

面向复杂三维场景的高质量纹理映射

姜翰青 王博胜 章国锋 鲍虎军

(浙江大学计算机辅助设计与图形学国家重点实验室 杭州 310058)

摘要 为三维模型创建纹理贴图是真实感图形学领域的一个重要课题,然而现今针对大数据的复杂结构的自然场景的纹理映射方法还很少. 该文提出了一种新的多视图纹理映射方法,能够利用多角度拍摄的图像序列为三维模型进行高真实感的纹理贴图. 相比现有的方法,该文的方法在针对模型中存在复杂遮挡的纹理映射以及大纹理数据的融合这两方面有着明显的优势. 首先将多视图纹理映射转化为一个马尔可夫随机场问题,并用能量优化的方法获得模型每个面片最优的纹理图像对应关系;然后将映射的纹理数据进行精简化,并对部分不可见区域的无效纹理进行修复;最后,利用梯度融合的方法消除纹理拼接边界的颜色差异,最终获得无缝隙的纹理贴图效果. 一系列复杂场景的实例证明了该文方法的有效性和鲁棒性.

关键词 多视图纹理映射;纹理映射最优化;无缝隙纹理融合

中图法分类号 TP391 DOI号 10.11897/SP.J.1016.2015.02349

High-Quality Texture Mapping for Complex 3D Scenes

JIANG Han-Qing WANG Bo-Sheng ZHANG Guo-Feng BAO Hu-Jun

(State Key Lab of CAD&CG, Zhejiang University, Hangzhou 310058)

Abstract Texture mapping of 3D models is an important problem in photorealistic graphics. However, few methods are proposed to deal with large-scale texture mapping of natural scenes with complicated structures. This paper presents a novel multi-view texture mapping method, which can recover photorealistic texture maps for 3D models using multi-view images. Compared to the existing methods, our method has obvious advantages in texture mapping with complicated occlusions and fusion of large-scale texture data. We first formulate multi-view texture mapping as a Markov Random Field problem, and minimize the energy function to get an optimal texture image map for each face on the 3D model. Then, we simplify the mapped texture data, and restore the invalid textures in some invisible regions. Finally, we eliminate color differences along the texture stitching boundaries by gradient fusion to achieve seamless texture mapping. A variety of challenging examples demonstrate the effectiveness and robustness of the proposed method.

Keywords multi-view texture mapping; texture map optimization; seamless texture fusion

1 引言

纹理映射是建立三维物体表面和二维图像空间

像素坐标对应关系的过程. 在三维图形学中,纹理映射允许加载用户采集的真实的自然场景照片来作为二维纹理数据,这对于表达具有真实感的三维物体模型具有非常重要的意义. 对于自然界的复杂场景,

收稿日期:2014-05-12;最终修改稿收到日期:2015-07-01. 本课题得到国家科技支撑计划(2012BAH35B02)、国家“八六三”高技术研究发展计划项目基金(2013AA013802)、国家自然科学基金(61272048,61103104)、全国优秀博士学位论文作者专项资金(201245)和新加坡新科劲力有限公司资助. 姜翰青,男,1984年生,博士,主要研究方向为基于视频的三维重建与分割. E-mail: jiang.hanqing@foxmail.com. 王博胜,男,1991年生,博士研究生,中国计算机学会(CCF)会员,主要研究方向为基于图像的三维重建和纹理映射. 章国锋(通信作者),男,1981年生,博士,副教授,中国计算机学会(CCF)会员,主要研究方向为摄像机跟踪、三维重建、视频分割与编辑、增强现实. E-mail: zhangguofeng@cad.zju.edu.cn. 鲍虎军,男,1966年生,博士,教授,博士生导师,中国计算机学会(CCF)理事,主要研究领域为计算机图形学、三维视觉、虚拟现实和增强现实.

单幅或者少数几幅照片往往不能涵盖场景的完整信息,往往需要更多角度的图像数据才能够实现完整的纹理映射.本文的研究目标便是如何利用多角度的图像序列来实现高质量的三维模型纹理映射.

近年来,随着三维重建技术的迅速发展^[1],从多视图图像序列或者视频中重建高精度的自然场景的三维模型已经越来越成熟.然而,要实现对自然三维场景的逼真展示,仅恢复其三维几何模型是不够的,还需要恢复其表面的真实纹理.因此如何有效地为重建的三维模型贴上真实的纹理同样是一个值得关注的课题.虽然已有一些工作^[2-3]尝试利用多视图图像序列来实现自然场景三维模型的纹理映射,并且有些方法^[4-5]已经能够创建高真实感的纹理贴图,但是这些方法往往只针对于结构简单的三维物体模型,难以很好地处理复杂的大尺度自然场景数据.

本文提出了一种纹理拼接优化的方法来实现三维模型的高真实感纹理映射.如图 1 所示,输入一段多视角拍摄并且每帧摄像机参数和深度已知的纹理图像序列,以及与该序列场景世界坐标系对齐的三维几何模型,我们的目标是为三维模型恢复完整的真实感纹理贴图.与文献^[4-5]类似,我们的方法将纹理拼接优化问题转换为一个马尔可夫随机场问题,并用能量优化的方法对其进行求解,如图 1(c)所示.然而,我们方法的不同之处在于考虑了遮挡和大尺度场景等问题的处理,因此能够很好地解决结构复杂的大尺度场景的纹理贴图,这是现有的方法难以做到的.

我们首先对三维模型中的每个三角面片选择最佳的纹理图像,从而实现纹理映射的整体最优化;然后我们对纹理数据进行分块和有效区域裁剪,以最小化整体纹理数据;接下来,我们对纹理图像的颜色进行调整,从而消除来自不同纹理图像帧的纹理色差,实现三维模型的无缝隙纹理映射.

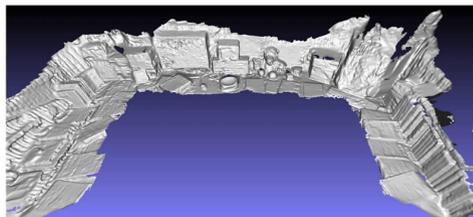
我们方法的主要创新点如下:

(1) 现有的纹理映射方法对于自遮挡结构的纹理贴图存在多义性,而我们的方法能够利用深度信息来有效地处理自遮挡,因此适用于结构复杂的三维模型的纹理贴图;

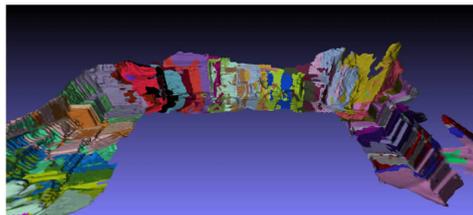
(2) 由于内存的限制,现有的大多数方法仅能够处理结构简单的小规模三维模型,缺乏一种面向大尺度复杂三维场景的纹理映射方法,而我们的方法通过迭代求解来分片处理模型的纹理,因此适用于大尺度场景三维模型的纹理贴图应用.



(a) 输入视频序列及其深度图序列



(b) 输入三维模型



(c) 纹理拼接最优化标记



(d) 纹理贴图结果



(e) 图(d)中的方框区域的局部放大

图 1 “桌面”实例的多视图纹理映射结果

2 相关工作

视频的每帧图像仅仅能捕获到场景的一个局部区域,因此需要结合视频的多帧图像来创建三维模型的完整纹理贴图. Wang 等人^[2]的方法和 Baumberg^[3]的方法用加权混合的方式结合多帧纹理图像,为每帧图像求解合适的权重以获得最佳的混合效果.然而,如果纹理图像之间存在偏移或者颜色差异,加权混合后会产生明显的走样. Niemi 和 Broszio^[6]的方法为每个三角面片选择其可见的某帧纹理图像进行映射,并用贪心的方式调整每个三角面片使得相邻的三角面片尽可能来自同一幅纹理图像,该方法的结果很难完全保证相邻三角面片的纹理一致性.与文献^[6]不同, Lempitsky 和 Ivanov^[4]提出将

三维模型的多视图纹理映射视为一个图像拼接问题; 整体模型的纹理映射即为每个三角面片选择合适的纹理图像, 并且保持相邻三角面片纹理接缝颜色的一致性. 该问题可转换为一个马尔可夫随机场问题进行求解. Gal 等人^[7]对于文献[4]的方法做了改进, 对于三角面片在每个帧纹理图像上的映射考虑了局部位移, 从而能够处理摄像机参数和三维模型存在的误差偏移问题, 但是求解空间扩大会影响该方法的运行效率. 文献[4, 7]提出的方法计算时间和空间复杂度均很高, 因此仅能够处理形状简单的小规模模型. 此外, 上述这些方法均没有考虑三维模型的自遮挡情况. 我们的方法与文献[4, 7]类似, 不同之处在于能够处理模型自遮挡导致的歧义性, 并且能够处理大尺度复杂场景三维模型的纹理映射, 这是上述方法所不能做到的.

除了文献[7]之外, 近年来还有一些纹理映射方法也能够较好地处理误差偏移问题, 从而实现高质量的真实感纹理贴图效果. 例如, Eisemann 等人^[8]利用多帧纹理图像之间的光流来实现图像对齐以解决误差偏移问题, 其运行效率依赖于 GPU 的加速. Goldluecke 等人^[5]则提出了超分辨率的纹理映射方法. Zhou 和 Koltun^[9]提出了利用颜色一致性调整摄像机参数并且对图像帧进行形变来解决误差偏移问题. 由于计算时间和空间上的复杂性, 这些方法都难以处理大尺度复杂场景的纹理贴图.

还有一些结合用户交互的方法^[10-12]提供了半自动的工具允许用户在模型面片和图像区域之间指定匹配点, 从而将纹理图像贴至三维模型表面上. 与这些交互工具不同, 我们的方法能够全自动地为三维模型的面片选取合适的纹理贴图区域, 无需繁琐的用户交互.

3 算法目标和系统概述

给定一段包含 n 帧的自然场景图像序列 $I = \{I_t | t=1, \dots, n\}$ 以及视频每帧对应的摄像机参数和深度图, 同时输入该场景的三维几何模型 $M = \{F_i | i=1, \dots, m\}$ (其中 m 为模型中三角面片的总数, F_i 为第 i 个三角面片, 其包含 3 个顶点 $\{V_k^i | k=1, 2, 3\}$, 顶点的位置与该序列场景世界坐标系对齐), 我们的目标是场景模型恢复完整的真实感纹理贴图以及纹理坐标数据. 视频第 t 帧的摄像机参数表示为 $C_t = \{K_t, \mathbf{R}_t, \mathbf{T}_t\}$, 其中 K_t 为内参, \mathbf{R}_t 为旋转矩阵, \mathbf{T}_t 为平移向量, 第 t 帧的深度图表示为 D_t . 我们分别

用 $I_t(x)$ 和 $D_t(x) = 1/Z_t(x)$ 来表示第 t 帧的像素点 x 对应的颜色和视差 (即深度的倒数).

这里的三维模型既可以通过深度相机 (如 Kinect 或 ToF 相机) 直接扫描得到 (如图 2), 也可以通过多视图立体方法 (如文献[13]或 PMVS2^[14]) 来重建, 甚至可以通过人工交互建模的方式得到 (如图 3 的家具模型). 采用多视图立体方法来重建, 一般先恢复每帧的摄像机参数和深度图 (在实验里, 我们采

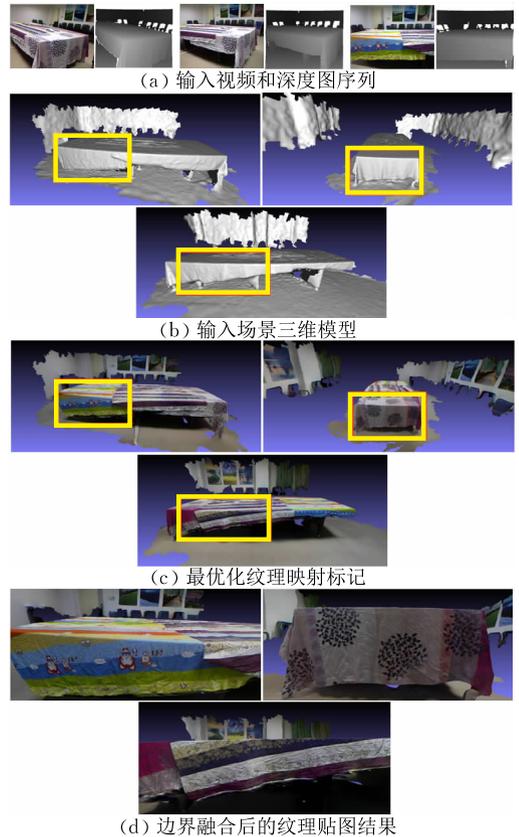


图 2 Kinect 数据的纹理映射结果

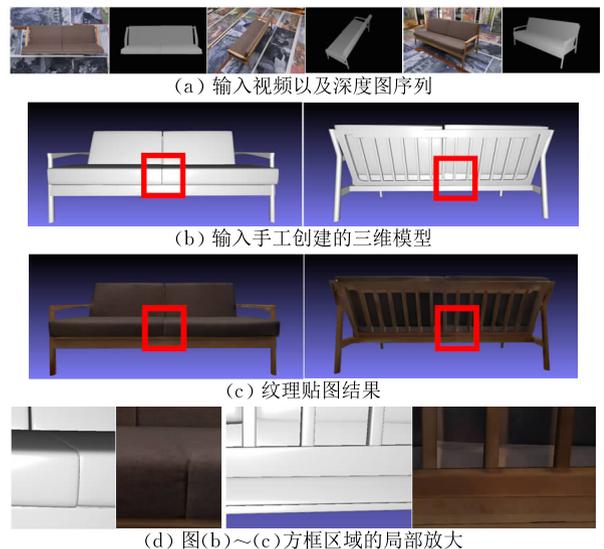


图 3 “家具”模型的纹理映射结果

用 ACTS 软件^①来恢复相机的参数和每帧的深度图像). 而对于人工交互建模的模型, 可能没有深度图, 可以通过将三维模型在每帧图像对应的视角下进行深度绘制来得到每帧的深度图像.

图 4 展示了系统的整体框架. 我们首先用能量优化方法为三维模型中的每个三角面片选择最佳的纹理图像, 从而实现纹理映射的整体最优化(见

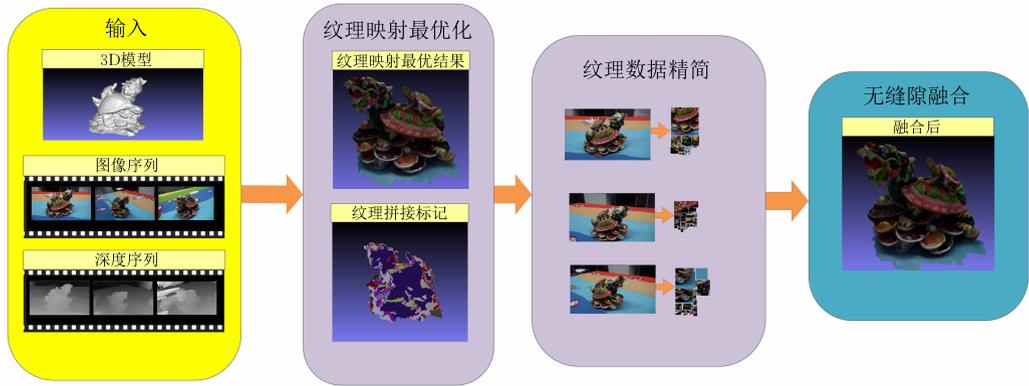


图 4 系统框架

4 纹理映射优化

我们输入的纹理数据是多视角拍摄的图像序列, 因此对于三维模型的每个三角面片, 可映射的纹理图像不止一幅, 我们所要做的是为每个三角面片选择一幅最优的纹理图像进行映射. 换句话说, 对于每个三角面片 F_i , 从输入图像序列 $I = \{I_t | t=1, \dots, n\}$ 中寻找一幅最优的纹理图像 $I_{l(i)}$, 其帧号 $l(i) \in \{1, \dots, n\}$ 可视为关于 i 的标记函数. 最优化的纹理映射结果即为所有三角面片最优纹理帧标记的集合 $L = \{l(i) | i=1, \dots, m\}$. 由此, 我们将纹理映射最优化问题转化为求解 L 标记问题. 此标记问题有两方面的目标: (1) 选择纹理映射信息最丰富的纹理图像; (2) 保证来自不同纹理图像的相邻三角面片之间的纹理缝隙颜色一致性. 类似于文献[4, 7], 我们可以将此标记问题视为一个马尔可夫随机场问题, 并用能量最小化的方法进行求解. 我们的能量方程表示如下:

$$E(L) = \sum_{i=1}^m (E_d(l(i)) + \sum_{F_j \in N(F_i)} E_s(l(i), l(j))) \quad (1)$$

其中: $N(F_i)$ 表示与 F_i 相邻的三角面片集合; 数据项 E_d 衡量纹理图像 $I_{l(i)}$ 在三角面片 F_i 上映射的纹理信息丰富程度, 平滑项 E_s 衡量模型表面不同纹理图像缝隙的颜色一致性.

第 4 节); 然后, 我们按照纹理映射结果对模型的三角面片进行分块, 同时对纹理图像的有效区域进行裁剪, 以此精简化模型整体的纹理数据(见第 5 节); 接下来我们对纹理图像的颜色进行调整, 从而消除来自不同纹理图像帧的纹理接缝色差, 得到三维模型最终无缝隙的纹理图像和纹理坐标(见第 6 节).

与文献[7]类似, 我们从两方面来衡量纹理映射信息的丰富程度: (1) 三角面片映射纹理图像的分辨率(可以用三角面片到纹理图像摄像机位置的距离来衡量); (2) 纹理图像视线方向与三角面片法向一致性(正视角度的纹理信息最丰富). 结合上述两方面标准, E_d 可以按照以下公式定义:

$$E_d(l(i)) = - \sum_{k=1}^3 d(V_i^k, l(i)) a(V_i^k, l(i)) \quad (2)$$

$d(V_i^k, l(i))$ 是关于顶点 V_i^k 到纹理图像摄像机位置距离的函数, 用来衡量三角面片映射纹理图像的分辨率, 其定义如下:

$$d(V_i^k, l(i)) = \begin{cases} \frac{1}{z(P_{l(i)}(V_i^k)) \delta_d} + 1, & z(P_{l(i)}(V_i^k)) \leq Z_{l(i)}(p_{l(i)}(V_i^k)) \\ 0, & z(P_{l(i)}(V_i^k)) > Z_{l(i)}(p_{l(i)}(V_i^k)) \end{cases} \quad (3)$$

其中, $P_{l(i)}(V_i^k)$ 表示 V_i^k 在第 $l(i)$ 帧摄像机坐标系下的坐标, $p_{l(i)}(V_i^k)$ 表示该帧上的二维投影坐标, 函数 $z(\cdot)$ 表示取三维点的 z 轴坐标. 可以看到, $z(P_{l(i)}(V_i^k))$ 越近, 纹理映射的分辨率越高, $d(V_i^k, l(i))$ 越大. 用距离的计算来代替纹理分辨率的统计比文献[7]中提出的颜色梯度积分法更为简单直观, 而且计算效率更高. 此外, 与文献[4, 7]不同的是, 我们对 $d(V_i^k, l(i))$

① ACTS 2.0: Automatic Camera Tracking System. <http://www.zjucv.g.net/acts/acts.html>

的定义引入了遮挡的判断:如果 $z(P_{l(i)}(V_i^k))$ 比预知的投影位置 $p_{l(i)}(V_i^k)$ 的深度值更远,可以判定 V_i^k 在该帧被遮挡,该帧的纹理图像对于 V_i^k 无效,因此 $d(V_i^k, l(i))$ 默认为 0. 在能量优化的过程中,为了使目标能量降至最低,优化器会避免选择无效的纹理信息.

$a(V_i^k, l(i))$ 是关于三角面片法向与纹理图像视线方向夹角的函数,其定义如下:

$$a(V_i^k, l(i)) = \begin{cases} \langle \hat{v}(V_i^k, l(i)) | \hat{n}_i' \rangle^2 + 1, & \langle \hat{v}(V_i^k, l(i)) | \hat{n}_i' \rangle > 0 \\ 0, & \langle \hat{v}(V_i^k, l(i)) | \hat{n}_i' \rangle \leq 0 \end{cases}$$

$$\hat{v}(V_i^k, l(i)) = (-\mathbf{R}_{l(i)} \mathbf{T}_{l(i)} - V_i^k) / \|\mathbf{R}_{l(i)} \mathbf{T}_{l(i)} - V_i^k\|,$$

$$\hat{n}_i' = \sum_{F_j \in N(F_i)} \hat{n}_j / \left\| \sum_{F_j \in N(F_i)} \hat{n}_j \right\| \quad (4)$$

其中, $\langle \cdot | \cdot \rangle$ 表示两个法向的点积, $N(F_i)$ 表示 F_i 的 3 个顶点的一环邻域所包含的三角面片集合. 如果该帧摄像机视线方向与 \hat{n}_i' 的夹角大于 90° , 则三角面片 F_i 在该视角下被遮挡不可见(通常处于三维模型的背面), 该帧的纹理图像对于 V_i^k 无效, 因此 $a(V_i^k, l(i))$ 默认为 0, 同样可以避免无效纹理信息的影响. 值得注意的是, 这里我们利用三角面片邻域的表面法向平均值 \hat{n}_i' 来替代该三角面片的真实法向 \hat{n}_i . 平均法向在邻域范围内具有一致性, 因而可以避免表面法向不规整所产生的影响.

E_d 中将两类函数相乘, 可以同时考虑到上述的两方面标准, 从而有效地衡量纹理信息的丰富程度, 来作为最优帧选择的参考. 图 5(d) 展示了利用 E_d 获得的最优帧映射结果. 相比于利用真实法向计算的结果, 邻域平均法向获得的映射结果更为规整.

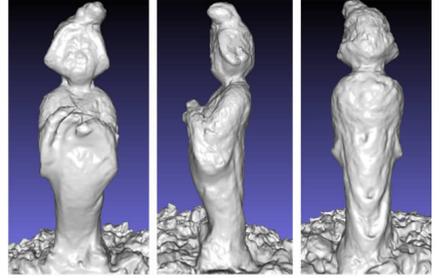
E_s 统计相邻三角面片边缘的纹理颜色一致性, 从而保证无缝隙的纹理拼接. 其定义如下:

$$E_s(l(i), l(j)) = [l(i) \neq l(j)] \left\| I_{l(i)} \left(\frac{(k-1)p_{l(i)}(V_{ij}^1) + (r-k)p_{l(i)}(V_{ij}^2)}{r-1} \right) - I_{l(j)} \left(\frac{(k-1)p_{l(j)}(V_{ij}^1) + (r-k)p_{l(j)}(V_{ij}^2)}{r-1} \right) \right\|_2 \quad (5)$$

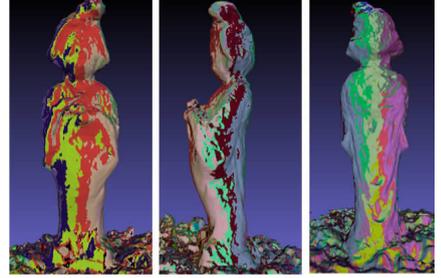
其中 (V_{ij}^1, V_{ij}^2) 表示 F_i 和 F_j 的公共边, r 表示边投影的采样点数. 可以看到, 仅当两个相邻三角面片映射自同一幅纹理图像时, E_s 为 0, 从而鼓励相邻的三角面片尽可能映射自相同的纹理图像, 减少纹理缝隙的出现. 对于处在纹理缝隙上的相邻三角面片, E_s 衡



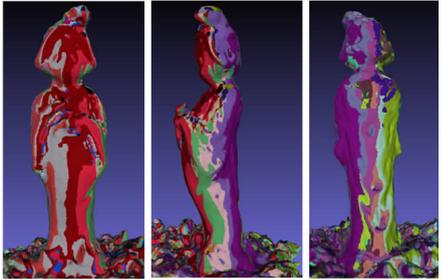
(a) “仕女”实例输入视频以及深度图序列



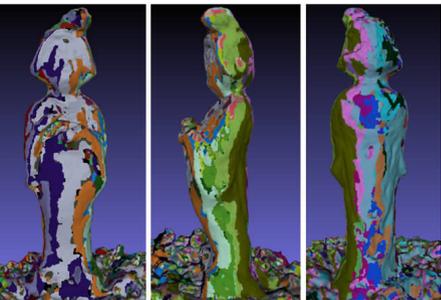
(b) 输入三维模型



(c) 利用三角面片真实法向计算的最优映射标记(无平滑项约束)



(d) 利用邻域平均法向计算的最优映射标记(无平滑项约束)



(e) 结合平滑项约束的最优化纹理拼接标记



(f) 相应的纹理贴图结果

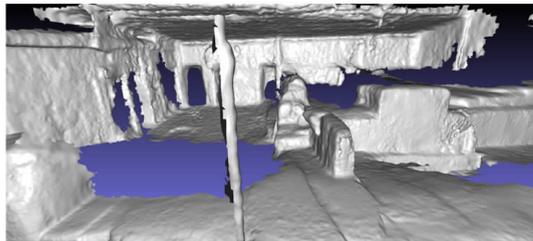
图 5 纹理映射最优化示意

量其公共边在各自纹理图像上的投影颜色差异(在公共边的投影位置均匀采样 r 个点进行颜色比较),从而保证纹理缝隙处的颜色尽可能一致,实现最大程度的无缝纹理拼接.

我们利用 Graph Cuts 算法^[15]求解能量方程最小化.对于面片数目较多的大场景模型,输入纹理帧数过多会导致候选标记数目过多,从而导致内存过大.为解决此问题,我们在求解过程中采用了标记压缩法,对于每个三角面片仅考虑数据项最小的 5 个纹理帧,Graph Cuts 算法则在压缩后的纹理帧标记范围内求解,这样便可在内存允许的情况下获得近似的最优标记结果.最优化后的纹理映射标记结果如图 5(e)所示,比图 5(d)中的无平滑项约束的映射结果更为规整化.此外,如图 6 所示,由于函数 $d(\cdot)$ 和 $a(\cdot)$ 均考虑了遮挡的情况,因此我们的纹理映射最优化方法能够很好地处理三维模型中存在的复杂结构,这是文献[4,7]的方法难以做到的.



(a) “Dionysus”实例输入视频的遮挡区域



(b) 输入三维模型的遮挡区域



(c) 无遮挡处理的纹理贴图结果



(d) 引入遮挡处理的纹理贴图结果

图 6 遮挡区域的纹理映射结果比较

纹理映射最优化计算的时间和空间瓶颈在于 Graph Cuts 求解,其中,数据项的计算时间复杂度为

$O(mn)$,能量方程的求解根据文献[15]中的算法描述,其时间复杂度为 $O(m)$ (理论上为 $O(|L|^2 m)$),由于候选标记数 $|L|$ 在我们实验中取常量 5 而省略),空间复杂度为 $O(m)$,因此该环节的总时间复杂度为 $O(mn)$,空间复杂度为 $O(m)$.

5 纹理数据精简化

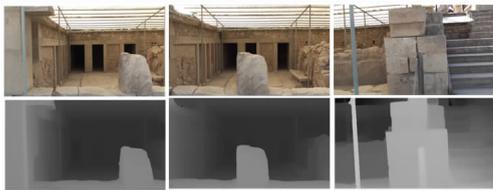
对于长序列的自然场景视频来说,纹理图像集合的数据量很大.例如,图 7 展示的“Dionysus”实例拍摄了雅典卫城的“Dionysus”歌剧院遗址内景,其中包含 343 帧图像的长序列实例,需要 107MB 大小的源纹理图像数据,这会导致真实感三维模型渲染时的纹理数据载入过程负担较大.幸运的是,根据第 4 节的纹理映射最优化算法获得的结果可以发现,三维模型并没有利用每帧纹理图像的全部数据,而是仅仅映射了一些局部区域,因此我们可以对纹理图像进行裁剪,只保留有效区域的数据,同时更新三角面片的纹理坐标,便能够在不影响纹理映射结果的前提下对纹理数据进行精简化.

首先对三维模型上的三角面片进行分块,使得每一块包含所有相邻的并且纹理映射图像相同的三角面片.为此我们构造一个无向无权图 $G = \langle \mathcal{V}, \mathcal{E} \rangle$, \mathcal{V} 中的每个顶点代表一个三角面片, \mathcal{E} 的每条边连接相邻的并且映射自同一帧纹理图像的三角面片对.然后,我们寻找 G 中的所有连通分量 $C = \{C_i | i = 1, \dots, q\}$, 每个连通分量 C_i 中包含的三角面片是相连接的整体,并且映射自同一幅纹理图像 $f(i)$.我们将 C_i 包含的三角面片集合投影至相应的纹理图像 $I_{f(i)}$ 上,计算所有三角面片投影的包围盒区域 B_i ,将包围盒所在的纹理图像块 $I_{f(i)}(B_i)$ 作为 C_i 纹理映射的有效数据单元,我们称之为“纹理块”,同时依据 B_i 的坐标位置重新计算所有三角面片的纹理坐标.所有连通分量对应的纹理块集合 $\{I_{f(i)}(B_i) | i = 1, \dots, q\}$ 即为精简的纹理数据.

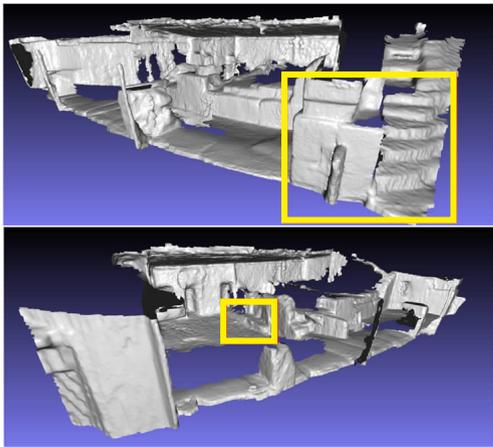
利用上述方法处理后,“Dionysus”实例的纹理数据精简至 55 MB.该环节的运行时间主要与模型面片数量有关,复杂度为 $O(m)$.

5.1 不可见纹理的修复

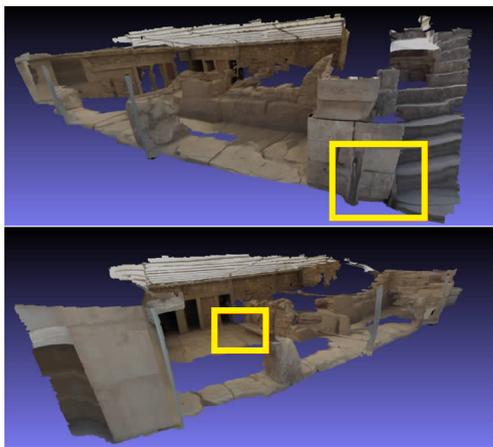
由于输入纹理视频存在遮挡情况,三维模型的某些局部区域在纹理图像序列中是不可见的,无法获得这些区域的纹理映射信息.这些区域的纹理贴图结果会呈现出“空洞”,如图 8(a)所示,这会对模型的真实感视觉效果产生很大的影响.因此,需要对



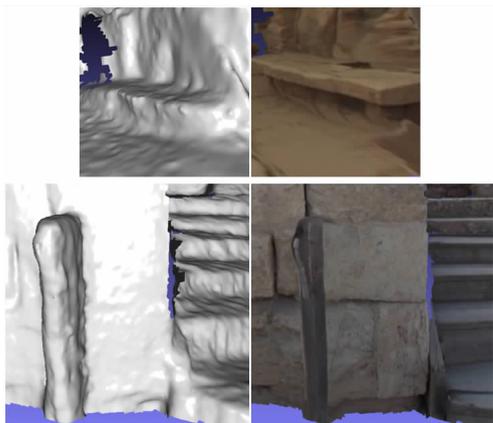
(a) 输入视频以及深度图序列



(b) 输入场景三维模型



(c) 纹理贴图结果



(d) 图(b)~(c)方框区域的局部放大

图 7 “Dionysus”实例的纹理映射结果

这些区域的纹理进行修复。我们注意到,不可见的模型区域往往是小片区域,通常不包含丰富的纹理结构信息。对于这些区域可以用简单的扩散填补方式进行修补。

对于模型中某个不可见区域 Ω , 首先获得 Ω 中面片法向的平均值 \bar{n} ; 然后, 将 Ω 邻域 (Ω 中三角面片顶点的一环邻域) 的纹理颜色正交投影到与 \bar{n} 垂直的平面上获得一张正投纹理图; 接下来, 迭代地扩散填补正投纹理图中空洞区域的边界像素点颜色, 对于每一个空洞边界上的像素点 p , 利用以 p 为中心 3×3 窗口内已知像素点颜色的平均值对其进行填补, 直到空洞区域的所有像素点都被填补完毕; 最后将该纹理图作为新的纹理块加入模型的纹理块集合中, 并计算 Ω 中三角面片的纹理坐标。扩散填补方法的修补结果如图 8(b) 所示。



(a) “桌面”实例的不可见纹理区域



(b) 修复后的纹理贴图结果

图 8 不可见纹理修复示意

6 纹理边界无缝隙融合

由于光照变化、拍摄角度的不同、表面高光 and 阴影等多种因素的影响, 模型的局部区域在多个纹理块图像中的颜色可能会不一致。尽管纹理映射优化过程中的平滑项尽可能地保持纹理边界的颜色一致性, 然而映射到三维模型上的纹理边界不可避免地会存在一些色差缝隙, 如图 9(a) 所示。因此, 还需要对精简后的纹理块颜色进行调整以消除这些缝隙, 即对纹理边界的颜色进行无缝隙融合。

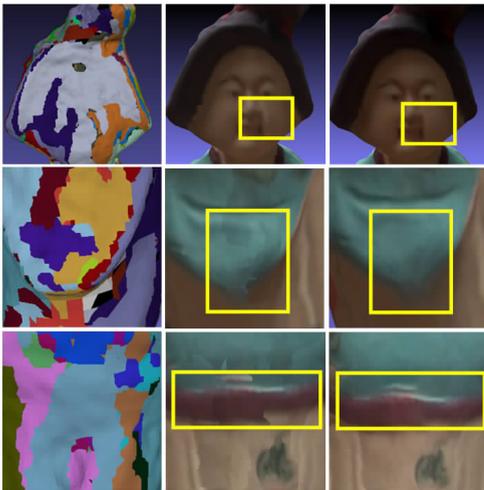
纹理边界融合需要满足 3 方面条件: (1) 只是做微小的调整, 调整后的纹理块颜色尽可能和源图像的颜色一致, 不要有太大的变动; (2) 调整后的纹理颜色梯度和源图像的颜色梯度尽可能一致; (3) 调整后的纹理块在相邻的三角面片边界上尽可能保持颜色一致。类似于文献[16], 上述条件下的纹理边界无缝隙融合问题可以通过求解一组带约束的泊松方程组来解决。该方程组求解纹理块集合 $\{I_{f(i)}(B_i) \mid i =$



(a) “仕女”实例边界融合前的纹理贴图结果



(b) 边界融合后的纹理贴图结果



(c) 图(a)~(b)中方框区域的局部放大

图 9 纹理边界融合示意

$1, \dots, q\}$ 中包含的所有像素点的颜色,每个像素点的颜色对应一个变量(我们对 RGB 的 3 个通道独立求解). 方程组包括以下 3 个部分方程式,分别对应上述 3 方面条件:

(1) 对于任意一个纹理块中的每个像素点 $x \in I_{f(i)}(B_i)$, 我们有

$$\omega_s I_{f(i)}^*(x) = \omega_s I_{f(i)}(x) \quad (6)$$

其中: $I_{f(i)}^*(x)$ 是待求解的 x 点颜色变量, 约束其其与源图像的颜色 $I_{f(i)}(x)$ 接近, ω_s 是一个权重值(在实验中取 0.01, 表示第 1 类条件仅起到很小的约束作用);

(2) 对于任意一个纹理块中的每个像素点 $x \in I_{f(i)}(B_i)$, 我们有

$$\begin{aligned} |N(x)| I_{f(i)}^*(x) - \sum_{y \in N(x)} I_{f(i)}^*(y) = \\ |N(x)| I_{f(i)}(x) - \sum_{y \in N(x)} I_{f(i)}(y) \end{aligned} \quad (7)$$

其中 $N(x)$ 表示 x 的四邻域像素点集合(仅在包围盒 B_i 范围内的有效), 这部分方程式约束 x 点颜色变量的拉普拉斯梯度和源图像保持一致;

(3) 任意一对在三维模型上相邻的纹理块 $I_{f(i)}(B_i)$ 和 $I_{f(j)}(B_j)$ (相应的三角面片集合存在相邻关系), 将 $I_{f(i)}(B_i)$ 和 $I_{f(j)}(B_j)$ 相邻纹理边界中的每一条边 (V_{ij}^1, V_{ij}^2) 投影到两幅纹理图像 $I_{f(i)}$ 和 $I_{f(j)}$ 上, 并对二维投影边 $(p_{f(i)}(V_{ij}^1), p_{f(i)}(V_{ij}^2))$ 和 $(p_{f(j)}(V_{ij}^1), p_{f(j)}(V_{ij}^2))$ 均匀采样 r 个点, 对于每一对采样点我们有

$$\begin{aligned} I_{f(i)}^* \left(\frac{(k-1)p_{f(i)}(V_{ij}^1) + (r-k)p_{f(i)}(V_{ij}^2)}{r-1} \right) = \\ I_{f(j)}^* \left(\frac{(k-1)p_{f(j)}(V_{ij}^1) + (r-k)p_{f(j)}(V_{ij}^2)}{r-1} \right) \end{aligned} \quad (8)$$

其中 $k \in \{1, \dots, r\}$, 这部分方程式约束纹理边界两侧的颜色变量保持一致.

由于以上方程组是过约束的稀疏线性方程组(方程数大于变量数), 用共轭梯度法对其进行求解, 并将求解获得的颜色写回相应的纹理块中, 便可获得纹理边界无缝融合的效果, 如图 9(b) 所示.

6.1 大数据纹理的迭代融合

对于长序列的大数据纹理图像, 稀疏线性方程组的规模会很大, 求解方程容易造成内存溢出. 例如, 图 1 所示的“桌面”实例的方程组规模达到 20649637×20649637 之大. 对于此类大纹理数据场景, 无法一次性求解所有的纹理块, 需要对纹理块集合进行分批求解, 每次求解纹理块的像素点总数不超过 10000000. 为此, 我们对纹理融合算法进行改进, 采用一种迭代融合的策略. 每次迭代选取若干个纹理块用上述方法进行融合求解, 并将求解完成的纹理块标记为“已求解”. 在下次迭代过程中, 用“已求解”的纹理块颜色作为除了上述 3 类约束条件之外的附加约束条件: 如果待求解的纹理块 $I_{f(i)}(B_i)$ 的某个相邻纹理块 $I_{f(j)}(B_j)$ 标记为“已求解”, 则我们将其相邻纹理边界中的每一条边 (V_{ij}^1, V_{ij}^2) 投影到两幅纹理图像 $I_{f(i)}$ 和 $I_{f(j)}$ 上, 并在二维投影边 $(p_{f(i)}(V_{ij}^1), p_{f(i)}(V_{ij}^2))$ 和 $(p_{f(j)}(V_{ij}^1), p_{f(j)}(V_{ij}^2))$ 上均匀采样 r 个点, 对于每一对采样点我们有以下方程式:

$$I_{f(i)}^* \left(\frac{(k-1)p_{f(i)}(V_{ij}^1) + (r-k)p_{f(i)}(V_{ij}^2)}{r-1} \right) =$$

$$\hat{I}_{f(j)} \left(\frac{(k-1)p_{f(j)}(V_{ij}^1) + (r-k)p_{f(j)}(V_{ij}^2)}{r-1} \right) \quad (9)$$

其中 $\hat{I}_{f(j)}$ 表示修改后的第 $f(j)$ 帧的纹理颜色。如此便可约束纹理边界一侧带求解的颜色变量和另一侧已求解的颜色常值保持一致, 以保证每次迭代求解的颜色结果可以融合到接下来的迭代运算中。我们按照广度优先顺序遍历选取纹理块作为每次迭代的求解对象, 以保证下一次迭代选取的纹理块和当前迭代选取的纹理块有相邻, 从而保证求解结果能够不断地传递扩散遍整个三维模型, 保证整体模型纹理边界实现无缝隙融合, 如图 10(d) 所示。

迭代融合的算法伪代码如算法 1 所示。该算法的时间瓶颈在于线性方程组的共轭梯度求解环节, 其时间复杂度为 $O\left(\sum_{i=1}^q |I_{f(i)}(B_i)|\right)$, 其中 $\sum_{i=1}^q |I_{f(i)}(B_i)|$ 为需要迭代融合的纹理块像素点数总和。该迭代融合算法能够在有限的内存下处理大规模的纹理数据, 这是现有的算法难以做到的。

算法 1. 纹理数据迭代融合算法。

遍历纹理数据中的每一个纹理块 $I_{f(i)}(B_i)$:

如果 $I_{f(i)}(B_i)$ 未被标记为“已访问”:

1. 创建队列 $Q = \{I_{f(i)}(B_i)\}$, 并创建纹理块集合 $S = \emptyset$, 初始为空, 并将 $I_{f(i)}(B_i)$ 标记为“已访问”;

2. 迭代执行以下操作:

2.1. 取出 Q 队首的纹理块 $I_{f(h)}(B_h)$:

2.1.1. 如果 $I_{f(h)}(B_h)$ 中的像素点数与 S 中的像素点数总和大于 10 000 000 个:

2.1.1.1. 对 S 中的纹理块集合进行融合求解, 其中方程组的创建方法如下:

对于 S 中的每个纹理块 $I_{f(s)}(B_s)$ 创建式(6)、(7), 并遍历每个与其相邻的纹理块 $I_{f(n)}(B_n) \in N(I_{f(s)}(B_s))$:

1) 如果 $I_{f(n)}(B_n)$ 为“已求解”, 则创建式(9);

2) 如果 $I_{f(n)}(B_n) \in S$, 则创建式(8);

2.1.1.2. 将 S 中的所有纹理块标记为“已求解”, 并清空 S ;

2.1.2. 将 $I_{f(h)}(B_h)$ 加入 S ;

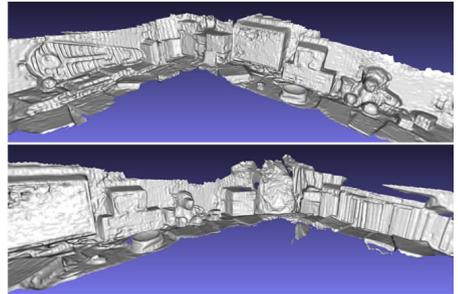
2.2. 遍历每个与其相邻的纹理块 $I_{f(n)}(B_n) \in N(I_{f(h)}(B_h))$:

如果 $I_{f(n)}(B_n)$ 未被标记为“已访问”, 则将其添加至 Q 队尾, 并将其标记为“已访问”。

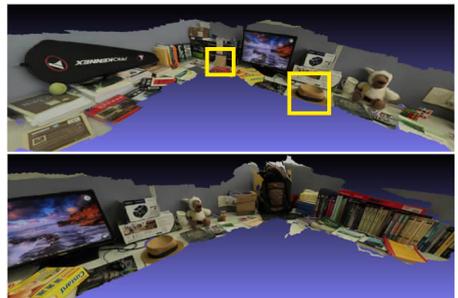
最后, 我们用 Zhou 等人^[17] 提出的网格参数化方法(该算法已经在 Direct3D 9 中实现)将融合后的所有纹理块信息展开至一幅完整的纹理图像上(如图 10(e) 所示), 并重新计算三角面片的纹理坐标。



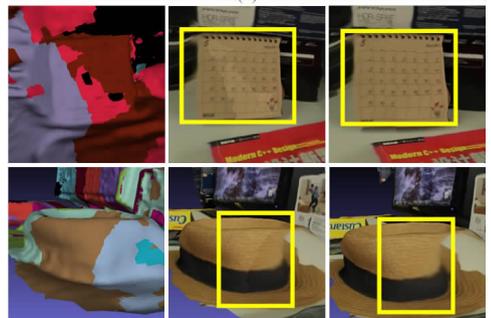
(a) “桌面”实例的输入视频以及深度图序列



(b) 输入场景三维模型



(c) 场景的纹理贴图结果



(d) 图(c) 方框区域的局部放大图, 展示边界融合前后的纹理贴图细节比较, 其中每组图分别为纹理映射最优优化标记、边界融合前的纹理以及融合后的纹理结果



(e) 参数化展开后的纹理图像

图 10 大数据实例的纹理映射

7 实验结果

我们实验了几组视频数据及其三维模型。所有

实验都是在一台主频为 3.20 GHz CPU 的台式机上运行的. 对于图 1 的“桌面”实例, 包含 211 帧分辨率为 960×540 的序列, 纹理映射的时间需要 14 min, 实际内存占用仅 770 MB. 表 1 给出了本文主要实例的运行时间和内存占用情况. “桌面”、“仕女”、“Dionysus”三组实例以及图 11 和图 12 展示的两组实例的输入三维模型均是用多视图三维重建的方法^[14]生成的. 其中, “龙龟”实例的三维模型包含一些复杂的自遮挡结构, 并且纹理视频颜色还包含非朗伯的光照变化, 而我们的方法能够很好地处理这些问题并获得无缝隙的纹理贴图结果, 如图 11(e)

表 1 本文所有实例运行时间和内存占用情况

实例(分辨率 960×540)	帧数	模型大小 (三角面片个数)	精简纹理		所需 时间/min	内存 占用/MB	纹理映射优化环节		无缝隙融合环节	
			大小/MB	精简比例/%			时间/s	内存/MB	时间/min	内存/MB
桌面	211	1043007	60	93	14	770	84	770	12	733
仕女	551	486921	18	18	5	545	92	545	3	492
Dionysus	343	1633989	55	51	15	1325	287	1325	10	900
龙龟	434	782302	6	9	2	300	22	300	1	157
Dionysus2	200	562186	48	80	9	736	7	550	8	736

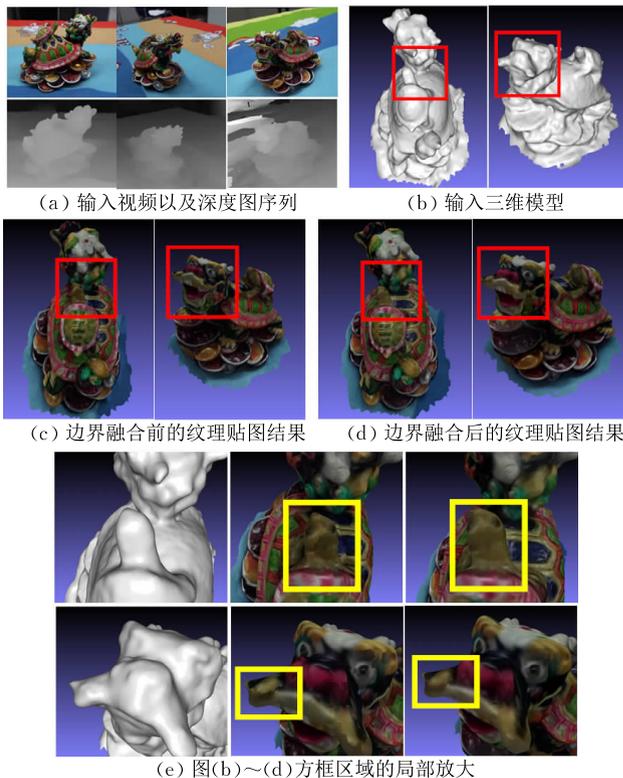


图 11 “龙龟”模型的纹理映射结果

8 结论和未来工作展望

本文提出了一种三维模型的纹理映射方法, 能够利用多视角拍摄图像序列实现高真实感的纹理贴图. 我们将纹理拼接最优化问题转换为一个马尔可

所示. 图 12 展示的大规模场景模型的实例拍摄了“Dionysus”歌剧院遗址外景, 我们的方法能够精确地还原场景模型的完整纹理贴图以及局部的纹理细节. 上文图 3 展示的家具三维模型是通过手工建模获得的, 跟实际家具的三维结构难免会有细小的出入, 而我们的方法能够很好地处理三维模型的结构误差并获得真实感的纹理贴图效果. 上文图 2 还展示了一组 Kinect 捕获的室内场景实例, 我们的方法同样能够实现高质量的纹理映射. 我们的方法对于各类复杂的自然场景模型均能够实现高质量的完整纹理映射, 这是现有多视图纹理映射方法较难做到的.

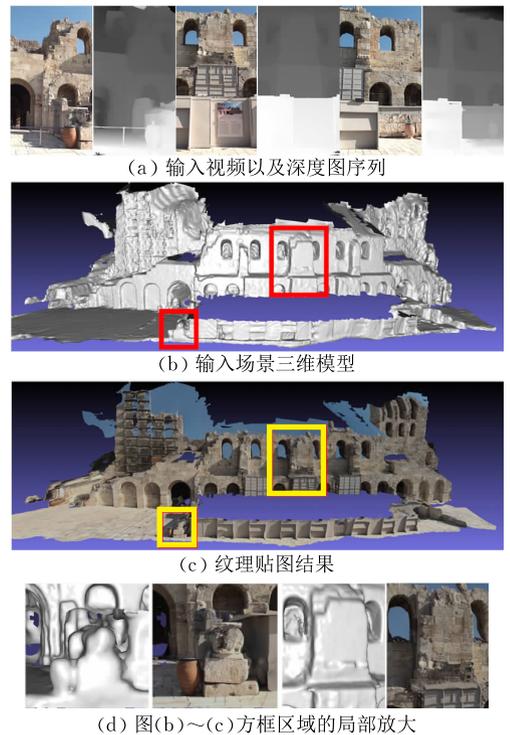


图 12 “Dionysus 2”实例的纹理映射结果

夫随机场问题, 并通过求解能量方程实现最优视图的纹理映射. 我们的方法能够很好地处理自然场景的自遮挡等复杂结构的纹理贴图. 对于不同视图纹理拼接边界产生的色差, 我们提出了一种迭代融合的算法以消除颜色差异, 能够实现大数据的场景的无缝纹理拼接.

本文的实验数据输入的三维模型的坐标位置以

及在每帧纹理上的投影均较为精确,然而实际应用中可能存在三维模型无法精确对齐多个视角的纹理数据的情况,如何处理模型自身的误差使得纹理贴图的算法更为鲁棒是我们接下来将要解决的问题.如果输入的模型存在局部破损结构,现有的方法无法恢复这些破损区域的纹理,如何利用多视图的纹理信息来指导模型结构的修补,从而实现完整的纹理修复以满足实际工业应用要求,也是未来我们的研究方向.此外,我们在纹理图像的选择中仅考虑了图像的分辨率和视线方向,实际应用中光照条件等拍摄环境因素同样会影响视图中纹理的颜色和信息丰富度,物体本身的非朗伯材质导致的高光效果也会掩盖物体本身真实的纹理色彩,如何恢复环境光照模型以及物体表面材质系数,从而还原场景真实的纹理颜色信息,这些都是我们未来的研究方向.

致 谢 审稿专家提出了宝贵意见,新加坡新科劲力有限公司的王英杰博士和 Soh Ling Min 博士对此工作给出了建议,杭州深粉象数字科技有限公司提供了家具模型和照片,在此一并致谢!

参 考 文 献

- [1] Seitz S M, Curless B, Diebel J, et al. A comparison and evaluation of multi-view stereo reconstruction algorithms// Proceedings of the CVPR. New York, USA, 2006; 519-528
- [2] Wang L, Kang S B, Szeliski R, Shum H-Y. Optimal texture map reconstruction from multiple views//Proceedings of the CVPR. Kauai, USA, 2001; 347-354
- [3] Baumberg A. Blending images for texturing 3D models// Proceedings of the BMVC. Cardiff, UK, 2002; 404-413
- [4] Lempitsky V, Ivanov D. Seamless mosaicing of image-based texture maps//Proceedings of the CVPR. Minneapolis, USA, 2007; 1-6
- [5] Goldluecke B, Aubry M, Kolev K, Cremers D. A Super-resolution framework for high-accuracy multiview reconstruction. International Journal of Computer Vision, 2014, 106(2): 127-191
- [6] Niem W, Broszio H. Mapping texture from multiple camera views onto 3D-object models for computer animation// Proceedings of the International Workshop on Stereoscopic and Three Dimensional Imaging. Santorini, Greece, 1995; 99-105
- [7] Gal R, Wexler Y, Ofek E, et al. Seamless montage for texturing models. Computer Graphics Forum, 2010, 29(2): 479-486
- [8] Eisemann M, De Decker B, Magnor M, et al. Floating textures. Computer Graphics Forum, 2008, 27(2): 409-418
- [9] Zhou Q, Koltun V. Color map optimization for 3D reconstruction with consumer depth cameras. ACM Transactions on Graphics, 2014, 33(4): 1-10
- [10] Kraevoy V, Sheffer A, Gotsman C. Matchmaker: Constructing constrained texture maps. ACM Transactions on Graphics, 2003, 22(3): 326-333
- [11] Zhou K, Wang X, Tong Y, et al. TextureMontage. ACM Transactions on Graphics, 2005, 24(3): 1148-1155
- [12] Tzur Y, Tal A. Flexistickers: Photogrammetric texture mapping using casual images. ACM Transactions on Graphics, 2009, 28(3): 1-10
- [13] Zhang G, Jia J, Wong T T, Bao H. Consistent depth maps recovery from a video sequence. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2009, 31(6): 974-988
- [14] Furukawa Y, Ponce J. Accurate, dense, and robust multiview stereopsis. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010, 32(8):1362-1376
- [15] DeLong A, Osokin A, Isack H N, Boykov Y. Fast approximate energy minimization with label costs. International Journal of Computer Vision, 2012, 96(1): 1-27
- [16] Perez P, Gangnet M, Blake A. Poisson image editing. ACM Transactions on Graphics, 2003, 22(3): 313-318
- [17] Zhou K, Synder J, Guo B, Shum H-Y. Iso-charts: Stretch-driven mesh parameterization using spectral analysis// Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing. Nice, France, 2004; 45-54



JIANG Han-Qing, born in 1984, Ph. D. His main research interests include video-based 3D reconstruction and segmentation.

WANG Bo-Sheng, born in 1991, Ph. D. candidate. His main research interests include video-based 3D reconstruction.

ZHANG Guo-Feng, born in 1981, Ph. D., associate professor. His main research interests include camera tracking, 3D reconstruction, video segmentation and editing, and augmented reality.

BAO Hu-Jun, born in 1966, Ph. D., professor, Ph. D. supervisor. His main research interests include computer graphics, 3D vision, virtual reality and augmented reality.

Background

The problem studied in this paper belongs to texture mapping in computer graphics, especially video-based multi-view texture mapping of 3D models, which is an important problem in photorealistic graphics.

To recover the complete texture information of a 3D scene, multi-view images are usually needed. However, how to seamlessly fuse textures of multiple images into a single object model is a difficult problem worthy of study. Although some existing methods can automatically achieve photorealistic texture mapping using multi-view images, most of them turn out to handle simple objects. There are seldom any methods which can deal with large-scale texture mapping of natural scenes. Besides, current methods usually cannot successfully handle 3D models with complicated structures. Some other works provide semi-automatic interactive tools to combine user assistances for texture mapping. However, requirements of complex user interactions make these tools impractical.

This paper presents a novel multi-view texture mapping method, which effectively recovers seamless photorealistic texture maps of 3D models from multi-view captured videos.

The proposed method can not only handles mapping problem of large-scale texture data, but also performs well with complicated structured objects which often appears in natural scenes. The recovered texture maps are automatically stitched from multi-view frame images with the best resolutions and viewing directions, and a texture fusion process is performed to eliminate all the stitching seams. For large-scale texture data, an iterative texture fusion framework is proposed to solve memory limitation problem.

This work is supported by the National Science and Technology Support Plan Project (2012BAH35B02), the National High Technology Research and Development Program (863 Program) of China (2013AA013802), the National Natural Science Foundation of China (Nos. 61272048, 61103104), A Foundation for the Author of National Excellent Doctoral Dissertation of PR China (No. 201245), and Singapore Technologies Dynamics PTE Ltd.

Therefore, achievement of this work can contribute to the photorealistic texture mapping study.