

# 基于视频纹理的自然场景建模方法

李 骋, 钱文华, 徐 丹

LI Cheng, QIAN Wen-hua, XU Dan

云南大学 计算机科学与工程系, 昆明 650091

Department of Computer Science and Engineering, Yunnan University, Kunming 650091, China

E-mail: licheng19841984@yahoo.com.cn

**LI Cheng, QIAN Wen-hua, XU Dan. Algorithm for natural phenomena simulation based on video texture. Computer Engineering and Applications, 2010, 46(12): 157-160.**

**Abstract:** This paper proposes an improved video-based texture synthesis approach for simulating realistic natural phenomena. The algorithm measures similarity of every two frames to determine an initial playing sequence and then divides the initial playing sequence into several subsequences with different frame-to-frame delay, so as to remove visual discontinuities. For removing the abrupt change from the last frame to the first frame in the video loop, at first the algorithm determines the cycle of the initial playing sequence. Then the algorithm chooses several frames from the initial playing sequence based on the cycle and divides every single chosen frame or subsequences into patches, which depends on the cycle of video. At last the algorithm re-organizes these patches into a final playing sequence. The experiment results indicate that the algorithm is easy to synthesizing realistic effects of natural surrounding video which has repetitive or quasi-repetitive motions.

**Key words:** video texture; texture synthesis; cycle; initial playing sequence; final playing sequence; subsequence

**摘 要:** 提出了一种改进的基于视频纹理合成的自然场景建模方法。算法通过衡量图像之间的相似度来确定一个视频的初步播放序列,然后将这个初步播放序列划分成几个子序列并调整视频播放过程中的帧与帧之间的延迟,从而消除视频播放中的跳变。为了消除视频循环播放中最后一帧到第一帧的跳变,首先确定初步的播放序列的周期,然后根据周期选取用于合成最终播放序列的图像并对选取的图像根据周期的长短选择以单帧图像或子序列为单位进行图像剖分,最后把剖分后的图像重新组合成最终的播放序列。算法对于各种有周期性或近似于有周期性运动的自然场景都能合成贴近真实的视频效果。

**关键词:** 视频纹理; 纹理合成; 周期; 初步播放序列; 最终播放序列; 子序列

**DOI:** 10.3778/j.issn.1002-8331.2010.12.047 **文章编号:** 1002-8331(2010)12-0157-04 **文献标识码:** A **中图分类号:** TP391.1

## 1 介绍

计算机越来越多地用于表现和模拟现实的场景,因此在计算机中通常可以用图像和视频来表现现实的场景。但是,图像具有不随时间而改变的特点,无法反映动态的场景,自然界中的许多场景,一张甚至多张图像都无法完整地刻画其壮观和美丽。而视频虽然具有动态性,但是由于存储介质容量的限制,视频只能记录某段时间的流体场景的信息,其长度会受到一定的限制。视频片断有开始,有过程,有结束,因而无法表现场景运动的“无限性”。为解决这个问题,Arno Schodl<sup>[1]</sup>在2000年的SIGGRAPH年会上提出了一种新的媒体,这种媒体被称为视频纹理。视频纹理同时具有图像和视频的特征:时序稳定性和内容动态性。利用视频纹理,可以实现利用有限样本表达无限信息的目标。此外,利用视频纹理可以合成很多电影特效而不用每次需要特效时都要花高昂的代价去拍摄。视频纹理适用于有

周期性或近似于有周期性运动的视频,比如,流水、火焰、喷泉等,但是不适用于不具有周期性运动的视频,比如拥挤的商场。

## 2 相关工作

自然场景的模拟一直是图形学领域中的热点问题。近年来,研究者们对基于图像的建模和绘制技术做了深入的研究<sup>[2]</sup>。在此基础上发展的视频纹理方法则可看作是基于图像的绘制方法在时间域的扩展。Schödl等<sup>[3]</sup>提出视频纹理的概念,并通过循环播放视频帧来合成无限长的钟摆视频。Schödl等<sup>[4]</sup>把视频纹理应用到视频精灵, Schodl的方法通过L2距离衡量帧与帧之间的相似度来确定视频的播放顺序。这种算法成功地实现了高度真实感的景物动画, Kwatra等<sup>[5]</sup>提出基于图切技术的图像视频合成算法,将输入图像或视频的某个部分经过转换然后拷贝到输出图像上,这些图像碎片通过优化算法在输出图像上弥

**基金项目:** 国家自然科学基金(the National Natural Science Foundation of China under Grant No.60663010); 云南省自然科学基金(the Natural Science Foundation of Yunnan Province of China under Grant No.2006F0017M)。

**作者简介:** 李骋(1984-),男,硕士研究生,主要研究方向为图像处理,视频纹理等;钱文华(1980-),男,讲师,主要研究方向为图像处理,计算机非真实感等;徐丹(1968-),女,教授,博士生导师,主要研究方向为基于图像的建模与绘制,虚拟现实等。

**收稿日期:** 2008-10-10 **修回日期:** 2008-11-28

合成为一个整体,能将输入图像合成任意大的输出图像。2004年, Bhat<sup>[6]</sup>提出一种简单的编辑自然场景视频的算法,算法通过分析自然的运动规律,然后对视频的每帧剖分成几个部分,最后根据场景的运动规律对剖分后的部分重组成新的视频。

基于 Arno Schodl 和 Bhat 等人的思想提出了一种改进的视频纹理合成方法,和以往的算法不同的是:算法在合成视频前先确定初步播放序列的周期,然后根据初步播放序列的周期选取用于合成最终的播放序列的图像。此外,算法还根据周期的长短选择以一个单帧图像或者子序列作为单位对被选取的图像进行剖分,然后重组来合成一个最终的无限长的播放序列。

### 3 算法概述

视频纹理合成的最大挑战在于以下两个问题:(1)如何克服播放过程中相邻的两帧图像之间的跳变,使合成结果保持时空一致性并且视频效果贴近真实的自然场景。(2)因为视频纹理是通过循环播放合成视频所用的图像来实现视频在时间上的无限,所以要解决好播放中首尾两帧的跳变问题。算法按如图 1 所示的步骤解决上面两个问题。和以往的算法相比,该文提出的算法有以下几个贡献:

(1)根据初步播放序列的周期选取合成视频的图像并使得最终的播放序列的长度为一个周期,减弱了播放过程中的跳变。

(2)基于纹理的复杂与否,选择子序列或单一图像作为剖分的单位,扩大了视频纹理的处理范围,解决了以往的视频纹理算法只适用于纹理简单的周期短的视频的问题。

(3)根据划分好的子序列来控制视频播放过程中相邻帧之间的延迟时间长短,解决了视频的跳变、不连贯和视频的播放速度过快的问题。

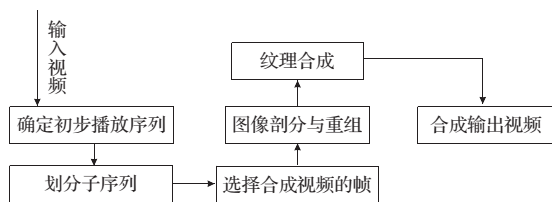


图1 系统框架图

### 4 确定初步的播放序列

视频纹理合成的关键在于调整视频帧的跳变,使合成结果平滑。为此需要确定视频的一个初步的播放序列,算法通过衡量每两帧图像的相似度来确定初步的播放顺序。定义  $D_{ij} = \|I_i - I_j\|_2$ , 其中  $\|I_i - I_j\|_2$  是帧  $I_i$  与帧  $I_j$  的 L2 距离。两帧图像的 L2 距离越小这两帧图像的相似度就越高。在计算图像的 L2 距离时,首先将图像从 RGB 空间转换到 YCbCr 空间,从 YCbCr 空间中提取图像的亮度信息,然后基于提取出来的亮度信息计算图像  $I_i$  和图像  $I_j$  的 L2 距离,一旦每两帧图像之间的 L2 距离计算完,结果即被保存在矩阵  $D_{ij} = \|I_i - I_j\|_2$  中。播放了图像  $I_i$  后,从矩阵  $D_{ij}$  中找到和图像  $I_i$  的 L2 距离最小的图像作为图像  $I_i$  的后续帧,由此可以得到一个播放序列。这个播放序列即为一个初步的播放序列。但是仅播放这个初步的播放序列不能完全解决视频的跳变问题。下面将进一步讨论解决视频跳变的问题。

### 5 划分子序列

在合成视频的时候不仅需要计算图像的相似度,还要消除

播放中的跳变和视频速度过快的问题。如图 2 所示视频纹理的初步播放序列由  $n$  帧图像组成并且播放顺序为  $I_1, I_2, \dots, I_n$ 。顺序播放这  $n$  帧图像并且每相邻的两帧图像之间没有时间延迟,这将产生视频跳变和播放速度过快的问题。

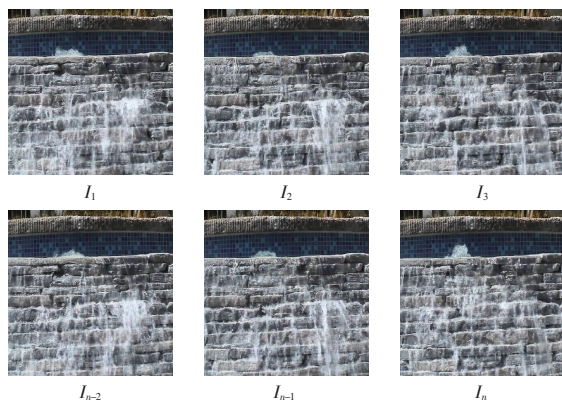


图2 播放顺序图

为了解决这个问题,需要把图像按相似度进行分类。如果在视频的播放顺序中相邻两帧图像的 L2 距离小于某一阈值  $K$ ,就把这两帧图像归为一个子序列。用公式(1)来表示阈值  $K$ ,其中  $n$  表示视频帧的数目,  $m$  表示视频序列中的某一帧。

$$k = \frac{1}{n-1} \sum_{m=1}^{n-1} D_{m,m+1} \quad (1)$$

在播放过程中每相邻两帧图像之间都会有延迟,因为序列内部相邻两帧的相似度大于序列间相邻两帧的相似度,所以同一个序列内相邻两帧之间的延迟时间要短于不同序列之间相邻的两帧。这样合成的视频比较流畅,更能贴近真实的自然场景。解决了视频播放相邻两帧图像的跳变后,下一步要解决视频的首尾两帧的跳变问题。

### 6 确定最终的播放序列

纹理视频是通过循环播放视频片段来实现视频的无限长,如果循环播放初步的播放序列,那么在播放中首尾两帧图像的突变将会很大,影响了视频的连贯性。为了解决这个问题,算法先确定初步播放序列的周期。满足公式(2)的  $I_1, I_2, \dots, I_m, I_{m+1}, \dots, I_n$  这  $n$  帧图像组成了初步播放序列的周期(假定  $I_1, I_2, \dots, I_m, I_{m+1}, \dots, I_n$  为初步播放序列中的正确的播放顺序),其中  $I_1$  为周期的第 1 帧,  $I_n$  为周期的最后一帧。初步播放序列的周期为  $n$ 。表 1 中记录了图 2 所示的视频的第 1 帧和后续帧的 L2 距离。

$$D_{1,2} < D_{1,3} < \dots < D_{1,m} > D_{1,m+1} > \dots > D_{1,n} < D_{1,m+1} \quad (2)$$

表1 第1帧和后续帧的 L2 距离

第1帧	后续帧							
	2	3	4	5	6	7	8	9
1	4 776	6 206	7 069	7 124	7 125	7 454	7 453	7 114
第1帧	后续帧							
	10	11	12	13	14	15	16	17
1	6 952	6 117	5 822	5 549	5 049	5 305	5 309	5 879

由表 1 可知:  $D_{1,2} < D_{1,3} < \dots < D_{1,7} > D_{1,8} > \dots > D_{1,14} < D_{1,15}$ 。所以  $I_1, I_2, \dots, I_7, I_8, \dots, I_{14}$  组成了初步播放序列的周期,其中  $I_1$  为第 1 帧,  $I_{14}$  为最后一帧,其周期为 14。由公式(2)可知一个周期的首尾两帧图像很相似,由大量实验数据表明:相隔一个周期长度的两帧图像很相似,比如图 2 中不仅第 1 帧和第 14 帧很相似,第 2 帧和第 15 帧也很相似。通过实验表明流水、喷泉、瀑布、火



焰视频都有上述的周期性规律,只是不同的视频它们的周期长度不同而已。因此可以选择相隔一个周期长度的两帧图像作为播放序列的首尾两帧,这样可以在一定程度上解决首尾两帧的突变,但是还达不到希望的效果。Bhat 在 2004 年提出了一种通过图像剖分和重组解决首尾两帧跳变的算法。算法将合成视频的图像进行剖分,然后重组成新的视频。这里根据相隔一个周期长度的两帧图像很相似并结合 Bhat 的方法来实现。如图 3 所示,图像被分成三部分,1-1 表示合成视频的第 1 帧图像的上部部分;1-2 表示合成视频的第 2 帧图像的中间部分;1-3 表示合成视频的第 3 帧图像的下面部分。 $M-N$  表示合成视频的第  $M$  帧图像的第  $N$  部分。如果周期为  $n$ ,就把图像剖分成  $n$  部分。

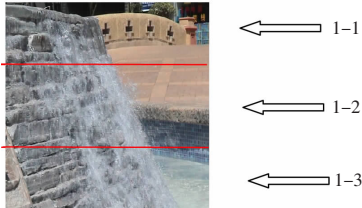


图 3 网格剖分示意图

然后把剖分后的部分重新组合来模拟自然场景。如图 4 所示,(a)为流水随时间向下流动,矩形表示剖分后的流水的某一部分随时间  $t$  流到不同位置。(b)为随着时间  $t$  的变化,剖分后的流水的某一部分在时间  $t=1$  时处于第 2 帧的第 1 部分的位置,在时间  $t=2$  时处于第 3 帧的第 2 部分的位置,那么时间  $t=5$  时处于第 6 帧的第 5 部分的位置。(c)为对一个周期为 3 的流水的初步播放序列,取流水的三部分来模拟流水的变化,其中同一颜色表示剖分后的流水的同一部分随时间变化。如 1-1、2-2、3-3 表示剖分后的流水的同一部分在时间  $t=1$  时处于第 1 帧的第 1 部分,时间  $t=2$  时处于第 2 帧的第 2 部分, $t=3$  时处于第 3 帧的第 3 部分。那么对于周期为  $n$  的情况,就要取流水的  $n$  部分来模拟流水的变化。如表 2 所示,对于周期为  $n$  时,算法中选取流水的  $n$  部分来模拟流水并循环播放这  $n$  部分来组成一个无限长的视频,同样的,同一颜色表示流水的同一部分。表 3 表示了当周期为 3 时,选取流水的 3 部分然后循环播放这 3 部分来组成无限长的视频。在表 2 和表 3 中每一列为播放中的每一帧。从表 3 可知,重组后的播放序列通过重复播放中间的帧、第 5、第 6 这 3 帧图像来实现无限长的视频。表 2 中同样也是通过重复播放中间的  $n$  帧图像来实现无限长的视频。表 2 和表 3 中的重复播放的部分即为最终的播放序列。最终的播放序列是通过图像剖分后重组而形成的一个周期长的播放序列。通过循环播放最终的播放序列可以实现一个贴近真实感的无限的视频。

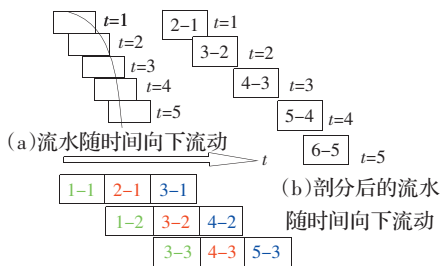


图 4 模拟自然场景

表 4 是没有进行图像剖分和重组的情况,在表 3 中第 3 帧和第 4 帧之间有一跳变(在表中用一条黑色竖线表示)而在表

表 2 周期为  $n$  时图像剖分重组后的播放序列

1-1	2-1	...	$n-1$	1-1	2-1	...	$n-1$	1-1	2-1	...
	2-2	3-2	...	$(n+1)-2$	2-2	3-2	...	$(n+1)-2$	2-2	3-2
		...	...	...	...	...	...	...	...	...
			$n-n$	$(n+1)-n$	...	$(2n-1)-1$	$n-n$	$(n+1)-n$	...	$(2n-1)-1$

表 3 图像剖分重组后的视频播放序列

1-1	2-1	3-1	1-1	2-1	3-1	1-1	2-1	3-1
1-2	2-2	3-2	4-2	2-2	3-2	4-2	2-2	3-2
1-3	2-3	3-3	4-3	5-3	3-3	4-3	5-3	3-3

2 和表 3 中跳变就分散到了整个视频中(用黑色竖线表示跳变的地方)。在表 3 中第 3 帧和第 4 帧之间只有上面部分有跳变,第 4 帧和第 5 帧之间只有中间部分有跳变,由此可见跳变是均匀分散的。而且跳变的部分都是相隔一个周期长度(如 3-1 到 1-1)就使得跳变比较少而且还被均匀分布。因此整个视频就非常流畅。表 2 和表 3 中的重组方法适用于视频中的自然场景向下运动,比如流水、瀑布。但是对于视频中的场景向上运动的情况,比如喷泉、火焰,根据其运动特点对重组顺序进行了改动,如表 5 所示。

表 4 原始视频播放序列

1-1	2-1	3-1	1-1	2-1	3-1
1-2	2-2	3-2	1-2	2-2	3-2
1-3	2-3	3-3	1-3	2-3	3-3

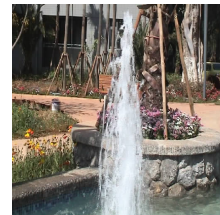


图 5 向上运动的视频

表 5 网格剖分重组后的视频播放序列(向下)

1-3	2-3	3-3	4-3	5-3	3-3	4-3	5-3	3-3
1-2	2-2	3-2	4-2	2-2	3-2	4-2	2-2	3-2
1-1	2-1	3-1	1-1	2-1	3-1	1-1	2-1	3-1

以上情况使用于视频的初步播放序列的周期比较短的情况,但对于纹理比较复杂的视频,其视频的初步播放序列的周期比较长,如周期为 20 帧图像,不可能将图像剖分为 20 个部分再重组。对于这种情况在前面的工作中已经把视频划分成了几个子序列,以子序列为单位进行剖分和重组来合成新的视频序列。表 2 和表 3 中的每一帧被相应的子序列所取代。比如第 1、2、3、4 帧图像组成第一个子序列,而第 13、14、15、16 帧图像组成第 4 个子序列,那么表 4 中的第 4 帧图像由第 1 个子序列和第 4 个子序列重组后取代,如表 6 所示。对图像剖分重组后会导导致图像的内部不连贯,因此在重组的时候要进行纹理合成。

表 6 第 1 个序列和第 4 个序列重组

1-1	2-1	3-1	4-1
13-2	14-2	15-2	16-2
13-3	14-3	15-3	16-3

### 7 纹理合成

在剖分后的图像重组的时候帧与帧之间会有重叠区域,在

决定重叠区域像素来源于哪个区域时采用函数

$$M(s, t, A, B) = \|A(s) - B(s)\| + \|A(t) - B(t)\| \quad (3)$$

来决定。 $A$  和  $B$  表示有重叠区域的两帧图像,  $s$  和  $t$  表示重叠区域中相邻的两个像素,  $A(s)$  表示像素  $s$  在图像  $A$  的颜色,  $B(s)$  表示像素  $s$  在图像  $B$  的颜色,  $A(t)$  表示像素  $t$  在图像  $A$  的颜色,  $B(t)$  表示像素  $t$  在图像  $B$  的颜色。如图 6 所示 1、2、3、4、5、6、7、8、9 为图像  $A$  和图像  $B$  的重叠像素。计算每一横排的相邻像素的函数之间的  $M(s, t, A, B)$  值。然后找出每横排中最小的相邻像素的  $M(s, t, A, B)$  函数值, 在这对相邻像素之间的连线上划一条竖线。竖线左边的像素颜色由图像  $A$  提供, 竖线右边的像素颜色由图像  $B$  提供。如图 6 所示, 1、2、3、5、6 由图像  $A$  提供, 4、7、8、9 由图像  $B$  提供。最终的合成效果图如图 7 所示。图 7 中左边和中间的图像是通过视频的第 1 帧图像和第 4 帧图像如表 4 所示网格重组后进行纹理合成而得到的最终结果图。而右边的图像是视频中的第 5 帧图像和第 2 帧图像网格剖分后进行纹理合成后得到的最终结果图。从图 7 中可以看出纹理合成后消除了图像剖分重组后图像的内部不连贯。

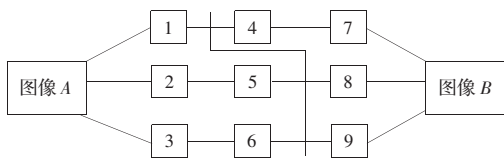


图 6 重叠区域的像素



图 7 纹理合成的图像

## 8 结语

提出了一种改进的纹理视频合成算法, 算法通过衡量图像之间的相似度来确定视频的初步播放序列, 然后将初步播放序列划分成几个子序列并调整视频播放过程中的帧与帧之间的延迟, 与以往的算法不同的是: 算法在合成视频前先确定初步

播放序列的周期, 然后根据初步播放序列的周期, 选取用于合成最终的播放序列的图像。此外, 算法还根据周期的长短选择以一个单帧图像或者子序列作为单位对被选取的图像进行剖分, 然后重组来合成一个最终的无限长的播放序列。

该文提出的算法能在简单的纹理视频和复杂的纹理视频上都能有很好的效果, 使得视频纹理贴近真实的效果。

在未来的工作中, 要将纹理视频应用于能和用户交互的全景漫游场景中。纹理视频和非真实感绘制相结合也是未来值得研究的方向。

## 参考文献:

- [1] Fournier A, Fussell D, Carpenter L. Computer rendering of stochastic models[J]. Communications of the ACM, 1982, 25(6): 371-384.
- [2] Debevec P, Bregler C, Cohen M, et al. Image-based modeling and rendering[C]//Proceedings of ACM SIGGRAPH 1999, Los Angeles, California, 1999.
- [3] Schödl A, Szeliski R, Salesin D H, et al. Video textures[C]//Proceedings of ACM SIGGRAPH 2000, New Orleans, Louisiana, 2000: 489-498.
- [4] Schödl A, Essa I. Controlled animation of videosprites[C]//Proceedings of ACM SIGGRAPH 2002, San Antonio, Texas, 2002: 121-127.
- [5] Kwatra V, Schödl A, Essa I, et al. Graphcut textures: Image and video synthesis using graph cuts[C]//Proceedings of ACM SIGGRAPH 2003, San Diego, California, 2003, 22(3): 277-286.
- [6] Bhat D H, Seitz S M, Hodgins J K, et al. Flow based video synthesis and editing[C]//Proceedings of ACM SIGGRAPH 2004, Los Angeles, California, 2004, 23(3): 360-363.
- [7] Reeves W T. Particle systems-A technique for modeling a class of fuzzy objects[J]. ACM Transactions on Graphics, 1983, 2(2): 91-108.
- [8] Smith A R. Plants, fractals and formal languages[C]//Proceedings of ACM SIGGRAPH 1984, Minneapolis, Minnesota, 1984, 18(3): 1-10.
- [9] Doretto G, Chiuso A, Soatto S, et al. Dynamic textures[J]. International Journal of Computer Vision, 2003, 51(2): 91-109.
- [10] Boykov Y, Kolmogorov V. An experimental comparison of min-cut/max-flow algorithms for energy minimization in computer vision[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004, 26(9): 1124-1137.
- [11] Agarwala A, Zheng Keolin. Panoramic video textures[C]//ACM SIGGRAPH 2005 Conference Proceedings, 2005: 821-827.

(上接 156 页)

性退化模型求取出参数。该方法的特点是操作简便, 不需要其他的辅助信息, 其主要的不足总结为以下几个方面:

(1) 只讨论了人脸左右方向的旋转, 并假设所有输入图像没有俯仰旋转。实际上人脸姿态的变化是非常复杂的, 可绕空间正交的 3 个轴同时进行旋转, 如此使得系统的实用性受到了一定程度的限制。

(2) 样本图像是对特定人群和特定姿态的采集而得到的, 图像质量较高, 但任意的待合成图像会由于噪声、面部表情或服饰遮盖物、成像条件差等因素造成失真。

(3) 作为一种基于统计意义的方法, 其效果在很大程度上受样本分布情况的影响, 如性别、年龄、种族等, 而样本数量的增加势必也会增加计算开销。

下一步的研究重点是建立更合理更稳定的统计数据模型, 随着面貌特征分析使用的基础数据进一步丰富、翔实, 以及面

部采集系统的不断普及和完善, 进一步提高合成效果的准确度。

## 参考文献:

- [1] 李月敏, 陈杰, 高文, 等. 快速人脸检测技术综述[C]//全国第 16 届计算机科学与技术应用(CACIS)学术会议论文集, 2004.
- [2] 朱长仁, 王润生. 基于单视图的多姿态人脸识别算法[J]. 计算机学报, 2003, 26(1): 104-109.
- [3] 黄万军, 尹宝才, 陈通波, 等. 基于三维可变形模板的眼睛特征提取[J]. 计算机研究与发展, 2002(4): 495-501.
- [4] Cootes T F, Edwards G J, Taylor C J. Active appearance models[C]//Burkhardt H, Neumann B. Proc of the 5th European Conf on Computer Vision. [S.l.]: Springer-Verlag, 1998: 484-498.
- [5] Hou X, Li S, Zhang H, et al. Direct appearance models[C]//Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition(CVPR). Hawaii: IEEE, 2001: 828-833.
- [6] 周伟华. 基于统计模型的人脸图像理解[D]. 2002.
- [7] CVL Face database[EB/OL]. <http://www.lrv.fri.uni-lj.si/facedb.html>.