2) 再将  $T'$  映射到  $T''$

其中:

$$T' = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ c_{p1} & c_{p2} & \cdots & c_{pn} \end{bmatrix}$$

$$T = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ c_{m1} & c_{m2} & \cdots & c_{mn} \end{bmatrix}$$

同时,在将  $S \rightarrow T'$  和  $T' \rightarrow T''$  的转换过程中,考虑到图象格式文件的存贮顺序对图象处理难易程度的影响,在变换过程中,还要作些调整。

以位图格式文件为例。因为一幅 BMP 图象文件主要包括 3 部分内容,即:

位图文件头;位图信息;位图阵列。

在位图阵列中记录了图象的每一个象素值。其记录顺序是从左至右,从下到上将图象的象素值一一记录下来的。即位图阵列中的第一行象素值字节在生成图象时,是对应于屏幕的最下一行图象。反之,位图阵列文件的最后一行象素值,对应于图象的最顶一行。也就是说,特征矩阵  $S$  中的每行元素,在文件中是连续存放的。因此,在将  $S$  转换到  $T'$  时,只要分别读取每行图象的象素值,再采用插值或者挤压的方法,对该行象素进行操作。

在将  $T'$  转换到  $T''$  时,由于需要分别对  $T'$  中的每列象素进行操作,而在位图文件中,这些象素的存放是不连续的。因此,为了处理的方便,应该先将特征矩阵  $T'$  转置得到  $T'_i$ ,形成一个临时文件,再将  $T'_i$  按照  $S \rightarrow T'$  的转换方式转换到对应的  $T''$  的转置区域中。当处理完毕以后,再转置过来,从而得到  $T''$ 。

#### 3.2 向 $T$ 的部分区域 $Q$ 映射

通过上述讨论,很清楚地看到,在将全区域映射后获得的结果,向  $T$  的部分区域  $Q$  映射的处理方式,与第一步有许多相似之处,即:只要考虑图象在一个方向(行)的拉伸或挤压情况。唯一的区别是前者的每行图象的拉伸或挤压比相同,而后者由于  $Q$  区域并非是个矩形,因此,每行图象的拉伸、挤压比是不同的。

以  $W_1$  表示原始图象的宽度,  $W_2$  为  $Q$  区域对应的宽度,  $\beta_1$  为该行图象的挤压、拉伸比。  $\beta_2$  是  $\beta_1$  取整后的值。即:

$$\beta_1 = \frac{W_2}{W_1}, \beta_2 = \text{int}(\beta_1)$$

在  $\beta_1 < 1$  的情况下, 该行象素在  $Q$  区域被挤压。映射后的结果为:

$$T = \sum_{i=0}^{m-1} B_i / m$$

其中:  $m$  为受挤压的点数,  $B_i$  为被挤压的各个象素点值。

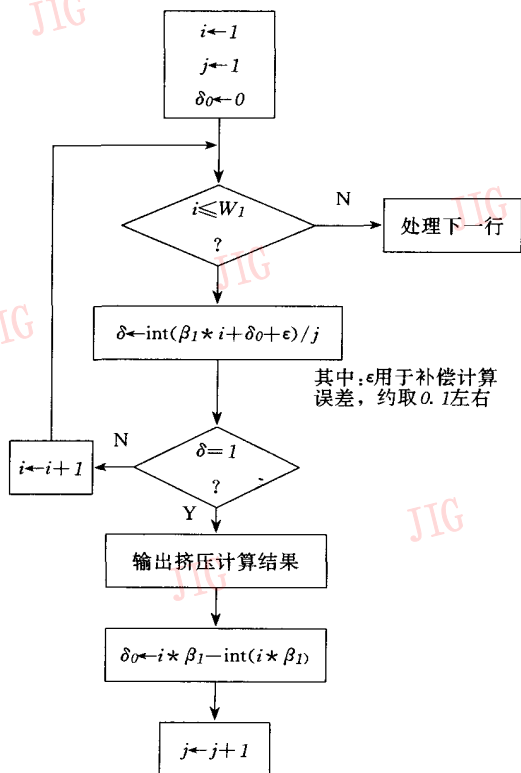
在  $\beta_1 > 1$  的情况下,  $Q$  区域的该行象素被拉伸。映射后的结果为:

$$T = (1 - \frac{i}{W}) * B_0 + \frac{i}{W} * B_1$$

其中:  $i$ : 为插值节点,  $i=0, 1, \dots, n-1$ 。  $W$ : 为插值区间长度,  $W=n-1$ 。  $B_0, B_1, T$ : 分别为插值区间的起始点象素值、终止点象素值、以及映射结果。

$$B_0 = \begin{pmatrix} b_{0r} \\ b_{0g} \\ b_{0b} \end{pmatrix}, B_1 = \begin{pmatrix} b_{1r} \\ b_{1g} \\ b_{1b} \end{pmatrix}, B_i = \begin{pmatrix} b_{ir} \\ b_{ig} \\ b_{ib} \end{pmatrix}, T = \begin{pmatrix} t_r \\ t_g \\ t_b \end{pmatrix}$$

图 3 和图 4 是处理一行图象受拉伸或挤压作用的流程图:



其中:  $\epsilon$  用于补偿计算误差, 约取 0.1 左右

图 4 图象受挤压作用的流程图

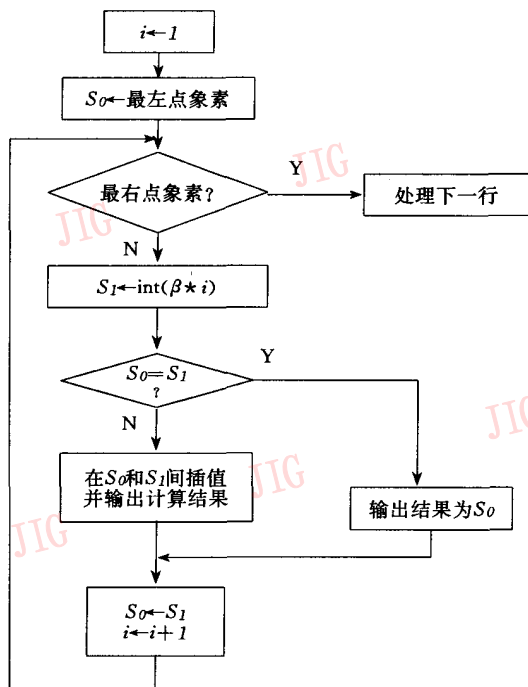
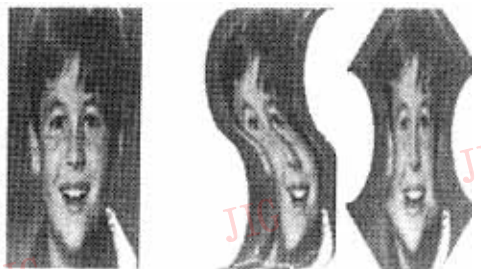


图 3 图象受拉伸作用的处理流程



(a) 原始图象

(b) 宽度方向挤压图象



(c) 高度方向拉伸图象

(d) 向区域映射的结果

图 5 变形图象

图 5 是根据上述算法, 以图 5(a) 为原始图象, 经编程处理获得的几种映射区域的变形图。

## 4 结论

图象变形技术是一门非常复杂的学科。同时由

于各种图象格式不同,也为图象的处理增加了难度。本文从图象映射方面入手,在图象的变形方面作了一些尝试,事实证明,上述算法是非常有效的。

### 参考文献

- 1 王新成. 多媒体实用技术. 成都: 电子科技大学出版社, 1995.
- 2 董士海. 郑全战等. 图象格式编程指南. 北京: 清华大学出版社, 1994.
- 3 Lee Adams 著. Visual Basic 动画编程. 施小龙等译. 北京: 学苑出版社, 1994.

- 4 Charles Parly 著. Visual Basic 3.0 程序设计大全. 梁念蓉等译. 北京: 电子工业出版社, 1994.

何国辉 1983 年本科毕业于沈阳航空工业学院, 1991 年获北京航空航天大学制造工程系航空宇航制造工程专业硕士学位, 其后一直从事图象处理、CAD/CAM、多媒体技术方面的研究和开发工作。



## Study of A Mapping Algorithm for Image Morphing

He Guohui

(Department of Computer Engineering, Wuyi University, Jiangmen City, Guangdong 529020)

**Abstract** This paper discusses the principle of image morphing, presents an algorithm, in which two-dimension image in square region is mapped to any two-dimension outline region, and gives some results of the algorithm by use of BMP file. Examples demonstrate that this algorithm is extremely valid.

**Keywords** Image morphing, Image processing, Mapping

## ELSA 新系列图形加速卡满足绘图处理需求

在图形处理与数字通信两大领域领先业界的艾尔莎(ELSA)国际科技股份有限公司,日前在北京举办了题为“艾尔莎国际科技拓展中国市场策略发布会”,全面加强大中国地区 OEM 业务及与经销商伙伴的合作关系,并致力提供给亚洲客户全球性的行销支持服务。

为了全面开拓亚洲及中国市场,艾尔莎已开始销售其 ELSA GLoria 2D/3D 家族最新的图形加速卡产品。其第一种产品是 GLoria-XL 高档 3D 图形加速卡,它是实现虚拟现实的最佳工具,备有快速而大容量的贴图及材质存储器,是 3D CAD /CAM 高级处理的利器。GLoria-XL 采用 3Dlabs MX 芯片,最高分辨率可达 1900×1200。

第二种产品是 GLoria-L 3D 图形加速卡,它可轻而易举地将 Intel 或 DEC-Alpha 系统升级为高档 3D 工作站。

第三种产品是 GLoria Synergy,它搭载知名的绘图芯片制造厂商 3Dlabs 的 PERMEDIA 2 绘图芯片,可支持 Windows 95、Direct3D,以及 Windows NT 平台上专业的 OpenGL 介面。GLoria Synergy 图形加速卡的主要应用是在 CAD 或设计专业领域。它是 OpenGL 上 3D 应用的理想图形加速卡。

第四种产品是 Winner 2000/Office,它为办公商业应用市场的图形处理性能创建了新标准。它在文书处理及制表软件上的超强的 2D 性能、集成的视讯功能,可提供高品质的全屏 MPEG 播放功能,快速的表面材质贴图(Texture Mapping)功能,在运行最新 Windows 95 游戏软件时,更是如虎添翼。

此外,新上市的还有 128 位 3D 游戏加速卡 Victory Erazor,它支持 Direct 3D、Windows 95、Windows NT 处理能力,可带给 3D 游戏全新的感受。