

分类号: TP391.72

单位代码: 10335

密 级: _____

学 号: 10806072

浙江大学

博士学位论文



中文论文题目: 基于不同维度间特征转移的建模方法

英文论文题目: Modeling via Feature Transfer across Dimensions

申请人姓名: 周世哲

指导教师: 刘利刚 教授

专业名称: 应用数学

研究方向: 计算机图形学

所在学院: 理学院

论文提交日期: 二〇一一年四月

基于不同维度间特征转移的建模方法



论文作者签名: _____

指导教师签名: _____

论文评阅人 1: _____

评阅人 2: _____

评阅人 3: _____

评阅人 4: _____

评阅人 5: _____

答辩委员会主席: _____

委员 1: _____

委员 2: _____

委员 3: _____

委员 4: _____

委员 5: _____

答辩日期: _____ 二〇一一年六月 _____

Modeling via Feature Transfer across Dimensions



Author's signature: _____

Supervisor's signature: _____

External Reviewers: _____

Examining Committee Chairperson:

Examining Committee Members:

Date of oral defence: June 2011

浙江大学研究生学位论文独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得浙江大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

学位论文作者签名：

签字日期： 年 月 日

学位论文版权使用授权书

本学位论文作者完全了解浙江大学有权保留并向国家有关部门或机构送交本论文的复印件和磁盘，允许论文被查阅和借阅。本人授权浙江大学可以将学位论文的全部或部分内内容编入有关数据库进行检索和传播，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

（保密的学位论文在解密后适用本授权书）

学位论文作者签名：

导师签名：

签字日期： 年 月 日

签字日期： 年 月 日

致 谢

值此毕业论文完成之际, 我又不自主的回顾起了在浙江大学读博士的三年时光. 在这三年学习、生活以及论文的写作中, 我要感谢诸多良师益友的深切鼓励与关怀, 因为正是在他们的帮助下, 我才得以不断获得进步与成功, 坚持并顺利走下这人生中重要的一段旅程.

首先, 衷心感谢恩师刘利刚教授对我的淳淳教诲和悉心关怀. 三年来, 刘老师以全面系统的专业知识对我毫无保留地倾囊相授, 他勤勉认真的工作态度也时刻鞭策着我. 本文中的每一章的研究工作的构思, 实施和写作等过程都是在刘老师悉心而具体的指导下完成的. 在学术上, 刘老师不仅功底深厚而且总能抓住计算机图形学前沿的热点, 他还是我研究的引路人, 他让我知道了怎样做高水平高效率的研究. 刘老师的思路开阔, 平易近人每一次与刘老师的讨论都能打开思路, 而这对我来说是莫大的帮助. 在此论文完成之际, 谨向恩师致以最衷心的感谢和诚挚的祝福.

衷心感谢香港城市大学的傅红波老师给我提供了宝贵的学习交流机会. 在有幸同傅老师合作科研项目的近一年时间里, 他细致而全面的分析能力, 极富创意的思维方式不仅给我留下了深刻的印象, 也让我懂得了做高水平学术研究所必须具备的学术素养.

感谢计忠平、陈中贵、张磊、陈仁杰、张冬梅、徐岗、陆利正、辛士庆、曹娟、胡健伟、朱平、方林聪、曾兰玲、武丹、陈军、周联、沈莞蕾、孔丁科等各位师兄师姐对我的鼓励和关怀. 各位师兄师姐丰富的科研经验和学识, 为我的学习和进步提供了强有力的支持. 诚挚的感谢孟敏、蔡鸿杰、徐寅、吴金亮、朱薇、刘刚、李果、郑翰林、刘堃、包力、施平、黄伟贤、魏炜立、韩晓光、樊鲁斌、吴晓堃、金聪健、黄海斌、楼奇、王骁、邓绍辉、吴晓群、金亮、孟凡慧、荣娟、谭一鸣、张春霞、胡瑞珍、沈小勇、谭富文、袁泉等各位同学对我的帮助和支持. 在学习上我们互相讨论, 互相勉励; 在生活上我们互相关心, 互相照顾, 因为你们, 在浙大的这三年时光是充实而快乐的.

特别感谢Daniel Cohen-Or, Kai Xu, Hao Zhang和 Tao Ju等各位前辈与同行对我科研上的启迪, 与你们合作项目是一段难忘而宝贵的学习经历.

致最敬爱的父母, 儿行千里母担忧, 父爱如山, 在我疲惫的时候, 只要想起你们我就重拾信心. 感谢家人和朋友长久以来对我的理解和支持, 你们为我增添了前进的动力.

最后, 感谢抽出宝贵时间评阅博士论文和参加论文答辩各位专家和教授, 感谢你们百忙之中给予的指导.

摘 要

于不同维度间进行特征转移的思想能引导出包括图像编辑和几何重建在内的许多计算机图形学的建模问题的解决方法. 通过将原始数据的特征转移到另一个维度, 我们常常能发现某些隐藏的信息, 而这些信息对我们解决问题往往能发挥关键的作用. 本文的各项工作都受到该思想的启发. 我们将特征转移的方法论与实际应用相结合, 对基于图像语义内容的参数式建模, 基于纹理合成的三维形状外观建模, 以及实时的数字浅浮雕建模这三个问题进行了研究.

我们首先利用二维到三维的特征转移方法解决了一个针对图像内容建模的新问题: 保持真实感的重塑单张图像中的人物体型. 用户使用目前的专业图像编辑软件来进行体型重塑往往需要执行上百次的局部编辑操作才能获得一个可接受的重塑效果; 由于希望达成的重塑效果很难事先进行视觉想象与估计, 因此一旦得到的效果不能满足要求, 当前的重塑效果是无法重用的. 我们使用一种基于模型的方法将这个二维的图像操控问题转移到三维来解决. 我们将一套三维人体全身可形变模型和一种新的图形变形方法进行结合并开发了一套直观易用的人体重塑系统. 三维人体全身可形变模型在我们的系统中发挥了两方面的作用: 第一是从图像中重建出一个三维人体, 第二是利用三维人体的重塑效果来驱动图像发生相同的体型变化. 由于这个可形变模型将人体体型的几何变化模式参数化到了一组直观的语义属性上, 因此我们的系统允许用户以参数可控的直观方式对图像中的人体进行重塑. 我们算法的核心是一种新型的身体感知的图像变形技术, 它即使在模型上存在一定拟合误差时也能鲁棒的将重塑效果从三维模型忠实的转移到二维图像. 该变形技术的要点在于我们观察到大部分的重塑现象直接表现为人体在平行或垂直于骨架方向上的长度变化, 因此我们将图像的重塑化归为一个保持二维和三维间长度比率的平面三角网格形变问题. 通过在一系列例如穿有宽松衣物以及带有复杂遮挡的人物图片上进行试验以及普通用户实际使用的测试结果都表明, 我们的参数式体型重塑系统能够产生各向异性但是全局协调的体型重塑效果, 使用户轻松达成图像重塑要求.

我们的第二方面的贡献是提出了一种特征对齐的曲面纹理合成新方法. 我们观

察到二维纹理样本和三维模型上存在许多特征线，而传统的纹理合成方法忽略了这些纹理特征线在曲面上摆放的位置，这导致曲面的三维形状常常被纹理特征混淆和遮盖了起来。此外，曲面特征线的定向对最终的合成质量也有较大的影响。使用不相容的特征线定向将插值出不光滑的向量场，而由这样的向量场引导合成的纹理会包含许多奇异点和扭曲。为了解决以上问题我们做出了两个主要的贡献。首先我们提出了一个自动的特征线定向算法，该算法通过快速的极小化一个向量场的相容性度量得到优化的特征线定向与对应的光滑向量场。然后，我们引入了一种新的纹理优化方法，该方法在最大化样本和结果间的全局相似度的同时使得纹理样本的特征线和曲面特征线精确的对齐。试验结果证明我们的方法能利用纹理加强对形状外观特点的装饰与表达。

最后，我们用特征转移的思想用来解决到数字浅浮雕模型的建模问题。从三维场景生成浅浮雕模型实际上就是一个将几何特征从流形曲面转移到高度场的过程。我们首先引入了一类新的动态范围压缩函数并证明该函数能比传统方法更好的保留输入模型的细节特征。同时，我们注意到传统的浮雕建模方法并不提供实时的视角选取和细节程度的调节，为此我们提出一个实时的建模系统，该系统利用多目标渲染的纹理运算技术将浅浮雕建模的全过程集成到图形系统上完成，从而使得用户能实时的选取其偏好的建模视角同时对浅浮雕各方面的细节特征进行微调。

关键词： 特征转移 参数式图像操控 人体图像变形 纹理合成 特征精确对齐
浅浮雕生成 实时建模

Abstract

The idea of feature transfer between different dimensions can lead to solutions to a variety of modeling problems spanning from image editing to geometric reconstruction. By transferring features of the original data into another dimension, one can discover extra information which are missing from the input dimension but helpful for solving the problem. In this work we have developed several methods inspired by the idea of feature transfer across dimensions, such as intuitive image manipulation, shape appearance modeling using texture synthesis and digital bas-relief sculpture modeling.

First, we address the image content modeling problem of realistic reshaping of human bodies in a single image. With professional image processing software packages, users need to employ hundreds of local editing operations in order to achieve a visually pleasing reshaping effects; Moreover, since the requirement of desired degree of reshaping is often non-visual and hard to anticipate even for skillful user, this laborious editing process cannot be reused for alternative retouching tasks. We transfer this 2D image manipulation task into 3D domain using a model-based approach. We combine a 3D whole body morphable model with a novel image warping technique and introduce an easy-to-use system. This system integrates the 3D morphable model for two-fold usage: first fitting a 3D human model onto the image, then generating feature changes of body shape to guide the reshaping of the image. Since the morphable model parameterizes the body shape space onto a set of semantic attributes, our system allow user to reshape the body image in a semantically controllable fashion. The algorithmic core of our technique is a body-aware image warping which reliably transfers the reshaping effects from the model to the image. With the observation that many reshaping effects directly exhibit length changes along the direction parallel or orthogonal to the skeleton, we formulate our body-aware image warping as a 2D triangular mesh deformation which preserve relative length ratios on predefined edge pairs. Experimental and practical results show that our parametric reshaping system can

achieve anisotropic but global consistent editing effects in realtime and is able to handle images with moderate occlusions and large amount of clothes.

On the topic of surface texture synthesis, we focus on those feature curves existing both in the 2D texture exemplars and 3D shapes. We observe that existing texture synthesis methods ignore the importance of an accurate placement of the local feature of the exemplar onto the surface, the salient features of the 3D shape are often concealed by the texture patterns. Besides, the orientation of feature curve on the surface has strong impact of the synthesized texture. Inconsistent curve orientations will give rise to inconsistent vector fields which directly lead to texture with a lot of singularities and distortions. We make two main contributions for these problems. First we propose an automatic curve orientation algorithm which minimizes a consistency measure for computing a smooth vector field. Then we present a novel synthesis method which unify the global texture similarity and the exact alignment of 2D and 3D feature curves into an final optimization. Sufficient results show that our method can generate surface textures that reinforce the shape's essential characteristics.

Finally, we apply the idea of feature transfer onto a geometric modeling problem: the creation of digital bas-relief. We present a modeling system for interactively generating digital bas-relief sculpture. In order to preserve detail features of the input model in a better way, we introduce a new family of functions for dynamically compressing the height field. By integrating all the computation workload onto the graphic processing system, our system converts the bas-relief modeling into a multi-target texture computation process. Our system enables users to select the preferred viewpoint and adjust the amount of retained geometric features in realtime.

Keywords: Feature transfer, Parametric image manipulation, Human image warping, Texture synthesis, Exact feature alignment, Bas-Relief generation, Realtime modeling

目 次

致谢	I
摘要	III
目次	
1 绪论	1
1.1 相关工作	2
1.1.1 基于特征的数字几何处理	3
1.1.2 基于特征的图像编辑与纹理映射	5
1.2 章节安排	7
2 从三维到二维的特征转移:参数式重塑图像中人物体型	11
2.1 问题介绍	11
2.2 相关工作	13
2.3 算法概述	16
2.4 全身人体可形变模型	16
2.4.1 带有语义信息的三维人体扫描模型数据库	17
2.4.2 人体姿势空间	18
2.4.3 人体体型空间	23
2.4.4 语义参数控制的体型变形	25
2.5 视角相关的模型和图像拟合	26
2.5.1 姿势拟合	27
2.5.2 体型拟合	29
2.6 身体感知的图像变形	33
2.6.1 直接方法	34
2.6.2 身体感知的图像变形	35
2.7 实验结果和讨论	40
2.8 本章小结	44

3	从二维到三维的特征转移:特征对齐的纹理合成	47
3.1	问题介绍	47
3.2	相关工作	48
3.2.1	纹理合成	48
3.2.2	向量场生成	50
3.2.3	基于特征的形状感知	50
3.2.4	其他相关工作	51
3.3	曲线导引的曲面向量场生成	52
3.3.1	网格上特征线的定向优化	53
3.3.2	受约束的向量场生成算法	55
3.4	特征对齐的纹理优化	57
3.4.1	样本分析和网格预处理	58
3.4.2	网格曲面上的纹理优化	58
3.4.3	特征对齐约束优化	60
3.5	实验结果和讨论	61
3.6	本章小结	64
4	从流形到高度场的特征转移:实时数字浮雕建模	65
4.1	问题介绍	65
4.2	相关工作	66
4.3	浅浮雕建模原理	68
4.4	高细节动态范围压缩函数	70
4.5	基于GPU实时建模	72
4.6	交互方式与细节控制	74
4.7	本章小结	75
5	总结与展望	79
5.1	工作总结	79
5.2	后续工作	80
	参考文献	81

简历	91
攻读博士学位期间主要研究成果	92

图目录

1.1	不同类别属性的三维模型: (a)人造物体模型; (b)生物体模型; (c)用于几何与计算数学研究的概念模型.	2
1.2	曲面上的特征线: (a)外轮廓线(<i>exterior silhouette</i>); (b)有被遮挡的轮廓线(<i>occluding contours</i>); (c)提示线(<i>suggestive contours</i>) ^[1] ; (d)脊线(紫色)和谷线(黄色)(<i>ridges and valleys</i>) ^[2] ; (e)显著脊线(<i>apparent ridges</i>) ^[3] ; (f)等照度线(<i>isophotic lines</i>) ^[4]	3
1.3	基于特征线的曲面编辑 ^[5, 6] : (a)通过编辑外轮廓线变形曲面; (b)通过勾画特征线框进行曲面建模.	4
1.4	基于特征线的形状分析 ^[7] : (a)局部特征线; (b)利用(a)中的特征线检测出重复的形状模式; (c)最终检测结果.	5
1.5	扩散曲线Diffusion curve ^[8, 9] : (a)特征线上定义的颜色被光滑传播到整幅图像; (b)由diffusion curve算法渲染的一个结果; (c)迭代式GPU求解提高渲染速度.	6
1.6	基于特征的纹理插值 ^[10] . 在每一列中,上图是使用最近点或双线性插值格式得到的纹理放大结果,下图为使用考虑特征距离的像素插值格式的纹理放大结果.	7
1.7	带有特征曲线的实时纹理渲染 ^[11] : (a)传统方法在特征线附近不平滑; (b)考虑特征线距离的法向插值格式的新算法能提高光照平滑度; (c)新算法对带有明显特征纹路的模型的渲染结果; (d)(e)特征曲线处放大观察.	8
2.1	我们的参数式体型重塑技术使用户只需要操纵一小组滑杆来控制一组关于体型的语义特征,例如身高、体重和腰围等,就能容易的对单张图片中的人物体型进行重塑编辑. 左1为输入图片.	11

2.2	Youtube上两段流行的关于重塑单张图片中人物体型的视频的截图: (左)输入图片; (中)使用Photoshop [®] 编辑结果; (右)我们方法的结果. . .	12
2.3	系统概览: (a)输入图片; (b)我们的系统首先从一个三维全身人体扫描数据库中学习出一个三维全身人体可形变模型, 该数据库包含姿势(上)和体型(下)两套数据集; (c)借助用户交互完成可形变模型和输入的人物图片之间的校准拟合; (d)通过减小体重语义特征重塑拟合上的人体模型; (e)三维模型体型变化前后的轮廓线; (f)通过应用我们的身体感知的图像变形, 我们将三维的特征变化转移到二维得到最终的结果.	13
2.4	三维人体扫描模型数据库: (a)部分姿势集合; (b)部分男性体型集合; (c)部分女性体型集合.	17
2.5	模板网格分割和嵌入的骨架: (a)分割; (b)嵌入的骨架包含18个关节; (c)从另一个视角观察.	18
2.6	体型集合中的样本和模板样本间也存在一定的姿势差异.	22
2.7	根据语义参数调整体型, 方框内为初始体型.	25
2.8	视角相关的模型拟合流程: (a)借助用户交互得到的初始姿势; (b)图像人物身体轮廓线 S_{img} 与三维模型轮廓 S_{shp} 间的最优对应关系(箭头指示); (c)只使用上半身轮廓线拟合的体型; (d)使用全身轮廓线更新的拟合结果.	27
2.9	我们的方法可以处理复杂姿势和较多遮挡的图片. 对每一行的三元组: (左)输入图片; (中)模型拟合结果; (右)体型重塑结果. (前(后)2个为增加(减少)体重).	28
2.10	我们的拟合方法对姿势歧义是不敏感的. 我们为(a)中输入图像同样高质量的拟合上两个人体模型(b)(c), 尽管在侧视图可以看出(d)(e)中它们有不同的姿势, 我们的方法能从它们得到几何完全相同的重塑结果(f)(g).	30
2.11	呈带状分布的模型轮廓点: (a)当前视角下的模型轮廓点(红色); (b)(c)局部放大.	31

2.12	Viterbi算法求HMM隐状态链: (a)需要被匹配的 S_{img} (紫色曲线)和 S_{shp} , 注意 S_{shp} 可以位于不连续的网格区域; (b)动态规划运行中的某一条状态链; (c)最终匹配结果, 注意从 v_d 到 v_b 发生的跳跃.	32
2.13	与直接方法产生的重塑效果的比较: (a)输入图像; (b)我们的身体感知的图像变形的结果; (d)使用Flexisticker ^[12] 方法对拟合模型贴上纹理的结果(红点为映射时的标记点); (e)为对(d)中三维带纹理的拟合模型重塑后的三维模型; (c)为RBF ^[13] 图像变形结果; (f)为MLS ^[14] 图像变形结果; 注意(c)和(f)中使用了相同的控制点约束: (c)中身体轮廓外的蓝色点 \mathbf{p}_k 移动到 \mathbf{q}_k (仅显示了若干代表点对).	34
2.14	身体感知的图像变形算法示意图: (a)红色表示重塑前人物轮廓; (b)蓝色表示体重减轻后的人物轮廓.	36
2.15	能量项 $E_{ske\perp}$ 示意图.	37
2.16	能量项 E_{ske} 示意图.	38
2.17	能量项 E_{ske} 示意图.	39
2.18	能量项 E_{reg} 和 E_{dis} 用于保证光滑变形.	39
2.19	同时调整身高、体重和腰围得到的多个体型重塑结果, 左一是输入. . .	41
2.20	调整单个语义特征得到的体型重塑结果.	42
2.21	对单张图像中多个人物同时重塑的结果, 第一行为同时减少体重, 第二行为同时增加体重.	43
2.22	通过调整身高和体重得到的一系列代表性的体型重塑结果, 按照姿势复杂度(从上至下增加)和遮挡的多少(从左至右增加)的顺序排列.	45
3.1	(a): 鱼模型及其上的特征曲线(蓝色是谷线, 红色是脊线); (b)对带有特征的纹理使用传统的合成方法容易隐藏曲面上丰富的几何细节特征; (c)我们的特征对齐的纹理合成能使得这些三维特征凸显出来, 注意曲面的三维特征线和样本中的二维特征线的精确对齐.	47

3.2	网格曲面存在各类特征线: (a)外轮廓线(<i>exterior silhouette</i>); (b)有被遮挡的轮廓线(<i>occluding contours</i>); (c)脊线(紫色)和谷线(黄色)(<i>ridges and valleys</i>) ^[2] ; (d)显著脊线(<i>apparent ridges</i>) ^[3] ; (e)平均曲率为零的等值线; (f)分界线(<i>demarcating curves</i>) ^[15] (红色)与谷线(蓝色).	50
3.3	特征线的定向优化提高了曲面向量场的质量: (a)采用任意定向(曲线上绿色箭头)的特征线来引导生成调和扩散的向量场常包含许多奇异点和弯折扭曲(黑色箭头); (b)我们的迭代式曲面定向算法, 每次仅使用当前已被定向的曲线(黑色曲线)作为约束更新向量场; (c)然后计算新的向量场和未定向曲线的相容性度量(按右上标度显示为曲线的颜色); (d)然后选择相容性度量最高的一根曲线((c)中红色曲线)为之定向; (e)最终迭代结果, 向量场更光滑且包含较少的奇异点.	52
3.4	将我们的曲线定向算法和3种向量场生成算法相结合进行特征线定向优化得到的光滑向量场: (a)文献[16]中的算法; (b)文献[17]中的算法; (c)文献[18]中的算法.	56
3.5	我们的纹理合成算法流程: (a)输入模型及其表面的特征线(红色曲线); (b)由特征线插值出的光滑向量场; (c)传统的曲面纹理合成结果; (d)如果纹理样本中存在显著的特征(红色直线段), 我们的算法使得纹理二维特征和曲面三维特征精确对齐.	57
3.6	特征对齐的纹理优化: (a)包含明显的特征线的纹理样本; (b)被曲面特征线穿越的网格采样栅格块, 其中心点为 p ; (c)我们的算法在优化时只选择样本中那些也被特征线所穿越的邻域栅格块(蓝色方框).	59
3.7	特征线任意的定向(a)与优化的定向(b)在与特征对齐的纹理优化后的合成结果, 注意红色方框内标准的奇异点.	60
3.8	特征对齐的纹理合成结果, 在每个三元组中: (左)我们方法的合成结果; (中)没有特征对齐的合成结果; (右)输入模型及其特征线(蓝色是谷线, 红色是脊线).	62

3.9	我们的方法使用不同的特征线集合可以生成多分辨率层次的效果: (a)不使用特征对齐得到的合成结果; (c)使用较低特征分辨率对应的 特征曲线(b)的合成结果, 注意鼻和嘴的部分细节清晰,但是眼睛处 被奇异点遮盖了原有的细节; (e)使用较高特征分辨率对应的特征曲 线(d)得到合成结果, 与(c)的结果比较眼睛处细节得到了特征对齐纹 理准确的装饰.	63
4.1	浮雕是一种常见的实体雕塑艺术形态: (a)亚述人狩猎图浮雕; (b)中 国敦煌飞天浮雕; (c)钱币浮雕.	65
4.2	从三维模型生成浅浮雕模型 ^[19] : (a)输入模型; (b)对输入模型在当前 视角下的高度场进行动态范围压缩我们得到浅浮雕模型(下方为侧视 图,可以看到其厚度较小); (c)电脑数控机床依据数字浅浮雕模型在石 灰石平板上进行加工; (d)加工完成的浅浮雕实体.	67
4.3	浅浮雕建模算法流程: (a)输入模型; (b)在当前视角下的高度场; (c)对 高度场进行高动态范围压缩; (d)使用我们的GPU建模系统得到的浅 浮雕模型(下图为平视图).	68
4.4	(a)(c)文献[19]方法的结果 $c = 0.2$; (b)(d)我们方法的结果, (b)的参数 为 $\alpha = 1.7, \beta = 4.24$; (d)的参数为 $\alpha = 1.03, \beta = 3.8$	70
4.5	函数 $a(x)$ (4.6式)和 $\ln(1+x)$ 的函数曲线比照.	71
4.6	GPU算法流程与数据纹理布局.	73
4.7	通过 β 调整浮雕细节(从左至右为增大 β).	74
4.8	通过 α 调整浮雕的整体厚度(从上至下为增大 α).	75
4.9	调整阈值 φ_{thred} 以消除瑕疵(从上至下为减小 φ_{thred}).	76
4.10	我们的实时浮雕建模系统得到的结果. 在每一列中下图为上图的平视图.	76
4.11	我们的建模系统允许用户从不同的视点实时得到反馈(每一行为同一 模型不同视角下的浅浮雕).	78

第1章 绪论

特征是基于视觉的多媒体内容中最富表现力的部分，也是人们在对这些内容进行观察后最容易记住的元素。在计算机图形学研究中我们经常和多媒体内容打交道，如二维的图形，三维的网格模型和四维的视频等。高速发展的计算机技术使得这些多媒体数据日渐丰富，而其增长速度之快已经超过其能被全面理解和处理的速度。但这些庞大的数据并非没有冗余。从研究的角度来看，一种好的方法论是以数据中的特征作为着眼点。原因有两点：一是特征能让我们避开冗余的干扰专注于研究数据的模式，二是人类的观察和感知方式是对特征敏感的，而发掘出新的特征更进一步加深我们对数据的理解。因此，本文的研究工作启用了这样一种方法论：先从数据中抽取具有相同特征的类别分开对待，然后观察到不同类别间在某些特征上的内在联系，最后利用这种联系将一个类别中出现的问题化归到另一个类别中来解决。

在数据采集端，我们首先需要对真实世界中的物体进行数字化的表达与存储。由于全息扫描或者照相等采集手段的不同，我们能得到各种不同维度的数据。例如人体在三维中可表达为一个三维曲面的网格或点云模型，而在二维中则是一幅人物图像。因此，属于被表达物体的某些特征在不同的维度上有着不同的数据形式。例如人物的腰围特征在三维模型上是由其腹部曲面的向外弯曲程度所反应的，而在二维图像中则是反应在躯干部分图像的宽度上。又例如一个人的脸部表情反应在尺寸为 $100*100$ 的24位RGB图像上是240000维的数据，但是在语义空间我们实际上可以简单的用处于四种不同情绪下的标志性表情：“快乐”、“悲伤”、“惊恐”和“平静”以及位于这四种表情中间的过渡性表情来进行描述，因此从语义特征的角度来考虑人的脸部表情实际上可以看成是一种4维数据。

我们观察到数据和维度的关系并加以利用。尽管在不同的维度空间数据有着完全不同的形式，但当它们所表示的都是真实世界中的某一类物体所蕴含的属性内容时，这些数据在不同维度中的特征之间存在紧密的联系。本文讨论了基于不同维度间的特征的关联和转移的建模方法。我们主要研究如下几种形式的特征：人体体型的语义特征，曲面特征线和纹理样本中的特征线，以及三维场景的高度场的梯度特

征；并将这些特征分别用于解决图像中人物体型的参数式重塑的图像建模问题，用纹理来精确装饰形状的的曲面外观建模问题以及数字化浅浮雕的实时建模问题. 我们通过解决这些问题的过程证明，基于不同维度间特征转移的建模方法论在计算机图像学研究的各个领域中都能发挥其独特的作用.

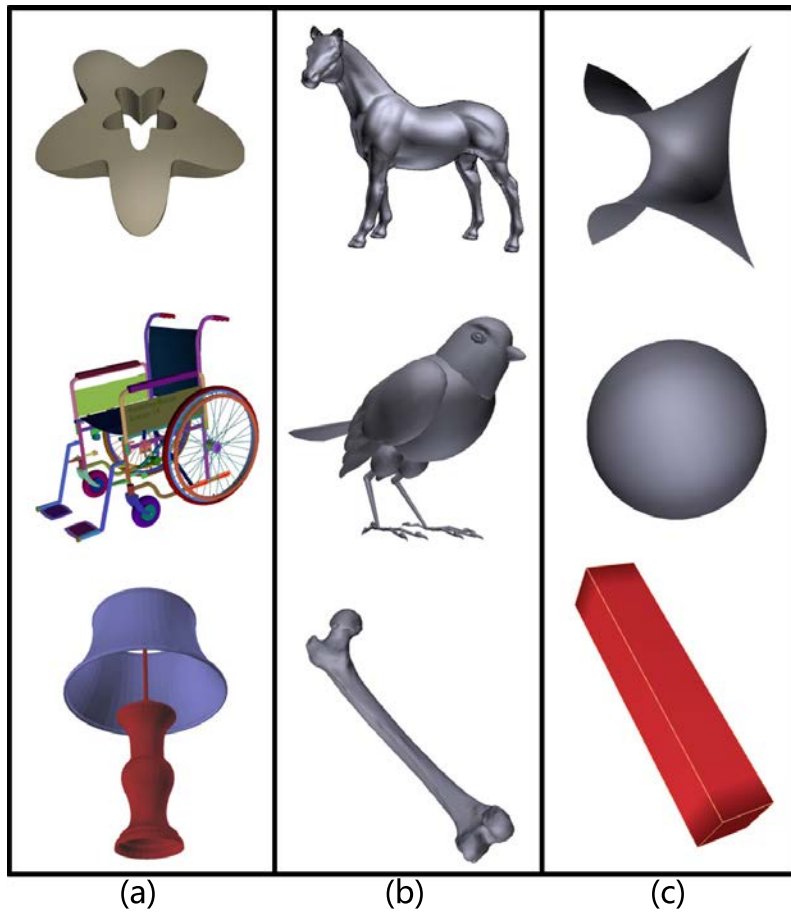


图 1.1 不同类别属性的三维模型: (a)人造物体模型; (b)生物体模型; (c)用于几何与计算数学研究的概念模型.

1.1 相关工作

近年来计算机图形学界出现了一些着眼于特征的研究工作. 从研究的角度来分, 这些工作可分为特征检测, 特征分析与特征利用. 按照研究的具体对象来分, 这些工作又横跨了数字几何处理^[20], 数字图像处理和视频处理等几乎计算机图形学研究的所有领域. 下面我们对数字几何处理领域和图像及纹理映射领域与特征关系密切的研究工作进行简单的回顾.

1.1.1 基于特征的数字几何处理

在数字几何处理领域,特征是一个多尺度的概念,既有宏观的全局特征也有微观的细节特征.以网格模型为例,其宏观特征可以等价地看成是其对应的现实世界的物体的属种类别.如图1.1所示,进行数字几何处理常用的网格模型大体可分为三类:人造物体模型,生物模型和用于几何研究的概念模型.细节特征则是每个大的属种类别内的个体之间的微观差异.例如有的模型表面非常光滑,而有的模型表面则比较粗糙布满细节.无论是光滑的网格模型还是粗糙的网格模型,其细节特征通常和其表面存在的某些特征曲线有关.如图1.2所示,这些特征曲线能以一种紧凑而有效的形式抓住曲面形状外观的精髓,例如外轮廓线可以反映模型的整体形状和当前被观察的视角,而脊线与谷线则反映了模型的凹凸结构.

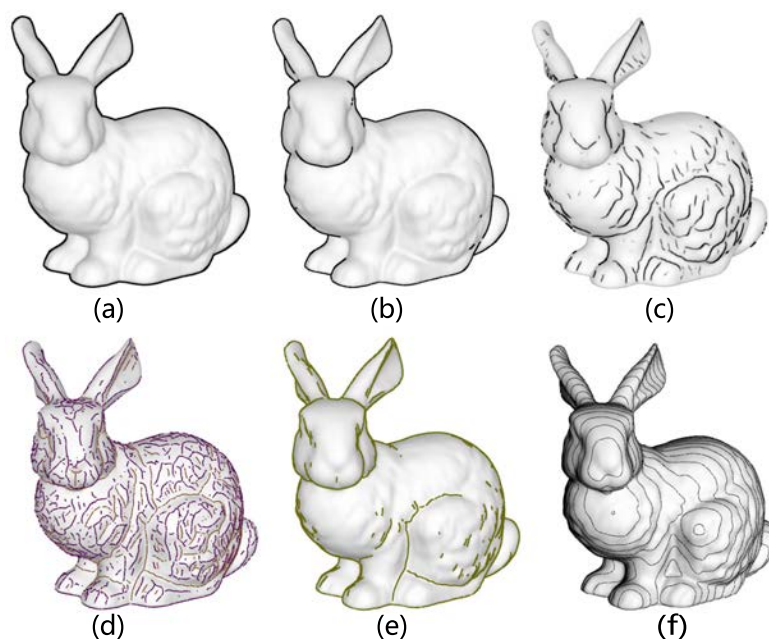


图 1.2 曲面上的特征线: (a)外轮廓线(*exterior silhouette*); (b)有被遮挡的轮廓线(*occluding contours*); (c)提示线(*suggestive contours*)^[1]; (d)脊线(紫色)和谷线(黄色)(*ridges and valleys*)^[2]; (e)显著脊线(*apparent ridges*)^[3]; (f)等照度线(*isophotic lines*)^[4].

近年来,研究者利用包括这些特征曲线在内的各种特征量来进行网格模型建模与编辑,对称检测,数字城市建模等方面的研究.在网格建模与编辑方面,由于特征

的低维特点,使得建模系统可以搭建在一个低维的参数化的设计空间之上,这样就提高了建模系统的易用度和可用性.当设计者为了建模的需要,对例如三维网格模型或者数字图像等媒体内容进行改变的时候,往往试图去保持其全局特征,以保证变化后的内容在其类别属性上的可辨识度,同时又试图去改变那些细节的特征以反应其设计意图.比如,Nealen等人^[5]开发了一种利用特征线来进行网格曲面建模的工具,用户可以通过勾画指定在当前视角下的特征线,主要是轮廓线和附加的鼓出部分的分割线,然后它的曲面重建算法通过极小化一个薄膜能量泛函^[21]来自动的得到以这些特征线为框架的三角网格,如图1.3(b)所示;其建模结果是与视角无关的,而是由特征线框架的几何与结构完全决定.文献[6]中则介绍了一种用户勾画的网格变形操作界面,能Laplacian^[22]变形一个原始模板网格使其具备新添加的特征线,如图1.3(a)所示.最新的方法开始着眼于使用体现网格结构的较高级的特征,例如结构性,对称性和平行性等,来进行形状编辑.Gal等人^[23]提出了利用特征线以及他们之间的相互关系来引导网格变形的的方法,他们的算法先从网格中提取出特征线集合并对特征线之间的位置,对称,平行以及聚类等相互关系进行自动分析,然后用户操纵某一根特征线或者特征线上的某一段发生改变,然后算法自动的将改变传播到整个特征线集合,最后由特征线驱动网格变形.整个编辑过程特征线充当了控制器的角色.Zheng等人^[24]进一步将人造物体模型的各个有语义的部分作为控制器,由于以部分这种高阶特征作为形变的基本单位,其方法不仅能提高编辑系统的易用程度,同时还能较好的保持部分内部的细节.

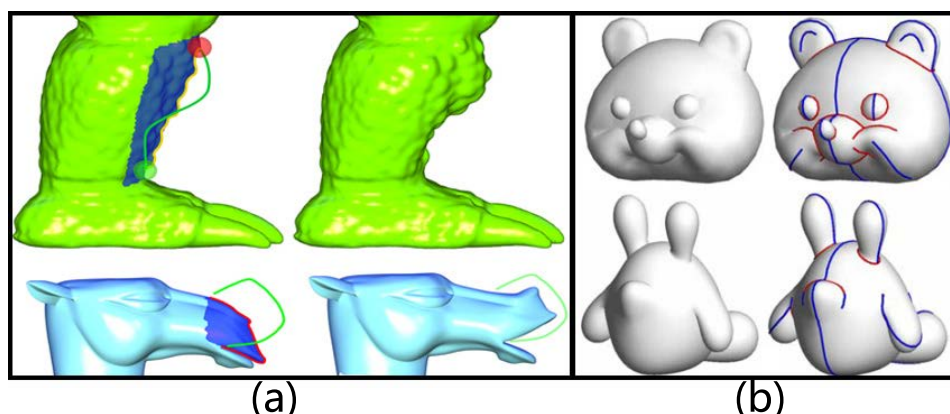


图 1.3 基于特征线的曲面编辑^[5,6]: (a)通过编辑外轮廓线变形曲面; (b)通过勾画特征线框进行曲面建模.

在形状分析和检索方面, 特征也扮演着重要的角色. 由于曲面的几何特征一般是简洁清晰的几何元素例如特征点和特征线, 对形状进行分析往往基于或者等价于对这些特征进行分析. 例如在进行点云的对称以及重复模式检测时, 提取特征点和特征线往往是算法的第一阶段. 比如Bokeloh等人^[11]提出一种从点云数据中检测出重复结构的算法(如图1.4所示). 他们先从点云中抽取碎特征线, 然后利用迭代最近线算法(ICL: Iterative Closest Lines)从中找出连续对称的特征模式, 最后利用这些特征从原始点云中找出那些全等或镜像对称的部分. Sunkel等人^[25]则根据用户给定的特征模板通过有监督学习的方法从点云数据搜索出重复的特征, 并利用这些特征来进行点云曲面的对称检测.

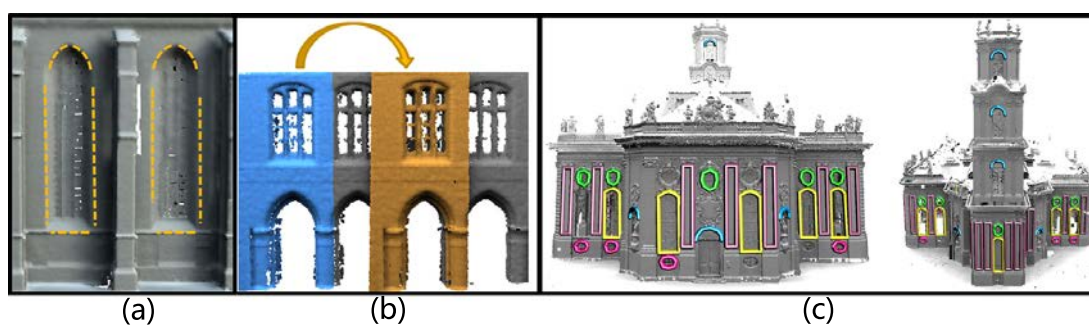


图 1.4 基于特征线的形状分析^[7]: (a)局部特征线; (b)利用(a)中的特征线检测出重复的形状模式; (c)最终检测结果.

1.1.2 基于特征的图像编辑与纹理映射

特征在图像处理领域也能发挥重要作用. 将图像特征与颜色扩散格式结合起来能得到多种对图像的压缩表达和操控方法. 常用的图像特征可分为线形特征和结块特征两类. 我们这里主要介绍与前者相关的研究. 线形特征包括图像中物体的轮廓和对比度较强的边界线, 这些特征线也往往是图像中前景物体的分界线. 利用这些特征线, Orzan等人^[8]提出了扩散曲线(Diffusion curve)作为一种新的矢量表达用于生成色调平滑但前景和背景内容分块清晰的图像. 扩散曲线不仅定义了图像域的一个划分, 曲线的不同侧还定义了不同的颜色, 这些颜色沿着曲线可以发生平滑的渐变, 此外颜色从一侧变化到另一侧的变化锐度也可以被控制. 给定这样一组曲线, 通过求解一个以这些扩散曲线上的颜色梯度为约束的Poisson方程我们就能得到最终的

图像. Jeschke等人^[26]则在扩散曲线的基础上提出了一种鲁棒的基于Voronoi图的光栅化方法来将扩散曲线上的连续边界值离散化,这种光栅化方法能解决传统方法的误差问题;然后在GPU上来求解Poisson方程^[27]扩散颜色,通过定义一个可变的镂空模板使得其GPU求解过程一般只需要进行8次迭代就能快速得到精确收敛的结果.文献[9]进一步将扩散曲线推广到了曲面上并扩展了原来的颜色扩散算法,其中包括使用一个与视角相关的变形算法动态的为占据较大面积的特征部位分配较多的纹理,以及使得其整个框架支持直接渲染置换映射^[28]和几何图像^[29].

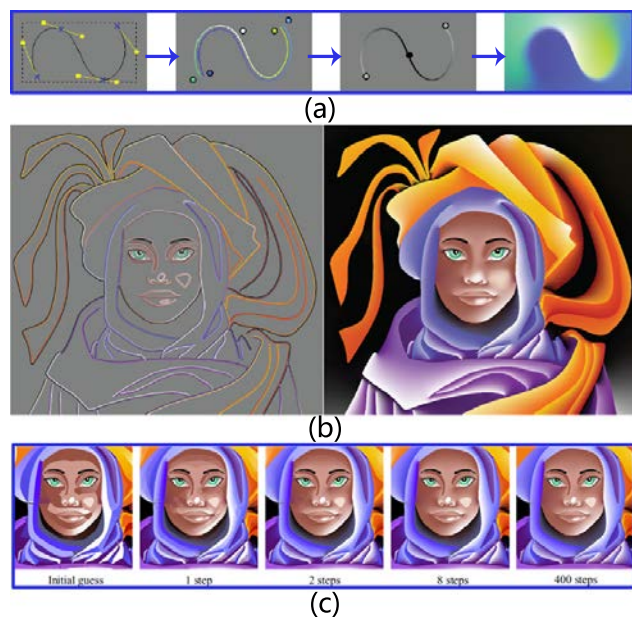


图 1.5 扩散曲线Diffusion curve^[8,9]: (a)特征线上定义的颜色被平滑传播到整幅图像; (b)由diffusion curve算法渲染的一个结果; (c)迭代式GPU求解提高渲染速度.

对模型或者纹理中的特征进行增强或者其他处理能得到风格化的渲染效果. 比如在纹理映射中,让纹理样本中的特征和模型上的那些锐利的几何特征对齐,能使得模型的三维形状得到正确的装饰和表达. Ramanarayanan等人^[10]将纹理样本中的特征线和像素联合使用,在纹理映射插值的时候使用特征线作为分界线,只使用位于同一区域内的样本像素来插值,从而提高了渲染效果的锐度. Parilov等人^[11]观察到传统的双线性插值和法向图映射会在锐利的特征处造成视觉瑕疵,而这在渲染那些带有多而明显的折痕皱纹和凹槽的模型时更为严重.为了解决该问题,他们将特征附近的法向量表达为到特征线距离以及特征线上点的位置的向量函数,从该向量

函数中能采样出与网格分辨率无关的光滑过度的法向量，从而实现了对特征部分的实时精确渲染。

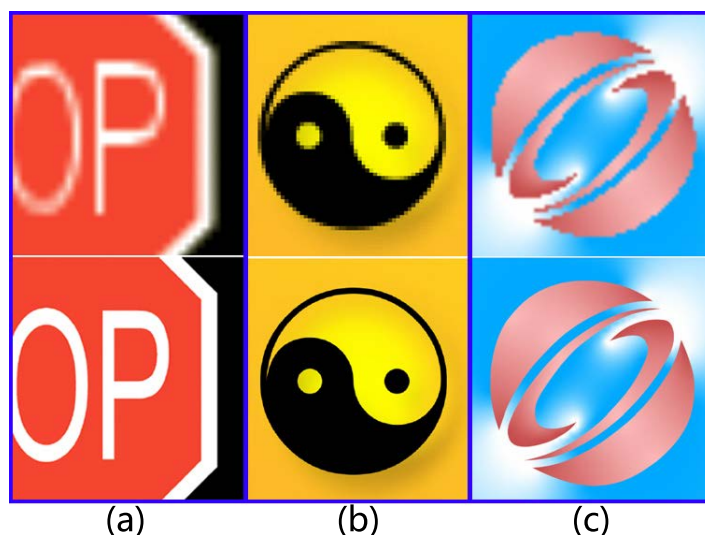


图 1.6 基于特征的纹理插值^[10]. 在每一列中,上图是使用最近点或双线性插值格式得到的纹理放大结果,下图为使用考虑特征距离的像素插值格式的纹理放大结果.

1.2 章节安排

本文中我们利用特征在计算机图形学研究中的特殊作用,配合我们观察到的特征在不同纬度空间中的相互关系来解决图像编辑,曲面纹理合成以及数字几何建模中的若干问题.其实研究一开始的时候我们并没有把不同纬度间的特征联合起来考虑作为一种解决方案,从表面上来看,问题本身也没有横跨多维度的范畴.但是通过实验我们发现,将问题放到另外一个维度进行思考的确能带来突破,而且往往能搜索到可行的解决方案.这主要是因为问题的输入数据可能就是被降维或升维过的,对它所对应的真实世界的事物而言,在一个高于输入维度的空间里能更好的展现其本来面貌和内在特征.

在本文的前三章我们分别介绍利用不同维度间特征转移的思想来解决的三个问题.在第二章中我们处理的是传统的图像编辑问题.图像编辑本是一个纯二维的问题,大部分关于图像编辑的研究工作都是在图像颜色域或者频率域展开的,比如在图像域中嵌入四边形或多边形网格来进行图像重映射^[30,31]以及在频率域中对图像

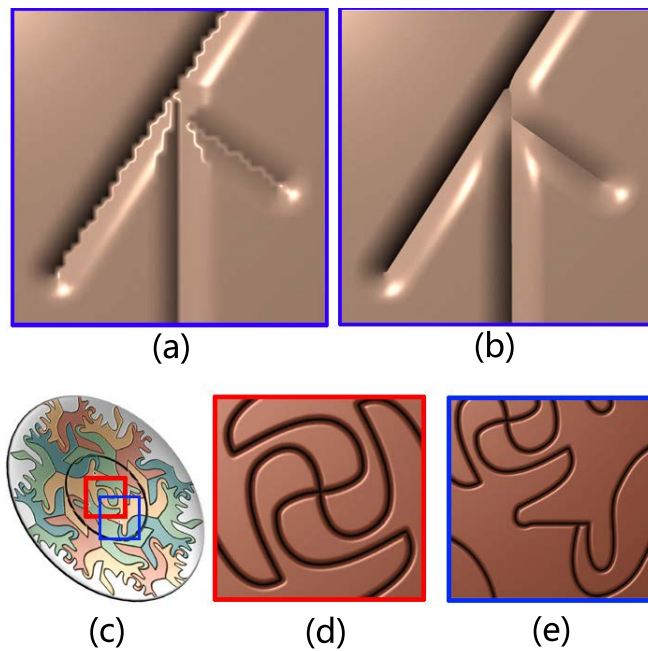


图 1.7 带有特征曲线的实时纹理渲染^[11]: (a)传统方法在特征线附近不平滑; (b)考虑特征线距离的法向插值格式的新算法能提高光照平滑度; (c)新算法对带有明显特征纹路的模型的渲染结果; (d)(e)特征曲线处放大观察.

进行细节增强^[32]等. 但是这些工作都是针对图像的低层次的特征进行修改, 比如图像重映射是修改图像的空间分布特征, 而细节增强是修改像素的颜色特征. 但是我们需要处理的是对人物图像进行体型重塑问题, 这是一个对图像的语义特征进行编辑的任务, 而这些语义特征实际上是存在于一个三维的人体模型空间, 因此我们将二维问题提升到三维空间中来处理, 然后利用特征转移的思想将三维的变形转移到二维图像上来最终解决问题.

在第三章我们进行特征对齐的纹理合成的研究. 我们观察到很多光滑模型, 特别是那些人造物体的模型表面实际上可分为两种区域: 一是清晰而形成网络的特征线, 二是相邻特征线之间的平滑曲面. 这些特征线对模型的形状表达与感知无疑是非常重要的. 传统的纹理映射着重于考虑如何尽可能的减少映射的目标域在表面上的扭曲程度以及合成结果和输入样本之间的全局相似度, 而忽略了样本在表面上的位置摆放和朝向等因素对最终结果的特殊影响. 我们观察到当使用各项异性的纹理覆盖到模型表面时, 如果纹理中的特征线不能和曲面的特征线保持一致, 反而容易

混淆对三维形状的表达.为了解决这个问题,我们利用维度转移的思想,将二维纹理特征转移到三维与曲面上特征线进行优化置放,提出了特征对齐的纹理优化算法.通过大量实验,我们不仅证明了特征对齐的合成结果能增强三维形状的表现力,同时还证明了对不同维度的特征做出正确的处理可以对视觉效果带来显著的提升.

在第四章中我们研究一种新型的建模方法:浅浮雕建模.浮雕可以看成是一种介于三维雕塑造型和二维绘画间的艺术形式,该形式力图在接近平面的素材上反映本体形状的三维特征,因此浮雕建模的过程就是一种在不同维度间做特征转移的过程.我们注意到传统的浅数字浮雕建模方法并不提供实时的视角选取和细节程度的调节,为此我们提出一个实时的建模系统,该系统利用多目标渲染的纹理运算技术将浅浮雕建模的全过程集成到图形系统上完成,从而使得用户能实时的选取其偏好的建模视角同时对浅浮雕各方面的细节特征进行微调.此外我们还引入了一类新的动态范围压缩函数并证明该函数能比传统方法更好的保留输入模型的细节特征.

第2章 从三维到二维的特征转移:参数式重塑图像中人物体型*



图 2.1 我们的参数式体型重塑技术使用户只需要操纵一小组滑杆来控制一组关于体型的语义特征,例如身高、体重和腰围等,就能容易的对单张图片中的人物体型进行重塑编辑.左1为输入图片.

2.1 问题介绍

计算机图形学的内在动因之一是满足对视觉审美的追求.随着该学科的发展,人们开始尝试用软件方法进行人物图像的修饰与增强,例如软件Adobe Photoshop[®]已成为图像编辑的工业标准软件^[33].由于现存的修饰工具主要是为底层的编辑任务而设计的,常常只能用来进行对人物的皮肤进行一些局部的修改,如去除疤痕和皱纹等等.如果要进行人物体型的修改,就要求进行全局协调的编辑操作,而这对熟练掌握软件的用户来说都是一项单调而且繁琐的任务.从观察互联网上专业的艺术家使用Photoshop[®]来进行人物体型修改的视频(图2.2)我们可以发现,完成一次视觉可信的细微的体型重塑往往需要进行上百次的局部编辑操作.如果修改的体型效果不够满意,又需要重复这样耗时的过程来修描出一个新的体型.一种较为高级的方案是对身体各部分分别进行操作,即按照四肢和躯干的骨骼的方向来对他们各自部分

*本章内容发表在:Parametric Reshaping of Human Bodies in Images. ACM Transaction on Graphics,2010,29(4).

的图像进行缩放变形. 然而这种方法存在两大难点: 第一, 真实感的体型重塑常常要在身体各部分满足人体测量学(Anthropometry)的不同尺度的缩放^[34]; 第二, 如何使得每个部分之间的形变保持全局的连贯. 而第二个问题在那些互相遮挡的区域将更为棘手. 我们采用一种模型驱动的方法来解决这些问题. 通过嵌入一个三维全身可形变模型, 我们实现了身体各部分为不同尺度但同时全局协调的形变效果. 我们的方法基于如下两点观察. 一是虽然目前的从图像恢复三维人脸或者体型的研究着眼于尽可能真实的重建三维模型本身, 但是这里我们仅仅是执行图像的平面域上的形变, 这就允许我们只需要在三维模型和二维图像之间建立起一种视角相关的匹配(View-dependent matching); 二是我们发现人体体型在图像中的视觉感官基本上由对应于身体骨架的身形的二维轮廓线决定的, 专业艺术家在使用Photoshop[®]来进行人物体型修改的过程也其实就是在对这些二维轮廓线进行修整(图2.1). 由于以上两点观察, 我们认为最终的体型重塑是在图像平面域内做编辑操作.

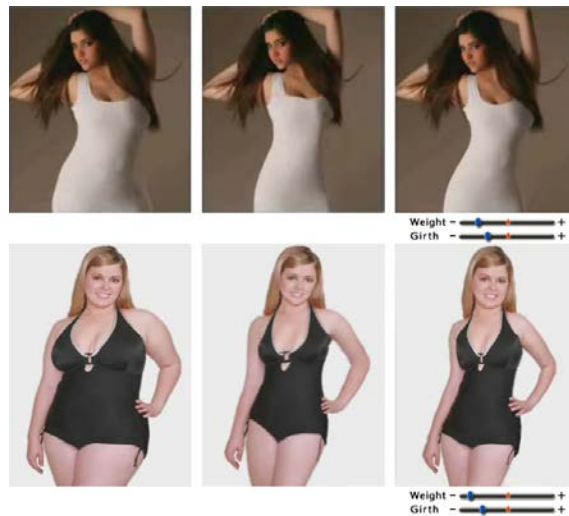


图 2.2 Youtube上两段流行的关于重塑单张图片中人物体型的视频的截图: (左)输入图片; (中)使用Photoshop[®]编辑结果; (右)我们方法的结果.

在问题被化归到图像平面域之后, 我们将人体体型重塑最终当作对图像执行一个二维图像扭曲操作, 并提出一种新的人身感知(**Body-Aware Image Warping**)的图像扭曲技术来将三维全身可形变模型的形变量转移到图像中. 该方法一个特点是能容忍模型和图像间建立的视角相关的匹配中存在中等的匹配误差. 我们的系统的工作流程如图2.3所示:借助一定量的用户交互, 我们首先将三维全身可形变模型大概

的匹配到图像人物上;接着用户通过调整一小组有语义的属性,如身高体重等,来重塑与图像匹配的三维模型的人体体型;最后,我们通过人身感知的方式对人体部分做图像扭曲,使得图像中人物的变化,特别是人物轮廓对应于人体骨架的变化能和三维模型的变化尽可能的相似.这套工作流程绕开了精确重建带有纹理的三维人体模型这个难点,使得我们的系统可以顺利的对身着宽松服装并摆出各种姿势的人物进行体型重塑.在使用的速度上,由于我们的方法是参数型的,用户只需要操纵若干个滑杆来调整他们各自对应的体型语义属性.在使用的速度上,由于我们的快速图像扭曲算法可以提供实时的反馈,用户可以轻松的取得期望的塑身效果,而采用以往的局部编辑的方法则很难做到这一点.

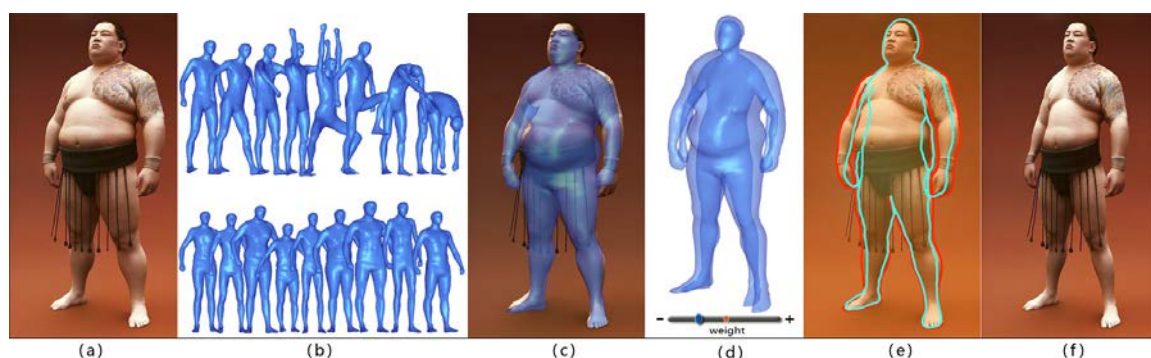


图 2.3 系统概览: (a)输入图片; (b)我们的系统首先从一个三维全身人体扫描数据库中学习出一个三维全身人体可形变模型,该数据库包含姿势(上)和体型(下)两套数据集; (c)借助用户交互完成可形变模型和输入的人物图片之间的校准拟合; (d)通过减小体重语义特征重塑拟合上的人体模型; (e)三维模型体型变化前后的轮廓线; (f)通过应用我们的身体感知的图像变形,我们将三维的特征变化转移到二维得到最终的结果.

2.2 相关工作

图像修饰. 修饰一幅图像常常需要使用一系列的编辑工具,比如,色调调整与重着色,图像修复与去噪,图像拼接与图像扭曲等等.每一类工具都有相关的研究工作,这里我们不一一展开讨论.现存的大部分工具都是在像素级别对图像进行编辑,他们在进行较低级别的编辑任务时是很方便的,例如利用图像修复与去噪来进行皮肤

皱纹的去处^[33]等;而如果要完成较高级别的修饰任务比如美化人脸和重塑体型等操作时,这些工具将显得很不方便,因为用户必须保证许许多多的局部操作之间的连贯性.而针对研究高级别的编辑工具的研究是很少的,因为它们都需要解决一个基本问题:如何处理被编辑物体的语义特征^[35].

图像扭曲和缩放. 现存的许多图像扭曲的算法实质上都是将用户制定的控制点处的形变传播到图像的其余部分.针对这个传播过程提出了许多算法,例如基于散乱数据点插值(其中包括径向基函数插值法^[13]和移动最小二乘法^[14]等),或者变形优化方法(其中包括尽可能刚性的形状操控法^[36]).最近,一类新图像扭曲方法被提出用来解决内容敏感的图像与视频缩放问题^[37],而我们提出的图像扭曲方法其实也可以被看成是一种图像缩放方法,但是是沿着人体骨骼轴向对人体图像进行各向异性的缩放.

数据驱动的人脸修饰. 采用数据驱动的方法来进行图像中人脸的修饰被证明是一条有效的途径. Blanz等人^[38]用样本学习的方法搭建了一个三维可形变的人脸模型来进行真实感的图像操控,包括对单张图片中的人脸进行修饰(例如让脸变胖或者变瘦等).值得注意的是,他们的方法先是从一张人脸图片中恢复出带有纹理信息的三维人脸模型,然后将形变后的模型直接渲染回原来的图片中.我们的方法则没有采用这个流程,因为将其推广到人体体型重塑这个问题上需要精确的恢复一个带有纹理的人体三维全身模型.我们实验发现,即使很小的误差也会在渲染回原来的图像时留下明显的瑕疵.因为人脸模型较为简单,不存在关节连接的结构,所以最新的人脸修饰方法通过直接采用二维人脸图像作为训练集合来得到中间的驱动模型.例如, Leyvand等人^[39]提出一种增强人脸美观程度的方法,此方法先从一系列人脸图像中学习训练出一个二维图像扭曲函数,然后用它对嵌入到输入图片的脸部模型按照人脸美观程度的标准进行变形.

三维人体全身可形变模型. 近年来在三维人体体型空间的分析上取得的进展为本工作的研究铺平了道路.一些早期的研究工作如文献^[40, 41]将人脸的可形变模型的建模方法推广,用来捕获不同人体的体型变化.其方法生成的三维人体全身可形变模型是参数型的,允许用户通过直观的控制一两个参数来浏览整个体型空间.但是他们的方法在生成这个体型空间时必须使用保持相似的姿势的人体模型,同时这些“直

观”的参数并非我们所希望的“语义”参数.进一步的,文献[42, 43]提出了两种类似的方法将体型的变化同姿势的变化分离开来,并将剥离出的体型变化作几何编码.这样得到的可形变模型可以用来导出任意姿势和任意体型的三维人体模型.这两种方法的一个最大区别就是文献[43]的方法将体型和姿势的变化一起进行编码,而由Anguelov^[42]提出的SCAPE全身可形变模型则去除了这两者之间的耦合,为姿势和体型提供了两套独立的参数来解析他们各自引入的三维模型的几何变化.解耦合的姿势和体型参数使我们接下来做视角相关的匹配的优化变得较为方便.

从图像中估计人体姿势.在计算机视觉领域对从单张图像中恢复三维姿势展开了许多研究工作,提出了很多全自动的算法^[44].然而由于这个问题本身的复杂性,如果要恢复出真实合理的姿势,往往需要一定的用户交互(例如手动的指定预定义的骨架和图像之间的关节对应关系^[45-48]).

从图像中恢复人物模型.由于人体是一种非刚性的支架结构,目前仅有少量的技术能图像中恢复出细节完善的人体模型.其中的大多数技术是基于多视角动态人体体型的视频采集的.他们一般使用一个对被采集人物的预先扫描好的模板模型^[49, 50],或者使用一个SCAPE全身可形变模型^[51, 52]去和视频中的每一帧进行匹配.然而由于问题本身明显的欠约束,从单张图像中的恢复出细节完善的人体模型往往是十分困难的.最近, Guan等人^[53]首次尝试了从单张图像中恢复细节完善的人模型.他们的算法首先需要在用户交互的帮助下得到一个三维人体姿势和体型的初始预测,然后使用一套复杂的优化格式来求解最优的SCAPE模型的姿势和体型参数.他们的优化考虑了包括边界重叠约束,身高约束和平滑光照约束在内的一系列约束.他们的方法主要处理裸体或者只穿少量衣物的人物,而不以处理带有复杂衣服的人体为目标.

此外我们的工作还和视角相关的一般三维物体建模技术有关,特别是从单张图片中借助一个简单的三维模板模型来建模的技术^[54, 55].类似文献[49, 56],这些技术通过形变模板模型来匹配目标轮廓.在我们的体型拟合步骤中,我们使用了^[54]中的方法来完成将一个三维可形变模型拟合到二维身体轮廓的任务.

形变迁移技术常被用来将一个模型的三维动态重现到另一个模型上.这种转移可以发生在三维网格之间^[57],从卡通动画到三维模型^[58]或者从视频到视频^[56].为了解决姿势驱动的动画中保持形状的问题, Hornung等人^[48]提出了一个角色动画系统,

该系统通过将三维动作捕捉(Mocap)数据转移到二维的角色动画上来从单张图片中生成动画效果. 然而我们的方法首次尝试将三维模型上发生的体型变化转移到单张人物图片上.

2.3 算法概述

重塑单张图片中的人体可看成一种内容敏感的图像变形^[31, 59]. 为了得到全局协调和视觉可信的重塑效果, 人体的不同部分需要进行不同尺度的修改. 艺术家使用传统软件往往需要进行一些列局部操作来达成目的. 然而, 我们提出了一个先拟合后形变的方法通过在图片中嵌入一个三维全身可形变模型来控制参数化的体型重塑.

我们应用SCAPE模型作为我们的可形变模型, 并将之拟合入单张人物图片. 这个拟合过程等价于在图像中二维人物形态的约束下求解SCAPE中的姿势和体型参数. 我们借助用户交互来求解这个困难的问题. 我们通过设计一个用户界面使得全过程只需要少量的用户输入, 同时还能允许交互式的修正拟合结果. 通过分开求解两类参数, 我们的系统能在较快时间内得到可以接受的拟合结果. 在重塑时, 用户只需指定和体型有关的语义特征的期望值, 例如增加5厘米身高, 系统就能实时的给出相应的重塑效果. 为了做到这一点, 我们的系统首先让拟合好的三维人体模型根据语义特征发生形变, 然后用这种变化来驱动二维图像产生同等程度的重塑.

基于特征转移的思想, 我们提出了身体感知的图像变形技术. 我们观察到无论是在三维还是二维, 体型特征的变化在方向上和程度上都满足下列规律: 重塑都是身体的各部分沿着平行或者垂直骨骼的方向发生变形, 同时各部分的变形程度之间的比例关系在三维和二维间是相等的(此规律在任何视角下皆成立). 利用这种关系, 身体感知图像变形技术根据三维上的长度变化全局协调的重缩放图像人物的各个部分. 我们的变形能实时的给出反馈来帮助用户尽早达成期望的重塑效果

2.4 全身人体可形变模型

我们使用SCAPE模型作为我们的全身可形变人体模型. SCAPE模型是一种对人体全身曲面在不同姿势和不同体型的情况下的几何形状的变化进行参数化建模的方

法, 将该方法应用到一套三维人体扫描模型数据库上, 可以得到一个低维的参数可控的可形变人体模型; 在数学形式上, SCAPE模型最终能表示成 $M = M(\theta, \beta)$, 其中的 θ 和 β 分别表示可形变人体的姿势和体型参数, 通过这两套独立的参数我们能够分别控制模型在姿势上和体型上的发生形变. 在本工作中, 我们则能利用这种三维人体的体型变化来驱动在二维图像中人物体型的重塑.

下面我们详细的叙述利用SCAPE模型在一套人体扫描网格模型数据库上的建模流程. 如前所述SCAPE模型提供姿势和体型两套独立的参数, 为了得到这样的参数我们首先从数据库中提取出相互解耦合人的体型和姿势几何变形数据, 然后再分别以合适的参数化方法对他们进行编码.

2.4.1 带有语义信息的三维人体扫描模型数据库

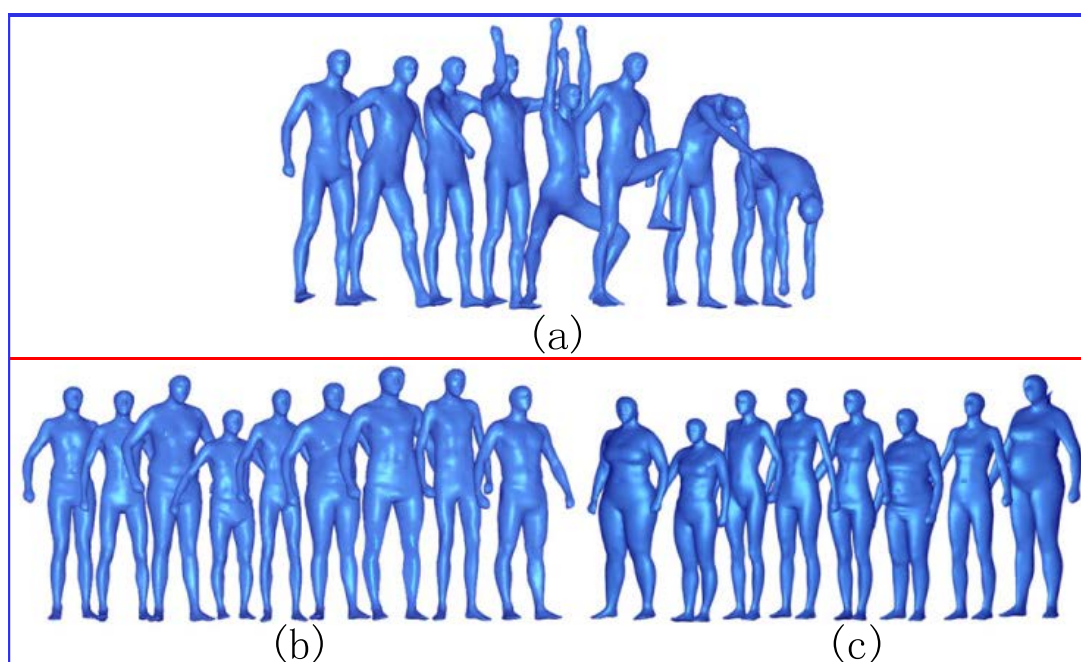


图 2.4 三维人体扫描模型数据库: (a)部分姿势集合; (b)部分男性体型集合; (c)部分女性体型集合.

我们使用由文献[43]中提供的人体全身扫描模型数据库. 它包括111个人物在多种不同的姿势下的扫描重建模型, 整套数据库一共有550个网格模型. 此外它还包含了每个被扫描人物的身体属性, 比如身高体重腰围等等, 以及他们的生理信息, 比如脂肪含量和血型等等. 在这些模型中, 我们把一个身高180厘米, 体重65.5千克的

男性在标准站立姿势下的模型设定为模板. 在预处理阶段, 除模板之外的549个网格都通过一个非刚性注册的过程与模板建立起了顶点对顶点的完全对应关系; 同时, 所有模型都被校准到同样的定向以保证他们的以胸腹平面彼此平行朝向一致. 由于扫描技术的局限, 其中某些模型有较为明显的不规则瑕疵, 我们在预处理阶段对这些瑕疵进行了手动处理. 在本章中, 我们记模板网格上第 k 个三角形为 T_k , 我们称 T_k 在数据库中其他模型上对应的三角形为 T_k 的实例三角形, 每个网格都由 V 个顶点和 N 个三角形构成.

从这个数据库中, 我们获得了两个数据集(图2.4): 一是姿势集合(图2.4(a)), 它由模板人物在35种不同的姿势下的模型组成; 二是体型集合(图2.4(b)(c)), 该集合由与模板相同姿势的57位男性和54位女性的模型组成. 由于男性和女性在体型上存在本质的差别, 故而我们将体型集合按照性别分成两个子集. 在接下来的论述中如不特别说明, 我们都以男性体型集合为例. 对每一个网格, 我们使用文献[60]的方法嵌入了一套人体骨架到其中, 该骨架由17块骨骼组成, 共有18个关节. 利用该骨架我们对网格进行分割, 分割出的每个关节部分都和与之最接近的一块骨骼相对应, 如图2.5所示.

2.4.2 人体姿势空间

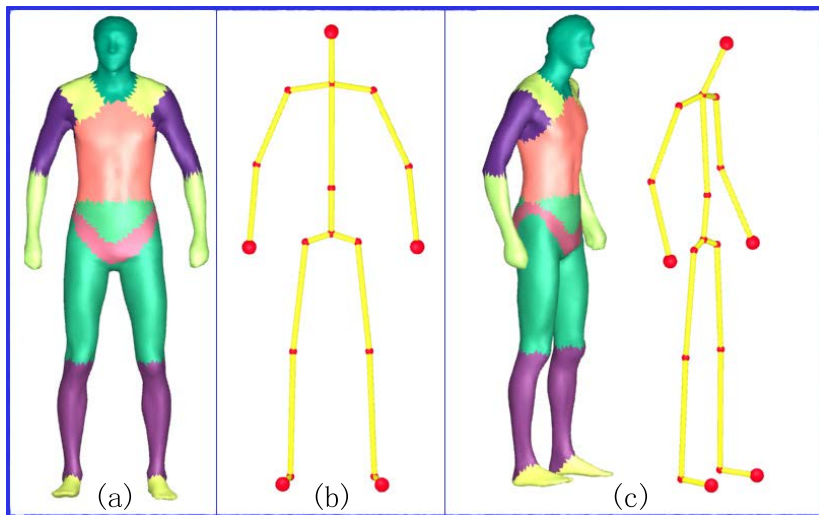


图 2.5 模板网格分割和嵌入的骨架: (a)分割; (b)嵌入的骨架包含18个关节; (c)从另一个视角观察.

人身体中一共有206块骨骼, 由于这些骨骼绕各自的关节转动而形成了不同的姿势. 在姿势变化时, 人身体曲面的几何变化可以看成是由骨骼绕关节转动带来的刚性旋转以及附着于骨骼之上的肌肉收缩而发生的非刚性变换共同作用的结果. 因此我们假设网格上每个三角形都受到刚性和非刚性两种变换的共同作用. 我们设定姿势集合中的模板模型作为姿势变形的源模型, 其余的姿势都能通过模板姿势变形得到.

姿势变形格式

我们将姿势变形局部化到每个三角形上来分析. 我们用 $p[k]$ 表示三角形 T_k 所属的身体部分, 假设其三个顶点是 $v_{k,1}, v_{k,2}, v_{k,3}$. 显然的, 姿势变形是平移不变的, 故而我们将 T_k 平移至以 $v_{k,1}$ 为原点的坐标系到来考虑, 设 T_k 从 $v_{k,1}$ 发出的两条边 $e_{k,j} = v_{k,j} - v_{k,1}, j = 2, 3$, 我们把作用在 T_k 上的姿势变形格式定义为它的两条边向量的变换:

$$e_{k,j} = R_{p[k]}Q_k\hat{e}_{k,j}, j = 2, 3 \quad (2.1)$$

按照这个定义每个三角形先被应用一个 $3*3$ 的线性变换 Q , 它所对应着由肌肉所产生的非刚性变形. 对于每种不同的姿势以及在同一种姿势下位于人体上不同位置的三角形, Q 都是不一样的. 接下来, 每个部分上所有的三角形将被应用一个刚性旋转 R , 它等于该姿势与模板姿势对应骨骼向量间的刚性旋转.

这个变形格式有两方面的用途. 第一个应用是如果已知对应某种姿势的 R 和 Q , 我们可以通过模板合成新的姿势的模型. 由于2.1式对不同的三角形将算出不相容的变换, 因此为了得到一个光滑的模型, 类似于基于微分域坐标的形变方法^[22], 我们求解一个以未知网格顶点坐标为未知数的线性系统来极小化整体的拟合误差:

$$\arg \min_{\{v_0, \dots, v_v\}} \sum_k \sum_{j=2,3} \|R_{p[k]}Q_k\hat{e}_{j,k} - (v_{j,k} - v_{1,k})\|^2 \quad (2.2)$$

注意到2.2式中的 R , S 和 Q 都是平移不变的变换, 因此该线性系统带有一个平移的自由度. 为了使得2.2式能定解我们必须固定一个参考网格上的已知顶点. 另一个应用是如果仅仅已知 R , 我们可以通过分析姿势集合提取出 Q . 具体方法将在下一节给出.

提取姿势变形

在体型不变的前提下, 所有姿势变形的集合实质上形成了一个关系数据库. 建造这个数据库的关键在于建立骨骼刚性形变和肌肉非刚性形变之间的关系. 在此之前我们必须首先提取出姿势集合中每个模型的 R 和 Q . R 的计算较为简单, 它等于每块骨骼与模板骨架的对应骨骼之间的绝对旋转. 对于 Q , 我们通过与姿势集合中的模型进行拟合来求得. 具体来说, 若以非刚性变换 Q 为未知数, 2.2式可被改造成:

$$\arg \min_{\{Q_0^i, \dots, Q_T^i\}} \sum_k \sum_{j=2,3} \|R_{p[k]}^i Q_k^i \hat{e}_{j,k} - e_{j,k}\|^2 + w_s \sum_{k_1, k_2 \text{adj}} I(p[k_1] = p[k_2]) \|Q_{k_1}^i - Q_{k_2}^i\|^2 \quad (2.3)$$

其中的指示函数 $I(p[k_1] = p[k_2])$ 当且仅当 $p[k_1] = p[k_2]$ 时为1, 其余情况为0. 将2.3式应用到姿势集合中除模板之外的每个模型上我们都可以求出与 R 对应的 Q . 类似于文献[40, 57, 61], 这里我们加上一个光滑项用于保证在属于同一个关节部分的相邻三角形受到的变换尽可能相等. 我们设定该项的权因子 $w_s = 2.0$.

由于决定人体姿势的不是骨骼的绝对朝向, 而是相邻的骨骼之间的夹角, 例如腿部的姿势是由膝关节弯曲的角度决定的. 因此, 我们实际上应该将人体姿势定义在关节而不是骨骼上. 如果相邻的两块骨骼的绝对旋转是 R_{l_1} 和 R_{l_2} , 那么我们定义这两根骨骼相连接的关节处的相对旋转为 $R_{l_2}^T R_{l_1}$. 这里我们的骨架总共包含18个关节. 这里需注意胸关节和腰关节实际上对应着多根骨骼, 如图2.5. 与胸关节相连的总共有左右锁骨、头骨和胸骨4根骨骼. 为避免混淆, 我们这里指定的左右锁骨互为相邻骨骼; 头骨和胸骨互为相邻骨骼. 指定腰骨只有胸骨一根相邻骨骼. 此外, 对于位于肢体末端的左右前臂骨、左右脚骨和头骨, 只有一根骨骼与之相邻.

相对旋转的编码方式是比较重要的. 因为在之后生成不同姿势的模型时我们需要对这些旋转进行线性插值, 而这这就要求对该编码做线性插值后能忠实的反应原旋转矩阵插值的结果. 3×3 旋转矩阵本身明显是不适合做线性插值的; 四元数虽然表示紧凑但是并非对任意的插值都有效; 欧拉角则存在gimbal lock的问题[62]. 所以我们使用rotation vector(又称做twist坐标)来表示旋转矩阵. 对于一个 3×3 的旋转矩阵 $[M]$, 设其第 i 行第 j 列元素为 m_{ij} , twist坐标 t 的计算方法如下式:

$$t = \frac{\|\theta\|}{2 \sin \|\theta\|} \begin{bmatrix} m_{32} - m_{23} \\ m_{13} - m_{31} \\ m_{21} - m_{12} \end{bmatrix}$$

$$\text{其中的 } \theta = \cos^{-1} \left(\frac{\text{tr}(M) - 1}{2} \right)$$

(注意到旋转矩阵的twist向量还存在另一种等价的计算方式为:

$$t = r * \text{atan2}(\|r\|, m_{11} + m_{22} + m_{33} - 1) / \|r\|$$

$$\text{其中的 } r = \begin{bmatrix} m_{32} - m_{23} \\ m_{13} - m_{31} \\ m_{21} - m_{12} \end{bmatrix}$$

两种计算方式等价.)

twist表达的优势之一是几何意义明显: t 的方向代表转轴的方向, 而其模长表示旋转的角度(右手系下逆时针方向). 通过twist对关节相对旋转进行编码, 我们对姿势定义的总自由度从使用3*3旋转矩阵时的 $18*9 = 162$ 降为 $18*3 = 54$. 在下一节我们将看到使用twist编码还降低了获取姿势变形函数这个学习问题的维度.

学习姿势变形函数

为了得到骨骼刚性形变和肌肉非刚性形变之间的关系, 我们使用统计学习的方法为每个三角形训练出一个线性回归函数来从 R 估计出 Q . 由于人的肢体是一种分节铰接的结构(Articulated structure), 因此我们假设对网格上每个三角形其经受的非刚性形变只由与它所属关节部分的邻接的两个关节决定. 因此我们为每个身体部分定义一个由两个twist向量构成的6元向量:

$$\Delta R_p = (\Delta R_{p_1}, \Delta R_{p_2})$$

按假定, 我们为矩阵 Q 中的9个元素中的每一个元素都关联一个回归向量, 并用它建立 Q 和 ΔR_p 之间的多项式线性回归关系为:

$$a_{k,mn}^T \cdot \begin{bmatrix} \Delta R_{p_1}^i \\ 1 \end{bmatrix}, m, n = 1, 2, 3$$

按照回归学习的惯例, 我们在 ΔR_p 后附加了一个常偏差项1. 故对每个三角形我们一共需要求出 $7*9=63$ 个回归系数.

由于之前我们已经使用2.3式从姿势集合中提取出了 R 及其对应的 Q , 我们可以对这些数据做最小二乘拟合来求出回归系数. 特别的, 我们求解一个线性方程极小

化如下的二次拟合误差:

$$\arg \min_{a_{k,mn}} \sum_i \left(q_{k,mn}^i - a_{k,mn}^T \cdot \begin{bmatrix} \Delta R_{p[k]}^i \\ 1 \end{bmatrix} \right)^2 + \lambda \|a^T a\|^2, m, n = 1, 2, 3 \quad (2.4)$$

上式中的 $q_{k,mn}^i$ 是从姿势集合中提取出的 Q 的分量. 为了控制过拟合, 我们这里加上正则项 $\|a^T a\|^2$, 其权系数我们设定为0.01. 注意到方程2.4是对每个三角形的 Q 中的每个分量单独求解, 因此我们对每个三角形需执行9次稠密矩阵最小二乘求解. 注意到, 由于之前对 R 使用的了twist向量形式的编码, 使得该稠密矩阵只有7个未知数, 这在一定程度上节省了计算量.

回归向量 a_k 建立起了骨骼形变 R 和肌肉形变 Q 之间的线性关系:

$$Q_k = \Psi_{a_k} (R_{p[k]}) \quad (2.5)$$

它使得我们在生成新的人体姿势模型时能快速的由骨骼向量算出每个三角形在该姿势下应当经受的非刚性肌肉变形 Q . 然后, 将新的 R 和 Q 带入方程2.2就可以求出新的姿势模型.

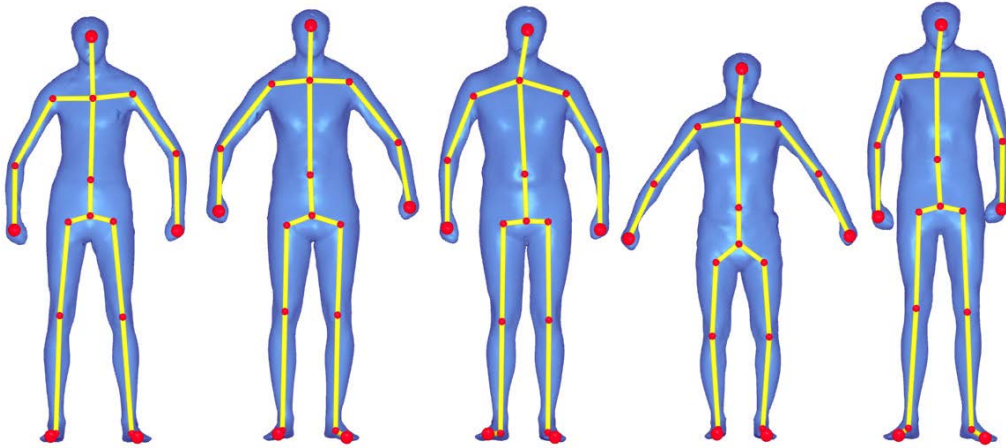


图 2.6 体型集合中的样本和模板样本间也存在一定的姿势差异.

样本局限性. 注意到我们的姿势集合仅仅包含35个样本(见图2.4(a)), 而且其中大部分的样本都是双手不超过肩部的姿势. 这导致我们用在生成与该样本空间差异较大的姿势的模型时变形质量不够理想. 我们通过实验发现, 如果我们在学习姿势变形函数时, 增大与输入姿势相近的样本在式2.4中的权因子, 可以缓解这个问题.

由于肌肉的非刚性形变最终也通过骨骼三维旋转进行表达,因此SCAPE方法建立起了以骨骼三维旋转为参数的姿势模型,我们将这些骨骼三维旋转统一记为姿势参数 θ .

2.4.3 人体体型空间

体型变形格式

影响人体表面几何形状的另一因素是人物的体型.通过改造姿势变形格式2.1,我们得到同时应用姿势变化和体型变化的变形格式为:

$$e_{k,j} = R_{p[k]} S_k Q_k \hat{e}_{k,j}, j = 2, 3 \quad (2.6)$$

其中的 S_k 是三角形 T_k 上的体型变形矩阵,由此得到的同时应用姿势变化和体型变化的网格重建方程是:

$$\arg \min_{\{v_0, \dots, v_v\}} \sum_k \sum_{j=2,3} \|R_{p[k]} S_k Q_k \hat{e}_{j,k} - (v_{j,k} - v_{1,k})\|^2 \quad (2.7)$$

变化连续应用的顺序的重要性.注意到在2.6式我们将体型变换矩阵 S_k 放到了骨骼刚性旋转 $R_{p[k]}$ 和姿势变化引入的非刚性变化 Q 之间.这样安排的顺序主要是基于如下考虑:在从姿势集合中提取姿势变形时,我们显式的去除了骨骼旋转,因此提取出的 Q_k 的旋转度相对较小;而 S_k 则不然,它表示的是体型变化对三角形的影响,其旋转度和伸缩度都比较大:试想象如果三角形 T_k 所属的身体部分是腹部,那么它在一个肥胖的人和模板人物的实例三角形之间存在明显的旋转与缩放.我们通过对 S_k 和 Q_k 做极分解发现, S_k 包含的旋转比 Q_k 包含的旋转大了1到2个数量级.如果我们先应用 S_k 到 $\hat{e}_{k,j}$ 上,其乘积向量 $S_k \hat{e}_{k,j}$ 和 $\hat{e}_{k,j}$ 在方向上有较大的差异,那么之后对 $S_k \hat{e}_{k,j}$ 和 $\hat{e}_{k,j}$ 应用 Q_k 将导致较大的差异,这就意味着体型的变化对姿势的变化产生了较强的影响,而这与我们希望姿势和体型相互解耦合的初衷矛盾.因此我们把 S_k 放在 Q_k 之后.对于骨骼刚性旋转 $R_{p[k]}$,由于它直接决定肢体部分的朝向,有着三个变换矩阵中最大的旋转度,因此我们必须在变形格式的最后应用之.

抽取体型变形

类似于提取姿势变形的方法, 我们改造2.7式将其未知数换成体型变形矩阵 Q_k , 就得到了用于提取体型变形矩阵的方程:

$$\arg \min_{\{S_0^i, \dots, S_T^i\}} \sum_k \sum_{j=2,3} \|R_{p[k]}^i S_k^i Q_k^i \hat{e}_{j,k} - e_{j,k}\|^2 + w_s \sum_{k_1, k_2 \text{adj}} \|S_{k_1}^i - S_{k_2}^i\|^2 \quad (2.8)$$

这里第二项为光滑项, 用于保证在属于同一个关节部分的相邻三角形受到的体型变换尽可能相等. 我们设定该项的权因子 $w_s = 3.0$. 将2.8式应用到姿势集合中的57个样本上, 就得到了他们各自对应的体型变形矩阵 S_k . 我们通过观察发现, 虽然体型集合中的所有样本的姿势虽然相似, 但还是存在明显的差异, 特别是一些比较高或比较胖的样本, 如图2.6所示. 为了尽量去除姿势的影响, 我们首先利用每个样本中嵌入的骨架计算出 R , 再利用姿势变形函数2.5得到 Q_k , 最后带入2.8式求出样本的对应的体型变换矩阵 S_k .

参数化体型变形空间

现在我们获得了体型集合中56组体型变化, 我们希望能从这些数据中得到一个连续的体型空间. 体型空间的真实维数和网格的三角形采样密度应该是无关的. 考虑到人类体型的相似性以及不同体型之间差异并不大, 我们可以假设这个空间能从一个较紧的低维的子空间映射得到. 为此我们使用主成份分析(PCA: Principle component analysis)的方法来得到这个子空间. 对体型集合中的第 i 个样本, 我们首先将其每个三角形的 S_k 拉成一个9维列向量, 这样每个体型样本都对应着一个 $9N$ (N 为三角片总数)的体型变化向量 S^k . 将56个这样的列向量组成一个矩阵 U , 对 U 执行PCA分解我们得到它的一个低维表达:

$$S^i = \Theta(\beta^i) = U\beta^i \quad (2.9)$$

其中的矩阵 U 是由数据矩阵 U 的协变方阵的前 n 个最大的特征值所对应的特征向量组成. 对男性和女性体型集合我们选择 $n = 50$, 以使得能保留原数据集中90%以上的

模式变化. β^i 是第 i 个样本对应的体型参数. 给定一个新的体型参数 $\hat{\beta}$, 就能映射出一组新的体型变形矩阵 $\hat{S} = \Theta(\hat{\beta})$.

到此, 我们就完整的叙述了SCAPE模型 $M = M(\theta, \beta)$ 的建模过程. 在使用SCAPE模型时, 用户只需通过骨架指定一组新的姿势参数 $\hat{\theta}$ 和体型参数 $\hat{\beta}$, 就能通过求解重建方程式2.7合成出数据库中原本不存在的新的身体模型 $\hat{M} = \hat{M}(\hat{\theta}, \hat{\beta})$. 注意到该模型关于体型参数 β 是线性的, 所以能支持实时的体型变形; 但是该模型关于姿势参数 θ 是非线性的, 其非线性主要来自于将骨骼旋转矩阵 R 转换成twist向量时使用了非线性运算如 acos .

2.4.4 语义参数控制的体型变形

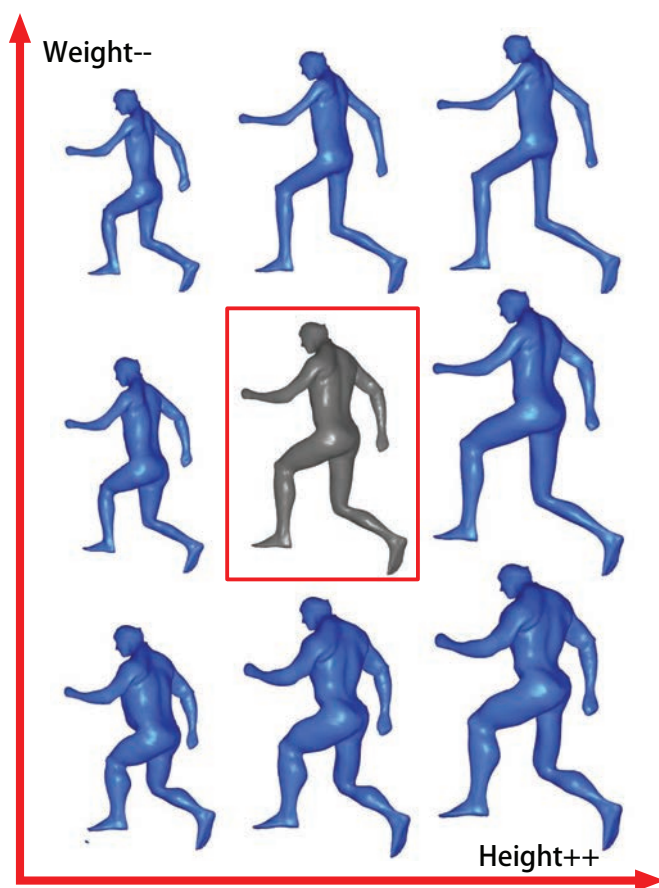


图 2.7 根据语义参数调整体型,方框内为初始体型.

尽管SCAPE提供了一个人体几何空间的紧凑的表达, 但是由PCA分析得到的体型参数 β 不具有语义性, 我们还不能通过一种有语义的方式来浏览这个空间. 直

观来看, 我们希望能以增加体重或者减小腰围等更为直观的方式来修改模型的体型(图2.7). 由于数据库提供了每个被扫描人物的与体型相关的语义参数, 因此我们再次使用线性回归来建立一个从语义参数到体型参数之间的线性映射^[43]:

$$\beta = f(\gamma)$$

其中的 $\gamma = [\gamma_1, \dots, \gamma_l, 1]^T$ 为语义参数, 表示样本的各种身体属性, 在我们的实现中 $l = 10$. 一般情况下, 任意给定的图像中的人物的身体属性一般是未知的, 因此在体型重塑时用户只能给出一组变化量而不是指定一组目标值来修改这些语义参数. 我们定义这样的一组变化量为:

$$\Delta\gamma = [\Delta\gamma_1, \dots, \Delta\gamma_l, 0]^T$$

其中的 $\Delta\gamma_i$ 就是在重塑时对应的语义参数 γ_i 的变化量. 对一个初始模型 $M = M(\theta, \beta)$, 通过修改 $\Delta\gamma$ 我们能得到直接反应体型变化需求的新模型 $M^{\Delta\gamma} = M(\theta, \beta + f(\Delta\gamma))$. 例如, 假设参数 γ_1 对应的语义是人的体重, 那么我们可以通过设定 $\Delta\gamma = [\pm 20, 0, \dots, 0, 0]^T$ 来让人物增加或较少20千克的体重, 如图2.7. 通过以上步骤, 我们得到了语义参数化的SCAPE模型 $M = M(\theta, f(\gamma))$.

2.5 视角相关的模型和图像拟合

图像体型重塑的一个关键步骤是将学习出的男性或者女性可形变模型 $M = M(\theta, \beta)$ 拟合到单张图像上. 注意到这里的一个误区是使用语义参数化的SCAPE模型 $M = M(\theta, f(\gamma))$ 直接做拟合, 也就是说通过拟合直接得出图像中人物的语义属性. 由于我们在建立从语义参数到体型参数之间的映射时所做的线性回归实质上是一个降维的过程, 也即将40维体型参数 β 映射到了10维的语义参数 γ . 因此在体型变化上, 非语义参数模型 $M = M(\theta, \beta)$ 比语义参数模型 $M = M(\theta, f(\gamma))$ 包含了更多的变化模式, 从而能更好的完成与任意图像中人物的体型的对应. 所以我们没有使用降维过的 $M = M(\theta, f(\gamma))$ 模型来做拟合.

文献[53, 55]中提出了一类轮廓线变形算法. 这类算法首先让用户选择出一个特征部位并勾画出该部位的新轮廓, 然后将模板模型变形到与图像拥有相同的轮廓,

这些工作给了我们一些灵感. 我们使用与文献[53]类似的方法来求解未知的姿势和体型参数. 由于他们的方法主要是针对姿势简单而且接近裸露的人体图像, 而我们希望处理姿势复杂同时身着较宽松服饰的情况, 所以我们的方法与之相比有两点根本的不同. 首先, 我们没有使用光照信息的辅助而是专注于图像中的人物轮廓信息来作为我们拟合的根据, 这因为大多数图片都不是在理想的光照环境下拍摄的, 其包含的光照信息不准确, 同时用光照信息对做视角相关的拟合辅助效果不大. 使用人物轮廓作为拟合根据的优点还在于即使对于身着宽松服饰的人物其轮廓也很容易通过用户交互获得. 其次, 文献[53]中通过缓慢的随机搜索算法同时求解姿势和体型参数, 我们则分开求解姿势和体型参数以提高拟合步骤的效率.

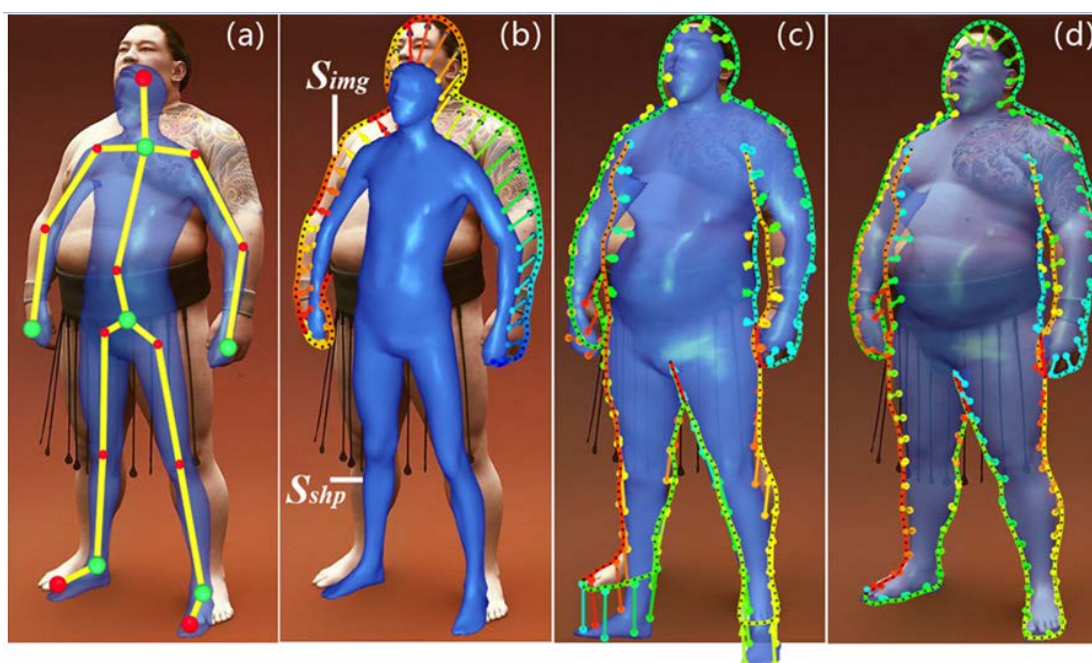


图 2.8 视角相关的模型拟合流程: (a)借助用户交互得到的初始姿势; (b)图像人物身体轮廓线 S_{img} 与三维模型轮廓 S_{shp} 间的最优对应关系(箭头指示); (c)只使用上半身轮廓线拟合的体型; (d)使用全身轮廓线更新的拟合结果.

2.5.1 姿势拟合

我们首先求解姿势参数 θ . 按定义 θ 是关节相邻骨骼间的三维旋转, 因此求解 θ 等同于恢复与图像中人物对应的三维骨架的朝向.



图 2.9 我们的方法可以处理复杂姿势和较多遮挡的图片. 对每一行的三元组: (左)输入图片; (中)模型拟合结果; (右)体型重塑结果. (前(后)2个为增加(减少)体重).

为了能简单而且较快速的达到目的,我们参考文献[45]中的方法,在图像为弱透视投影的假设下求出最优拟合的三维骨架几何.其方法基于这样的观察:铰接结构的三维物体(如人体骨架)的各铰接部分的长度比率,在正交投影或者弱透视投影下保持不变.以人体数据库中模板模型的骨架为模板骨架,我们事先利用用户交互得到了各三维骨骼关节在图像中对应的位置,然后根据这种对应关系确定模板骨架的各根骨骼在相机坐标系中的朝向.在许多图片中,我们利用皮肤检测和人脸检测的辅助来减少需要用户直接标注的对应关节的个数^[44],在图像满足辅助的情况下,例如皮肤颜色清晰的图片,用户只需要交互的给出6个关节的对应像素点(胸关节、腰关节、双手关节和双脚关节),系统将自动算出其余关节的对应点.此外我们的系统还允许用户通过修改相机模型的缩放因子等操作进一步调整骨架的质量^[45, 46].

文献[53]中指出进行姿势拟合时存在一个所谓姿势歧义的问题:即在轮廓相同的情况下,恢复出的模型可能有不同的三维姿势,如图2.10所示.由于我们的目标是完成视角相关的拟合,我们并不需要显式的处理该问题.因为只有当我们从不同于照片的相机的其他视角观察模型时才会发现模型三维姿势上的差别,而我们只要关心以相机做视角的情形下的模型配准.

在求出姿势参数 θ^* 后,我们就得到了初始的人体姿势模型 $M(\theta^*, \beta)$,下面的任务是进行体型拟合.

2.5.2 体型拟合

这一节我们描述体型拟合的方法.我们希望在给定姿势参数 θ^* 的情况下求出 $M(\theta^*, \beta)$ 中的体型参数 β .我们在SCAPE模型中搜索一个三维人体模型以最优的匹配图像数据.我们注意到图像中人物身体的轮廓线不但非常容易被辨析,而且它还在很大程度上刻画了人物的体型.所以我们将匹配的误差定义为图像中人物身体的轮廓线 S_{img} 与三维模型轮廓 S_{shp} 在图像上的投影曲线 $S_{M(\theta^*, \beta)}$ 之间的距离:

$$\beta^* = \arg \min_{\beta} \|S_{M(\theta^*, \beta)} - S_{img}\|^2 \quad (2.10)$$

这里的图像轮廓 S_{img} 包括身体的外轮廓、互相遮挡的内轮廓和被衣服完全遮盖

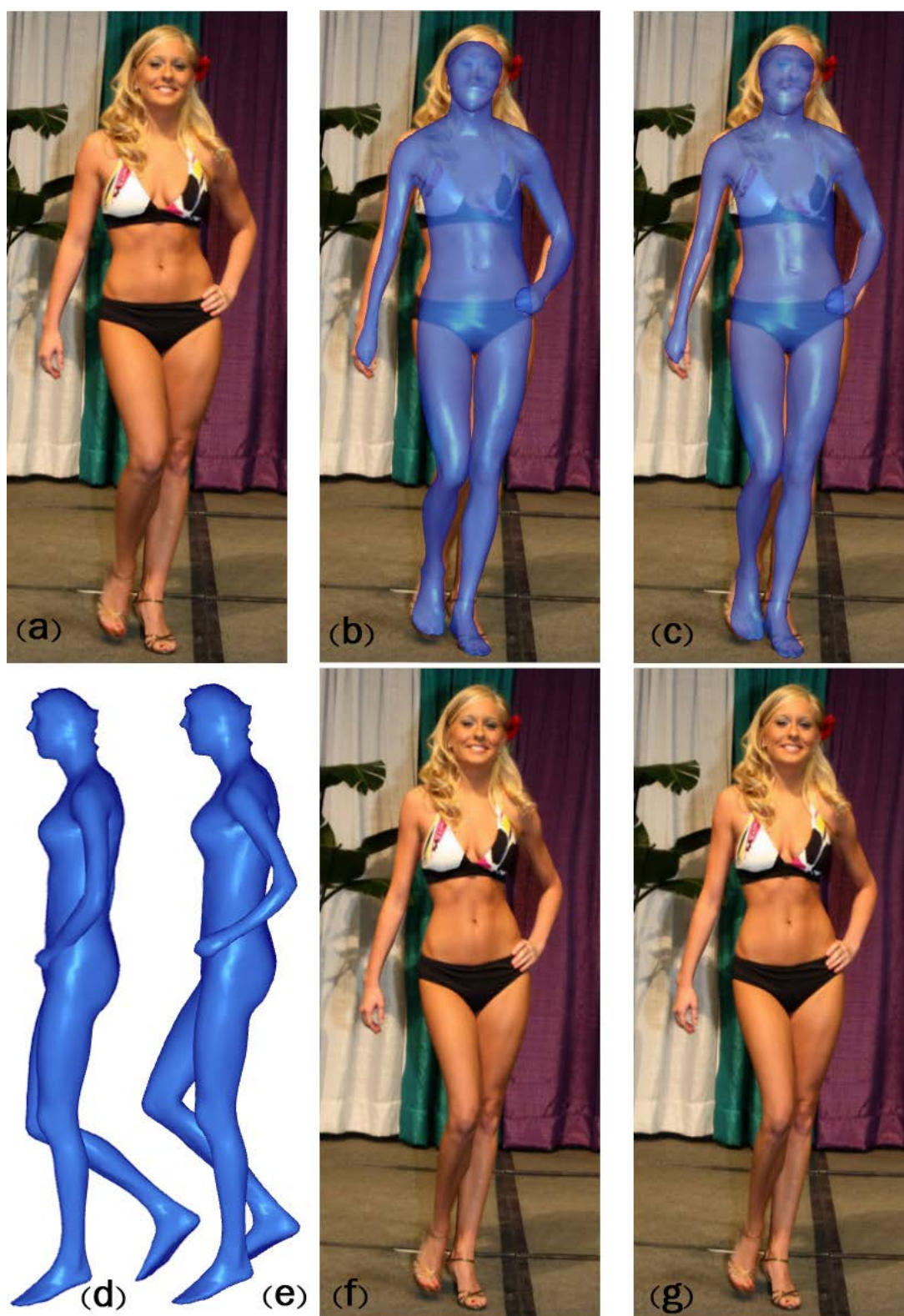


图 2.10 我们的拟合方法对姿势歧义是不敏感的. 我们为(a)中输入图像同样高质量的拟合上两个人体模型(b)(c), 尽管在侧视图可以看出(d)(e)中它们有不同的姿势, 我们的方法能从它们得到几何完全相同的重塑结果(f)(g).

住的身体部分的轮廓. 它可以通过标准的交互式扣图算法如Lazy Snapping^[63]或者智能剪刀^[64]求得. 借助这些半自动的交互算法, 我们的系统能较好的处理人物身体轮廓被遮挡的图像. 注意2.10式将距离定义在图像轮廓线和投影轮廓线对应的顶点点对上.

为了得到体型最优拟合的模型, 我们采用一种类似于文献[54]中提出的方法来迭代优化体型参数 β . 该方法首先从用一组平均体型参数 β^0 得到一个初始模型 $M^0 = M(\theta^*, \beta^0)$, 然后作匹配和变形两步迭代. 在匹配阶段, 我们建立起 S_{shp} 和 S_{img} 上顶点的最优对应关系; 在变形阶段, 我们求解2.10使得 $S_{M(\theta^*, \beta^0)}$ 上的顶点向对应的 S_{img} 上的顶点移动一段距离(该距离等于变形前两者之间距离的一半)得到新模型 $M^1 = M(\theta^*, \beta^1)$. 重复上面的步骤我们能继续求出 $\beta^1, \beta^2, \dots, \beta^*$, 直到最终的 $M^* = M(\theta^*, \beta^*)$ 和 S_{img} 之间的距离小于给定的阈值.

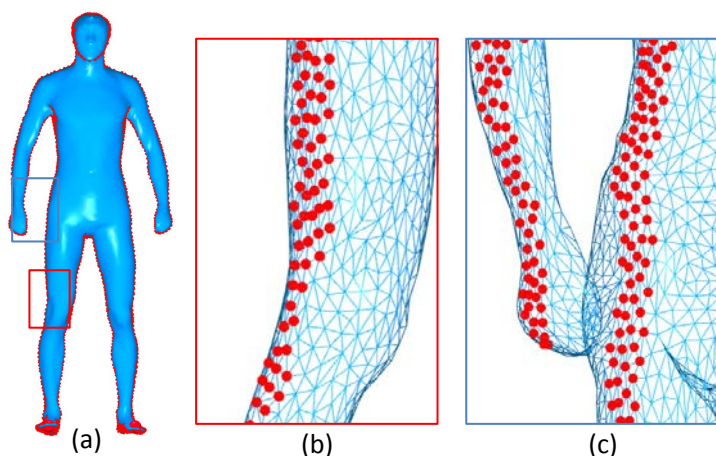


图 2.11 呈带状分布的模型轮廓点: (a)当前视角下的模型轮廓点(红色); (b)(c)局部放大.

匹配阶段. 我们使用基于隐式马尔可夫模型HMM(Hidden Markov Model)的Viterbi算法^[65]来求解 S_{shp} 和 S_{img} 上顶点的最优对应关系. HMM模型是统计分析领域的一个概率模型, Viterbi算法的核心思想是利用动态规划来搜索一个HMM模型的最优参数, 以使得该模型生成的一串状态链能最优的匹配观测到的数据的模式. 在轮廓匹配问题中, 我们将图像轮廓 S_{img} 上顶点作为观测到的数据, 将三维模型轮廓 S_{shp} 上的顶

点作为隐状态. 这样设定的原因在于HMM的观测数据必须为确定有序的, 因此只有二维的线性有序的图像轮廓能符合这个要求. S_{shp} 实则是位于与视线精确正交的轮廓线前后两侧的一条带状区域里的顶点, 如图2.11所示. 在网格上该带状区域有可能出现不连续的跳跃情形(a), 因此我们必须通过HMM模型来估计出其中一个确定有序的子集. 具体的, 我们希望从 S_{shp} 的带状区域中搜索出一条线性顶点序列与 S_{img} 上顶点形成一一对应. 为了避免在随后的变形阶段产生较大的移动误差, 我们要求该顶点序列与 S_{img} 有同样多的顶点数量, 同时对对应点之间的距离尽可能小. 为了进行优化, HMM模型需要定义发射概率和转移概率: 发射概率是由某个隐状态生成某个观测数据的似然率, 而转移概率则是任意两个隐状态之间单向转换的概率(可以理解为围绕原子核的电子在不同能量状态间发生跃迁).

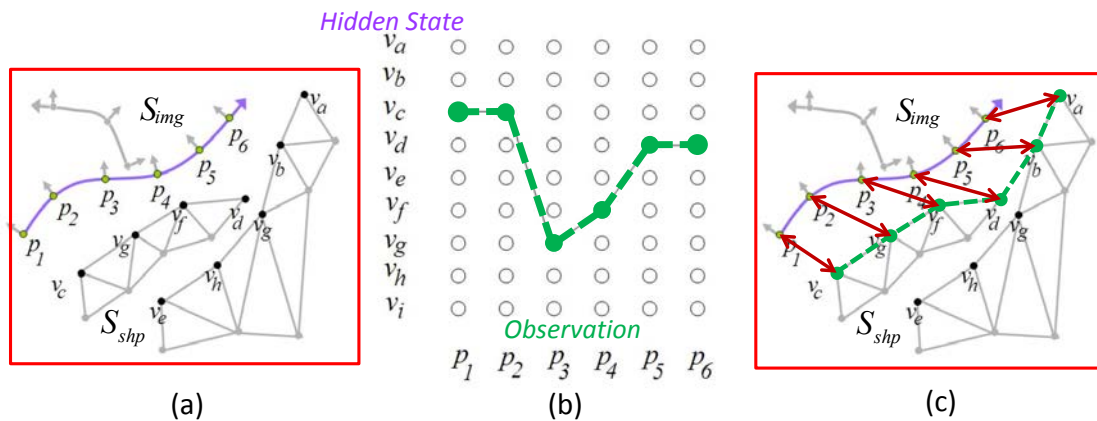


图 2.12 Viterbi算法求HMM隐状态链: (a)需要被匹配的 S_{img} (紫色曲线)和 S_{shp} , 注意 S_{shp} 可以位于不连续的网格区域; (b)动态规划运行中的某一条状态链; (c)最终匹配结果, 注意从 v_d 到 v_b 发生的跳跃.

类似于文献[54]我们用为 S_{shp} 上的顶点 v 和 S_{img} 上的顶点 p 的距离和法向来定义发射概率(Emission probability):

$$P(p_j | v_i) = e^{-\frac{1}{2} \left(\frac{d_p}{\sigma_N} \right)^2} e^{-\frac{1}{2} \left(\frac{d_N - 1}{\sigma_N} \right)^2}$$

上式中的 $d_p = (p^x - v^x)^2 + (p^y - v^y)^2$ 是轮廓点对在二维图像平面内的距离; $d_N = n^p \cdot n^v$ 用来度量法向的差异, 注意到在最优的匹配情况下 $d_N = 1$. 考虑法向差

异能让靠近轮廓平面的网格顶点优先被匹配上. 我们利用在图像平面内的二维距离来定义转移概率(Transmission probability):

$$P(v_i | v_{i-1}) = e^{-\frac{1}{2} \left(\frac{d_C - 1}{\sigma_C} \right)^2}$$

上式中的 $d_C = \|v_i - v_{i-1}\|^2 / \|p_i - p_{i-1}\|^2$, 用来保证 p_i 与 p_{i-1} 的距离和与它们分别对应的网格顶点 v_i 与 v_{i-1} 的距离尽可能相等, 以使得最后得到的 v 序列尽可能的平滑且与 S_{img} 保持全等. 我们在实现中设定 $\sigma_C = 0.5$, $\sigma_N = 1.0$. Viterbi 算法运行过程如图 2.12 所示.

变形阶段. 我们利用轮廓线驱动网格进行变形. 注意到方程 2.10 是关于 β 的线性系统. 原因有两点: 一是因为 SCAPE 的体型空间是关于体型参数 β 的线性空间, 所以在姿势参数已经确定的情况下, $M(\theta^*, \beta)$ 被简化为 β 的线性模型. 二是因为 $S_{M(\theta^*, \beta)}$ 是 S_{shp} 在图像平面的投影, 在弱投影或正交投影的情形下, $S_{M(\theta^*, \beta)}$ 的二维坐标和与之对应的 S_{shp} 的三维坐标之间也是线性关系(相差一个已知的 3×2 的投影变换), 故而 $S_{M(\theta^*, \beta)}$ 也是线性依赖于 β . 所以, 方程 2.10 是关于未知数 β 的线性方程. 只要给定 S_{img} 我们就能反复的快速的求解之.

这里值得注意的是, 由于人的体型和姿势会产生相互的影响(参考文献[43]中统一编码的模型), 因此最理想的拟合方法应该交替的拟合体型和姿势以获得全局优化的结果. 但是随后我们将会看到, 尽管我们的方法可能造成拟合精度上的误差, 但是我们的图像变形的算法仍然能较好的完成重塑任务(如图 2.9). 在实际使用中我们发现, 通过先拟合姿势后拟合体型的两步方法得到的结果模型 $M^* = M(\theta^*, \beta^*)$ 对随后的编辑任务来说, 在大多数情况下是足够好的. 在少数拟合结果不够理想的情况下, 我们的系统允许用户通过调整语义参数控制体型变形来修改体型拟合结果, 或者调整骨架关节的位置得到新的初始模型来重新进行体型拟合.

在得到与图像配准的三维人体模型 M^* 之后, 我们希望能将三维模型的身体特征的变化转移到二维图像上来.

2.6 身体感知的图像变形

令 $M^* = M(\theta^*, \beta^*)$ 为我们拟合好的三维模型(如图 2.3(c)), 令 $M^{\Delta\gamma} = M(\theta^*, \beta^* +$

$f(\Delta\gamma)$ 为在语义参数改变量为 $\Delta\gamma$ 的重塑后的三维模型(如图2.3(d)). 在本节中我们引入我们的身体感知的图像变形算法完成从三维到二维的重塑特征转移. 特别的, 首先我们需要澄清为什么直接的基于模型的方法和基于像素的图像变形算法难以完成此项任务.

2.6.1 直接方法

在设计我们的身体感知图像变形算法之前, 我们试验了两类直接方法: 直接纹理映射法(图2.13(d)(e))和不考虑人体结构的直接图像变形方法(图2.13(c)(f)).

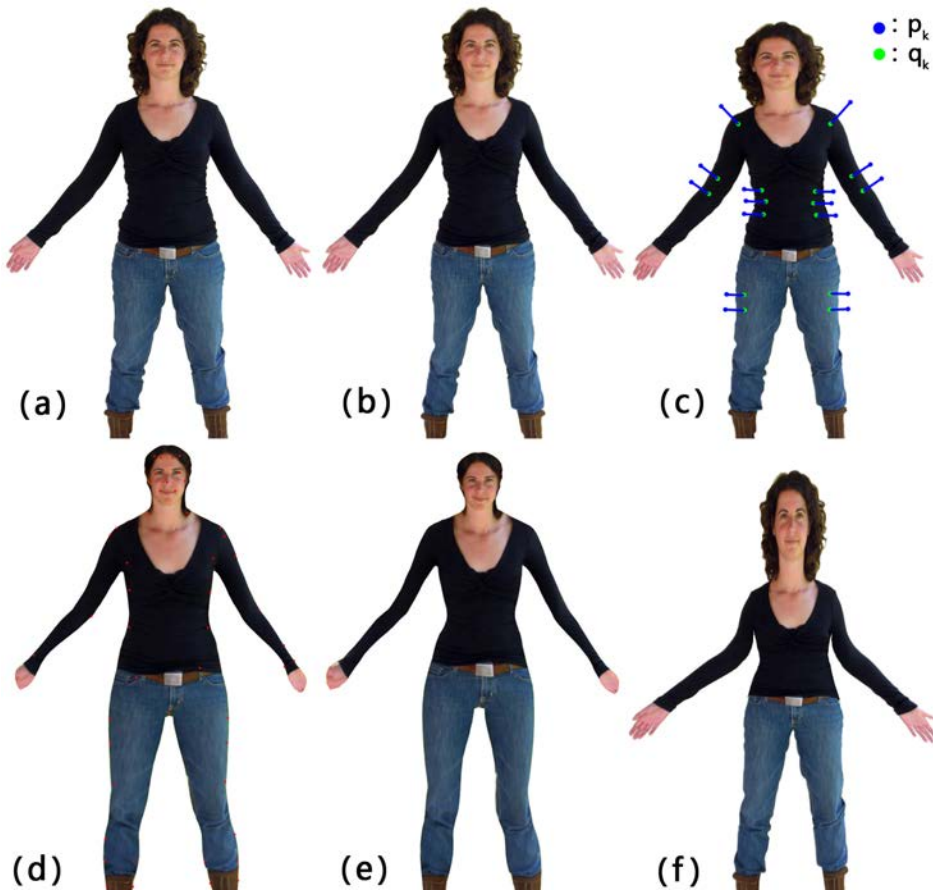


图 2.13 与直接方法产生的重塑效果的比较: (a)输入图像; (b)我们的身体感知的图像变形的结果; (d)使用Flexisticker^[12]方法对拟合模型贴上纹理的结果(红点为映射时的标记点); (e)为对(d)中三维带纹理的拟合模型重塑后的三维模型; (c)为RBF^[13]图像变形结果; (f)为MLS^[14]图像变形结果; 注意(c)和(f)中使用了相同的控制点约束: (c)中身体轮廓外的蓝色点 p_k 移动到 q_k (仅显示了若干代表点对).

直接纹理映射法. 在拟合阶段我们已经拿到了可信的三维全身人体模型, 因此一个自然而然的解决方案就是直接将该模型贴上衣服和皮肤的纹理, 然后把直接对贴有正确纹理的三维模型进行参数式体型重塑. 如果有纹理的模型渲染回图像之后能生成没有瑕疵的结果, 那么体型重塑问题相当于直接在三维中得到了解决.

在现存的工作中也的确有类似的方法. 其中, Blanz等人^[38]提出的可形变人脸模型在做人脸图像修饰时就使用了这样一种直接方法: 使用纹理映射技术直接生成编辑后的图像. 近来, Tzur等人^[12]提出的Flexisticker方法将拟合模型 M^* 贴上纹理(如图2.13(d)), 然后将重塑后的三维模型直接投影为二维图像作为重塑结果(如图2.13(e)). 然而, 由于人体模型本身的复杂度, 在纹理映射时很难保证人物图像能完美的覆盖三维模型. 由于图片中实际人体和我们拟合的模型不可能完全一致, 这常常导致在对初始拟合模型做纹理映射时就造成扭曲和瑕疵(注意图2.13(d)中的人物脖子部分). 其次, 直接纹理映射明显不能处理头发和宽松衣物等身体轮廓以外的部分, 而这在日常图片中随处可见.

不考虑人体结构的直接图像变形法. 另一种直接方法是使用 M^* 和 $M^{\Delta\gamma}$ 之间的变化量用作控制约束来在图像平面作基于像素的变形. 由于是在图像上操作, 这类方法可以将模型的变化传递到头发和宽松衣物等身体轮廓以外的部分. 为此我们试验了RBF^[13]和MLS^[14]两种方法. 我们将 M^* 和 $M^{\Delta\gamma}$ 之间的变化量定义为模型上三维的点在图像平面的投影点位置的绝对移动上, 即 $\mathbf{q}_k - \mathbf{p}_k$, 其中的 $\mathbf{p}_k \in \mathbb{R}^2$ 和 $\mathbf{q}_k \in \mathbb{R}^2$ 分别是 M^* 和 $M^{\Delta\gamma}$ 上顶点的二维投影坐标(如图2.13(c)). 然后我们用 $\mathbf{q}_k - \mathbf{p}_k$ 来驱动基于像素的图像变形. 然而, 这些绝对移动是由三维模型完全决定的, 因此要完成可信的图像变形需要完美的模型拟合; 再者, 这类方法没有考虑人体的身体结构而是对整个图像域做同样的对待, 因此它们无法保持人体体型, 如图2.13(c)(f)所示.

2.6.2 身体感知的图像变形

我们观察到许多重塑的效果, 例如改变身高, 展现出来的是长度的变化. 实际上, 对人的身体进行重塑在很大程度上是在特定的方向上重新调整身体的各个部分的长度和宽度. 这里的特定方向主要包括沿着骨骼的方向, 记为 \mathbf{d}_{ske} , 以及垂直于骨骼的方向, 记为 \mathbf{d}_{ske^\perp} , 这两种方向. 比如增加或减少人的身高主要是对身体各部分沿着其对应的骨骼方向 \mathbf{d}_{ske} 进行缩放变换, 而在 \mathbf{d}_{ske^\perp} 方向上缩放则能很好的模拟增加

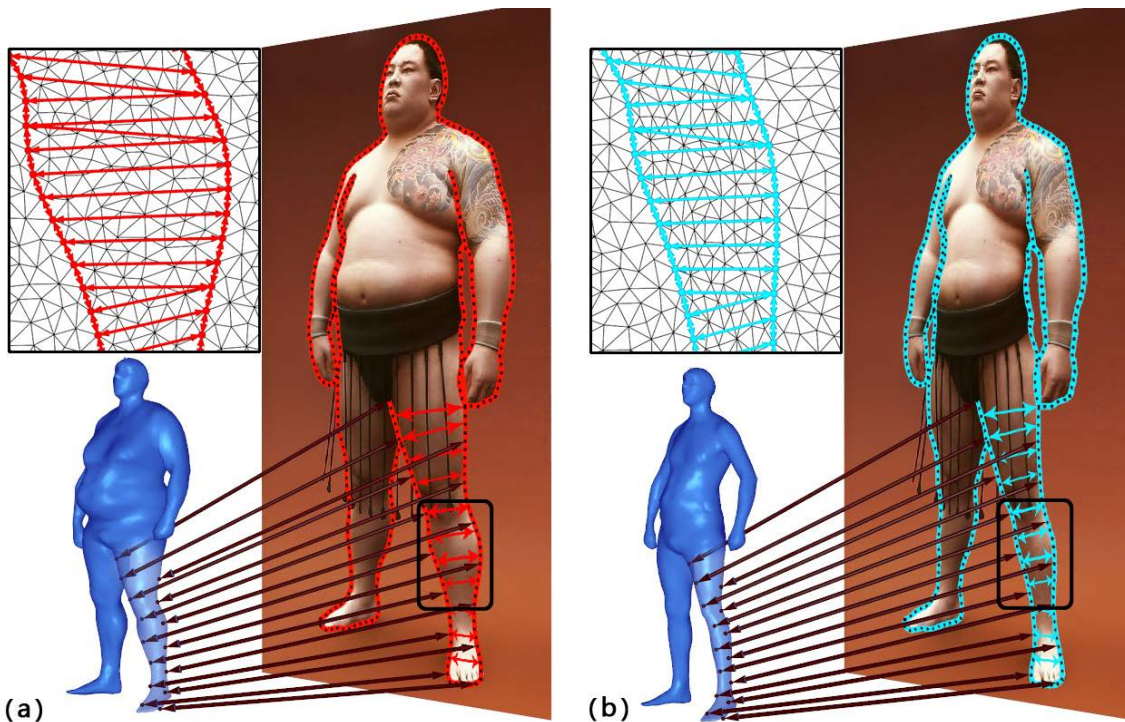


图 2.14 身体感知的图像变形算法示意图: (a)红色表示重塑前人物轮廓; (b)蓝色表示体重减轻后的人物轮廓.

或减低体重的效果. 基于以上的观察, 我们设计了一种骨架感知的图像缩放方法, 该方法的核心思想是将三维模型 M^* 在发生体型形变时在沿着方向 \mathbf{d}_{ske} 和 \mathbf{d}_{ske^\perp} 上的长度的变化转移到二维, 以保证图像发生尽可能相似的变形. 注意到我们的方法与传统的图像缩放方法的最大的区别就是: 传统的方法在 X 轴或 Y 轴方向上只有一个缩放因子, 而我们的方法在重塑体型时将在身体的不同部分应用不同的缩放因子, 以保持形变后的人体图像具有真实感.

为了保持缩放效果的连贯性, 我们将图像嵌入到一个二维三角形网格 Γ , 利用 Γ 驱动图像变形(如图2.14). 我们通过在图像的矩形域上执行一次带约束的Delaunay三角化得到 Γ , 约束是以线段的形式存在的, 包括在姿势拟合步骤中得到的图像轮廓线和图像矩形域的四条边界. 使用 S_{img} 作为三角化的约束能让体型重塑效果在网格 Γ 做变形时得到忠实的表达. 在执行三角化之后, S_{img} 上的采样点就是网格 Γ 中的顶点, 它们沿 S_{img} 顺次链接的边就是 Γ 中的边. 我们通过控制三角化生成的Steiner Point的数量来保证网格 Γ 的质量.

我们优化一个能量函数 E 来驱动 Γ 发生变形:

$$E = \omega_{ske}E_{ske} + \omega_{ske\perp}E_{ske\perp} + \omega_{sil}E_{sil} + \omega_{reg}E_{reg} + \omega_{dis}E_{dis} \quad (2.11)$$

E 由5个能量项加权而成. 在这5个能量项中, E_{ske} , $E_{ske\perp}$ 和 E_{sil} 这三项是专门针对我们的体型重塑问题设计的, 我们通过优化这三项能量来极小化三维和二维在长度度量上的缩放比率之间的差异, 以保证 M^* 和 Γ 上发生尽可能特征相似的形变. 这三项有相同的形式:

$$E_{\Delta len}(P) = \sum_i \left\| e_i^{\Delta\gamma}(\mathbf{V}) - \frac{\|g_i^{\Delta\gamma}\|}{\|g_i\|} e_i \right\|^2 \quad (2.12)$$

其中的 $P = \{(g_i, e_i)\}$ 是一个边向量二元组, $e_i \in \mathbf{R}^2$ 表示位于 Γ 所在的图像平面中的边向量, $g_i \in \mathbf{R}^3$ 表示与 e_i 对应的由模型 M^* 定义的边向量. $e_i^{\Delta\gamma}$ 和 $g_i^{\Delta\gamma}$ 是在重塑后发生改变 e_i 和 g_i . 注意到 $g_i^{\Delta\gamma}$ 是已知量, 它直接来自体型变化后的三维模型 $M^{\Delta\gamma}$; $e_i^{\Delta\gamma}$ 则是未知向量, 它是方程未知数 \mathbf{V} 的线性函数(\mathbf{V} 是变形后 Γ 上顶点的未知坐标). 使用不同的边向量二元组 $P = \{(g_i, e_i)\}$ 能完成不同的重塑操作, 下面我们分别介绍 E_{ske} , $E_{ske\perp}$ 和 E_{sil} 这三个能量项和他们使用的边向量二元组.

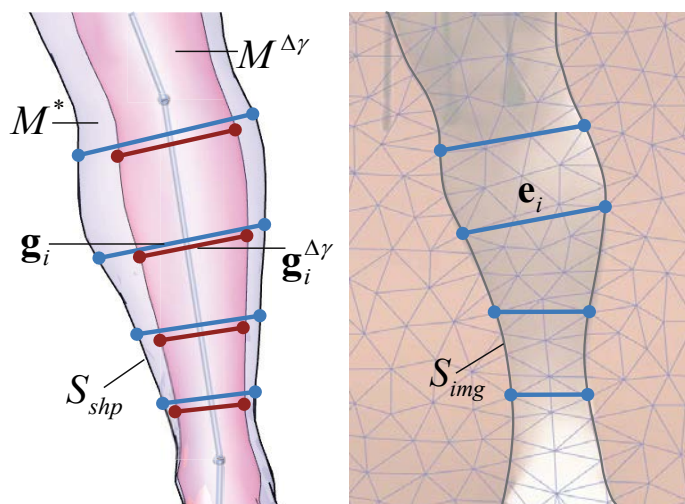


图 2.15 能量项 $E_{ske\perp}$ 示意图.

能量项 $E_{ske\perp}$: 我们观察到当人物发生体重上的改变时, S_{shp} 沿着 $\mathbf{d}_{ske\perp}$ 会发生横向偏移, 偏移的距离和体重的改变直接相关. 为了能将三维模型沿着 $\mathbf{d}_{ske\perp}$ 方向的变化

转移到到图像中的人物部分, 我们首先在 S_{shp} 的顶点中沿着 $\mathbf{d}_{ske\perp}$ 方向采样大量点对 $S_{pair.3d}$, 每个点对之间的三维连线与他们所属的身体部分的骨骼近似的垂直(如图2.15). 利用在体型拟合时得到的 S_{shp} 和 S_{img} 上顶点之间的对应关系, 对 $S_{pair.3d}$ 中的每一个点对, 我们都能从 S_{img} 上的顶点中取出对应的一个点对, 我们记在这些点对的集合为 S_{pair} . 因此 $S_{pair.3d}$ 与 S_{pair} 中的点对不但数量相同, 而且存在一一对应关系. 注意对于那些胸关节和腰关节附近的区域, 我们不做点对 $S_{pair.3d}$ 的采样, 因为这些关节的度数大于2, 我们无法唯一指定某一根骨骼来定义垂直关系.

利用 $S_{pair.3d}$ 与 S_{pair} 我们得到模拟体重变化的能量项 $E_{ske\perp} = E_{\Delta len}(\{(e_i, g_i)\})$. 其中的 e_i 是由 S_{pair} 中的点对相连而成的二维边向量. g_i 则是与 S_{pair} 相对应的 $S_{pair.3d}$ 中的点对相连而成的三维边向量, g_i 连接了 S_{shp} 的左右两侧. 注意 S_{pair} 中的点同时是 $Delaunay$ 三角化生成网格 Γ 时的约束点, 故而 S_{pair} 中的点就是 Γ 的顶点, 因此 $e_i^{\Delta\gamma}(V)$ 就等于变形后 Γ 中未知顶点位置之间的差.

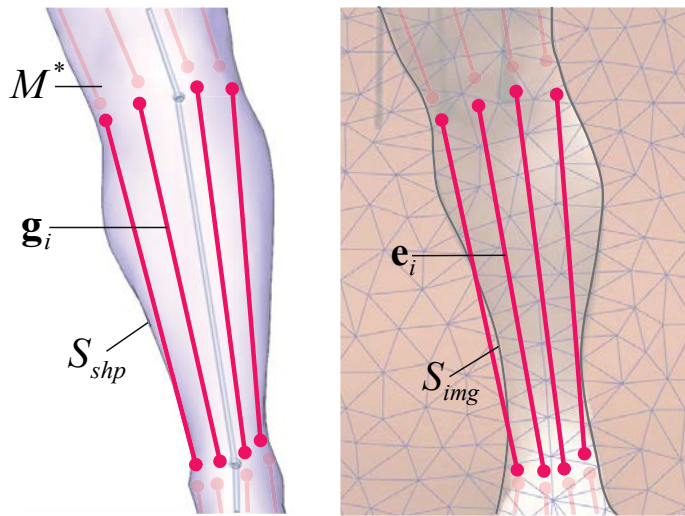


图 2.16 能量项 E_{ske} 示意图.

能量项 E_{ske} : 为了得到沿着骨骼方向的缩放效果, 我们需要沿着平行于骨骼方向采样一系列点对. 类似于定义能量项 $E_{ske\perp}$ 的做法, 我们在 Γ 和 M^* 上采样相同数量的点对. 对 M^* 上每个身体部分, 我们首先在它的两个末端边界区域采样一些顶点, 然后我们为每个顶点找到对面的末端边界上的一个对应点, 使得这两个点之间的连线基本平行于该身体部分的骨骼方向(如图2.16). 我们把这样得到的点对形成的边向

量记为 g_i , 把 g_i 在图像平面上的投影边向量记为 e_i . 考虑到有些图片中的人物只露出一部分身体, 因此我们抛弃那些两个端点都不在图像矩形域中的点对, 以解决可见性的问题. 然后我们定义 $E_{ske} = E_{\Delta len}(\{(e_i, g_i)\})$. 因为边向量 e_i 的两个端点有可能落在 Γ 的某个三角形之内, 我们用重心坐标将 $e_i^{\Delta\gamma}$ 表示为未知顶点 V 的坐标的线性组合.

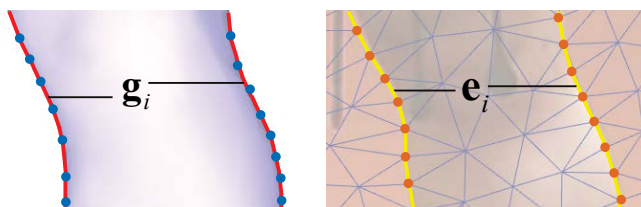


图 2.17 能量项 E_{ske} 示意图.

能量项 E_{sil} : 为了防止变形后的新人物轮廓发生剧烈的变化以至失去真实感, 我们引入能量项 E_{sil} 来考虑轮廓线本身发生的长度变化. 其形式为 $E_{sil} = E_{\Delta len}(\{(e_i, g_i)\})$. 其中的 e_i 是由 Γ 的在 S_{img} 上的相邻的顶点定义的(如图2.17), 由这些相邻顶点在 S_{shp} 上对应的顶点定义了边向量 g_i .

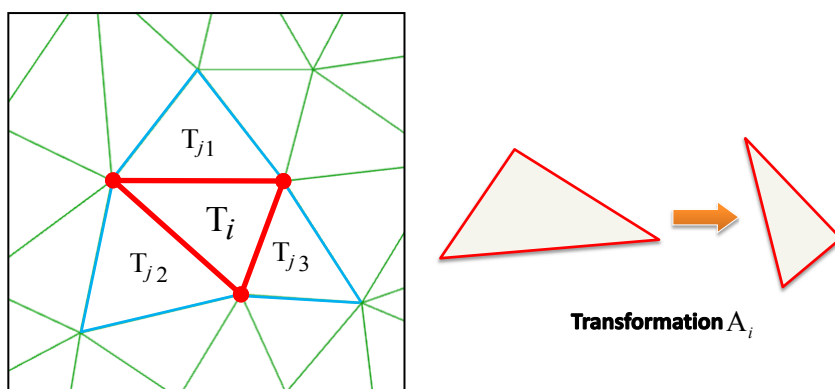


图 2.18 能量项 E_{reg} 和 E_{dis} 用于保证光滑变形.

能量项 E_{reg} 和 E_{dis} : 除了专门为体型重塑问题设计的三个能量项之外, 我们还使用两个相对一般化的能量项 E_{reg} 和 E_{dis} :

$$E_{dis} = \sum_i u_i \|A_i - I\|_F^2$$

$$E_{reg} = \sum_i \sum_{j \in N(i)} \|A_i - A_j\|_F^2$$

其中的 A_i 为在变形时 Γ 中的三角形 T_i 经受的二维仿射变换(如图2.18所示). A_i 作用在 T_i 的两条边向量 e_1 和 e_2 上:

$$\begin{cases} Ae_1 = e_1^{\Delta\gamma} \\ Ae_2 = e_2^{\Delta\gamma} \end{cases}$$

因此它可以表示成变形后未知顶点坐标的线性函数. 这里的 E_{reg} 用于极小化相邻的三角形所经受的变形的差别, E_{dis} 用于极小化变形和同一变化的差距以防止剧烈的形变, 它的权因子 u_i 等于三角形 T_i 包含的图像内容的重要程度. 对人脸与头发等不应随体型发生改变的区域的三角形, 我们使用较大的 u_i 来保护它不被扭曲. 类似于文献[37]我们的系统提供自动检测和手动交互两种方式生成图像重要度图(Saliency map).

最终的能量 E 是未知顶点坐标的二次函数, 极小化 E 等价于求解一个稀疏线性系统. 当用户调节语义参数时只会引起长度的改变, 所以这个线性系统只有右端向量受到影响. 对此我们预分解系统的左端矩阵以达到实时重塑的编辑效果.

权因子: 因为模型拟合的结果存在一定程度的误差, 因此保持相对长度变化的三个能量项 E_{ske} , $E_{ske\perp}$ 和 E_{sil} 实质上是作为软约束的形式存在的. 因此我们用权因子 ω_{ske} , $\omega_{ske\perp}$, ω_{sil} , ω_{reg} 和 ω_{dis} 来平衡他们在 E 中的强弱. 经反复试验, 我们在系统中设定 $\omega_{reg} = 1, \omega_{dis} = 1, \omega_{ske} = 8, \omega_{ske\perp} = 10$ 以及 $\omega_{sil} = (\omega_{ske} + \omega_{ske\perp}) / 4$ 为它们的默认值, 在随后将展示的大部分实验结果中我们采用默认值. 在所有的实验中我们固定 $\omega_{reg} = \omega_{dis} = 1$. 注意到 E_{ske} , $E_{ske\perp}$ 的强弱还和他们各自对应的点对的采样数量有关, 如果点对采样较多, 可以将 ω_{ske} 和 $\omega_{ske\perp}$ 调低. 用户还可以控制身高相对体重的调整效果的强弱程度. 如果希望固定住身高而只改变体重, 可以设定 $\omega_{ske\perp} = 0$. 最后, 注意到我们总是使用相对较小的 ω_{sil} 以确保当需要较复杂的体型变化(例如减小体重同时增大腰围)的时候, 网格能产生相应的光滑变形.

2.7 实验结果和讨论

我们对大量的图片做实验来验证我们的体型重塑系统的表现, 这其中包括各种

复杂姿势和自我遮挡的人物图像. 为了保证在调节后的人体体型仍具真实感, 我们的系统为每个语义参数以及对应的滑杆设定了可以调节的最大范围, 例如对体重参数我们限定其调节范围为 $[-30\text{kg}, 30\text{kg}]$. 这个范围设定还和我们的人体数据库有关, 例如男性数据库中体重的最大值, 最小值和平均值分别等于为110.4千克, 59.2千克和75.72千克, 这使得训练出来的SCAPE模型的体重范围和我们设定的调节范围是基本吻合的. 我们只有在确保三维可形变模型本身的视觉可信度的前提下, 才能将合理的重塑效果由模型转移到图像上. 我们的系统允许用户能通过操控一个滑杆或同时控制几个滑杆的方式来给出语义特征的偏移向量 $\Delta\gamma$, 来得到想要的体型重塑效果(如图2.19).

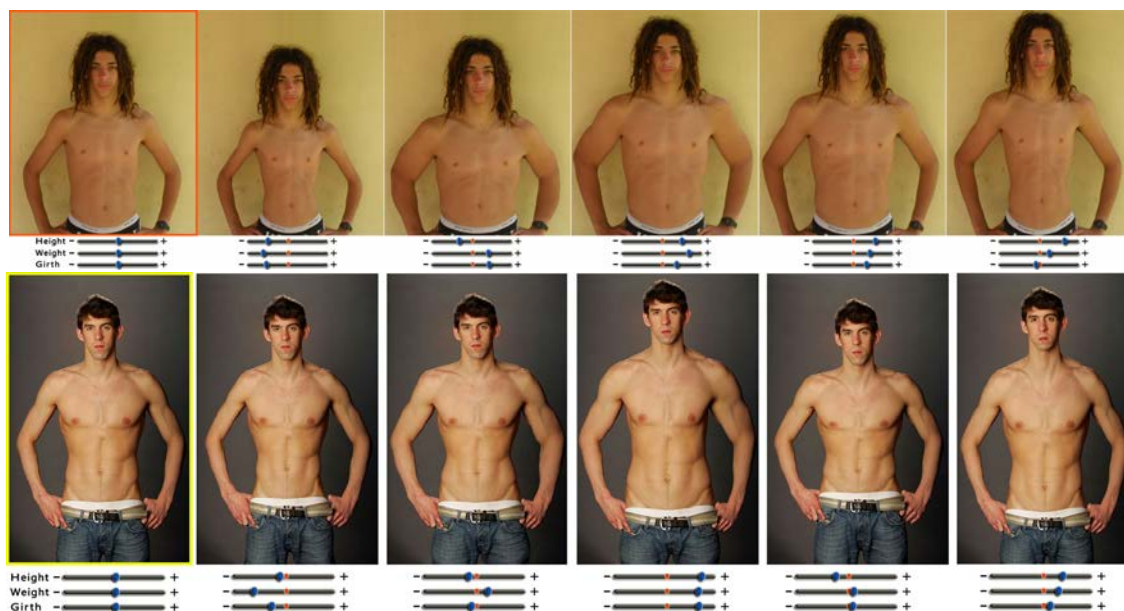


图 2.19 同时调整身高、体重和腰围得到的多个体型重塑结果, 左一是输入.

图2.22 展示了通过调节人物的身高和体重而得到的一系列重塑结果. 我们将它们按照从左到右遮挡逐渐增加和从上到下姿势逐渐变复杂的顺序进行了排列. 图2.20显示了对单个语义参数进行调节的重塑结果, 这里显示的是对腰围、胸围和腿长的调节. 由于我们的系统提供参数化的调节, 使得用户能通过任意的和连续的调节语义参数, 来快速的获得输入的图像的对应不同体型特征的效果, 如图2.1和图2.19所示.

我们的姿势和体型拟合步骤需要用户辅助, 对于较为简单的例子我们需要花费

数分钟的时间(如图2.20右和图2.9上), 对于复杂的例子(如图2.9下和图2.22(d3))则需要花费10多分钟的时间. 其中最耗时的部分就是用户需要交互的给出图像中人物的轮廓. 因为我们的系统允许先对部分的轮廓进行体型的拟合, 所以用户可以根据系统的反馈逐步的增加轮廓线来完善体系拟合的结果. 特别对那些遮挡较严重的例子, 用户可以首先给出容易辨认的区域的轮廓(比如裸露的皮肤区域), 如果初步拟合的结果不够满意, 则可继续为那些不确定区域(比如图2.22(a3)中被衣服遮挡的部分)添加轮廓线. 尽管如此, 我们必须指出在拟合过程中所需要的用户交互都是十分直观的, 因为它们都是在直接标定输入图像中的已有的特征(比如关节点和轮廓). 然而, 对人物进行体型重塑则是一项需要创意的任务, 因为最终的重塑的效果是无法预先看到的. 这也说明为体型重塑提供一套参数化的编辑方法是有用的.



图 2.20 调整单个语义特征得到的体型重塑结果.

我们的图像变形算法能较好的处理身体上的附属物体, 如衣服和头发等等, 甚

至当这些物体和身体交叠在一起的时候也是如此. 这就避免了在传统的方法中经常需要使用的一套比较麻烦的流程: 首先分割出人体的区域(一般包括了头发、衣服和装饰物等), 然后对分割出的部分进行体型重塑, 最后将重塑过的人物部分贴回原图像并执行补洞操作得到最终结果. 然而, 我们的图像变形可能会对那些带有结构特征的背景部分产生可以察觉到的扭曲, 例如图2.22中的地面和墙壁部分. 类似于已知的内容感知的图像缩放方法中提到那样, 让用户手动的指定图像的重要度图能减轻该问题. 我们的方法既能被应用到整幅图像也能仅对人体区域做编辑, 为了简单起见本章中的所有结果中我们都是对整幅图像进行编辑.

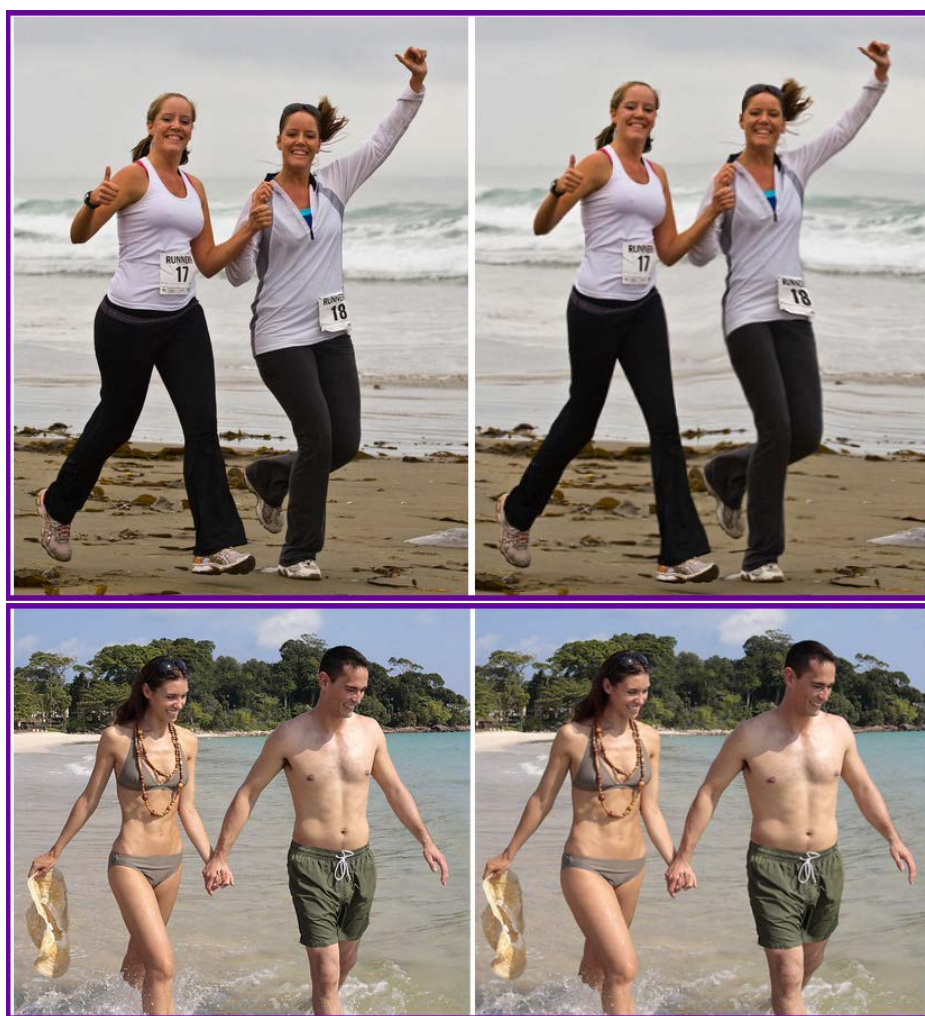


图 2.21 对单张图像中多个人物同时重塑的结果, 第一行为同时减少体重, 第二行为同时增加体重.

我们的系统还可以对单张图像中的多个人物进行重塑: 我们首先对每个人物完

成拟合, 然后所有模型变形引入长度变化将驱动图像网格完成一次性的优化变形. 试验结果如图2.21所示.

此外, 尽管对那些被遮挡的区域做图像变形时, 分层次的表达是更合适的^[48]. 然而我们通过实验发现我们的单层方法已经能足够好的处理图像中人物区域和背景区域间的关系. 其原因在于我们使用的三个专门为体型重塑问题设计的能量项 E_{ske} , $E_{ske\perp}$ 和 E_{sil} 是遮挡感知的, 它们的权因子远大于另外两个不考虑遮挡问题的能量项 E_{reg} 和 E_{dis} . 我们可以利用图2.20左边的图像来做进一步解释. 在这个例子中, 由三维模型引入的相对变化使得人物的腰部发生了水平的收缩. 但是放在人物腹部的手臂和手掌部分并没有发生变化, 这是由于拟合的三维模型并没有在这些部分发生形变.

我们将我们的方法做出的结果与专业的艺术家使用Photoshop[®]编辑得到的结果进行了对比. 读者可以访问我们的项目主页查看对比的结果. 通过操控两个语义参数我们就能得到使用Photoshop[®]需要许多局部编辑操作的效果. 在我们的方法完成对全局体型的编辑之后, 其他的编辑工具, 例如皱纹去除和人脸美化工具等, 能进一步用来做局部的修饰.

我们方法的局限性主要包括两点. 第一点是我们的全身可形变模型是由一个小型的人体扫描数据库训练的来的, 它显然无法涵盖整个人类体型. 因此用它来拟合某些特定姿势或体型的时候可能出现较大的误差. 例如, 我们的方法基本不能处理儿童的图像, 这是因为我们使用的人体数据库只包含成年人的扫描模型. 第二点是我们的方法是通过一个二维的图像变形来达成体型重塑效果的, 它无法完全的模拟三维的变形效果, 特别是当在平行于相机投影方向上的出现较大的形状变化时, 我们的方法无法对相应的光照变化进行编辑. 比如在图2.3(f)中可以看到体重减轻后的人物其腹部显得不够真实.

2.8 本章小结

在本章中我们提出了一种易于使用的图像修饰系统用来对单张图像中的人物体型进行重塑. 我们的核心思想是将三维的体型特征转移到二维图像上来指导重塑. 我们通过嵌入一个三维全身可形变模型将重塑的形变过程进行语义参数化, 同时获得

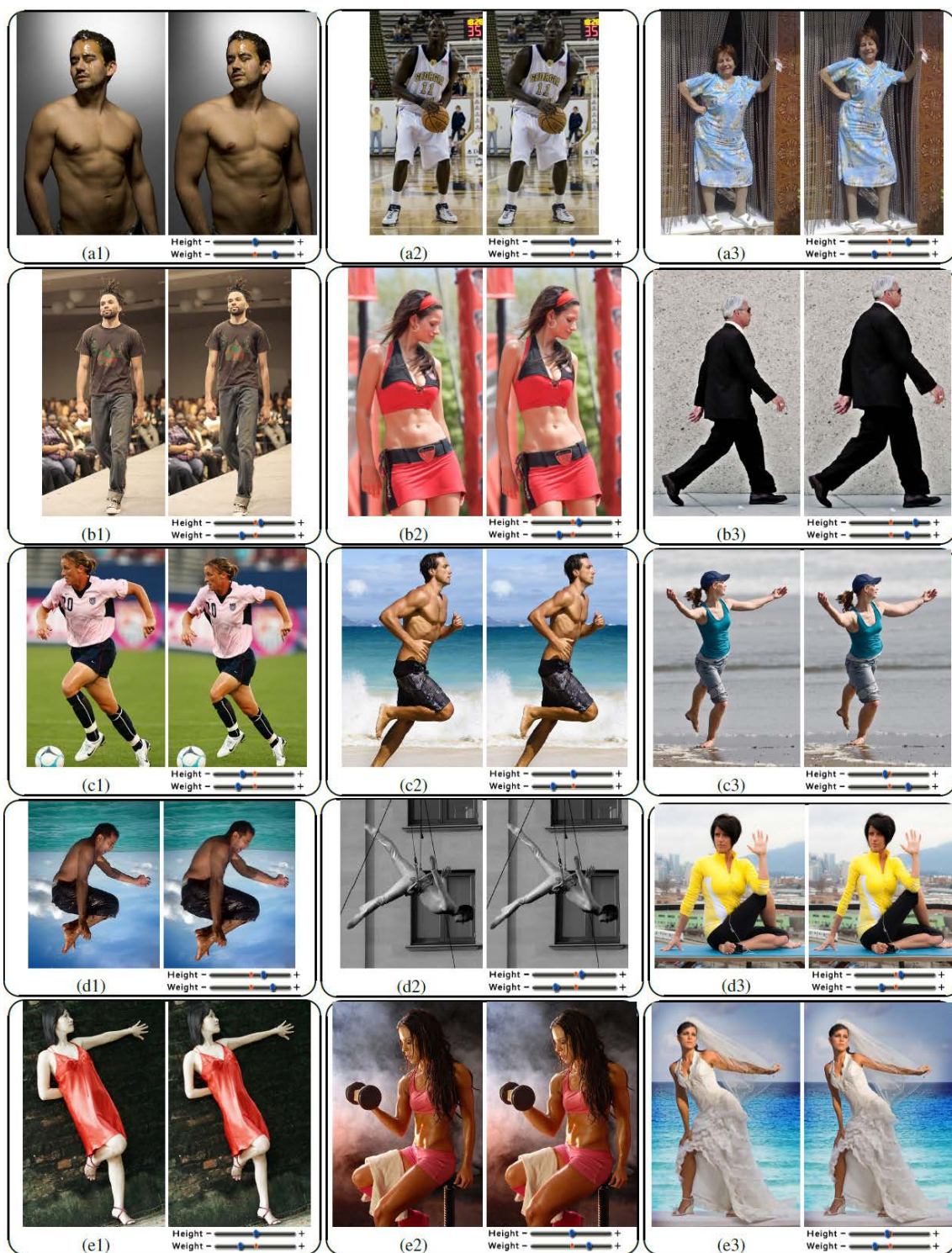


图 2.22 通过调整身高和体重得到的一系列代表性的体型重塑结果,按照姿势复杂度(从上至下增加)和遮挡的多少(从左至右增加)的顺序排列。

了全局协调而且视觉可信的编辑效果. 为了完成三维体型特征变化到二维图像的转移, 我们设计了一种新的身体感知的图像变形方法成功的克服了在嵌入三维模型时存在的拟合误差和图像本身存在的较为严重的遮挡问题. 在一定量的交互帮助下完成三维模型的拟合之后, 我们的系统允许用户通过控制若干个语义参数的滑杆来实时对图像中的人物进行体型重塑编辑.

利用三维模型驱动二维图像变形的思想为我们提供了未来的研究方向. 如果能将基于模型的编辑流程从人物图像扩展到其他内容的图像, 比如人造物体图像等, 相信能带来新的实用的参数式全局编辑方法和具有真实感的编辑效果.

第3章 从二维到三维的特征转移:特征对齐的纹理合成*

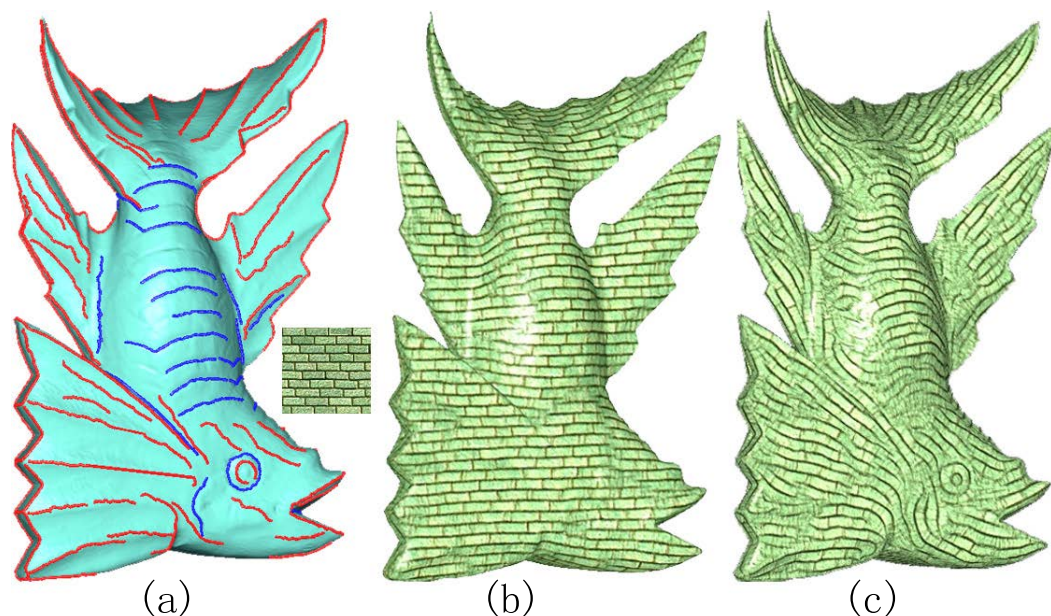


图 3.1 (a): 鱼模型及其上的特征曲线(蓝色是谷线, 红色是脊线); (b)对带有特征的纹理使用传统的合成方法容易隐藏曲面上丰富的几何细节特征; (c)我们的特征对齐的纹理合成能使得这些三维特征凸显出来, 注意曲面的三维特征线和样本中的二维特征线的精确对齐.

3.1 问题介绍

上一章我们讨论了如何用三维模型的特征变化转移到二维图像来驱动形变. 在本章中我们反过来研究二维的图像特征向三维模型的转移. 我们注意到在一些网格模型上, 存在一些明显的特征线, 这些特征线能够抓住曲面上最精华的几何特征. 很多复杂曲面的几何特征往往能通过若干条特征线进行表达, 因此在非真实感渲染领域二维图像中和三维模型表面的特征线是一类重要的研究对象. 比如说, 一张二维图像中的梯度值局部极大曲线能够提取出该图像的外观特点, 同时一张三维曲面能

*本章内容发表在:Feature-Aligned Shape Texturing. ACM Transactions on Graphics, 2009,28(5).

通过那些为与其上的隆起和凹陷部分的关键部分的特征线得到很好的可视化表达. 如果我们要使用颜色或者纹理信息来合理而准确装饰一个模型曲面, 我们应该好好的利用这些特征线的丰富的表现力. 换句话说, 当这些装饰信息和特曲面上的征线有较好的匹配的时候, 模型的外观才能得到准确的表达而不是被这些装饰掩盖起来.

基于上述考虑, 我们探索了曲面特征线在纹理合成领域的应用, 提出了一种能合成较为直观的方式反应三维形状的纹理的方法. 我们的方法遵循两个准则: 一是使得纹理样本中模式的方向和特征线的方向保持一致, 二是使得纹理中的主特征线和模型上的特征线精确的对齐. 我们的方法合成出的纹理不仅能与模型的几何自然的拟合, 同时能让曲面上那些最精华的几何特征在视觉上凸显出来(如图3.1). 我们把这种方法合成出的纹理称作特征对齐的形状纹理. 与传统的向量场导引的纹理合成方法相比, 我们的全自动方法引入了两个技术贡献: 一是特征线定向的优化方法, 二是特征对齐的纹理优化方法.

3.2 相关工作

我们的工作主要与如下三个领域相关: 纹理合成, 向量场生成以及基于特征的形状视觉表达.

3.2.1 纹理合成

在纹理合成方面, 计算机图形学界做出了大量的研究工作^[66, 67]. 给定一个纹理样本, 曲面纹理合成技术旨在网格表面生成一个纹理图像, 该图像和样本需要保持高度的相似性同时又带有足够丰富的变化. 该纹理图像以无缝的方式附着在网格曲面. 与平面纹理合成不同, 曲面纹理合成的纹理图像不是以像素为存储的基本单位, 而一般是以网格顶点的颜色为存储单位. 按照最小的合成单位, 已知的纹理合成算法可分为两类: 基于像素的纹理合成和基于块的纹理合成. 首先出现的是基于像素的合成方法, Wei^[68]和Turk^[69]等人将之推广到了曲面上. 由于基于像素的算法合成速度非常慢. 同时由于它是一种局部增量式的合成方法, 与合成的顺序相关. 在增量合成的过程中与样本间的相似度误差会不断的累计从而导致丢失纹理模式的特征和结构, 因此被后来出现的基于块的合成方法所取代.

基于块的方法每次向目标纹理中植入一个来自样本的邻域块^[70], 由于邻域块更

容易抓住纹理的结构,因此此类方法提高了合成结果与样本整体外观的相似度.此类方法中最早是由Praun^[71]提出的Lapped Texture方法,该方法将用户指定的不规则的样本块随机的粘贴在网格曲面上得到合成结果,块的边界不通过样本中texton的边界,同时块的相邻部分由不透明度度控制混合.其算法成功的要点在于人的视觉对不规则的不连续缝较为不敏感.Zelinka等人^[72]则执行合成之前先对样本执行一个预处理为每个块建立一个跳跃图(JumpMap),该跳跃图定义了样本中每个块与拥有相似邻域外观的块之间的映射关系,跳跃转移的几率是由邻域外观的相似度和块在样本中位置共同决定的.合成时通过在这个跳跃图中遍历一条随机游走的路径得到结果;但是该方法得到的结果中所有的像素都是直接来自于样本,不保证在相邻像素间有合理匹配,其结果存在较多破损.Soler等人^[73]和Nealen等人^[74]都使用了多尺度的块进行合成,并自适应的把匹配误差大过阈值的块分裂;后者进一步的从块重叠区域挑出那些错误像素并使用基于像素的纹理合成法加以修正.

大多数基于块的合成实际上都是在建立一个从目标块到样本块的单射,因此并不能保证样本中所有的块都出现在目标中,当样本为不满足马尔可夫随机场(MRF:Markov Random Field)假设的非均匀纹理时,单射合成会导致目标和样本外观差异较大^[75].文献[76-78]通过反过来建立一个从样本到目标的映射来完善两者间的相似关系.其中文献[76]中将大尺寸的非均匀纹理压缩成一个较小的纹理样本,他们通过向文献[79]中提出的纹理能量中加入一个反向纹理能量以保持原样本中的变化度不丢失,从而得到的小样本在能合成出于原样本相同品质的结果,而小样本的合成速度是明显要更快的.文献[78]中则观察到非均匀纹理如铁锈等,实际上是位于不同薄层的不同材质的叠加,每一层上不仅纹理属性不同还有不同的二维形状,上层嵌套在下层之内.故而合理的手段是采用分层合成,具体的他们先借用文献[79]中的优化思想合成新的层的形状边界,然后在新边界内合成纹理,最后将新层按原顺序叠压组成新的纹理.

近年来随着图形硬件性能的大幅提升,过程纹理作为一种新的纹理形式受到研究者的关注.Han等人^[80]提出一种合成超大分辨率纹理的方法,他们将一组不同细节层次的样本联成一个样本图,当用户到不同细节层次时就使用不同的样本合成结果.他们为样本图中每一个样本附上一个Gauss Stack用以实现不同层次间的平滑过渡.

Lefebvre等人^[81]则专注于基于样本的大规模虚拟环境如数字城市等的纹理的生成和渲染,从少量的样本中切割出可重复的和不可重复的部分,在根据目标纹理的尺寸进行优化拼贴组装得到结果.该重组过程最终化归为一个图上最短路径遍历问题.其方法不仅能使用较为节省的GPU显存来渲染高品质的城市纹理,同时还允许用户将选中的细节部分合成到指定的位置.

3.2.2 向量场生成

向量场在控制曲面的纹理和细节方面扮演了关键的角色.现存的方法可以在平面和二维流形曲面生成向量场.文献[71]使用径向基函数插值选定约束点处的向量约束,这些约束点一般是较分散而且稀疏的.而文献[69]的方法是首先通过网格简化生成多层次的金字塔结构,然后通过在该层次级别中反复的上采样和下采样来得到网格曲面上的纹理定向和尺度.文献[68]也使用了类似的方法,所不同的是他们直接在切空间中插值采样向量.Fisher等人^[16]使用离散外微分工具^[82]来直接计算切向量场.Zhang等人^[17]中使用测地极映射和平行传输将特定点上的切向量约束光滑的插值到网格曲面上的其余区域而直接得到光滑的切向量场,其计算格式还允许用户把各种不同类型的奇异点置放到指定的位置.Chen等人^[83]中开发了一个虚拟街道和城市建模的框架,该框架通过在平面域上生成一个光滑的张量场,然后将该张量场中的拓扑结构映射为街道与城市.

3.2.3 基于特征的形状感知

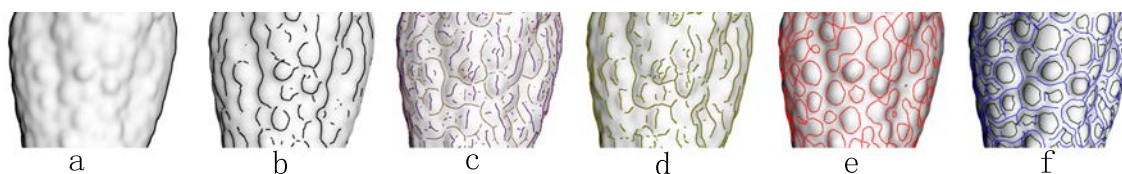


图 3.2 网格曲面存在各类特征线: (a)外轮廓线(*exterior silhouette*); (b)有被遮挡的轮廓线(*occluding contours*); (c)脊线(紫色)和谷线(黄色)(*ridges and valleys*)^[2]; (d)显著脊线(*apparent ridges*)^[3]; (e)平均曲率为零的等值线; (f)分界线(*demarcating curves*)^[15](红色)与谷线(蓝色).

使用曲面上的特征线来反应和表达模型的三维形状是一种惯用的做法.首先

是Hertzmann等人^[84]提出了一种通过检测模型的轮廓线,然后利用几何对偶性将这些轮廓线分割为具有常可见性分段的方法.他们的方法能实时的检测出所有的轮廓线以及位于轮廓线两侧的影线来对模型进行一种线条艺术的风格化渲染.文献[1]中的方法则没有走使用轮廓线的老套路,他们开发了一种新类型的提示轮廓(Suggestive contour),这种提示轮廓能够将模型中明显可见的部分表现出来,而光使用轮廓线将忽略被轮廓圈所包围的那些部分的形状.Ohtake等人^[2]则通过拟合隐式曲面的方法计算出网格上的脊线和谷线(图3.2(c)),由于这些线是通过曲面的一阶和二阶曲率导数定义的,因此它们是于视角和尺度无关的曲面的内蕴特征.Judd等人^[3]中认为人眼对形状的观察较容易受光照和阴影变化的影响(图3.2(d)),因此他们利用法向定义了与视角相关的曲率并基于这种曲率定义了与视角相关的特征线-显著谷线(Apparent ridges).显著谷线涵盖了并进一步完善了前述3种特征线对曲面形状的表现力.文献[15]中沿用了显著谷线的思想提出了另外一种与视角相关的特征线-分界线(图3.2(f)),这种线定义为沿着曲率的梯度方向的法曲率为零的点的轨迹曲线.由于这种线能反映曲面上凸出和凹陷的部分的边界,故而能用于提取并强调展示文物残片模型上的突出细节.Pottmann等人^[4]则提出了等照度量(Isophotic metric)以及等照度线,在这种度量下曲面上曲线的长度不仅依赖于曲率还和沿着该曲线的法向的变化率有关.Cole等人^[85]则从艺术的角度对特征线进行了研究,并引导了一个对艺术创作者在什么位置画线以及被描画的三维场景的几何,视角以及光照特点的调查分析.另一方面,纹理信息能较大的影响人对三维形状的感知,Gorla等人^[86]调查了当各向异性的纹理样本的走向和曲面主曲率方向的弯曲方向相容时,人利用纹理的走向来感知三维形状的准确性问题.与我们的工作较为相关的是文献[87]中提出的一种为连续流体生成沿着特征排布的纹理的方法,但是他们的方法没有考虑如何使三维几何特征和二维纹理特征精确的对齐.

3.2.4 其他相关工作

与给网格模型贴纹理相关的另一个研究领域是网格四边形化.文献[88-90]分别提出了各自的四边形化算法,他们生成的四边形网格都能与两个正交方向场对齐.Dong等人^[91]提出了一种基于谱分析的网格重四边形化算法,他们通过梯度流将Laplacian特征谱函数的极值点连接起来得到四边形网格.

Orzan等人^[8]提出了一种基于特征线的图像恢复和创建方法. 他们使用一组特征线以及定义在这些线上的颜色, 再通过Poisson方法^[27]将颜色光滑插值到整个图像域. 本章的研究工作受到了这个思路的启发.

3.3 曲线导引的曲面向量场生成

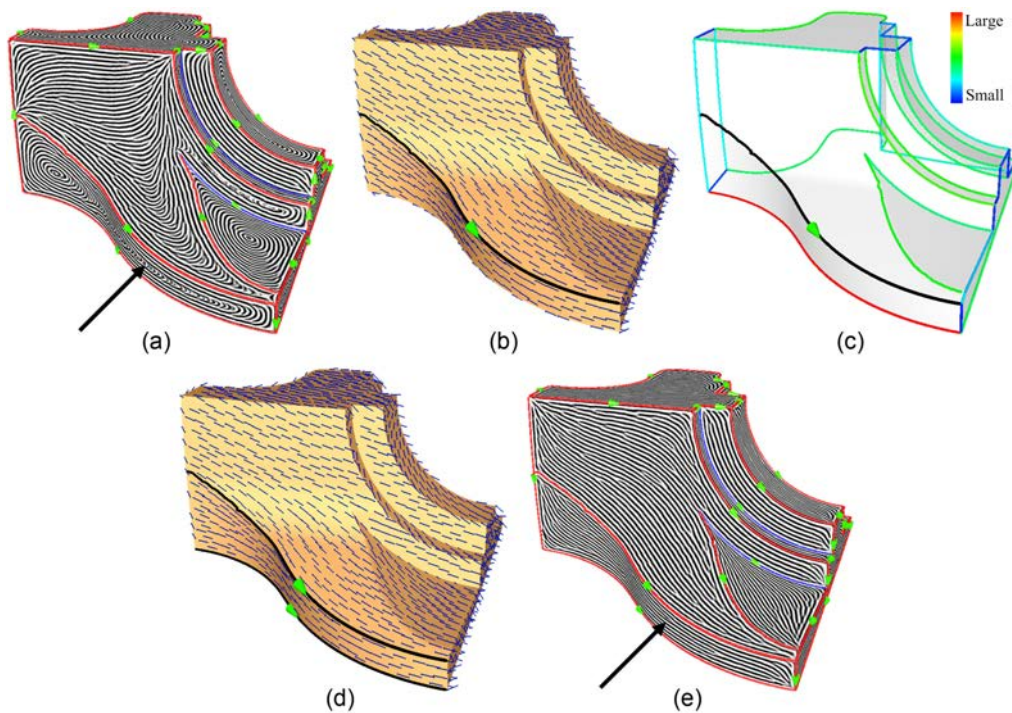


图 3.3 特征线的定向优化提高了曲面向量场的质量: (a)采用任意定向(曲线上绿色箭头)的特征线来引导生成调和扩散的向量场常包含许多奇异点和弯折扭曲(黑色箭头); (b)我们的迭代式曲面定向算法, 每次仅使用当前已被定向的曲线(黑色曲线)作为约束更新向量场; (c)然后计算新的向量场和未定向曲线的相容性度量(按右上标度显示为曲线的颜色); (d)然后选择相容性度量最高的一根曲线((c)中红色曲线)为之定向; (e)最终迭代结果, 向量场更光滑且包含较少的奇异点.

在曲面上纹理合成一般需要一个向量场来决定纹理的局部定向. 为了获得连续的, 特征对齐的纹理, 我们希望得到一个和特征线的走向基本保持一致的光滑向量场. 有许多方法可以计算在用户提供的方向约束下的光滑向量场, 例如文献[16, 71]. 使用特征线作为约束带来的一个主要问题是特征线是不定向的, 因此沿

着特征线我们可以定义两个方向的切向量. 如果随意指定任意定向的切向量做为约束, 得到的向量场可能存在过多的奇异点和复杂的拓扑结构. 如图3.3(a)所示.

我们将在下一节提出一个决定曲面上未定向的特征线的切向量的方向的算法, 该算法希望极小化由这些切向量约束形成的向量场的复杂度和奇异点数量. 我们的算法的一个特点就是它可以与任何一种具体的向量场生成方法相结合, 并对其得到的向量场进行优化. 在我们的实现中, 我们采用了一种的扩散方法来从给定的约束中得到调和的导引场. 该方法速度较快, 其算出的向量场将作为我们下一步进行纹理合成所用.

3.3.1 网格上特征线的定向优化

给定三维表面上的 N 条曲线, 每一条线都是由一系列两两连接的顶点组成, 我们希望上面的每一个顶点分配一个切向量, 以使得用这些切向量作为约束得到的向量场尽可能的光滑. 这个问题本质上是一个组合优化的问题, 该问题的可行解共有个 2^N , 一个直接解法是在可行解集中穷举搜索到一个最优解极小化曲面上向量场的某种光滑性度量. 然而, 一个全局的向量场光滑性度量是难于定义的. 我们通过实验发现, 通过一种贪婪算法来优化一个局部的增量式的度量不但能快速求解, 而且能得到自动得到光滑的向量场.

我们的算法增量式地一次只为一条曲线给定方向. 注意这里我们以曲线而不是组成曲线的顶点为考虑的最小单位, 因为后者得到的优化结果将有可能使得同一条曲线上发生切向量定向的反转. 当相邻顶点上的切向量由于噪声或离散化的缘故而变化较大时, 定向的反转就更容易出现. 这是我们所不希望的. 由于现有的网格上特征线的提取算法得到的结果存在一定的噪音和瑕疵, 我们在预处理中先沿着一条光滑而明显的特征区域将那些相邻但是断开的曲线段连接起来, 然后将那些过长的曲线在它们的高曲率部分打断. 这里的连接和打断操作是分别根据用户提供的距离和曲率上的阈值进行的.

图3.3演示了我们的增量式算法流程. 作为起点, 我们首先选择一条特征线然后为之指定一个任意的方向. 如图3.3(b)所示我们选择最长的一条, 因为在一般的情况下, 曲面上最长的那根特征线对最终合成的成果有最强的影响. 因此我们先固定这跟曲线的方向. 以之为约束我们能计算出一个光滑的向量场, 如图3.3(b)中的蓝色箭

头所示. 之后我们进行迭代优化. 在每一步优化中, 我们选择一条还未被定向的曲线加入被定向的曲线集合, 并为之定向. 被选择的曲线以及它的定向应该使得扩展的曲线集合生成的新向量场与原向量场之间的扭曲度最小. 具体的, 我们为每一条未定向的曲线计算一个数值(如图3.3(c)中的颜色所示). 拥有最大数值的曲线将被定向然后加入集合, 更新后得到的向量场如图3.3(d)所示. 迭代重复进行直到所有的曲线都被定向, 如图3.3(e)所示. 后面我们将会看到, 只要在迭代过程中的中间向量场都足够光滑, 我们的方法能最终得到一个插值所有特征线定向的光滑向量场.

我们的增量式算法的关键在于定义一个扭曲度的度量. 该度量可以告诉我们一个插值定向曲线集合 Θ 的切向量的向量场来说, 当我们向 Θ 中添加一条定向曲线会使向量场增加多少的扭曲, 或者说将在多大程度上破坏原向量场的光滑程度. 我们定义的度量是基于一个简单的考量: 设原有的向量场为 \mathbf{u} , 场中顶点 p 处的向量值为 \mathbf{u}_p . 现在我们在顶点 p 处添加一个约束向量 v_p , 设得到的新向量场为 \mathbf{u}' . 如果 $v_p = \mathbf{u}_p$, 那么当然有 \mathbf{u} 全等于 \mathbf{u}' . 反之, 为了在 p 处插值约束向量 v_p , \mathbf{u}' 将包括在 p 处产生的一些扰动和弯曲, 假如 v_p 和 \mathbf{u}_p 间的角度越大, 所引入的弯曲也越严重.

基于上述推理, 我们为一条未定向的曲线相对于给定向量场定义一个扭曲值用以反映这条曲线和向量场的整体的相容度. 特别的, 给定向量场 \mathbf{u} , 我们定义一条曲线 c 关于 \mathbf{u} 的相容度为:

$$g(c, \mathbf{u}) = \max \left\{ \sum_{p \in V(c)} \frac{\pi - \theta_p}{\pi + \theta_p}, \sum_{p \in V(c)} \frac{\theta_p}{2\pi - \theta_p} \right\} \quad (3.1)$$

这里的 $\theta_p \in [0, \pi]$ 是曲线 c 在 p 处的切向量和 \mathbf{u}_p 之间的夹角, 求和子式的分母用于将 $g(c, \mathbf{u})$ 规范化到 $[0, 1]$. 实际上, 可以证明3.1式的定义是关于 c 的定向不变的. 因为如果曲线 c 的定向反转, 那么夹角 θ_p 将变成 $\pi - \theta_p$, 即将3.1式中的 θ_p 替换为 $\pi - \theta_p$, 易知变形后与原式恒等. 取最大值的操作能保证 c 的两种定向中和 \mathbf{u} 最为相容的一种定向被用来计算相容度. 注意这里我们使用了顶点累加的方式来计算相容度, 以便优先选择那些既相容于向量场又较长的线.

算法局限性. 我们的自动曲线定向算法对多数三维模型, 包括本章中的所有模型, 都能得到较好的结果. 然而, 由于我们的算法是一种增量式的贪婪算法, 因此在某些情形下会得到不够满意的结果. 一个反例是: 在约束特征线是从一个中心呈辐射状

发出的一系列直线的情形中, 我们的算法将试图为中心对称的两条直线分配相同的定向; 但是由于这种情形具有全局的对称性, 因此其最优的定向结果应该是中心对称的两条直线有相反的定向, 且相邻的两条直线的定向不发生反转; 由此算出的向量场将以辐射中心为唯一的奇异点, 且精确的插值每一条直线的方向约束. 目前我们的算法还没有对包括对称在内的全局信息加以考虑, 我们有兴趣在后续工作中研究.

3.3.2 受约束的向量场生成算法

上一节中我们提到我们的定向算法可以和任何生成向量场的算法结合起来使用, 例如文献[16, 17]中提出的算法. 因为我们的方法是一种迭代的方法, 因此我们希望能较快的生成并更新向量场以得到最后的结果. 所以我们希望能找到一种既快速同时又能得到和其他方法质量一样的向量场的算法.

利用调和函数的光滑性和插值性来生成带约束的向量场是一种经典的向量场生成算法[8, 92]. 我们为三维向量的三个分量 x, y, z 分别计算三个独立的调和标量场, 计算时它们都受到同样的约束向量的对应分量的约束. 这里我们使用了文献[18]中提出的惩罚方法(Penalty Method). 该方法能迅速的计算出向量场并根据约束向量的变化迅速的更新向量场. 如同文献[71], 我们将最后得到的向量场投影到网格的切平面上为随后的纹理合成做好准备.

在三角形网格曲面上求解调和方程等价于求解待一个带约束的Laplacian线性系统. 具体的, 对于网格 $G = \{V, E\}$. 设 \mathbf{x} 是 G 上未知向量场的 x 分量(y, z 分量相同的对待), 设 \mathbf{b} 是约束的向量场的 x 分量. 我们通过极小化下面的能量函数来计算 \mathbf{x} :

$$\mathbf{x} = \arg \min_{\hat{\mathbf{x}}} \left\{ \frac{1}{2} \hat{\mathbf{x}}^T \mathbf{L} \hat{\mathbf{x}} + \|\mathbf{P}^{1/2}(\hat{\mathbf{x}} - \mathbf{b})\|^2 \right\} \quad (3.2)$$

这里, \mathbf{L} 是基于余切权因子[93, 94]的Laplacian矩阵, 由网格顶点的相对位置与连接关系共同决定:

$$\mathbf{L} = \mathbf{D} - \mathbf{W}$$

其中矩阵 \mathbf{W} 的分量 $\mathbf{W}_{ij} = \begin{cases} \frac{1}{2}(\cot \alpha_{ij} + \cot \beta_{ij}), & \{i, j\} \in E \\ 0, & \text{otherwise} \end{cases}$, $\{i, j\}$ 表示顶点 v_i, v_j 之间的边, α_{ij} 和 β_{ij} 是以这条边为对边的两个夹角. 矩阵 \mathbf{D} 是一个对角阵, 其对角线

上元素 D_i 等于 \mathbf{W} 第 i 行元素的和. 对角矩阵 \mathbf{P} 是一个罚函数矩阵, 用于指定约束的逼近程度. 当且仅当在顶点 i 处存在向量约束时, $P_{ii} = \alpha \neq 0$. 本章中我们设定, $\alpha = 1.0 * 10^8$, 以惩罚约束点处的插值误差.

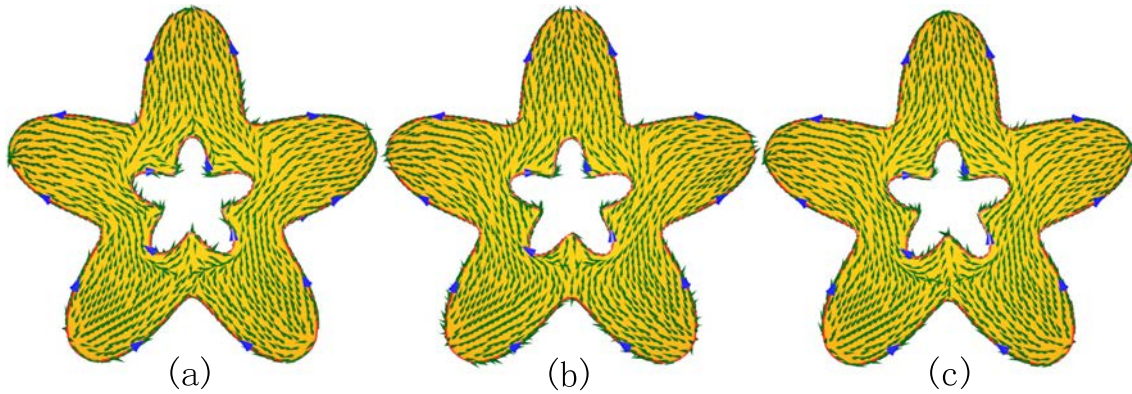


图 3.4 将我们的曲线定向算法和3种向量场生成算法相结合进行特征线定向优化得到的光滑向量场: (a)文献[16]中的算法; (b)文献[17]中的算法; (c)文献[18]中的算法.

极小化能量函数3.2等价于求解线性系统 $(\mathbf{L} + \mathbf{P})\mathbf{x} = \mathbf{P}\mathbf{b}$. 注意到我们实际上是把约束关系乘以一个权因子然后直接附加在 \mathbf{L} 的对应未知数的方程上, 所以当约束的个数和位置发生变化时, 系数矩阵 $\mathbf{L} + \mathbf{P}$ 和 \mathbf{L} 有相同的结构(即相同的非零元个数和位置). 由于 $\mathbf{L} + \mathbf{P}$ 的对称正定性我们采用Cholesky分解求解. 为了防止每次重新求解向量场, 我们对上一次的结果进行更新. 我们利用一种快速超节点法^[95, 96]来对上一轮的Cholesky分解结果进行更新. 注意到该快速超节点法要求对每一次更新时对系数矩阵的改变矩阵是一个低阶矩阵. 由于每一次被加入的曲线上的顶点个数相对于整个网格顶点数是比较小的, 因此我们的曲线定向算法中的每一步都能满足这个要求.

如图3.4所示, 在约束相同的情况下, 我们的向量场生成算法能得到和文献[16, 17]中展示的质量几乎相同的向量场. 前面提到我们的曲线定向算法可以和任意的向量场生成算法相结合. 从图3.4可以看到这里的3种方法通过我们的曲线定向方法将得到同样的定向. 另一方面, 我们的方法效率较高. 对于图3.4中的例子, 3种算法的完成时间分别是0.33秒、0.67秒和超过11分钟. 需要注意, 后面两种算法是直接得到切于平面的向量, 而我们的算法需要将最后得到的向量投影到网格的切平面上.

3.4 特征对齐的纹理优化

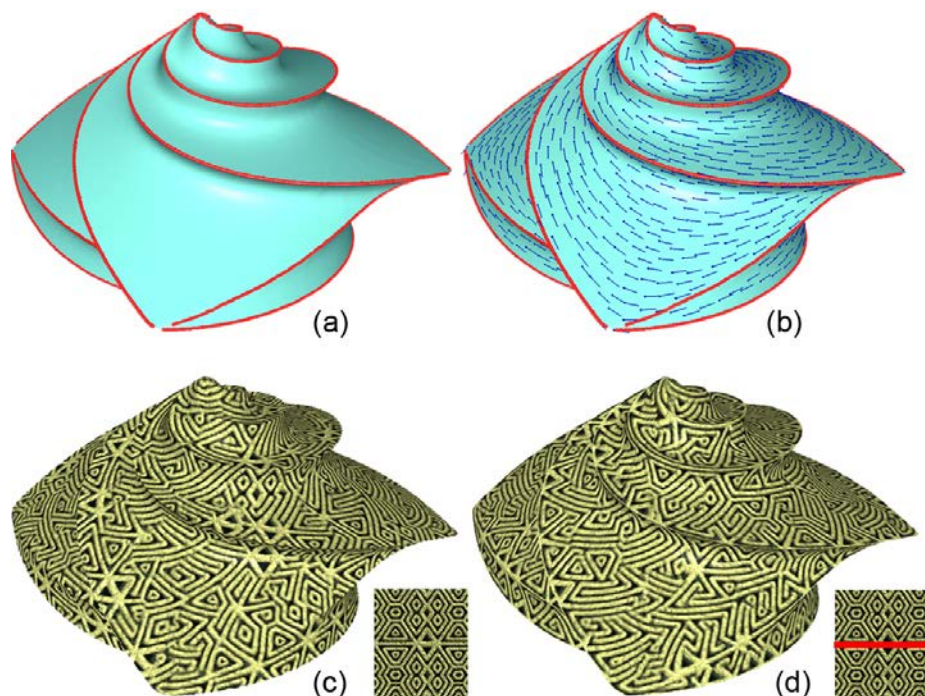


图 3.5 我们的纹理合成算法流程: (a)输入模型及其表面的特征线(红色曲线); (b)由特征线插值出的光滑向量场; (c)传统的曲面纹理合成结果; (d)如果纹理样本中存在显著的特征(红色直线段), 我们的算法使得纹理二维特征和曲面三维特征精确对齐.

在生成了和曲面特征线相容的光滑向量场后, 我们接下来进行纹理合成. 近来, 对向量场引导的纹理合成技术的研究提出了基于优化的纹理合成框架, 包括二维^[79]和三维^[97, 98]等. 我们观察到虽然这些无约束的优化方法能得到和曲面的整体形状大致相贴和的结果, 如图3.5(c)所示. 但是当使用的纹理样本中包含有明显的特征线的时候, 比如图3.6(a)所显示的那样, 如果我们能将纹理样本中的特征和曲面的几何特征精确的对齐, 则可以明显的增强模型的外观效果, 如图3.5(d)所示.

在优化框架上我们将纹理优化方法^[79]和外观空间表达方法^[99]进行结合, 并在其实现细节上进行了若干改进. 因此我们的方法是一种基于块的整体优化方法. 我们的算法可分别为三个阶段: 纹理样本分析、网格预处理和纹理优化. 接下来, 我们首先简单介绍我们的纹理样本分析与网格预处理过程. 然后引入我们使用的纹理优化方法, 最后介绍我们的特征对齐的纹理合成技术.

3.4.1 样本分析和网格预处理

在样本分析阶段, 我们首先将纹理样本逐个像素的转换到其外观空间. 为了使外观空间表达能更好以非局部的方式抓住样本的色彩特征, 像素的外观空间向量的维数一般要高于RGB的三维颜色表达(在本章的所有例子中我们使用5维外观空间). 这使得样本中的每个块叠加得到的外观向量维数过高, 以至于在随后纹理优化的匹配搜索中误差较高. 为了解决此问题, 我们用块中若干代表性像素来代替所有像素来得到一个较低维数的外观向量. 在此较低维的空间中, 熟知的近似最近邻域搜索算法如ANN能更好的发挥作用. 为了在之后的优化阶段进一步加快搜索, 我们借鉴文献[79]的方法, 对所有的块的外观向量进行K-means聚类, 然后将得到的聚类中心作为搜索的候选集. 搜索时, 先确定最近似候选集, 然后在候选集内搜索最相似的纹理块.

和文献[97]一样, 在网格预处理阶段我们先在网格曲面上随机而均匀的选择一些顶点. 以这些顶点为中心我们框选一些邻域栅格块. 为了保证这些块能完全覆盖整个曲面, 我们设定中心点与最近邻点之间的间距不超过栅格块变长的1/4. 为了进行均匀的纹理合成, 我们沿着向量场的积分线正交且等间距的取出栅格块覆盖整个网格. 我们把这些块保存起来, 在随后的特征对齐纹理优化中使用. 注意对于一个样本和一个模型, 样本分析和网格预处理只需执行一次.

3.4.2 网格曲面上的纹理优化

为了生成与样本外观相似而且无缝的曲面纹理, 我们对合成结果进行纹理优化. 给定一个纹理样本, 文献[79, 98]中提出的方法旨在最小化目标纹理和样本纹理中最相似的块之间的匹配误差. 形式上, 我们令输入样本为 Z , 我们要得到的曲面上的目标纹理为 X . 令 \mathbf{x} 为将 X 上像素RGB分量叠加排列得到的向量. 对 X 中一个像素 p , 我们可得到一个以 p 为中心宽度为 w 的领域像素栅格块 N_p . 记 N_p 上所有像素RGB分量叠加排列得到的向量为 \mathbf{x}_p . 则在样本 Z 中能找到一个与 N_p 外观最相似的一个像素领域块, 记其对应的向量为 \mathbf{z}_p . 我们定全局纹理能量函数为:

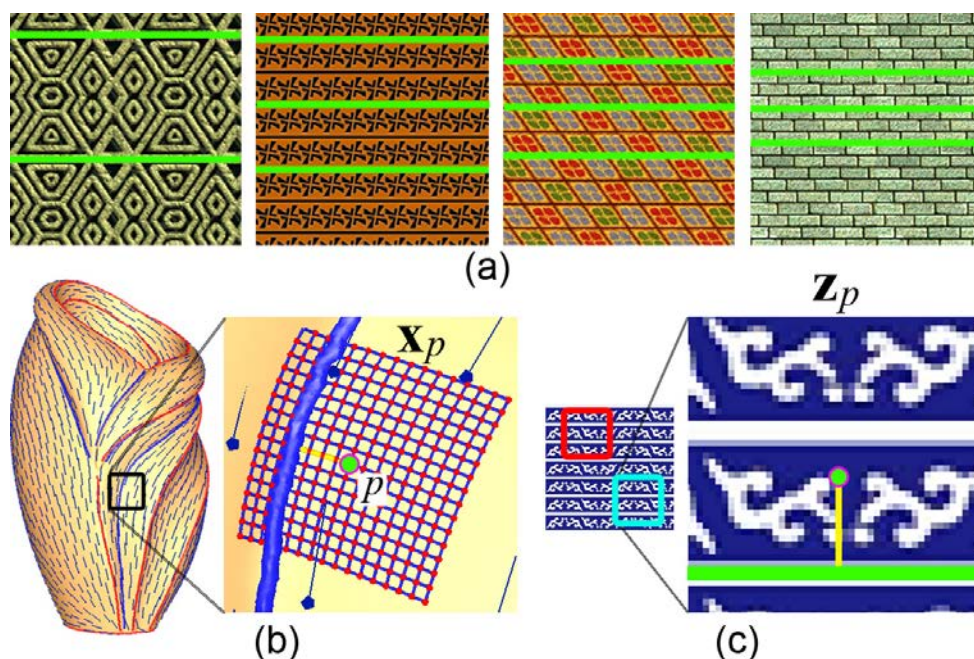


图 3.6 特征对齐的纹理优化: (a)包含明显的特征线的纹理样本; (b)被曲面特征线穿越的网格采样栅格块, 其中心点为 p ; (c)我们的算法在优化时只选择样本中那些也被特征线所穿越的邻域栅格块(蓝色方框).

$$E_t(\mathbf{x}; \{\mathbf{z}_p\}) = \sum_{p \in X^\dagger} \lambda_p \omega_p \|\mathbf{x}_p - \mathbf{z}_p\|^2 \quad (3.3)$$

注意这里我们使用欧式范数度量向量 \mathbf{z}_p 和 \mathbf{x}_p 之间的差异. 这里的 $X^\dagger \subset X$ 是目标像素点的一个子集. 这意味着该能量只考虑了 X 中的一部分邻域块. 实验证明, 只要 X^\dagger 对应的这部分邻域块相互间有足够的重叠部分, 并且能形成 X 的一个覆盖就能很好的定义 X 上的全局纹理能量. 在实现中, 我们选取 X^\dagger 为稀疏均匀的轴对齐的栅格顶点, 栅格间距为邻域块边长的 $1/4$. 这样的配置使得对于一个给定的像素, 同时影响该像素的只有 X 中相邻的若干个邻域块; 如果同时覆盖一个像素的邻域块过多, 则容易因为块与块之间的不匹配而合成出较为模糊的像素.

类似于文献[79], 我们使用基于期望最大化(EM: Expectation-Maximization)的方法对3.3式的能量进行极小化. 在所求解的问题上, 期望最大化算法和K-means算法有个共同特征: 即对于一个能量函数, 我们不仅需要找出使之最优化的变量, 同时该能量函数中的参数也是不确定的. 换言之, 我们是在一个泛函空间中寻找一个

最优解，而不仅仅是该空间中某一个指定的函数的定义域上来求取该函数的极小值. 在纹理合成问题中，变量对应的是目标纹理 \mathbf{x} ，参数对应的是样本中最相似邻域块集合 \mathbf{z}_p . 由于搜索空间从一个维度扩大到了两个或者多个维度，EM问题需要使用E步骤和M步骤交替迭代的非线性方法来进行优化. 为了启动算法我们首先用随机颜色填充 X . 在E步骤中，我们在选定 \mathbf{z}_p 的前提下求得使能量3.3为极小的 \mathbf{x} ，注意此时3.3式是以 \mathbf{x} 为变量的二次函数，故可线性求解；结合我们取栅格块的策略可知，最后 \mathbf{x} 中 p 点的最优像素为 \mathbf{z}_p 中覆盖 p 的所有最相似邻域块中对应像素的平均值. 为了避免这些块中存在过大的外观差异而得到不合理的像素值，3.3式中的权因子 ω_p 被设定成 \mathbf{z} 和 \mathbf{x} 差异的函数： $\omega_p = \|\mathbf{x}_p - \mathbf{z}_p\|^{-0.6}$. 由于我们在E步骤中改变了 X 上的每个块中的像素值，因此在接下来的在M步骤中，需要在 Z 中重新寻找最相似邻域块集合 \mathbf{z}_p ；M步骤可以看作一个重整理排列的过程，它通过匹配发现上一次E步骤得到的与周围领域不相容的块并将其替换为更合适的块. 简而言之，E步骤负责合成新像素而M步骤则负责找匹配. E和M两步骤交替迭代进行，直到M步骤使得 \mathbf{z}_p 不再发生变化时迭代停止.

3.4.3 特征对齐约束优化

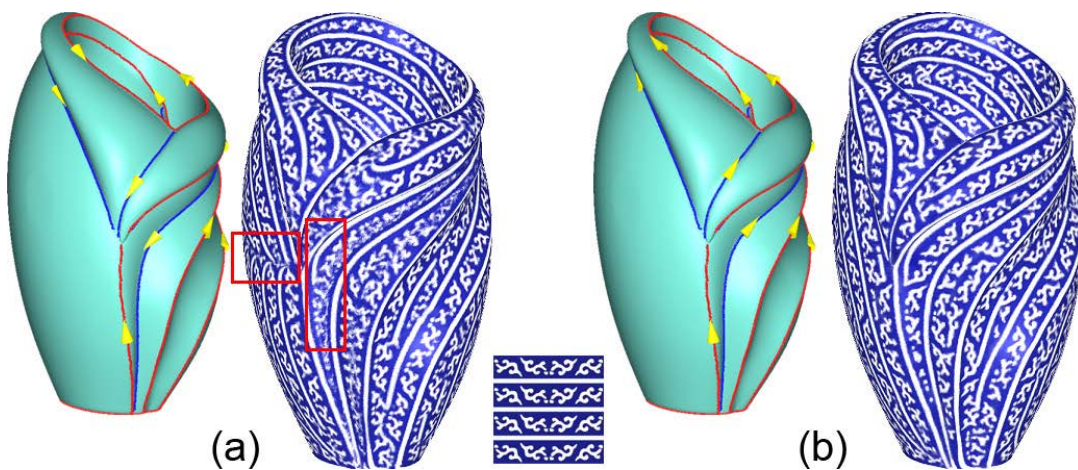


图 3.7 特征线任意的定向(a)与优化的定向(b)在与特征对齐的纹理优化后的合成结果，注意红色方框内标准的奇异点.

下面我们通过对纹理优化过程添加一个约束来控制合成过程使得纹理样本中的特征和曲面的特征能够精确的对齐. 给定一个纹理样本，我们提取出样本中的最显

著的分段线性特征, 这里我们使用文献[100]中提出的启发式算法, 该算法允许用户勾画式的交互. 如果有多条特征线被检测出, 我们从中选择出一条主特征线作为对齐的目标. 样本特征线如图3.6(a)所示.

由于M步骤确定了将要写入 X 中的纹理块, 而E步骤只是对这些纹理块的叠加部分进行加权平均操作. 因此每一个迭代周期合成出的纹理的外观基本上是由M步骤所决定的. 因此, 我们通过修改M步骤来解决特征对齐问题. 在M步骤找匹配时, 我们约束那些中心点位于曲面特征线附近的块只能在 Z 中的同样位于样本特征线附近的块中找寻外观最相似的块. 即我们约束集合 X^\dagger 中那些位于曲面特征线附近的点 $p \in X^\dagger$ 的纹理匹配过程. 特别的, 对曲面上所有的特征曲线集合 C , 我们定义点 p 到线集合 C 的最近测地距离为 $d(p, C)$. 如果 $d(p, C) < \omega/2$, 则说明 C 中至少有一条特征线穿过或有部分位于当前的栅格块 \mathbf{x}_p 所覆盖的曲面区域. 对于这样的 \mathbf{x}_p , 我们限制其搜索最相似邻域块 \mathbf{z}_p 的候选集为那些中心点离纹理特征线距离与 $d(p, C)$ 的差小于一个用户给定的阈值的样本纹理块. 该约束意味着即使 Z 中如果存在比样本特征线附近的块更加相似的块, 也无法被贴入 X 中, 这样能严格的保证特征对齐效果不受到保持外观尽可能相似的趋势的破坏. 例如, 对于图3.6(b)中的曲面栅格块, 只有图3.6(c)中蓝色外框内的样本块会被考虑为其候选集, 而红色外框内的样本将不被考虑. 为了进一步保证特征对齐的精度, 我们定义能量(2)中的权因子为:

$$\lambda_p = 1/(d(p, C) + \varepsilon)$$

其中我们设定 $\varepsilon = 1.0 * 10^{-6}$ 以保证上式分母 $(d(p, C) + \varepsilon)$ 不等于0. 引入 λ_p 可以保证靠近曲面特征线的纹理只能产生较小的匹配误差.

3.5 实验结果和讨论

我们通过在一系列三维模型表面进行合成来展示我们的特征对齐纹理效果. 在本章的所有例子中, 我们使用文献[101]中的算法和代码来快速且鲁棒的提取出模型上的特征线. 给定一个模型, 模型上的特征线, 和一个用户指定的纹理样本, 我们的算法将自动的完成纹理合成. 图3.7中显示了在花瓶模型上使用任意曲线定向图3.7(a)和用我们的特征线定向优化算法图3.7(b)优化后的特征线生成的向量场引导得到的纹理合成效果的对比. 可以看到图3.7(b)中纹理的扭曲和奇异点明显少于

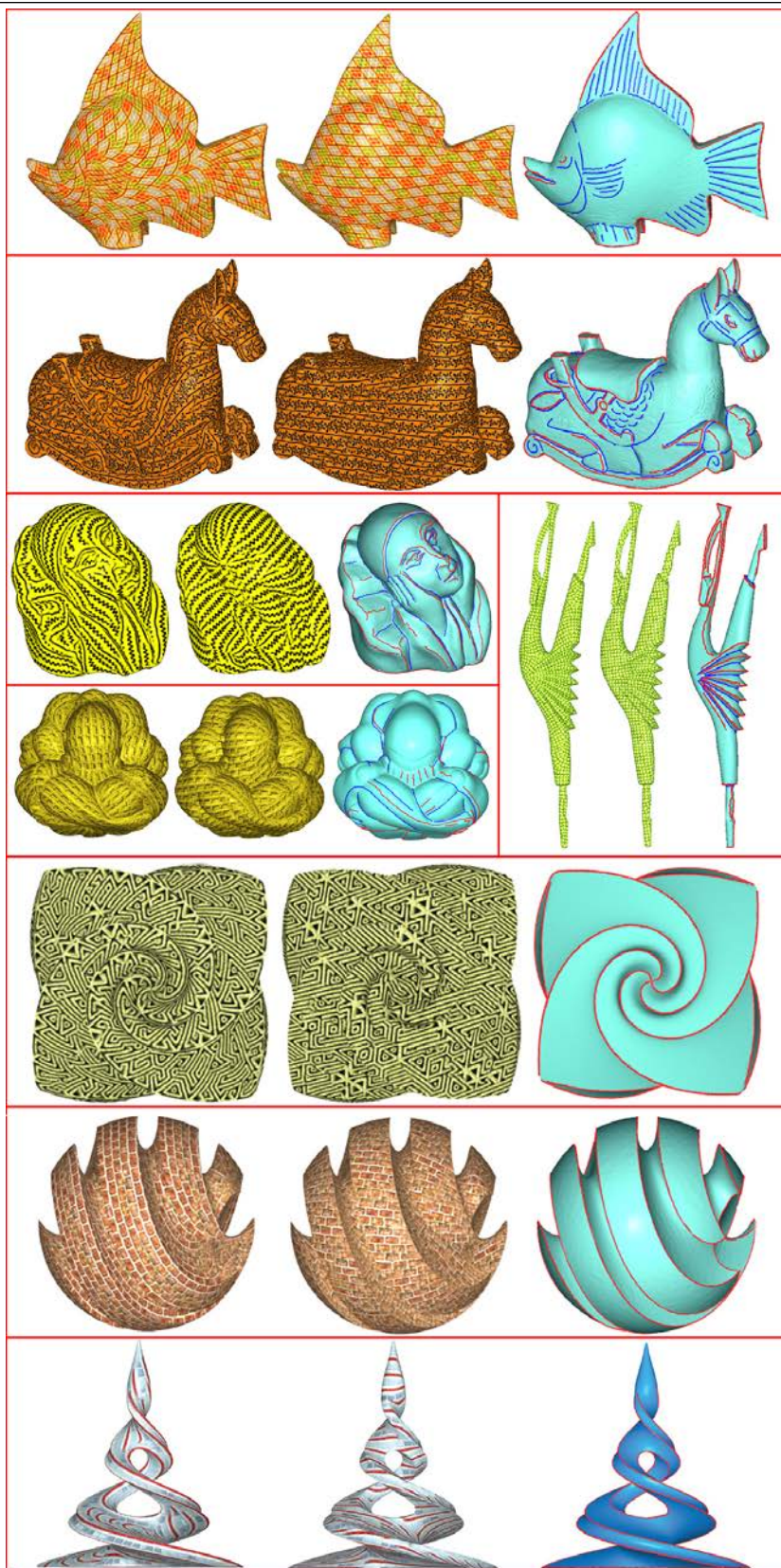


图 3.8 特征对齐的纹理合成结果, 在每个三元组中: (左)我们方法的合成结果; (中)没有特征对齐的合成结果; (右)输入模型及其特征线(蓝色是谷线, 红色是脊线).

图3.7(a). 在图3.3的CAD模型上的合成结果中我们也能看到同样明显的改进效果. 由于特征曲线集合 C 的拓扑特征, 即使我们为每根特征线都指定了合适的定向, 也很难完全消除奇异点. 注意到我们的定向优化算法是一种增量式的方法, 它能够通过优化特征线的定向使奇异点移动到两条或多条特征线汇合的角落处, 如图3.3(e)所示; 将图3.3(e)和图3.3(a)比较我们不难发现, 相比于那些较为平坦的区域, 让奇异点隐藏在这些角落处对视觉效果的影响是比较小的.

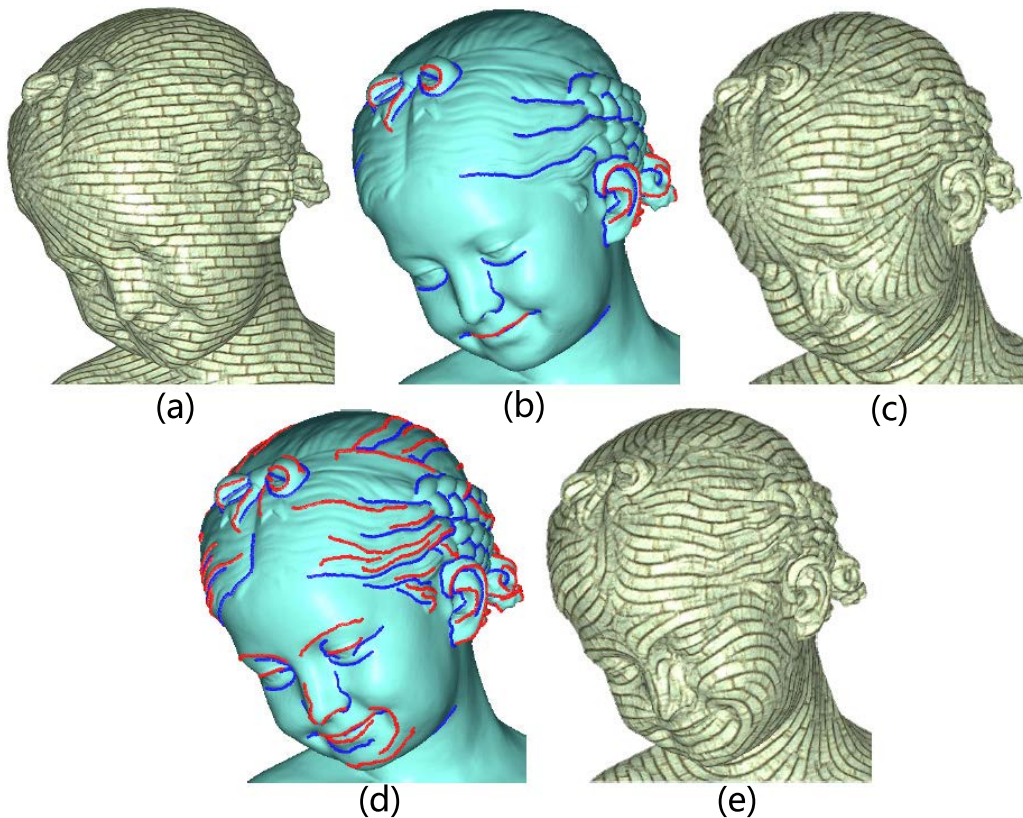


图 3.9 我们的方法使用不同的特征线集合可以生成多分辨率层次的效果: (a)不使用特征对齐得到的合成结果; (c)使用较低特征分辨率对应的特征曲线(b)的合成结果, 注意鼻和嘴的部分细节清晰,但是眼睛处被奇异点遮盖了原有的细节; (e)使用较高特征分辨率对应的特征曲线(d)得到合成结果, 与(c)的结果比较眼睛处细节得到了特征对齐纹理准确的装饰.

为了进一步展示特征对齐的纹理对于展示曲面形状的增强效果, 我们在图3.8中显示了一系列合成结果图. 对每个模型, 我们对比了使用和不使用特征对齐的纹理合成效果. 从这些结果可以看出, 通过对三维曲面上特征线的凸显和强调, 特征对

齐的纹理可以更好装饰模型的几何外观;另一方面,输入的纹理样本借助我们的特征对齐合成算法,也能更好的将自身的二维外观特征与三维模型结合.在图3.9中,我们展示了在提取了多个层次的特征线的人头部模型上的特征对齐合成效果.从图3.9(c)和(e)中可以看到,使用较低分辨率的特征线集合对模型的嘴部和使用较高分辨率的特征线集合对模型的眼睛和鼻梁部分的外观装饰效果的差别.而这种差别在没有使用特征对齐的纹理合成中是难于察觉的.

算法性能. 我们的纹理合成系统运行在2.5GHZ的Intel Core2 Duo PC上,系统内存为2GB.由于使用了快速的向量场生成算法,我们的系统在40万三角片的模型上只需要不到1分钟就能算出最终的优化特征线定向以及对应的向量场,其中每一次特征线定向更新耗时约为1秒.合成阶段耗时较长,在同等规模的模型上约需要20分钟.如文献[98]中所演示的,通过采取多分辨率的合成格式,我们的合成算法性能应该能得到一定提升.

3.6 本章小结

本章我们讨论了特征转移建模方法的另一种形式:将二维纹理特征转移到三维曲面进行物体形状的外观建模.我们观察到一个三维物体的形状的精华大都和其上的特征曲线密切相关,如果纹理能和这些特征线保持精确的对齐,那么二维纹理特征能很好的凸显并装饰三维形状特征.我们通过提出一种自动优化曲面特征线定向算法,以及一种将特征线匹配约束加入纹理优化的基于样本的纹理合成算法实现了我们所希望的外观建模效果.

本章研究的后续工作主要包括三个方面.在特征线定向算法上,如果能将一种考虑向量场全局结构协调性度量的定义结合进来,我们认为能提出一种更优秀的定向算法;在纹理合成方面,如前所述采用多分辨率网格的合成格式,或者在GPU上进行实现,可以为我们的系统带来性能上的提升;在人对形状的感知方面,针对二维纹理特征具体是如何增强人眼对三维形状的辨识度,以及纹理本身的颜色和结构信息如何装饰三维物体的外观美学展开调查分析是具有研究意义的.

第4章 从流形到高度场的特征转移:实时数字浮雕建模*

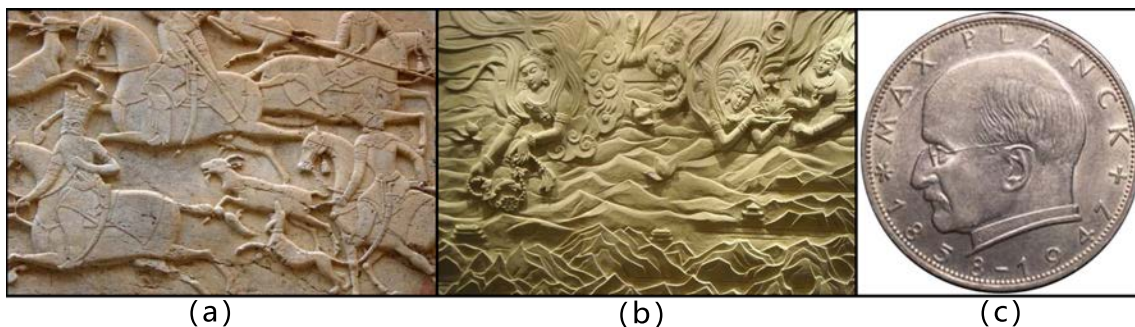


图 4.1 浮雕是一种常见的实体雕塑艺术形态: (a)亚述人狩猎图浮雕; (b)中国敦煌飞天浮雕; (c)钱币浮雕.

4.1 问题介绍

上一章我们介绍了特征转移方法对三维曲面纹理效果的提升. 在本章中, 我们继续沿着该思路来探索计算机图形学中的一个较为新兴的建模技术: 浮雕建模.

浮雕造型艺术是雕塑与绘画结合的实体艺术形态, 其特点是用压缩的办法来处理对象, 靠透视等因素来表现三维空间, 并只供一面或两面观看, 如图4.1所示. 它主要不是靠实体三维来营造空间效果, 而更多地利用绘画中的描绘手法例如透视错觉等来造成抽象的压缩空间. 该艺术形式源远流长, 早在美索不达米亚的古亚述人以及很擅长于浮雕创作; 在一系列的“亚述人狩猎图”中, 他们很好地运用浅浮雕手法, 富有节奏感和韵律感地表现出充满生气的人物和动物场景(图4.1(a)). 浅浮雕一般是附属在另一平面上的, 由于其压缩的特性, 所占空间较小, 钱币器物上也经常可以看到(图4.1(c)). 虽然它仅占据非常有限的空间范围, 但是浅浮雕造像能巧妙的展现出原三维场景的基本布局和视觉内容, 因此浅浮雕本身就是视觉与几何特征在不同维度间转移的一个范例.

近年来, 这门古老而成熟的艺术形式被引入了计算机图形学的数字几何处理领

*本章内容发表在:实时数字浮雕建模.计算机辅助设计与图形学学报, 2010, 22(3):434-439.

域.按照成品厚度的不同,浮雕可分为两类基本形态:高浮雕和浅浮雕.本章工作专注于浅浮雕建模.浅浮雕起位较低,形体压缩较大,平面感较强,更大程度地接近于绘画形式.传统的浅浮雕制作工艺完全依赖艺术家的空间想象,制作过程一般比较繁琐而漫长,而现在借助计算机和数控雕刻机床^[102],一般兴趣爱好者也可轻易地进行创作.Weyrich等人^[19]提出了一种基于三维场景生成数字浮雕的方法:即对一个模型,提取其观察视图的高度场,并对这个高度场进行梯度域压缩获取浅浮雕.由于最终归结为线性系统的求解,目前的数字浮雕算法普遍需要一个初始化的过程,即在生成浮雕前必须先分解一个大型稀疏线性矩阵;在用户每一次旋转视角,或是改变场景的尺寸和形状后,必须再进行一次求解,以获取这个视点的浮雕.在这样的框架下,创作人员无法实时地以任意角度进行浮雕建模,同时也不能在第一时间调整参数控制浮雕的建模效果.

为了解决这个问题,我们提出了利用GPU加速实时获取数字浮雕的算法.该算法以获取最大的并行度为目的,将整个运算过程放在GPU和其附带的帧缓存内进行,获得实时速度.我们通过设计一个高细节压缩函数,获得了较以前方法细节更明显的浮雕.实时交互上则采用了DtB(Drop to Best)的技术,该技术类似于Adobe Photoshop[®]中的双倍像素技术(Pixel Doubling),即在平移和缩放图片的过程中暂时性的使用下采样的低分辨率图片,等操作完成后自动回复原清晰度.在DtB技术的帮助下,我们的系统能使用户能更流畅地调整建模角度.在某些角度,模型生成的浮雕可能带有瑕疵.为此我们设计了多个参数调整界面,使得用户能对这些不满意的局部进行修改.整个修改过程也是实时地进行.我们的算法在较大规模,如1000*1000点的浮雕尺寸,也能进行实时建模,从而能方便获取更为细腻的浮雕效果.

4.2 相关工作

浮雕建模的过程应该使得浮雕尽量保持原有模型的细节和外观.模型高度场的动态压缩方法是其中关键的技术.因此浮雕建模的研究受到了高动态范围(HDR:High dynamic range)图像的压缩技术的启发.Fattal等人^[103]提出了在图像梯度域利用对数衰减函数来进行色彩强度缩减的思想,对数压缩的特点是对越大的值

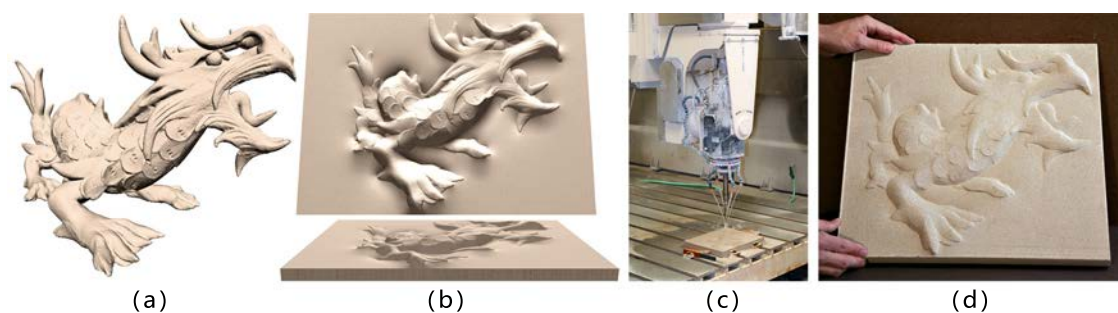


图 4.2 从三维模型生成浅浮雕模型^[19]: (a)输入模型; (b)对输入模型在当前视角下的高度场进行动态范围压缩我们得到浅浮雕模型(下方为侧视图,可以看到其厚度较小); (c)电脑数控机床依据数字浅浮雕模型在石灰石平板上进行加工; (d)加工完成的浅浮雕实体.

衰减越快,而较小的值则相对得到保护^[104]. 同样地,将这样的函数用于网格高度场压缩,则能在光滑的压缩外形的同时维持细节,文献^[103]中即采用了这种对数函数族来压缩高度场的梯度. Song^[105]等人的算法也使用了梯度域形状表示,不同的是他们结合了网格显著特征(Mesh Saliency)^[106]定义和用于处理网格表面信号的非尖锐遮罩(Unsharp-masking)^[107]技术来度量网格的视觉特征量,然后重建出保持原网格视觉特点和几何细节的浅浮雕模型.

另外,近年来GPU通用计算(GPGPU:General Purpose on GPU)的研究得到开展. 一些基于GPU的线性代数和偏微分方程的数值解法被实现. 如Krüger等人^[108]实现了基于GPU求解大型稀疏线性方程组的高斯-赛德耳迭代法和共轭梯度法; Cuntz等人^[109]在GPU上利用粒子水平集方法可视化三维流场; Bolz等人^[110]则实现了稀疏非结构化矩阵的共轭梯度法和多重网格法. Zhou^[111]等人在GPU上实现了W型多重网格方法并将它嵌入实时流体模拟,提高了模拟速度. 另外在纹理数据管理方面,目前由NVIDIA[®]公司推出的计算设施统一架构(CUDA:Compute Unified Device Architecture)技术能自动管理显存内数据,如Zhou等人^[111]利用CUDA来生成KD-Tree并将之用于实时全局光照的计算. 注意我们的系统并没有采用CUDA,而是直接使用了OpenGL的帧缓存扩展来完成纹理数据管理.

4.3 浅浮雕建模原理

目前提出的技术中，浅浮雕的数字化建模框架一般采用梯度域非线性压缩的曲面重建方法^[19]。其基本思想是采用高动态范围压缩原网格模型在当前视角下展现出的不同尺度的细节，来生成依附于一平面上的高度场模型，这个流程如图4.3所示。现将其算法做如下描述。

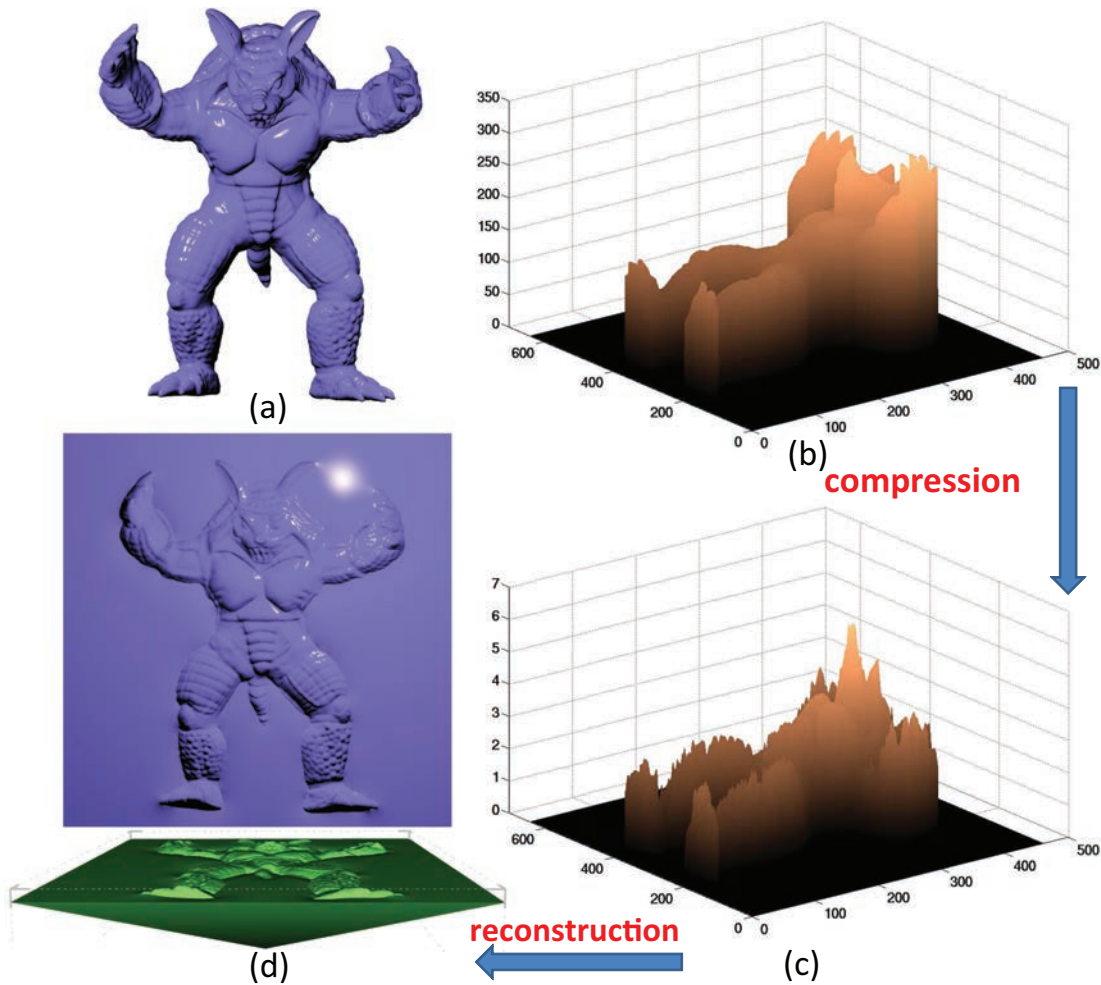


图 4.3 浅浮雕建模算法流程: (a)输入模型; (b)在当前视角下的高度场; (c)对高度场进行高动态范围压缩; (d)使用我们的GPU建模系统得到的浅浮雕模型(下图为平视图).

给定一三维场景(可以是单个模型或若干个的组合)以及一个视角(一般是选择一个能观察到该场景大部分视觉内容且构图优美的视角), 我们可得到对应此视角的

带有深度信息的二维视图,其深度信息是由一张坐落在屏幕坐标域 Π (即浮雕附着的零高度平面)上的高度场曲面所定义的(图4.3(b)),记为 $h(x, y)$.该高度场的梯度场为:

$$\mathbf{g}(x, y) = \nabla h(x, y)$$

将每一点的梯度写成梯度方向和模长的乘积形式:

$$\mathbf{g}(x, y) = \|\mathbf{g}\| \mathbf{d}(x, y)$$

在压缩时我们保持梯度方向 $\mathbf{d}(x, y)$ 不变,只在模长 $\|\mathbf{g}\|$ 上压缩:

$$\mathbf{g}'(x, y) = \|\mathbf{g}'\| \mathbf{d}(x, y) = C(\|\mathbf{g}\|) \mathbf{d}(x, y).$$

压缩函数 C 是至关重要的,它直接影响浮雕的轮廓线,局部细节和浮雕的厚度(图4.3(c)).较好的一种压缩函数是采用对数压缩^[103, 104],文献[19]中采用的压缩函数是:

$$C(x; c) = 1/c \ln(1 + cx) \quad (4.1)$$

其中参数 c 可用来控制压缩程度,越小的 c 压缩率越小,浮雕的细节也越突出.然而我们通过实验发现,当减小至 $c = 0.2$ 后细节增强已不明显.故本章以此参数下的结果作为比照对象.

得到了压缩后的梯度场之后,就可以对其积分来重建新的高度场.重建步骤等价于求解一个Poisson方程^[26, 27]:

$$\Delta r(x, y) = \nabla \cdot \mathbf{g}' \quad (4.2)$$

$r(x, y)$ 即新生成的浮雕高度场(图4.3(d)).该方程的解等价于使 $r(x, y)$ 的梯度和压缩后梯度 \mathbf{g}' 相等的变分最小问题(4.3式)的解:

$$r(x, y) = \arg \min_h \iint \|\nabla h(x, y) - \mathbf{g}'(x, y)\|^2 dx dy \quad (4.3)$$

所以其解能最大程度保持压缩后模型曲面的几何特点.如果选取保持细节的压缩函数,则生成的浮雕也能很好的维持原模型视觉特征和局部细节.

4.4 高细节动态范围压缩函数

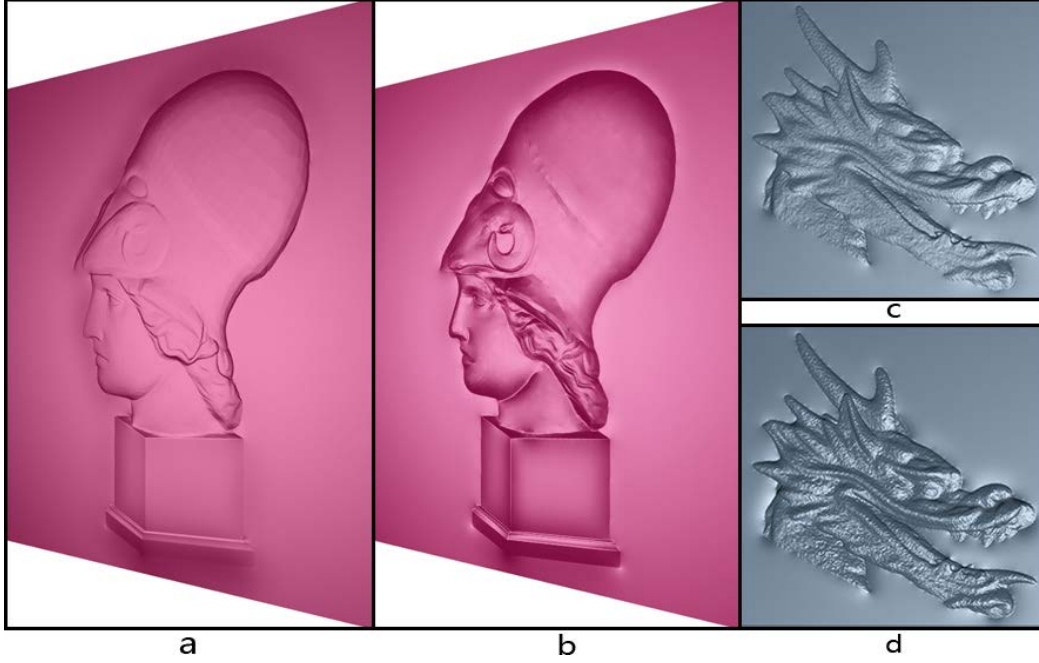


图 4.4 (a)(c)文献[19]方法的结果 $c = 0.2$; (b)(d)我们方法的结果, (b)的参数为 $\alpha = 1.7, \beta = 4.24$; (d)的参数为 $\alpha = 1.03, \beta = 3.8$.

梯度压缩函数对浮雕的厚度和细节起着决定作用. 我们研究发现, 一个保存细节同时减少厚度的压缩函数 $f(x)$ 应满足以下条件:

$$\begin{cases} f(x) \geq 0, f'(x) > 0, f''(x) < 0 & \forall x \geq 0 \\ f(x_1)/x_1 > f(x_2)/x_2 & \text{if } x_2 > x_1. \end{cases} \quad (4.4)$$

即非负区间, $f(x)$ 是非负的单调递增的上凸函数; 同时 $f(x)$ 还应动态压缩各个区间段. 这里我们对满足4.4式的函数 $f(x)$ 定义一个压缩率为:

$$R_{cpr}(f(x)) = f(x)/x, \forall x > 0. \quad (4.5)$$

经过仔细分析, 我们选择分段函数:

$$a(x; \alpha, \beta) = \begin{cases} \alpha \tan^{-1}(\beta x), & 0 \leq x \leq \varphi_{thred} \\ 0, & \text{else.} \end{cases} \quad (4.6)$$

作为我们浮雕高度场的压缩函数. 现对4.1式和4.6式表示的两族函数的压缩特性进行分析和比较. 我们可以证明,当参数 $\alpha\beta > 1$ 时两类函数在正实数区间存在唯一的交点 $x = i$ (如图4.5中黑色方点); 且交点左(右)侧 $a(x)$ 的压缩率小(大)于对数压缩, 即:

$$\begin{cases} R_{cpr}(a(x)) \leq R_{cpr}(C(x)) & , x \in (0, i] \\ R_{cpr}(a(x)) > R_{cpr}(C(x)) & , else. \end{cases} \quad (4.7)$$

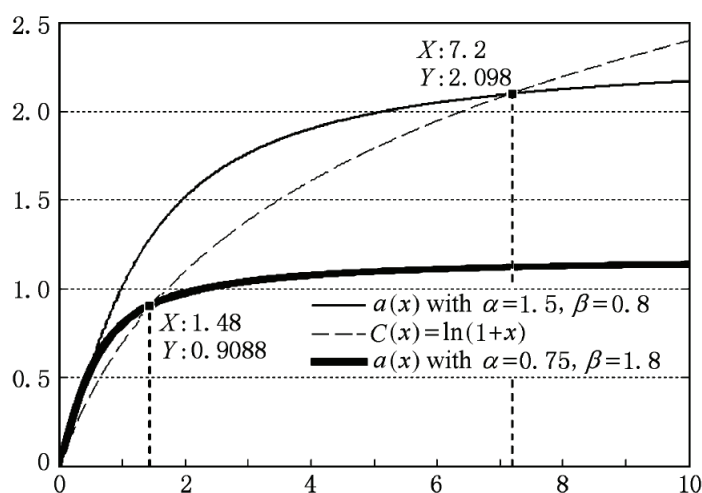


图 4.5 函数 $a(x)$ (4.6式)和 $\ln(1+x)$ 的函数曲线比照.

定理 4.1: 当参数 $\alpha\beta > 1$ 时两类函数4.1和4.6在正实数区间存在唯一的交点.

证明 4.1: 函数4.1和4.6的导函数分别是:

$$a'(x) = \alpha\beta / (1 + \beta^2 x^2)$$

$$C'(x) = 1 / (1 + cx)$$

有
$$a'(x) - C'(x) = -(\beta^2 x^2 - c\alpha\beta x - \alpha\beta + 1) / ((1 + \beta^2 x^2)(1 + cx))$$

记
$$d(x; \alpha, \beta, c) = \beta^2 x^2 - c\alpha\beta x - \alpha\beta + 1 \quad (4.8)$$

4.8式是 x 的二次函数, 由 $\beta^2 > 0$ 及 $c\alpha\beta/\beta^2 > 0$ 知方程 $d(x) = 0$ 必有正实根 x_d ; 再由假定 $\alpha\beta > 1$, 有 $d(0) = 1 - \alpha\beta < 0$, 所以当 $x \in (0, x_d)$ 时, $d(x) > 0$, 故:

$$a'(x) - C'(x) > 0, x \in (0, x_d) \quad (4.9)$$

又 $a(0) = C(0) = 0, a'(x) > 0, C'(x) > 0$, 故 $\exists x_1 \in (0, x_d)$, 使得

$$a(x_1) > C(x_1) \quad (4.10)$$

考虑到 $\lim_{x \rightarrow +\infty} a(x) = \alpha\pi/2$, $\lim_{x \rightarrow +\infty} C(x) = +\infty$, 故定能取到一个充分大的 $x_2 > x_d > x_1$, 使得

$$a(x_2) < C(x_2) \quad (4.11)$$

由4.10和4.11式, 推出两族函数在区间 $(x_1, x_2)_1$ 上必上有交点, 且显然的, 交点两侧两族函数的关系满足4.7式. 由 $a'(x) > 0, C'(x) > 0$ 知它们在区间 $x \in (0, +\infty)$ 上都单调递增且都无拐点, 因此两族函数在正数区间上有且仅有一个交点. 证毕.

注意该结论对这两族函数是普遍成立的, 并不受各自参数选取的影响. 该结论保证: 无论 $C(x; c)$ 中的参数 c 如何取值, 我们的函数通过调节 α, β 都可以到达浮雕细节比之更强的效果. 图4.5给出了我们的函数在 $\alpha = 1.5, \beta = 0.8$ 时的图像(图中较细的实线), $\alpha = 0.75, \beta = 1.8$ 时的图像(图中较粗的实线)与函数 $\ln(1+x)$ (图中虚线)的图像对比. 为图示清晰, 这里我们仅以 $C(x) = \ln(1+x)$ 为函数族 $C(x; c)$ 的代表. 可清楚的看到交点(图中黑色方点)将我们的函数划分为低压缩率段(交点至原点部分)和高压缩率段(交点以右部分). 这说明, 函数 $a(x)$ 在有效压缩梯度的同时能更好地保存并强调细节. 我们在图4.4中对比了采用不同压缩函数所建浮雕结果. 图4.4各分图中结果所用参数分别是: a, c幅: $c = 0.2$; b幅: $\alpha = 1.7, \beta = 4.24$; d幅: $\alpha = 1.03, \beta = 3.8$.

4.5 基于GPU实时建模

在上一节我们引入的高度场重建算法的核心是求解一个Poisson方程, 若以差分格式离散偏微分算子我们将得到一个大型稀疏线性系统; 传统方法采用矩阵分解法求解, 则需要一个初始化的过程. 该过程耗时与底座 Π 的尺寸相关, 我们的系统运行在Intel E7200 2.4Ghz CPU的平台上, GPU为Nvidia[®]公司的GTX9400. 我们在实现中

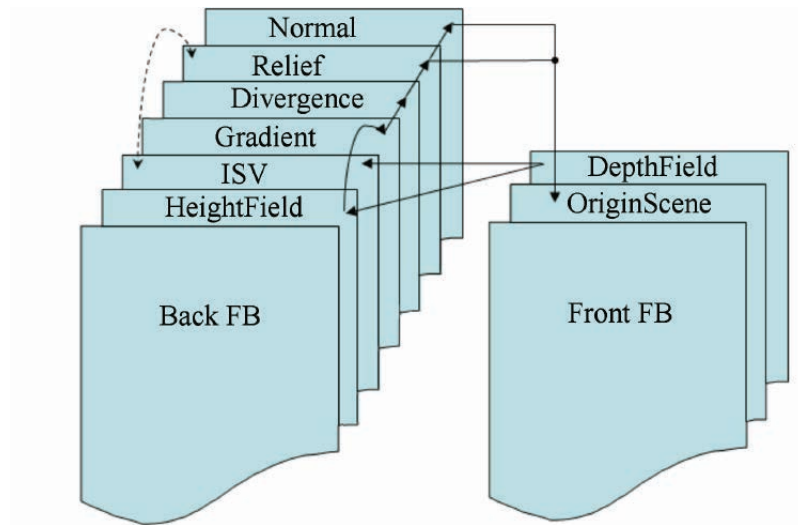


图 4.6 GPU算法流程与数据纹理布局.

发现生成尺寸为 100×100 到 800×800 的浮雕需时一般在10秒到1分钟. 值得注意的是, 我们的系统允许用户任意的旋转模型选择一个最合适的视角. 但是若改变 π 的形状或者缩放其尺寸, 都将改变输入的高度场, 因此需要再进行一次计算纹理的初始化.

为了达到实时建模, 我们在GPU上进行Jacobi迭代来求解4.2式, 也可使用加权的Jacobi迭代^[108]或多重网格法^[112]. 在GPU上每一次迭代归结为运行五次纹理采样函数, 效率非常高, 因此可在短时间内运行许多次. 迭代至收敛的次数与 π 的尺寸相关, 如表1所示. 使用Jacobi方法对迭代初值(ISV: Iterative starting value)的选取是十分敏感的. 我们实验发现若以原始高度场为ISV, 则无法收敛; 而采用 $h * 0.01$ 可迅速收敛. 原因是浅浮雕的厚度较小, 因此与估计值 $h * 0.01$ 是接近的.

表 1 实时浮雕建模速度

浮雕尺寸 /像素	平均迭代次数	平均帧率/帧/s	
		不使用 DtB	使用 DtB
1000 × 1000	650	10	18
500 × 500	180	17	32

我们的系统采用GPU帧缓存扩展(Framebuffer extension)^[108]的多重纹理计算技术, 在后台帧缓存BackFB中完成浮雕建模全部的计算工作并在前缓存FrontFB中

实时显示,如图4.6所示.值得注意的是我们的系统目前是使用OpenGL正交投影来渲染场景.从绘制场景中提取深度后,我们采用多目标渲染(MTR: Multi-targets rendering)的技术,利用一个shader同时计算出HeightField和ISV,保存到两幅纹理.然后继续利用MTR技术和多个shader依次计算压缩梯度和压缩散度准备迭代(虚线箭头).为了进一步加速,我们还用一个shader计算新生成的浮雕的法向,这样能再缩短每一帧浮雕的绘制用时.注意我们将后台数据纹理全存放在后台帧缓存BackFB中以最大程度地避免帧缓存切换耗时.

4.6 交互方式与细节控制

我们的系统为用户提供两类实时交互:场景视角旋转和浮雕细节参数调整.在用户实时交互阶段,我们采用DtB的技术来进一步提高系统运行的流畅程度(表1).具体是在用户改变场景视角时,系统自动降低迭代次数;用户松开鼠标交互确定之后,立刻增加迭代次数获取最精确解.这样用户在一次交互期内能流畅作业,并随时获得最佳浮雕结果.

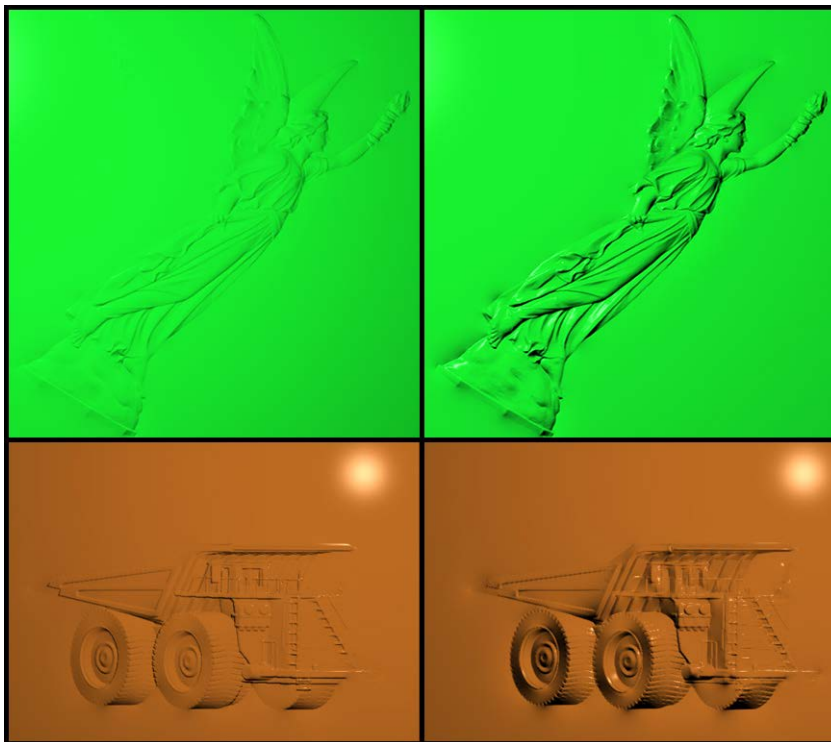


图 4.7 通过 β 调整浮雕细节(从左至右为增大 β).

我们的系统支持实时调整各种参数. 用户可以通过调节函数4.6中的两个实参数 α, β 来控制浮雕的厚薄与细节的强弱. 其中 α 为前置系数. 将4.6式带入4.2式可得:

$$\Delta r(x, y) = \nabla \cdot g' = \alpha(\nabla \cdot \tan^{-1}(\beta||g||)d(x, y)) \quad (4.12)$$

由4.12式知 α 可以线性地缩放4.2式的解, 因此 α 可以对浮雕的整体厚度进行线性地控制. 我们在实验中发现, 若生成边长为25cm的正方形浮雕, 取 $\alpha \in [1.0, 3.8]$ 可以获得最大处厚度在0.26cm到1.0cm之间的浅浮雕. 图4.8给出了取不同 α 时的浮雕厚度效果.

在实验中我们发现, 增大 α 和 β 都能增强浮雕上细节效果的突出程度. 但是如前所述 α 的增大会使浮雕厚度增加, 选取太大的 α 会使得生成的浮雕厚度过大, 有违浅浮雕的特点. 而通过调节 β 来控制细节的凸出程度则能避免该问题. 图4.7给出了调节 β 对浮雕细节程度的控制效果(α 维持不变). 因此在我们的建模系统中, 鼓励用户以调节 β 为主, 调节 α 为辅的方式来控制浮雕细节.

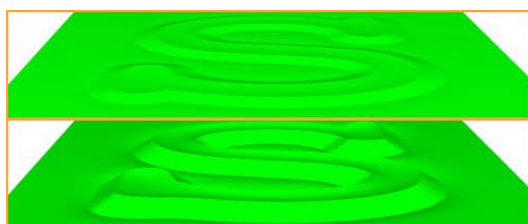


图 4.8 通过 α 调整浮雕的整体厚度(从上至下为增大 α).

此外, 在模型高度场中存在某些高度落差较大但小于 φ_{thred} 的区域, 在浮雕中会形成不连续的竖直凸出效果(以往方法^[19]的结果中也有出现), 这是由于选取了不恰当的 φ_{thred} 而无法正确识别浮雕边界的缘故. 本系统中用户可以通过实时调节 φ_{thred} 消除这种现象而仅轻微地影响其余的部分, 如图4.6所示. 注意图中原本不自然的“高墙”部分(红色标识区域)在调节后得到了平滑.

4.7 本章小结

本章承接在不同维度间进行特征转移的思路, 探索了利用三维模型场景快速生成数字浅浮雕的方法. 我们实现了利用GPU加速技术构建的实时数字浮雕建模系统; 我们还研究了影响建模效果的关键因素, 进而提出一种新的压缩函数以提高浮雕细

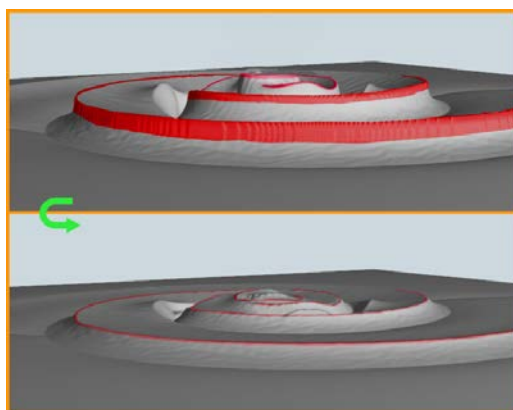


图 4.9 调整阈值 φ_{thred} 以消除瑕疵(从上至下为减小 φ_{thred}).

节度. 实验证明该系统不仅能使艺术创作者从任意三维模型场景中实时快速的制作大尺寸浅浮雕, 且建模效果好于传统的方法. 图4.10中显示了我们的系统从两个模型中制作尺寸为1000*1000像素的浅浮雕(上), 及它们在该角度建模效果的侧视图(下), 可以看到形成的浮雕具有较薄的厚度却能提供较好的视觉效果. 图4.11显示了系统对同一模型从不同视点进行建模的效果.

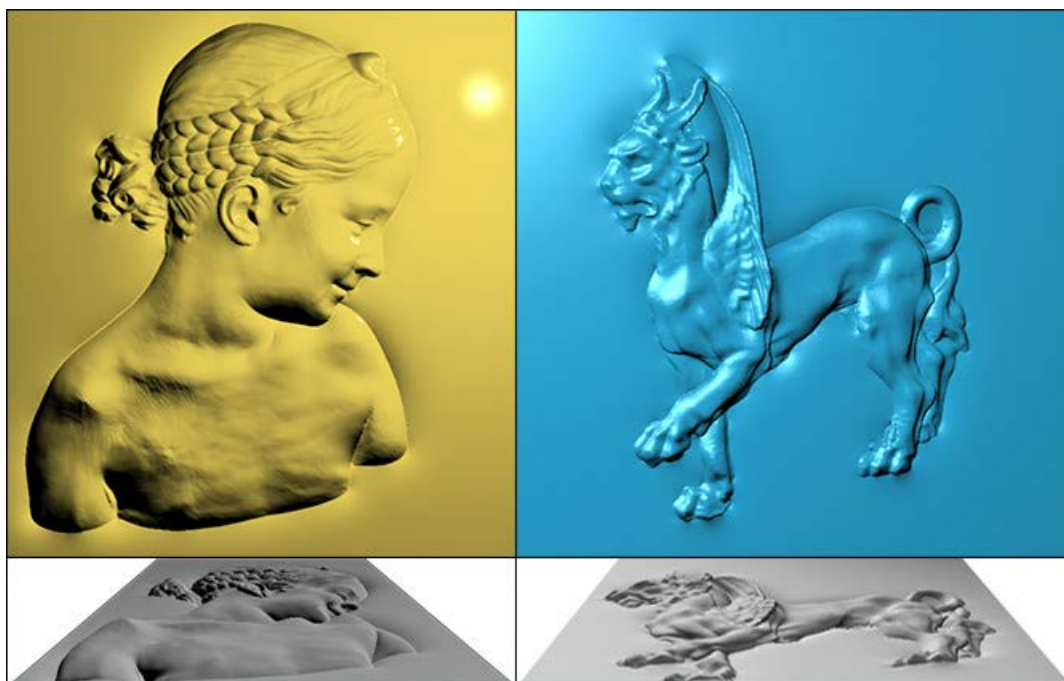


图 4.10 我们的实时浮雕建模系统得到的结果. 在每一列中下图为上图的平视图.

实时浮雕的创作过程实际上是一种浮雕风格的三维动画, 这种动画形式也是以

前没有出现过的. 未来我们将继续该方面的研究及其在浮雕动画和网格绘制方面的应用.



图 4.11 我们的建模系统允许用户从不同的视点实时得到反馈(每一行为同一模型不同视角下的浅浮雕).

第5章 总结与展望

5.1 工作总结

在对表达为不同维度的数据进行分析和处理时,原始信息的特征量往往是一个有效的切入点.计算机图形学发展到现阶段,研究者常常需要和各种不同维度的多媒体数据打交道,而这些数据之间存在紧密的联系.因此,本文的研究工作采用了这样一种方法论:利用数据特征在不同维度间的关联,将问题映射到另一个维度中进行解决.我们结合了数字几何处理中的网格形变技术,纹理合成中的纹理优化技术等一些成熟的算法和数值优化,离散优化等技术,研究了计算机图形学中的三个建模问题:针对图像内容的直观参数式建模,基于纹理合成的三维形状外观建模以及实时的数字浅浮雕建模.我们开发了这三个问题的解决方法,同时证明了进行维度间特征转移这种方法论的广泛适用性和有效性.本文工作取得了以下创新结果:

1. 开发了一套直观易用的人体重塑系统,用户只需要拖动一组对应于体型语义参数的滑杆就能轻松的对单张图片中的人物体型进行实时的重塑.我们的第一个贡献在于采用了一种新的三维模型驱动的图像变形流程,该流程首先将二维图像的语义特征转移到一个三维人体全身可形变模型上获得人类体型变化的参数式控制,然后将三维模型的几何变化以一种带特殊约束的能量优化方式映射回原图像驱动人物体型发生重塑变形.我们的第二个贡献是提出了一种新的身体感知的图像变形新算法以实现三维到图像的变形映射,该算法能鲁棒的将重塑效果从三维模型忠实的转移到二维图像.我们通过对一系列例如穿有宽松衣物以及带有复杂遮挡的人物图片进行试验证明了我们的参数式体型重塑系统能够产生真实可信的全局协调的体型重塑效果.相比于使用传统的图像编辑软件需要进行繁复的局部编辑的工作方式,我们的参数式体型重塑系统能让用户更直观而轻松的编辑人物体型.
2. 提出了一种特征对齐的曲面纹理合成新方法,该方法解决了使用带有明显特征线的各项异性的纹理样本做曲面纹理合成时,曲面本身的几何特征容易被纹理

特征混淆和遮盖的问题. 我们观察到该问题的原因在于传统的纹理合成方法忽略了纹理特征线在曲面上摆放的位置的重要性. 为此我们引入了一种新的纹理优化方法, 该方法在最大化样本和结果间的全局相似度的同时使得纹理样本的特征线和曲面特征线精确的对齐. 此外, 我们提出了一个给网格曲面的特征线定向的自动算法, 该算法通过极小化向量场的相容性度量得到曲面特征线的优化定向, 使用这样的曲面特征线能明显减少纹理合成结果中包含的奇异点和扭曲.

3. 我们注意到传统的浅数字浮雕建模方法并不提供实时的视角选取和细节程度的调节, 为此我们提出一个实时的建模系统, 该系统利用多目标渲染的纹理运算技术将浅浮雕建模的全过程集成到图形系统上完成, 从而使得用户能实时的选取其偏好的建模视角同时对浅浮雕各方面的细节特征进行微调. 此外我们还引入了一类新的动态范围压缩函数并证明该函数能比传统方法更好的保留输入模型的细节特征.

5.2 后续工作

本文工作主要有以下一些内容值得进一步的研究:

1. 利用三维模型驱动二维图像变形的思想为我们提供了未来的研究方向. 如果能将基于模型的编辑流程从人物图像扩展到其他内容的图像, 比如人造物体图像等, 相信能带来新的实用的参数式全局编辑方法和具有真实感的编辑效果. 此外, 由于目前我们使用的人体数据库主要是西欧地区人群的人体扫描模型, 因此进一步提高人体数据库的包容性和样本复杂度, 使得它能涵盖儿童, 以及亚洲等其他地区人群的扫描数据将能显著的提高模型拟合的准确程度.
2. 在曲面纹理合成方面的后续工作主要包括三个方面. 一是在特征线定向算法上, 如果能将一种考虑向量场全局结构的协调性度量定义结合进来, 我们认为能提出一种更优秀的定向算法; 二是目前的合成优化耗时较长, 如能采用多分辨率网格的合成格式, 或者在GPU上实现栅格采样和优化等步骤, 可以为我们的系统带来性能上的提升; 三是在人眼对于形状的感知方面, 针对二维纹理特征具体是如何增强人眼对三维形状的辨识度, 以及纹理本身的颜色和结构信息如何装饰三维物体的外观美学展开调查分析是具有研究意义的.

参考文献

- [1] DeCarlo D, Finkelstein A, Rusinkiewicz S, Santella A. Suggestive contours for conveying shape[J]. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 2003, 22(3):848–855.
- [2] Ohtake Y, Belyaev A, Seidel H P. Ridge-valley lines on meshes via implicit surface fitting[J]. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 2004, 22(3):609–612.
- [3] Judd T, Durand F, Adelson E H. Apparent ridges for line drawing[J]. *ACM Transactions on Graphics*, 2007, 26(3):19.
- [4] Pottmann H, Steiner T, Hofer M, Haider C, Hanbury A. The isophotic metric and its application to feature sensitive morphology on surfaces[C]// *Computer Vision - ECCV 2004, Part IV*, volume 3024 of *Lecture Notes in Computer Science*. Washington, DC: Springer, 2004: 560–572.
- [5] Nealen A, Igarashi T, Sorkine O, Alexa M. Fibermesh: Designing freeform surfaces with 3d curves[J]. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 2007, 26(3).
- [6] Nealen A, Sorkine O, Alexa M, Cohen-Or D. A sketch-based interface for detail-preserving mesh editing[J]. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 2005, 24(3).
- [7] Bokeloh M, Berner A, Wand M, Seidel H P, Schilling A. Symmetry detection using line features[J]. *Computer Graphics Forum (Proceedings of Eurographics)*, 2009.
- [8] Orzan A, Bousseau A, Winnemoller H, Barla P, Thollot J, Salesin D. Diffusion curves: A vector representation for smooth-shaded images[J]. *ACM Transactions on Graphics*, 2008, 27(3).
- [9] Jeschke S, Cline D, Wonka P. Rendering surface details with diffusion curves[J]. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 2009, 28(5).

- [10] Ramanarayanan G, Bala K, Walter B. Feature-based textures[C]// Proceedings of Eurographics Symposium on Rendering. 2004.
- [11] Parilov E, Zorin D. Real-time rendering of textures with feature curves[J]. ACM Transactions on Graphics, 2008, 27(1).
- [12] Tzur Y, Tal A. FlexiStickers - photogrammetric texture mapping using casual images[J]. ACM Trans. Graph., 2009, 28(3).
- [13] Arad N, Reifeld D. Image warping using few anchor points and radial functions[J]. Computer Graphics Forum, 1995, 14(1):35–46.
- [14] Schaefer S, McPhail T, Warren J. Image deformation using moving least squares[J]. ACM Trans. Graph., 2006, 25(3):533–540.
- [15] Kolomenkin M, Shimshoni I, Tal A. Demarcating curves for shape illustration[J]. ACM Transactions on Graphics (Siggraph Asia 2008), 2008, 27(5).
- [16] Fisher M, Schröder P, Desbrun M, Hoppe H. Design of tangent vector fields[J]. ACM Transactions on Graphics (Proc. SIGGRAPH), 2007, 26(3).
- [17] Zhang E, Mischaikow K, Turk G. Vector field design on surfaces[J]. ACM Transactions on Graphics, 2006, 25(4):1294–1326.
- [18] Xu K, Zhang H, Cohen-Or D, Xiong Y. Dynamic harmonic fields for surface processing[J]. Computers and Graphics (Special Issue of SMI 2009), 2009, 33(3):391–398.
- [19] Weyrich T, Deng J, Barnes C, Rusinkiewicz S, Finkelstein A. Digital bas-relief from 3d scenes[J]. ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH 2002, 2007, 26(3):32–38.
- [20] 胡事民, 杨永亮, 来煜坤. 数字几何处理研究进展[J]. 计算机学报, 2009, 08(5).
- [21] Botsch M, Sorkine O. On linear variational surface deformation methods[J]. IEEE Transactions on Visualization and Computer Graphics, 2008, 14:213–230.

- [22] Lipman Y, Sorkine O, Cohen-Or D, Levin D, Rossl C, Seidel H P. Differential coordinates for interactive mesh editing[C]// International Conference on Shape Modeling and Applications 2004 (SMI'04). 2004: 181–190.
- [23] Gal R, Sorkine O, Mitra N J, Cohen-Or D. iwires: an analyze-and-edit approach to shape manipulation[J]. ACM Transactions on Graphics (Proc. SIGGRAPH), 2009, 28(3).
- [24] Zheng Y, Fu H, Cohen-Or D, Au O K C, Tai. C L. Component-wise controllers for structure-preserving shape manipulation[J]. Computer Graphics Forum (Proc. Eurographics 2011), 2011.
- [25] Sunkel M, Jansen S, M.Wand, E.Eisemann, Seidel H P. Learning line features in 3d geometry[J]. Computer Graphics Forum (Proceedings of Eurographics), 2011.
- [26] Jeschke S, Cline D, Wonka P. A gpu laplacian solver for diffusion curves and poisson image editing[J]. ACM Transactions on Graphics (Proc. SIGGRAPH Asia), 2009, 28(5).
- [27] Perez P, Gangnet M, Blake A. Poisson image editing[J]. ACM Transactions on Graphics (Proc. SIGGRAPH), 2003, 22(3).
- [28] Wang L, Wang X, Tong X, Lin S, Hu S, Guo B, Shum H Y. View-dependent displacement mapping[J]. ACM Transactions on Graphics (Proc. SIGGRAPH), 2003, 22(3).
- [29] Gu X, Gortler S J, Hoppe H. Geometry images[J]. ACM Transactions on Graphics (Proc. SIGGRAPH), 2002, 21(3).
- [30] Wang Y S, Tai C L, Sorkine O, Lee T Y. Optimized scale-and-stretch for image resizing[J]. ACM Transactions on Graphics (Proc. SIGGRAPH Asia), 2008, 27(5).
- [31] Chen R, Freedman D, Karni Z, Gotsman C, Liu L. Content-aware image resizing by quadratic programming[C]// Proceedings of CVPR Workshop on Non-Rigid Shape Analysis and Deformable Image Alignment. 2010.

- [32] Subr K, Soler C, Durand F. Edge-preserving multiscale image decomposition based on local extrema[J]. *ACM Transactions on Graphics*, 2008, 28(5).
- [33] Eismann K, Palmer W. *Adobe Photoshop Restoration & Retouching*[M]. New Riders Press; 3 edition, 2006.
- [34] Barron C, Kakadiaris I A. Estimating anthropometry and pose from a single uncalibrated image[J]. *Computer Vision and Image Understanding*, 2001, 81(5).
- [35] Barrett W A, Cheney A S. Object-based image editing[J]. *ACM Trans. Graph.*, 2002, 21(3):777–784.
- [36] Igarashi T, Moscovich T, Hughes J F. As-rigid-as-possible shape manipulation[J]. *ACM Trans. Graph.*, 2005, 24(3):1134–1141.
- [37] Shamir A, Sorkine O. Visual media retargeting[C]// *SIGGRAPH Asia Course 2009*. 2009.
- [38] Blanz V, Vetter T. A morphable model for the synthesis of 3d faces[C]// *SIGGRAPH '99*. 1999: 187–194.
- [39] Leyvand T, Cohen-Or D, Dror G, Lischinski D. Data-driven enhancement of facial attractiveness[J]. *ACM Trans. Graph.*, 2008, 27(3):38.
- [40] Allen B, Curless B, Popović Z. The space of human body shapes: reconstruction and parameterization from range scans[J]. *ACM Trans. Graph.*, 2003, 22(3):587–594.
- [41] Seo H, Magnenat-Thalmann N. An example-based approach to human body manipulation[J]. *Graphical Models*, 2004, 66(1):1–23.
- [42] Anguelov D, Srinivasan P, Koller D, Thrun S, Rodgers J, Davis J. SCAPE: shape completion and animation of people[J]. *ACM Trans. Graph.*, 2005, 24(3):408–416.
- [43] Hasler N, Stoll C, Sunkel M, Rosenhahn B, Seidel H P. A statistical model of human pose and body shape[J]. *Computer Graphics Forum*, 2009, 28(2):337–346.
- [44] Hua Z, Wang G, Lin X, Yan H. Recovery of upper body poses in static images based on joints detection[J]. *Pattern Recognition Letters*, 2009, 30(5):503–512.

-
- [45] Taylor C. Reconstruction of articulated objects from point correspondences in a single uncalibrated image[J]. *Computer Vision and Image Understanding*, 2000, 80(3):349–363.
- [46] Davis J, Agrawala M, Chuang E, Popović Z, Salesin D. A sketching interface for articulated figure animation[C]// *SCA '03*. 2003: 320–328.
- [47] Parameswaran V, Chellappa R. View independent human body pose estimation from a single perspective image[C]// *CVPR '04*. vol 2. 2004.
- [48] Hornung A, Dekkers E, Kobbelt L. Character animation from 2D pictures and 3D motion data[J]. *ACM Trans. Graph.*, 2007, 26(1):1.
- [49] de Aguiar E, Stoll C, Theobalt C, Ahmed N, Seidel H P, Thrun S. Performance capture from sparse multi-view video[J]. *ACM Trans. Graph.*, 2008, 27(3):98.
- [50] Vlastic D, Baran I, Matusik W, Popović J. Articulated mesh animation from multi-view silhouettes[J]. *ACM Trans. Graph.*, 2008, 27(3):97.
- [51] Balan A O, Sigal L, Black M J, Davis J E, Haussecker H W. Detailed human shape and pose from images[C]// *CVPR '07*. 2007: 1–8.
- [52] Balan A, Black M. The naked truth: Estimating body shape under clothing[C]// *ECCV '08*. 2008.
- [53] Guan P, Weiss A, Balan A, Black M J. Estimating human shape and pose from a single image[C]// *ICCV '09*. 2009.
- [54] Kraevoy V, Sheffer A, van de Panne M. Modeling from contour drawings[C]// *SBIM '09*. 2009: 37–44.
- [55] Tan G, Chen W, Liu L. Image driven shape deformation using styles[J]. *Journal of Zhejiang University (SCIENCE C)*, 2010, 11(1):27–35.
- [56] Vlastic D, Brand M, Pfister H, Popović J. Face transfer with multilinear models[J]. *ACM Trans. Graph.*, 2005, 24(3):426–433.
- [57] Sumner R W, Popović J. Deformation transfer for triangle meshes[J]. *ACM Trans. Graph.*, 2004, 23(3):399–405.

- [58] Zhou K, Huang J, Snyder J, Liu X, Bao H, Guo B, Shum H Y. Large mesh deformation using the volumetric graph Laplacian[J]. *ACM Trans. Graph.*, 2005, 24(3):496–503.
- [59] Dong W, Paul J C. Adaptive content-aware image resizing[J]. *Computer Graphics Forum (Proc. Eurographics 2009)*, 2009.
- [60] Baran I, Popović J. Automatic rigging and animation of 3D characters[J]. *ACM Trans. Graph.*, 2007, 26(3):72.
- [61] Fu H, Au O K C, Tai C L. Effective derivation of similarity transformations for implicit laplacian mesh editing[J]. *Computer Graphics Forum*, 2007, 26(1):34–45.
- [62] Pletinckx D. Quaternion calculus as a basic tool in computer graphics[J]. *The Visual Computer*, 1989, 5:2–13.
- [63] Li Y, Sun J, Tang C K, Shum H Y. Lazy snapping[J]. *ACM Trans. Graph.*, 2004, 23(3):303–308.
- [64] Mortensen E N, Barrett W A. Intelligent scissors for image composition[C]// *SIGGRAPH '95*. 1995: 191–198.
- [65] L.R R, B.H J. An introduction to hidden markov model[C]// *Proceedings of the IEEE 77*. 1989: 257 - 286.
- [66] 肖春霞, 赵勇, 郑文庭, 彭群生. 三维离散点采样表面基于全局优化的纹理合成算法[J]. *计算机学报*, 2006, 12(5):2061–2070.
- [67] 韩建伟, 王青, 周昆, 鲍虎军. 基于wang tiles的几何纹理合成[J]. *软件学报*, 2009, 12(5):3254–3262.
- [68] Wei L Y, Levoy M. Texture synthesis over arbitrary manifold surfaces[C]// *Proceedings of the 28th annual conference on Computer graphics and interactive techniques (SIGGRAPH'01)*. New York: ACM Press, 2001.
- [69] Turk G. Texture synthesis on surfaces[C]// *Proceedings of the 28th annual conference on Computer graphics and interactive techniques (SIGGRAPH'01)*. New York: ACM Press, 2001.

-
- [70] Wei L Y, Lefebvre S, Kwatra V, Turk G. State of the art in example-based texture synthesis[J]. Eurographics 2009 State-of-the-art Report, 2009.
- [71] Praun E, Finkelstein A, Hoppe H. Lapped textures[J]. ACM Transactions on Graphics, 2000, 19(3):465–470.
- [72] Zelinka S, Garland M. Jump map-based interactive texture synthesis[J]. ACM Transactions on Graphics, 2004, 5(N):1–37.
- [73] Soler C, Cani M P, Angelidis A. Hierarchical pattern mapping[C]// Appolloni T. 29th International Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2002, July, 2002. San Antonio, Texas, Etats-Unis: ACM, 2002: 673–680.
- [74] Nealen A, Alexa M. Hybrid texture synthesis[C]// Christensen P, Cohen-Or D. Proceedings of Eurographics Symposium on Rendering. 2003: 97–105.
- [75] Efros A A, Leung T K. Texture synthesis by non-parametric sampling[J]. Computer Vision, IEEE International Conference on, 1999, 2:1033.
- [76] Wei L Y, Han J, Zhou K, Bao H, Guo B, Shum H Y. Inverse texture synthesis[J]. ACM Transactions on Graphics, 2008, 27(5):1–9.
- [77] Simakov D, Caspi Y, Shechtman E, Irani M. Summarizing visual data using bidirectional similarity[J]. 2008 IEEE Conference on Computer Vision and Pattern Recognition, 2008, 1–8.
- [78] Rosenberger A, Cohen-Or D, Lischinski D. Layered shape synthesis: automatic generation of control maps for non-stationary textures[J]. ACM Transactions on Graphics (Proc. SIGGRAPH), 2009, 28(5).
- [79] Kwatra V, Essa I, Bobick A, Kwatra N. Texture optimization for example-based synthesis[J]. ACM Transactions on Graphics (Proc. SIGGRAPH), 2005, 24(3).
- [80] Han C, Risser E, Ramamoorthi R, Grinspun E. Multiscale texture synthesis[J]. ACM Transactions on Graphics (Proc. SIGGRAPH), 2008, 27(3).

- [81] Lefebvre S, Hornus S, Lasram A. By-example synthesis of architectural textures[J]. ACM Transactions on Graphics (Proc. SIGGRAPH), 2010, 29(4).
- [82] Crane K, Desbrun M, Schröder P. Trivial connections on discrete surfaces[J]. Computer Graphics Forum (SGP), 2010, 29(5):1525–1533.
- [83] Chen G, Esch G, Wonka P, Müller P, Zhang E. Interactive procedural street modeling[J]. ACM Transactions on Graphics, 2008, 27(3).
- [84] Hertzmann A, Zorin D. Illustrating smooth surfaces[C]//. 2000.
- [85] Forrester Cole A G, Limpaecher A, Barros H S, Finkelstein A, Funkhouser T, Rusinkiewicz S. Where do people draw lines?[J]. ACM Transactions on Graphics (Siggraph 2008), 2008, 27(3).
- [86] Gorla G, Interrante V, Sapiro G. Texture synthesis for 3d shape representation[J]. IEEE Trans. Vis. Comput. Graph, 2003, 9(4):512–524.
- [87] Narain R, Kwatra V, Lee H P, Kim T, Carlson M, Lin M. Feature-guided dynamic texture synthesis on continuous flows[C]// Proc. Eurographics Symposium on Rendering. 2007: 361–370–105.
- [88] Alliez P, Cohen-Steiner D, Devillers O, Bruno Lévy M D. Anisotropic polygonal remeshing[J]. ACM Transactions on Graphics (Siggraph Asia 2003), 2003, 22(3).
- [89] Marinov M, Kobbelt L. Direct anisotropic quad-dominant remeshing[J]. Computer Graphics and Applications, Pacific Conference on, 2004, 0:207–216.
- [90] Kälberer F, Nieser M, Polthier K. Quadcover-surface parameterization using branched coverings[J]. Computer Graphics Forum, 2007, 26(3):375 – 384.
- [91] Dong S, Bremer P T, Garland M, Pascucci V, Hart J C. Spectral surface quadrangulation[J]. ACM Transactions on Graphics (Siggraph 2008), 2006, 25(3).
- [92] DeRose T, Meyer M. Harmonic coordinates[R]. Pixar Technical Memo #06-02, 2006.
- [93] Meyer M, Desbrun M, Schröder P, Barr A H. Discrete differential-geometry operators for triangulated 2-manifolds[J]. Visualization and Mathematics, 2002, 3:34–57.

- [94] Desbrun M, Meyer M, Schröder P, Barr A H. Implicit fairing of irregular meshes using diffusion and curvature flow[C]// Proceedings of the 26th annual conference on Computer graphics and interactive techniques(SIGGRAPH '99). 1999.
- [95] Davis T A, Hager W W. Modifying a sparse Cholesky factorization[J]. SIAM Journal on Matrix Analysis and Applications, 2006, 20(3):606–627.
- [96] Davis T A. User guide for CHOLMOD[R]. University of Florida, 2006.
- [97] Chen R, Liu L, Dong G. Local resampling for patch-based texture synthesis in vector fields[J]. International Journal of Computer Applications in Technology, 2009, to appear.
- [98] Kwatra V, Adalsteinsson D, Kim T, Kwatra N, Carlson M, Lin M. Texture fluids[J]. IEEE Trans. Vis. & Comp. Graphics, 2007, 13(5):939–952.
- [99] Lefebvre S, Hoppe H. Appearance-space texture synthesis[J]. ACM Transactions on Graphics, 2006, 25(3):541–548.
- [100] Wu Q, Yu Y. Feature matching and deformation for texture synthesis[J]. ACM Transactions on Graphics, 2004, 23(3):364–367.
- [101] Yoshizawa S, Belyaev A, Seidel H P. Fast and robust detection of crest lines on meshes[C]// Proc. of ACM Symposium on Solid and Physical Modeling. 2005: 227–232.
- [102] 蒋国兴, 朱登洁, 黄明吉. 基于图像信息的数控雕刻三维建模的研究[J]. 中国制造业信息化, 2007, 36(21).
- [103] Fattal R, Lischinski D, Werman M. Gradient domain high dynamic range compression[J]. ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH 2002, 2002, 21(3):32–38.
- [104] Ashikhmin M. A tone mapping algorithm for high contrast images[C]// Proceedings of the 13th Eurographics workshop on Rendering. 2002: 145–156.

- [105] Song W, Belyaev A, Seidel H P. Automatic generation of bas-relief from 3d shapes[C]// Proceedings of the IEEE International Conference on Shape Modeling and Applications. 2007: 211–214.
- [106] Lee C H, Varshney A, Jacobs D W. Mesh saliency[J]. ACM Transactions on Graphics, 2005, 24(3):659–666.
- [107] Luft T, Codditz C, Deussen O. Image enhancement by unsharp masking the depth buffer[J]. ACM Transactions on Graphics, 2006, 25(3):1206–1213.
- [108] Krüger J, Westermann R. Linear algebra operators for gpu implementation of numerical algorithms[J]. ACM Transactions on Graphics, 2003, 22(3):908–916.
- [109] Cuntz N, Kolb A, Strzodka R, Weiskopf D. Particle level set advection for the interactive visualization of unsteady 3d flow[J]. Computer Graphics Forum, 2008, 27(3):719–726.
- [110] Bolz J, Farmer I, Grinspun E, Schröder P. Sparse matrix solvers on the gpu: conjugate gradients and multigrid[J]. Computer Graphics Forum, 2003, 22(3):917–924.
- [111] Zhou K, Hou Q, Wang R. Real-time kd-tree construction on graphics hardware[J]. ACM Transactions on Graphics, 2008, 21(5):231–241.
- [112] 周世哲, 满家巨. 基于多重网格法的实时流体模拟[J]. 计算机辅助设计与图形学学报, 2007, 19(7):935–941.
- [113] Kim S, Hagh-Shenas H, Interrante V. Conveying shape with texture: experimental investigations of texture's effects on shape categorization judgments[J]. IEEE Trans. Vis. & Comp. Graphics, 2004, 10(4):471-483.
- [114] Interrante V, Fuchs H, Pizer S. Conveying the 3D shape of smoothly curving transparent surfaces via texture[J]. IEEE Trans. Vis. & Comp. Graphics, 1997, 3(2): 98-117.

简 历

基本情况

周世哲，男，浙江大学数学系博士研究生。

教育状况

2001年9月到2008年7月，湖南师范大学，本科，硕士，计算机科学与技术。

2008年9月到2011年7月，浙江大学，博士，应用数学。

研究兴趣

计算机图形学。

联系方式

通讯地址：杭州市浙江大学玉泉校区数学系，邮编：310027

E-mail: fleeting.ash.zsz@gmail.com

攻读博士学位期间主要研究成果

- 1 Shizhe Zhou, Hongbo Fu, Ligang Liu, Daniel Cohen-Or, Xiaoguang Han. Parametric Reshaping of Human Bodies in Images. ACM Transactions on Graphics (Proc. ACM SIGGRAPH), 29(4), Article No.126: 1-10, 2010.
- 2 周世哲, 刘利刚. 实时数字浮雕建模. 计算机辅助设计与图形学学报, 2010, 22(3):434-439.
- 3 Kai Xu, Daniel Cohen-Or, Tao Ju, Ligang Liu, Hao Zhang, Shizhe Zhou, Yueshan Xiong. Feature-Aligned Shape Texturing. ACM Transactions on Graphics (Proc. ACM SIGGRAPH ASIA), 28(5), Article No. 108: 1-7, 2009.