一种无标记点人脸表情捕捉与重现算法

吴晓军[1,2],鞠光亮[1]
(1. 哈尔滨工业大学深圳研究生院,广东深圳 518055;
2. 深圳先进运动控制技术与现代自动化装备重点实验室,广东深圳 518055)

摘 要: 提出了一种无标记点的人脸表情捕捉方法. 首先根据ASM(Active Shape Model)人脸特征点生成了覆盖人脸85%面部特征的人脸均匀网格模型;其次,基于此人脸模型提出了一种表情捕捉方法,使用光流跟踪特征点的位移变化并辅以粒子滤波稳定其跟踪结果,以特征点的位移变化驱动网格整体变化,作为网格跟踪的初始值,使用网格的形变算法作为网格的驱动方式. 最后,以捕捉到的表情变化数据驱动不同的人脸模型,根据模型的维数不同使用不同的驱动方法来实现表情动画重现,实验结果表明,提出的算法能很好地捕捉人脸表情,将捕捉到的表情映射到二维卡通人脸和三维虚拟人脸模型都能取得较好的动画效果.

关键词: 无标记点跟踪; 人脸表情捕捉; 表情动画重现
中图分类号: TP391 **文献标识码:** A **文章编号:** 0372-2112 (2016)09-2141-07
电子学报URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2016.09.018

A Markerless Facial Expression Capture and Reproduce Algorithm

WU Xiao-jun[1,2], JU Guang-liang[1]
(1. *Shenzhen Graduate School, Harbin Institute of Technology, Shenzhen, Guangdong* 518055, *China*;
2. *Shenzhen Key Laboratory for Advanced Motion Control and Modern Automation Equipment, Shenzhen, Guangdong* 518055, *China*)

Abstract: This paper presents a markerless facial expression capture and reproduce algorithm. Firstly, an uniform mesh model is built based on the feature points from ASM(Active Shape Model). It can cover 85 percent of the face. Then, a method to capture facial expression based on the face model is proposed. The optical flow tracking is used to track the feature points with particle filter for stablizing the result. The feature points' displacement can drive the overall mesh to deform as the initial value of the mesh tracking. Finally, the captured expression data is used to drive face models with different methods for the models of different dimensions, and then the facial animation can be reconstructed. Experimental results show that the proposed algorithm can capture facial expressions well, and the animation effect is good when mapping the captured expression to 2D cartoon or 3D virtual face models.

Key words: marker-less tracking; facial expression capture; facial animation reconstruction

1 引言

人脸表情捕捉是近年来兴起的一个热点研究课题. 首先,在动画产业中,为了使动画人物的表情更加逼真,对于表情捕捉技术的要求也更高. 其次,在其他领域也有着重要的作用,例如在虚拟现实、医疗模拟、人机交互、游戏娱乐等领域,人脸表情捕捉的研究越来越受到更多的重视. 作为这些领域中一种重要的数据信息获取方法,人脸表情捕捉技术涉及多个学科,如计算机视觉、图像处理、计算机图形学、人工智能等. 根据人脸表情捕捉的发展阶段可以分为两类:

基于标记点的捕捉算法: 多数学者使用比较成熟的商业捕捉系统(如Vicon等)作为他们的数据获取方法,例如:Deng[1]使用三个摄像机捕捉102个标志点的变化实现语音表情动画,Sifakis[2]也同样使用Vicon摄像机跟踪79个特征点实现人脸的医学重构,Curio[3]使用6个摄像机跟踪69个特征点实现人脸表情的交叉映射,等等. 然而采用商业系统,一方面价格较高,另一方面不能满足多样性的需求,所以不少学者研究低成本的人脸表情捕捉系统. 但是单从硬件设备而言,普遍选择的捕捉频率较高、性能较好的相机. 如Papic[4]等人只是用两个便携式摄像机,组合帧接收器将数据传输给计算机,使用直接线性变换将捕获的点的信息处理成可用信息. 微软公司的Guenter[5]等人使用6种不同颜

收稿日期:2015-01-09;修回日期:2015-10-13;责任编辑:梅志强
基金项目:深圳市科技计划项目(No. JCYJ20150928162432701, No. JCYJ20140417172417166)

色的标记点,6 个高分辨率的摄像机在不同的角度对人脸进行捕捉,获得数据后根据不同颜色点来识别并进行模板的训练,利用图像处理的技术计算三维点信息. Lin[6]等提出只用一台摄像机就可以获取三维信息的方法,使用两面镜子反射贴满荧光标记点的人脸. 在拍摄时,用紫光灯照射人脸,荧光标记点反射效果明显,在图像上有很高的对比度,便于跟踪. 然后利用空间几何原理计算点的三维信息,获得初始的捕捉数据. Bickel[7]等在人脸上涂抹彩色颜料,蓝色的标记点用于跟踪整体表情变化,而其它区域的彩色条纹用于跟踪细节皱纹的变化. 候文广等[8]提出通过投影不同类型的纹理实现真实人脸几何重建.

无标记点的表情捕捉算法:此类算法的研发还处于发展的初步阶段,发表的论文较少,比较典型的有:王玉顺等提出一种基于二维表情动作约束的三维人脸动画数据编辑与合成的有效方法,根据实现训练的人脸动画先验概率模型,将较少的用户约束传播到人脸网格的其他部分,从而生成完整生动的人脸表情[9]. 虽然文献[10,11]和本文都是无标识点的表情捕捉再现,但在算法本质上是不同的,文献[10,11]是基于学习的方法,首先利用了先验的三维表情数据库,以及 2D 表情图像标记点获得训练数据,通过训练图像和形状数据以及用户特定混合模型库(User-specific Blendshapes)进行 3D 回归学习,获得用户特定 3D 形状回归量,在运行时利用学习获得的回归量实现表情跟踪. 杜志军等提出基于梯度场修改方法表情映射的单张照片的人脸动画系统,采用 3D 人脸模型信息实现输入图像头部姿态的调整[12]. Weise 等[13]使用微软公司开发的 Kinect 体感外设作为他们的数据捕捉工具,Kinect 可以同时拍摄二维的图像和获得三维的位置信息,具有高速和同步性等优点,只是它所获取的三维信息的噪声和误差较大,需要软件方法去噪和修正. Weise 等使用训练的人脸表情模型,使用模型的匹配来限制噪声的影响,这样做的好处是有很好的实时性,但是人物的表情变化就有些单调,缺乏变化性和自由度. Zhang 等人[14]使用结构光照射在无标记点的人脸上,计算两幅图像序列的深度差,将深度与人脸模板做匹配,以便驱动模型变化. 此方法对光照的要求较高,在一般情况下使用则较为繁琐. Sibbing 等[15]使用 5 个同步相机拍摄无标记点的人脸,主要使用 Surfel fitting[16]人脸三维重建的方法,辅以二维网格跟踪建立帧与帧之间的联系实现人脸的表情动画重现,而由于需要重建人脸表情,所消耗的时间则较多. Hwang 等[17]使用特征点的跟踪方法来捕捉人脸的表情变化,并用捕捉的数据驱动人脸模型变化. 由于特征点只分布在局部区域,所描述的人脸表情也就存在局限性. 而本文的最终目的是将捕捉的表情传递给不同的模型,以网格作为表情的描述方法可以更高效的传递表情变化,所以本文选用网格跟踪作为主要捕捉手段,虽然本文使用的网格跟踪方法与 Sibbing 的方法相类似,但在速度方面有很大的提高. 本文也使用了特征点的跟踪,但只是作为网格跟踪的辅助,而网格可以覆盖大部分人脸面部区域,相对于特征点所能描述的表情更多、更真实.

2 算法概述

本文研究的内容主要围绕在无标记点人脸表情的捕捉及传递的算法的各个环节展开,包括人脸的表情建模、表情捕捉和人脸表情动画三个部分工作,主要包括:首先根据捕捉方法的不同,建立一个新的描述人脸表情的网格模型. 基于 ASM(Active Shape Model)算法获取人脸上特征区域的点的描述[18],根据特征点的位置对人脸区域进行均匀的三角网格化,以达到覆盖大部分人脸区域,可以描述绝大多数的人脸面部的表情变化. 然后,由于所拍摄的人脸是无标记点的,没有明显的跟踪目标,传统的跟踪方法失效. 本文采用两步法实现表情跟踪,首先采用传统的光流跟踪辅以粒子滤波的方法跟踪人脸的局部特征点,根据特征点的变化驱动网格点的位移变化. 然后采用基于灰度分布变化较小特性的网格跟踪方法,以求实现对人脸面部表情的全方位捕捉. 并且提高网格跟踪的速度. 最后,生成二维和三维脸部表情动画,二维的表情动画的目标人脸是单幅的卡通人物脸部图像,而三维的目标人脸是将捕捉的表情映射到三维人脸网格模型上. 在形成动画效果之前,首先要建立不同的人脸模型间的映射投影关系,主要采用的是径向基函数网络训练方法,而关系建立完成之后,两种模型的变化方式也有不同之处,二维模型除了平面空间内的网格位移还要添加网格内点的灰度值,而三维模型主要难点在于网格点的所在空间是三维的,主要采用求解曲面微分方程的方法计算网格点的位移.

3 人脸模型建立

在一张人脸的照片中,比较明显的特征有眼睛、鼻子、嘴巴和下巴的轮廓线,本文采用 ASM 算法来提取这些特征,如图 1 所示. 通过 ASM 算法可以得到 60 个点来描述人脸,但不是所有的点都会用到,本文只提取其中的部分点,从上到下依次是:眉毛 4 个、眼睛 8 个、鼻子 9 个、嘴巴 8 个和下巴 15 个,并且对部分点的位置进行了调整,以便于以后的处理和跟踪.

为了获得更多脸部的表情信息,本文将采用将面部三角网格化[19]的方法来描述人的表情. 三角网格化的基础就在于特征点的提取,以特征点为边界可以很快的生成三角网格,如图 2 所示,可以看出如此密集的


图1　ASM算法得到的点描述图

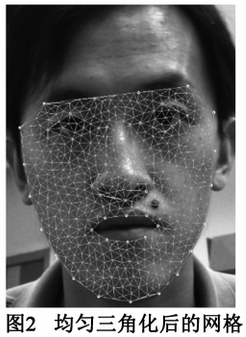


图2　均匀三角化后的网格

三角形覆盖了人脸的大部分区域,也是人脸表情最丰富的区域,这样就可以比较准确地描述人的面部表情了. 脸部的表情变化的同时三角形的大小、位置也同样跟着变化,以达到对表情捕捉的效果.

4　表情捕捉算法

4.1　特征点的跟踪

在逐帧的跟踪特征点的变化时,本文使用的是光流跟踪方法 LK[20](Lucas-Kanade). 在利用 LK 算法跟踪人脸上的特征点时,特征点之间是相互独立的,为每个特征点建立一个跟踪序列,将跟踪结果保存在序列中. 由于两幅图像之间的变化并不巨大,也就是说特征点的位移也不明显,所以可以用一个 $m \times n$ 的像素区域代替整张图像作为特征点的活动范围,以减小跟踪时的计算量. 根据点的位置不同设定的跟踪区域也不同,例如:眼睛上的点多为纵向移动,横向移动较小,区域设为 5×15 即可;而嘴巴上的点无论是横向还是纵向,移动距离都会较大,所以区域设为 15×15. LK 算法的优点就在于它的耗时较少,但是缺点也比较明显,就是跟踪的误差较大,稳定性不高,尤其是在像素点的灰度值区别度不高和变化幅度较大时,很容易出现错误跟踪,如图 3 所示:嘴巴张开后再闭合时,右侧的跟踪点出现在错误的位置,这种错误的跟踪点一旦出现就很难再挽回,以后的跟踪也就失去了意义,所以本文使用粒子滤波的方式来稳定 LK 的跟踪结果. 本文采用 CONDENSATION[21] 粒子滤波方法简洁、特征点跟踪稳定、计算效率高,具体的使用方法与文献[22]相似.

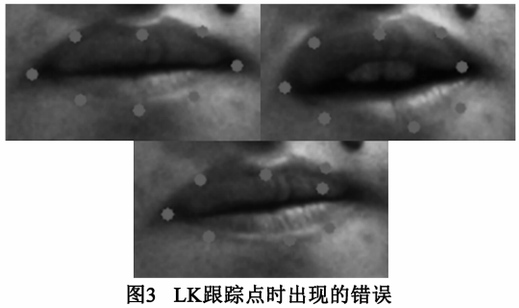


图3　LK跟踪点时出现的错误

4.2　网格的跟踪

网格跟踪的目的是建立帧与帧之间网格点的位置关系,也就是网格点的位置变化情况. 以建立的网格模型为基础,本文提出一种新的跟踪方法,达到对人脸大部分区域表情的捕捉效果. 主要的算法流程如图 4 所示.

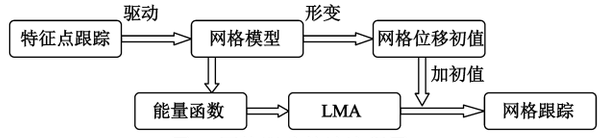


图4　网格跟踪算法流程图

定义初始的第一帧中生成的网格为 M_1,一系列连续的图片定义为:

$$I_1, \cdots, I_{n-1}, I_n, I_{n+1}, \cdots$$

在已知的两帧连续的图像 I_n 和 I_{n+1} 中,去寻找网格 M_n 的所有点的位置变化 $\boldsymbol{d}_i = [d_{i,x}, d_{i,y}]^{\mathrm{T}} \in R^2$,主要的根据就是在连续的两帧图像之间,对应的三角形中的人脸灰度值分布变化不大,如图 5 所示,虽然三角形的形状和位置都变了,但是三角形内的像素灰度却没有像位置和形状那样改变很多. 图像 I_n 中三角形的一个点 $\boldsymbol{p} = [x, y]^{\mathrm{T}}$ 可以通过线性映射 $f: R^3 \rightarrow R^3$,映射到变化后的图像 I_{n+1} 中的另一个三角形中,映射 f 可以通过对应的三角形的顶点求出:

$$\begin{pmatrix} \boldsymbol{X}_{n+1} \\ \boldsymbol{Y}_{n+1} \\ 1 \end{pmatrix} = f \begin{pmatrix} \boldsymbol{X}_n \\ \boldsymbol{Y}_n \\ 1 \end{pmatrix} \tag{1}$$

式中,$\boldsymbol{X}_{n+1}$ 为图像 I_{n+1} 中三角形顶点的 x 坐标向量,$\boldsymbol{X}_{n+1} = (x_{n+1}^0, x_{n+1}^1, x_{n+1}^2) = (x_n^0 + d_x^0, x_n^1 + d_x^1, x_n^2 + d_x^2)$;$\boldsymbol{Y}_{n+1}$ 为图像 I_{n+1} 中三角形顶点的 y 坐标向量,$\boldsymbol{Y}_{n+1} = (y_{n+1}^0, y_{n+1}^1, y_{n+1}^2) = (y_n^0 + d_y^0, y_n^1 + d_y^1, y_n^2 + d_y^2)$.

定义 $I_n(\boldsymbol{p})$ 为点 $\boldsymbol{p}$ 在图像 I 中的灰度值,那么两幅图像对应三角形之间的灰度差异可以表述为:

$$E_T = \left(\sum_{p \in T} I_n(\boldsymbol{p}) - \sum_{p \in T} I_{n+1}(f(\boldsymbol{p})) \right)^2 \tag{2}$$

式中,T 表示网格中的三角形,E_T 是三角形之间的灰度和差异.

与文献[15]中的不同之处在于,本文使用三角形内的灰度总和之差取代单个点的灰度值之差,那么两帧图像之间的灰度差异就是所有三角形的差异之和:

$$E_{\text{data}} = \sum_{T \in M_n} E_T = \sum_{T \in M_n} \left(\sum_{p \in T} I_n(\boldsymbol{p}) - \sum_{p \in T} I_{n+1}(f(\boldsymbol{p})) \right)^2 \tag{3}$$

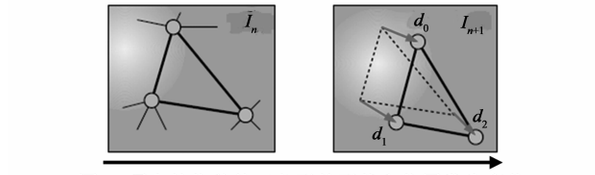


图5　顶点的位移使三角形的形状和位置发生变化

式中,M_n 是网格序列,E_{data}是网格之间的灰度和差异,将方程式(1)中所求出的f带入式(3)中即可得到由于位移 $\boldsymbol{d}_i$ 所产生的新的 E_{data}.

只要求出使差异 E_{data}最小的位移 $\boldsymbol{d}_i$,$\boldsymbol{d}_i$ 就是网格点的位移量,为了保证所有网格点的位移的平滑度本文还引入平滑函数[15]作为另一个能量方程:

$$E_{\text{smooth}} = \sum_{i\in V(M_n)}\left(\frac{1}{\omega_i}\sum_{j\in N_i}\omega_{i,j}\|\boldsymbol{d}_i - \boldsymbol{d}_j\|\right)^2 \tag{4}$$

式中,$V(M_n)$是网格 M_n 中点的序列号,N_i 是点 $\boldsymbol{p}_i$ 周围的一阶邻域,$\omega_{i,j}$ 则是标准的 chordal 权重 $\omega_{i,j} = \|\boldsymbol{p}_i - \boldsymbol{p}_j\|^2$,$\omega_i = \sum_{j\in N_i}\omega_{i,j}$.

将数据能量方程式(3)和平滑能量方程式(4)合并后即为最终的能量方程:

$$E = E_{\text{data}} + \lambda E_{\text{smooth}} \tag{5}$$

式中,λ 是控制平滑度的参数,可根据情况不同进行调整.

当能量函数式(5)取最小值时,所得到的位移就是网格点的跟踪结果. 而求最小值则使用 Levenberg-Marquardt(LM)算法. 当然 LM 算法有着明显的缺陷,就是它所求得的最小值依赖于起始点的位置,也就是说只有好的初始值,才能得到好的跟踪结果. 之所以不使用其他的求取最小值的方法的原因在于:一是求取全局最小值的方法势必要更多的计算,算法的时间复杂度也同样会升高;二是本文提出了一种计算较好的网格位移初值的方法,可以解决依赖初值的问题.

本文采用根据特征点的位移来计算网格点的初始位移,也就是说一种网格形变的方式,这种形变方法的优点是可以保持网格的连续平滑性和局部特征性,前提是将网格线性化,网格的形变分为伸展和扭曲两种类型. 而网格的伸展和扭曲的能量函数可以用位移矢量 $\boldsymbol{d}_i$ 的一阶和二阶偏导数的积分来表示. 由于本文中使用的网格是在平面区域内进行变形所以,变形方法与文献[7]中的有所区别,省略网格的扭曲,只保留网格的伸展能量函数:

$$\int_M\left(\left\|\frac{\partial \boldsymbol{d}_i}{\partial u}\right\|^2 + \left\|\frac{\partial \boldsymbol{d}_i}{\partial v}\right\|^2\right)\mathrm{d}u\mathrm{d}v \tag{6}$$

而求取使此能量函数最小的位移 $\boldsymbol{d}_i$ 时,可以将此函数转化为其相对应的 Euler-Lagrange 方程:

$$\Delta \boldsymbol{d}_i = 0 \tag{7}$$

式中,Δ 是离散的 Laplace-Beltrami 算子[21]:

$$\Delta(\boldsymbol{x}_i) := \frac{2}{A(\boldsymbol{x}_i)}\sum_{\boldsymbol{x}_j\in N(\boldsymbol{x}_i)}(\cot\alpha_{ij} + \cot\beta_{ij})(\boldsymbol{x}_j - \boldsymbol{x}_i) \tag{8}$$

式中,α_{ij},β_{ij}如图 6 所示,是线段 $\boldsymbol{x}_i\boldsymbol{x}_j$ 对应的两个夹角,$\alpha_{ij} = \angle(\boldsymbol{x}_i, \boldsymbol{x}_{j-1}, \boldsymbol{x}_j)$和 $\beta_{ij} = \angle(\boldsymbol{x}_i, \boldsymbol{x}_{j+1}, \boldsymbol{x}_j)$,$A(\boldsymbol{x}_i)$如图 7 所示,为顶点 $\boldsymbol{x}_i$ 周围的 Voronoi 面积[19].

在求解方程式(7)时,可以将其转化为离散的线性

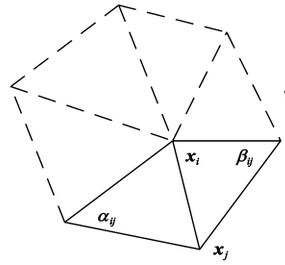


图6 相对应的角α和β

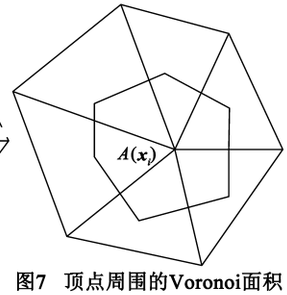


图7 顶点周围的Voronoi面积

系统:

$$\left(\begin{array}{c|c}\Delta & \\ \hline 0 & \boldsymbol{I}_H\end{array}\right)\begin{pmatrix}\boldsymbol{p}\\ \boldsymbol{h}\end{pmatrix} = \begin{pmatrix}0\\ \boldsymbol{h}_b\end{pmatrix} \tag{9}$$

式中,$\boldsymbol{p}$ 为跟随其他点移动的点位移,$\boldsymbol{p} = (p_1, \cdots, p_P)$,$P$ 等于网格点数与特征点数之差;$\boldsymbol{h}$ 为控制移动的点位移,$\boldsymbol{h} = (h_1, \cdots, h_H)$,$H$ 等于特征点的数量;$\boldsymbol{h}_b$ 是控制点位移的边界条件;$\boldsymbol{I}_H$ 是秩为 H 的单位矩阵.

方程式(9)的等号左边在第一帧图像时就可以确定,以后每一帧时只需要改变等号右边的边界条件. 由于等号右边的矩阵是稀疏的,在求解方程时可以利用系数矩阵的 Cholesky 分解,速度上可以满足要求. 求解次方程得到的结果就是网格上每个点的初始位移,可以作为 Levenberg-Marquardt 算法的初始值.

5 表情重现方法

在捕捉到人脸的表情变化之后,我们目的是让目标人脸模型做出与捕捉数据相同的表情变化,也就是人脸的表情传递过程. 这个过程在本文中分为两个部分:一是人脸模型间的相互对应关系的建立,二是人脸模型的表情驱动方法. 建立人脸间的对应关系的方法很多,由于采用网格化的方法来描述人脸,基于网格的人脸匹配常用是径向基函数(Radial Basis Function, RBF)方法. RBF 由于其很强的插值能力而为众人所熟知,尤其是在人脸的模型匹配领域也有着突出的作用. RBF 的基本形式如式(10).

$$f(\boldsymbol{x}_i) = \sum_{j=1}^{n} w_j h_j(\boldsymbol{x}_i) \tag{10}$$

式中,$\boldsymbol{x}_i$ 是输入的向量,$f(\boldsymbol{x}_i)$为对应的输出向量,w_j 为权重系数,n 是训练输入的数量,h 是基函数. 本文中采用 Multi-quadrics 作为 RBF 的基函数式(11).

$$h_j(\boldsymbol{x}_i) = \sqrt{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 + s_j^2} \tag{11}$$

式中,s_j 是 $\boldsymbol{x}_j$ 到离其最近的 $\boldsymbol{x}_i$ 的距离:$s_j = \min_{i\neq j}\|\boldsymbol{x}_i - \boldsymbol{x}_j\|$.

Multi-quadrics 基函数的优点在于可以使那些离训练中心点较近的点有着更明显更准确的变化,而使那些离中心点较远的点有着更平滑的变化[23]. 训练 RBF 网络时可以用已知对应关系的点的坐标作为输入向量,训练出权重系数 $\boldsymbol{w}$ 向量. 然后通过式(10)即可求出

其他对应点的坐标.

建立好对应关系后,下一步的网格的驱动就是希望网格按着给定的位移和方向做出变化,就二维网格而言,它的变化就是在一个平面内的变化,相对三维来说要简单些. 由于原始的人脸模型与目标人脸模型的所在的空间不同,尺度大小也就不同,所以需要进行驱动位移量的尺度调整. 原始网格捕获的位移为 $\boldsymbol{d}_o$,目标网格的位移为 $\boldsymbol{d}_t$,它们之间的转化关系为 s_o^t 用式(12)表示.

$$\boldsymbol{d}_t = s_o^t \boldsymbol{d}_o \tag{12}$$

而这个转化关系可以通过两个网格的平均边长获得:

$$s_o^t = \frac{m\sum_{i=1}^{n} b_{t,i}}{n\sum_{i=1}^{m} b_{o,i}} \tag{13}$$

式中,$b_{t,i}$为目标网格的三角形边长;$b_{o,i}$为原始网格的三角形边长;n 为目标网格三角形边的总数;m 为原始网格三角形边的总数.

相对于二维人脸的表情重现而言,三维模型的驱动要复杂些. 首先,需要采用双目相机的同步捕捉数据,根据网格对应点的关系和相机标定数据重建出捕捉的人脸表情. 同样使用 RBF 网络建立第一帧捕捉数据与目标 mesh 模型的对应关系,然后利用捕捉数据重建出低精度的三维人脸,根据重建后人脸的点位移向量驱动目标模型中的相对点做出相同的移动,目标模型的网格形变方法使用文献[7]中的方法. 由于形变方法中有平滑的效果,所以可以弥补重建人脸精度低的缺点. 在驱动网格变化之前,为了保持目标模型的某些平面的局部特征,需要调整驱动向量的方向,同时为了保持形变的逼真度,也需要调整驱动向量的尺度大小,调整方法可以参考文献[24].

6　实验结果

为验证本文算法的有效性,我们采用两台 Imaging-source 工业 CCD 相机搭建了图像实时采集平台,图像分辨率为 1024 × 768. 系统采用 VS2008 编程环境实现,图形和图像显示分别调用了 OpenGL 和 OpenCV2. 1 函数库. 计算机硬件配置为 Intel Core i5-3470 3. 2GHz CPU,4G 内存. 搭建好拍摄系统,在获取到第一帧图像后,识别出人脸位置,得到局部特征点并均匀网格化. 随后逐帧地对人脸进行特征点的跟踪和网格跟踪. 图 8 中特征点分布在鼻子、嘴唇和下巴等区域. 网格均匀分布在人脸面部,如图 8 中第二行图像所示. 400 帧图像总耗时 25. 489s,表情捕捉时间约为 63. 7ms/帧,可以做到实时捕捉. 文献[15]中的算法 400 帧图像总耗时 154. 394s,约为 386ms/帧.

将捕捉到的实时脸部表情映射生成二维表情动画时,选取动漫人物为目标人脸模型,将原始模型的网格利用 RBF 网络投影到目标模型上,产生新的网格,由于人物脸部特征的区别较大,产生的网格与有原始网格也有较大出入,但是并不影响动画效果,因为只要是三角形就可以相互映射,可以使用双线性插值方法对像素的颜色信息进行插值填补,由人物表情驱动的二维卡通动画效果如图 9 所示,动画人物与原始人脸表情上下相对应. 图 10 为第二组视频序列跟踪结果及生成的二维卡通动画.

图 10 中第二行为网格跟踪的映射动画效果、第三行则是只有特征点的跟踪动画效果. 可以看出如果只用特征点则表情的变化只取决于特征点的跟踪精度,

图8　特征点和网格的跟踪效果

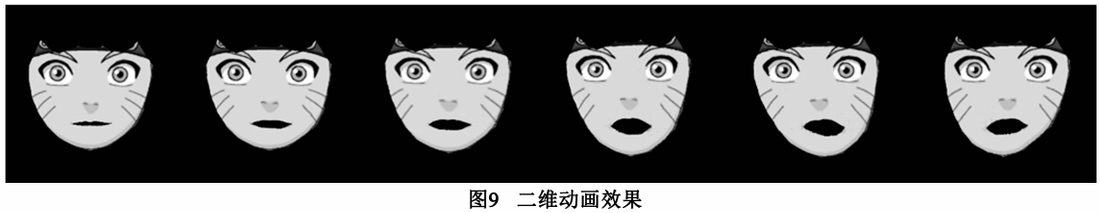


图9　二维动画效果

人脸面部尤其是颧骨部分则无法跟踪. 而网格跟踪算法则可以很好的弥补这一点,从卡通人物的胡须变化中可以看出两者的跟踪区别. 第四及第五行则是与二、三行相对应的表情的网格变化,从图中可以看出网格跟踪时网格的变形均匀和平滑.

生成三维动画时,首先也需要将第 1 帧图像重建好的三维人脸投影到目标人脸上,同样使用 RBF 训练网络. 图 11 中目标人脸上的红点即为原始模型的网格点所对应的驱动点. 将捕捉到的数据变换好后,驱动目标模型变化,目标模型就可以做出相同的表情变化.

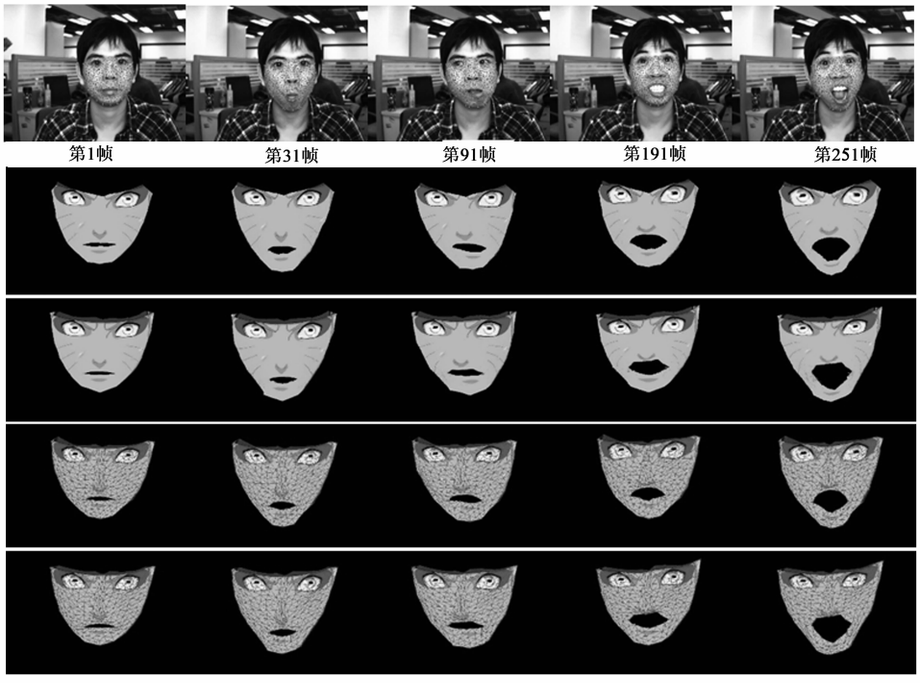


图10 第二组视频跟踪结果及二维动画效果

图11 三维表情动画效果

7 结论

本文针对无标记点人脸表情捕捉技术的几个重点问题进行深入研究,建立了一个新的人脸表情描述模型,直接利用被捕捉对象的表情数据和动画模型间建立表情映射,是一种新方法的探索和尝试. 根据新的人脸模型,提出一种基于特征点跟踪,驱动网格形变以完善网格跟踪的人脸表情捕捉算法. 使用 RBF 训练网络建立不同人脸模型间的网格点的对应关系,根据人脸模型的维数不同采用不同的表情驱动技术. 利用网格形变算法完成不同人脸模型的表情动画重现. 搭建实验系统对捕捉算法进行验证,取得比较好的实验效果.

本文算法存在的不足在于,捕捉和再现三维人脸的细节信息和稳定性有待提高,尤其是两幅图像序列同时捕捉并重建三维人脸模型时,需要保证三维模型的准确性和平滑度. 三维表情再现时使用双目视频作为数据源,深度信息较少,不能获取更为精细的三维人脸变形数据. 对三维人脸的细节表情捕捉和再现是本算法需要进一步研究的地方,未来的工作可以结合文献[10,11]中的特征点的跟踪方法稳定跟踪方法,并针对复杂表情以及面部的肌肤纹理变化做跟踪,会实现更逼真的动画效果.

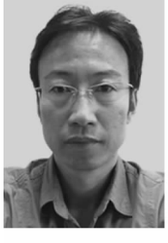
参考文献

[1] Deng Z, Chiang P, Fox P, et al. Animating blend shape faces by cross-mapping MoCap data[A]. ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games[C]. New York, USA: ACM, 2006. 43 - 48.

[2] Sifakis E, Neverov I, Fedkiw R. Automatic determination of facial muscle activations from sparse MoCap marker data[J].

ACM Transactions on Graphics,2005,24(3):417 - 425.
[3] Curio C, Breidt M, Vuong Q, et al. Senmantic 3D motion retargeting for facial animation[A]. ACM International Conference Proceeding Series[C]. New York, USA: ACM,2006. 77 - 84.
[4] Papic V, Zanchi V, Cecic M. Motion analysis system for identification of 3D human locomotion kinematics data accuracy testing[J]. Simulation Modelling Practice and Theory,2004,12(2):159 - 170.
[5] Guenter B, Grimm C, Wood D, et al. Making faces[A]. In Proceedings of SIGGRAPH[C]. Los Angeles, California, USA: ACM,1998. 55 - 66.
[6] Lin I C, Yeh J S, Ouhyoung M. Extracting 3D facial animation parameters from multiview video clips[J]. IEEE Computer Graphics and Applications,2002,22(6):72 - 80.
[7] Bickel B, Botsch M, Angst R, et al. Multi-scale capture of facial geometry and motion[J]. ACM Transactions on Graphics,2007,26(7):33 - 45.
[8] 侯文广,陈大为,丁明跃. 一种基于多重影像实现真实感人脸三维重建的方法,电子学报,2008,36(4):661 - 666.
Hou Wenguang, Chan Dawei, Ding Mingyue. A novel way achieving 3D Reconstruction of actual human face using multi images[J]. Acta Electronica Sinica, 2008, 36(4): 661 - 666. (in Chinese)
[9] 王玉顺,肖俊,庄越挺,王宇杰. 基于运动传播和 Isomap 分析的三维人脸动画编辑与合成[J]. 计算机辅助设计与图形学学报,2008,20(12):1590 - 1595.
Wang Yushun, Xiao Jun, Zhuang Yueting, Wang Yujie. Editing and synthesis of 3D facial animation by motion propagation and isomap analysis[J]. Journal of Computer Aided Design & Computer Gaphics,2008,20(12):1590 - 1595. (in Chinese)
[10] Yanlin Weng, Chen Cao, Qiming Hou, et al. Real-time facial animation on mobile devices[J]. Graphical Models, 2014,76(3):172 - 179.
[11] Chen Cao, Yanlin Weng, Stephen Lin, et al. 3D shape regression for real-time facial animation[J]. ACM Transactions on Graphics,2013,32(4):41:1 - 41:10.
[12] 杜志军,王阳生. 单张照片输入的人脸动画系统[J]. 计算机辅助设计与图形学学报,2010,22(5):1188 - 1193.
Du Zhijun, Wang Yangsheng. A facial animation system based on single image[J]. Journal of Computer Aided Design & Computer Gaphics,2010,22(7):1188 - 1193. (in Chinese).
[13] Weise T, Li H, Van Gool L, et al. Face/Off: live facial puppetry[A]. Symposium on Computer Animation[C]. New York USA: ACM,2009. 7 - 16.
[14] Zhang L, Snavely N, Curless B, et al. Spacetime faces: high-resolution capture for modeling and animation[A]. International Conference on Computer Graphics and Interactive Techniques[C]. New York USA: ACM,2004. 548 - 558.
[15] Sibbing D, Habbecke M, Kobbelt L. Markerless reconstruction and synthesis of dynamic facial expressions[J]. Computer Vision and Image Understanding, 2011, 115(5):668 - 680.
[16] Habbecke M, Kobbelt L. Iterative multi-view plane fitting[A]. Vision, Modeling and Visualization[C]. Aachen, German: AKA IOS,2006. 73 - 80.
[17] Hwang Y, Kim J B, Feng X, et al. Markerless 3D facial motion capture system[A]. The Engineering Reality of Virtual Reality, Burlingame[C]. California, USA: SPIE, 2012. 8289,6.
[18] Cootes T F, Taylor C J, Cooper D H, et al. Active shape models—their training and application[J]. Computer Vision and Image Understanding,1995,61(1):38 - 59.
[19] Shewchuk J R. Triangle: engineering a 2D quality mesh generator and delaunay triangulator[J]. Lecture Notes in Computer Science,1996,1148:203 - 222.
[20] Baker S, Matthews I. Lucas-Kanade 20 tears on: a unifying framework[J]. International Journal of Computer Vision,2004,56(3):221 - 255.
[21] Isard M, Blake A. Icondensation: Unifying low-level and high-level tracking in a stochastic framework[J]. Lecture Notes in Computer Science,1998,1406:893 - 908.
[22] Blake A, Isard M. 3D Position, attitude and shape input using video tracking of hands and lips[A]. Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques[C]. New York, USA: ACM, 1994. 185 - 192.
[23] Eck M. Interpolation methods for reconstruction of 3D surfaces from sequences of planar slices[J]. CAD and Computer Graphics,1991,13(5):109 - 120.
[24] Noh J Y, Neumann U. Expression cloning[A]. Proceedings of SIGGRAPH[C]. New York, USA: ACM,2001. 277 - 288.

作者简介



吴晓军　男,博士,副教授,1975 年 4 月出生,甘肃张掖人. 2001 年毕业于中国科学院沈阳自动化研究所,从事机器视觉、图像三维建模等方面的研究.
E-mail: hit. wu. xj@ gmail. com

鞠光亮　男,硕士,1987 年 5 月出生,黑龙江哈尔滨人,主要从数字图像处理、医学影像等方面的研究工作.