

面向网络应用的三维树木模型简化方法研究

董天阳 姚佳洁 纪磊

(浙江工业大学计算机科学与技术学院 杭州 310023)

摘 要 在网络环境下进行三维树木模型的绘制与传输,不仅对模型逼真度有着较高要求,还更需保证网络交互的实时性。由于现有的基于几何或图像的三维模型简化方法无法满足网络应用对三维树木的模型细节和网络传输速度的要求,因此提出了一种面向网络应用的三维树木模型简化方法。该方法对树木枝干部分的骨架节点进行提取,并采用非均匀的骨架简化策略对树木模型的枝干部分进行简化;此外,还利用纹理树冠方法对树冠部分进行简化,通过采用凸包纹理模板贴图重构树冠,使得重构后得到的模型与原始模型具有很好的相似度。实验和应用结果表明,面向网络应用的三维树木模型简化方法在保证视觉质量的基础上,能进一步减少模型文件的存储容量,从而提高树木模型的绘制效率及其在网络中的传输速率。

关键词 三维树木,模型简化,纹理模板,视觉感知

中图法分类号 TP391 文献标识码 A

Research on Three-dimensional Tree Model Simplification for Web Applications

DONG Tian-yang YAO Jia-jie JI Lei

(School of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China)

Abstract Construction and transmission of 3D tree models under network environments require not only fine model fidelity, but also high efficiency of real-time interaction. Because of the existing geometry-based or image-based 3D model simplifications unable to satisfy the model details and storage capacity requirements of 3D tree models, a novel simplification method of 3D tree model for web applications was proposed. This method extracts the skeleton nodes of branches of tree model, and simplifies braches by using non-uniform skeleton simplification method. Furthermore, it simplifies crowns of tree models by using crown textures and reconstructs them with convex hull based texture mapping, so as to improve the similarity of simplified 3D tree models. The applications show that this method can reduce storage capacity of model files, and improve the transmission efficiency of network on a basis of excellent visual quality.

Keywords 3D tree, Model simplification, Texture template, Visual perception

1 引言

由于自然界中树木的形态各异、种类繁多、拓扑结构复杂,若想构建出视觉效果逼真的三维树木模型,其建模过程非常繁琐且十分耗时。同时,当前网络带宽、延时和拥塞等网络因素和移动终端硬件条件等因素的限制,严重影响了三维植物模型网络化交互操作的实时性。因此,在保证三维树木模型视觉感知的基础上进行模型有效简化,对于三维树木模型在网络中的应用有着重大意义。

现有的树木模型简化方法主要包括:

(1) 基于网格的简化方法。2005年, Lee等人^[1]将网格显著度思想应用到网格简化过程中,得以在简化过程中更好地保持特征;2010年, Tarini等人^[2]提出了一种增量四边形网格简化方法,以局部操作来最大限度保持网格模型的表面细节质量。2011年,浙江大学的金勇等人^[3]根据变分网格逼近表示所定义的全局误差能量,提出一种局部贪心优化算法,该方

法具有计算量较小、效率较高等优点,能够有效地应用于几何造型系统中。此外,2013年,偶春生等^[4]提出了一种特征保持的网格简化新方法,在保证曲面特征细节的同时,具有良好的简化特性。2014年, Morig等^[5]提出了一种多层次的网格简化模型,对于不同复杂度的模型均有效果,尤其对于具有尖锐特征和平坦面的模型效果更好。

(2) 基于叶片裁减的简化方法。虽然网格简化方法能够快速生成保持原始模型外观的简化模型,但对于像树叶这种由离散网格构造的模型,简化效果并不理想。针对上述问题,2002年 Remolar^[6]提出了一种树叶简化算法(FSA),其简化过程以叶子为单位,反复地从所有候选的叶子中选取其中的一对叶子,合并这对叶子并生成新叶,同时新生成的叶子与原始叶子保持外观上的相似性。2011年,中国科学院的 Bao Guan-bo等人^[7]提出了匹配树叶纹理的叶片模型简化算法,并利用顶点缓存和树模型裁剪实时生成大规模的森林场景,平衡了视觉效果和模型复杂度。Bao Guan-bo等还在次年提

本文受国家自然科学基金(61202202)资助。

董天阳(1977—),男,博士,副教授,硕士生导师,CCF会员,主要研究方向为虚拟现实、计算机图形学,E-mail: dty@zjut.edu.cn;姚佳洁(1990—),女,硕士生,主要研究方向为计算机图形学;纪磊(1987—),男,硕士生,主要研究方向为计算机图形学。

出新的基于叶序层次细节 (LOD) 的建模方法^[8], 在呈现树叶模型的纹理的同时, 保证模型复杂度和视觉效果两者的平衡。此外, Lee 等人^[9] 提出了基于视点的可见性裁剪方法, Gumbau 等人^[10] 将随机排序方法应用到树叶裁减过程中。这些方法通过考虑植物器官层次、叶片密度、遮挡等因素, 将“裁减”引入植物叶片模型简化中, 从而提高了模型简化效率。

(3) 基于几何结构的模型简化方法。2003 年, 秦学英^[11] 等提出了 Quasi-3D Tree 方法, 能够高质量地绘制不同光线方向下的树木。Zeng Ji-guo 等^[12] 提出了一种从单一图片完成 3D 树木模型重构的方法。该方法是通过像素点的分析, 将给定的一张图片重构出 3D 树干模型, 通过对输入图像中树木的近似像素体的密度计算来确定最初的树枝位置, 然后使用植物生长规则和分支的角度产生几何形状的树枝骨架。

(4) 几何结构和图像绘制相结合的方法。基于几何结构的简化方法可以方便高效地对枝干部分进行简化, 但是对树叶的简化效果不够理想。而基于网格和基于叶片裁剪的简化方法虽然能够大幅度对树木模型进行简化, 但是对简化模型的近距离视觉观察效果仍不够理想。因此, 研究人员试图在树木模型的简化过程中使用基于几何和图像特征的混合方法, 在简化程度和视觉质量上找到一定的平衡。2011 年, Malheiros 等人^[13] 通过使用一个基于简单规则的 L 系统和一组预设的纹理, 采用基于几何和纹理 billboards 技术相结合的手段实现根据视点由远到近呈现不同树叶浓密程度的树木模型。Livny 提出了基于 Lobe^[14] 表示的快速 3D 树木建模方法, 该方法是在图片中提取树木点云信息, 并通过树冠重构和基于凸包的纹理树冠重构方法来表示三维树木模型, 然而其存储容量在整体建模过程中是递增的, 因此该方法并不适用于面向网络应用的三维树木模型简化。

鉴于此, 本文根据三维树木生长、几何规则和模型的内涵拓扑结构, 提出了一种面向网络应用的植物模型简化方法。该方法采用对模型枝干和树冠分别简化的策略, 对模型的树干部分采用基于骨架模型简化方法, 并利用基于圆柱体、梯形平面和插值的模型重构方法进行树木模型重构。对于树冠部分采用基于凸包纹理模板贴图的简化方法, 即通过计算树冠的凸包来表示树冠的整体拓扑结构, 然后在已有的纹理模板库中寻找符合标准的树枝模板进行填充。得利于三维树木模型内涵拓扑结构特有的视觉感知特征, 树木模型在简化的同时也保留了视觉感知特性。此外, 面向网络应用的三维树木模型简化方法还对模型采用不同等级的简化策略, 结合模型视觉重要区, 对视觉重要部分的简化程度适时降低, 从而保证高质的视觉效果。综上所述, 本文提出的面向网络应用的植物模型简化方法能够在保持树木的外部视觉感知特征的前提下, 最大程度简化树木模型, 提高网络应用中的三维树木模型的传输和绘制效率。

2 面向网络应用的三维树木模型简化过程

考虑到不同的三维树木模型给人以特有的视觉感知特性, 本文根据三维树木模型的视觉重要区域对模型进行拆分, 并在简化过程中采用不同级别简化策略。图 1 为三维树木模型简化整体流程, 根据树木模型整体拓扑结构, 采用将树干和

树冠分别简化重构的策略。树冠部分采用基于纹理模板贴图的简化方法, 即在提取出来的所有树冠纹理图片中选取一些具有代表性的纹理模板, 在树冠重构时用较少的纹理图片描述整个树冠轮廓。枝干部分采用基于骨架的简化策略, 枝干骨架能够描述树木模型的整体拓扑结构, 在提取出来的骨架模型的基础上可以利用不同的重构方法进行树干重构。最后将两部分相结合形成最终的简化模型。

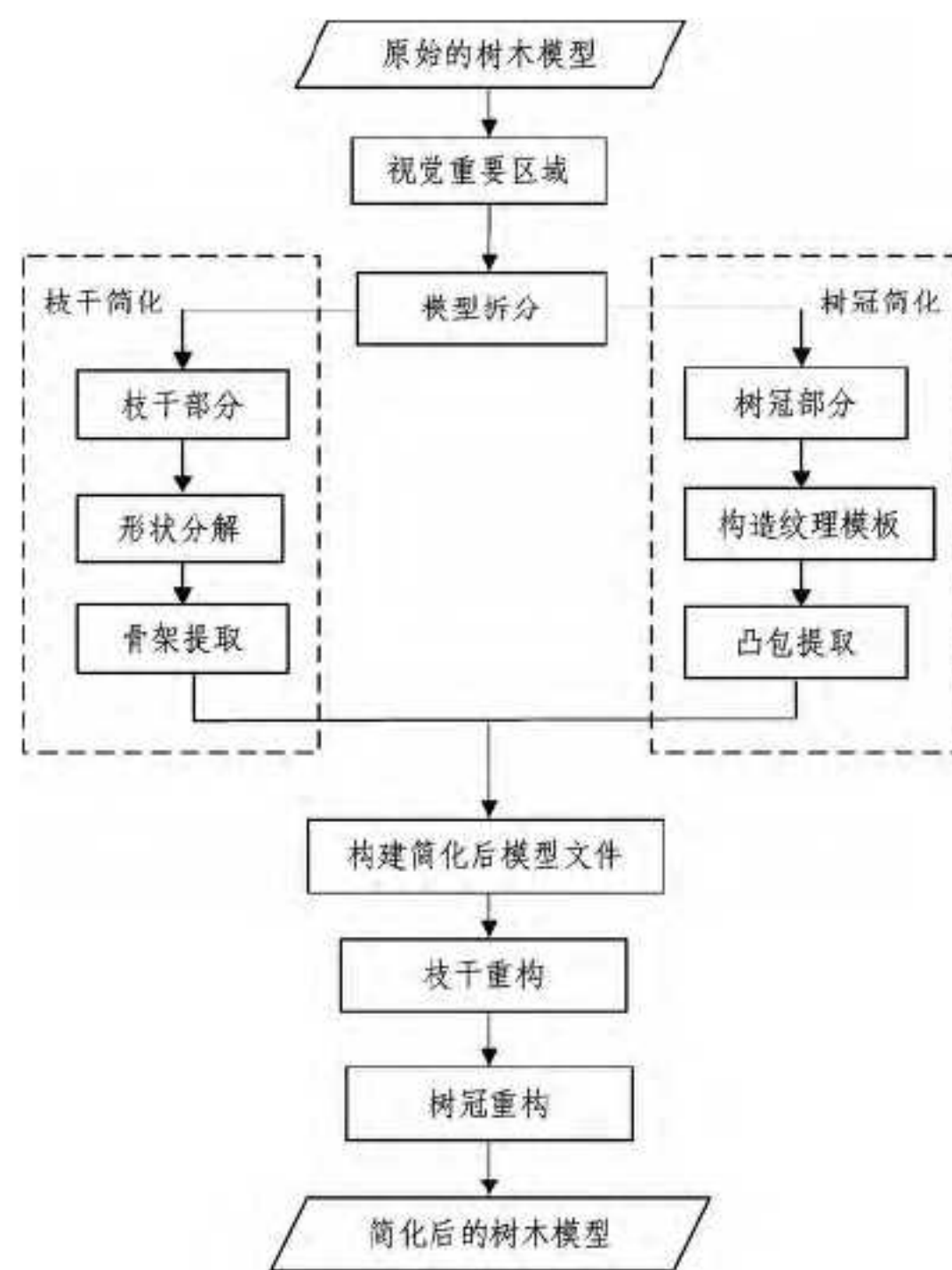


图 1 树木模型简化流程

3 基于骨架的三维树木模型枝干简化方法

骨架作为一棵树的基本骨骼结构, 描述了模型枝干部分的几何形态和拓扑结构。三维模型骨架结构广泛地应用在计算机图形学等相关领域, 如模型简化、模型检索、模型修复、模型分割、模型重构、形变控制和树木等复杂结构物体的三维重建。本文根据树木模型枝干的几何信息, 利用骨架提取算法提取其骨架节点, 并对三维树木模型进行重构, 实现用尽可能少的几何信息来表示复杂的模型枝干, 从而达到简化树木枝干的目的。

3.1 枝干简化过程

面向网络应用的三维树木模型的枝干简化过程主要包括枝干模型的骨架提取和基于骨架的枝干重构两个部分, 如图 2 所示。图 2 中骨架提取部分包括: 划分枝干模型拓扑层次、基于视觉重区域的非均匀简化和骨架组合 3 个步骤。首先, 本文根据三维模型的内部拓扑结构将模型划分主干、主枝、分支和枝条等子结构; 其次, 根据树木模型的视觉数据对枝干进行非均匀简化并提取骨架; 最后, 将所有分支提取出来的骨架组合成完整的模型骨架。模型重构部分包括: 基于圆柱体、基于梯形平面和基于线性插值 3 种模型重构方法。根据上述方法, 能够有效地提取出三维树木模型的基本骨架结构。此外, 本文引入基于视觉重要区域的非均匀分层简化策略, 使得提取出来的骨架能够保留原树枝模型的显著特性, 从而实现对树枝模型骨架的最佳提取。

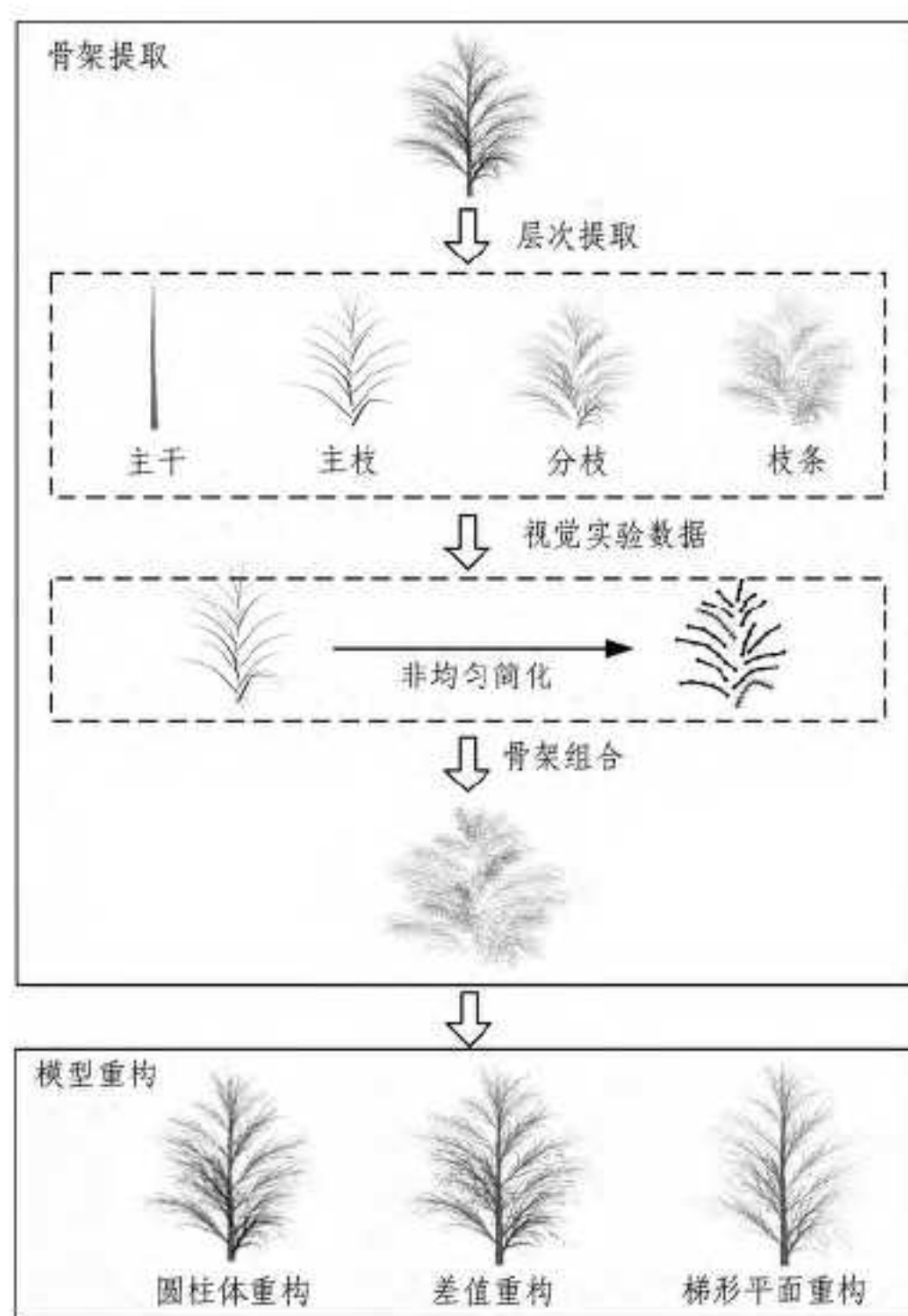


图2 枝干简化流程

3.2 骨架提取

由于三维树木生长、几何规则和模型的内涵拓扑结构,本文采用形状分解方法将三维树木模型枝干划分成不同层次的分支。在求取三维树木模型整体枝干的骨架时,按照相同的方法分别对不同层次的分支计算骨架节点,最后将所有不同层次分支的骨架节点组合成为三维树木模型完整的骨架。利用这种将枝干整体根据不同生长结构划分不同层次分支而分别提取骨架的方法,不仅大大减少了计算骨架的时间复杂度,而且降低了程序运行的空间复杂度。此外,将整体枝干拆分成不同细节分支的骨架提取方法,相比于直接求取模型整体骨架的方法,能够得到更加细致的结果,得到的骨架更能保存原始树木模型的几何和拓扑结构特征。

如图3所示为欧洲高山白蜡树模型经过形状分解的效果图。其中,图3(a)为三维树木原始模型,图3(b)~(e)为形状分解后结构。本文将原始模型划分成四层结构分别如图3(b)~(e)所示。从图中可以看出每一层次结构中包含了多个相互独立的子树枝,其中,在枝干简化时将分别对每根子树枝独立进行最短路径计算。

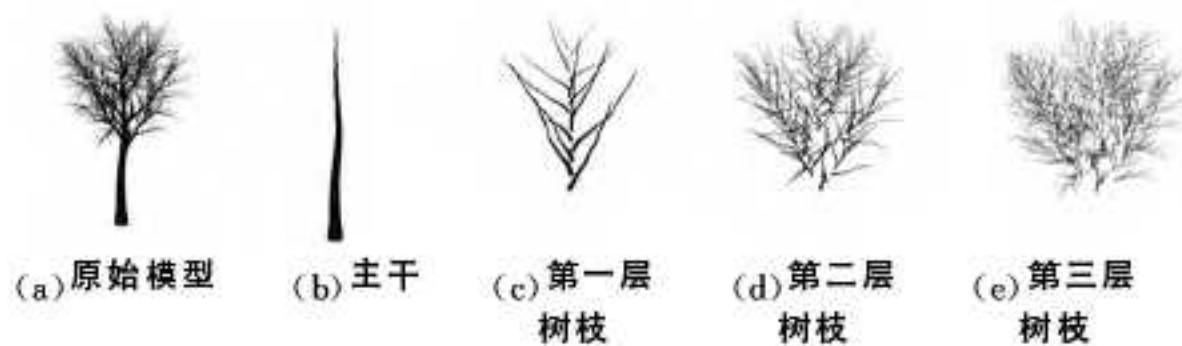


图3 欧洲高山白蜡树模型形状分解

本文在对模型进行形状分解的基础上,提出了均匀和非均匀的两种简化策略的概念。所谓均匀简化策略就是在提取枝干骨架节点时,对于所有层次的树枝采用均匀的分段方法。图4(a)为采用均匀简化策略的枝干简化示意图,该树木模型的枝干被划分为3个层次,图中(1)、(2)、(3)分别代表每个层次的序号。图4(a)中每根树枝上的每个骨架节点之间的距离都是相同的。

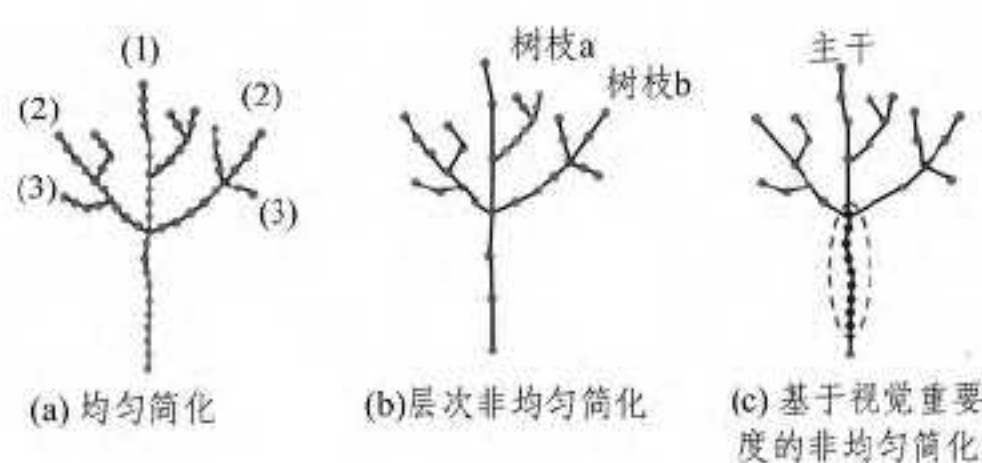


图4 枝干模型轻量化策略

非均匀简化策略的实质是在提取枝干骨架节点时对不同层次、不同树枝,甚至同一树枝的分段是非均匀的。本文的非均匀简化策略包含为两大类:层次间非均匀简化和基于视觉重要度的非均匀简化。图4(b)为枝干采用层次间非均匀简化策略的示意图,其本质是根据不同层次或不同树枝的树枝长度及弯曲程度等因素确定树枝的骨架节点。图4(b)中树枝_a和树枝_b属于同一层次,树枝_a比树枝_b长度短,但弯曲程度却更大。因此,为了能够保持树枝_a的弯曲程度,树枝_a提取骨架时应该具有更加密集的骨架节点。

考虑到不同的三维树木模型具有不同的视觉重要区域,本文还将模型视觉重要区域与非均匀简化策略相结合,为了尽可能降低简化的失真度,保持原始枝干模型的几何拓扑结构和视觉重要性,本文对模型中重要度较高的枝干进行较低程度的简化,而对于重要度较低或被遮盖的树枝进行较高程度的简化。如图4(c)中以椭圆虚线围成的区域就是枝干的视觉重要区域。因此,在计算该段树枝的骨架时应提取较多的骨架节点,而其他部分的骨架节点则相对稀疏。

基于视觉重要度的非均匀简化策略在保持原始模型视觉感知效果的基础上,对视觉重要和视觉非重要区域采取不同等级的简化标准,使得树木模型能够很好地保持原始模型的结构、视觉特性,从而有效地提高枝干模型的简化程度。

3.3 树干模型重构

目前,针对骨架模型的三维树木模型重构有比较多的方法,它们的重构效率和效果各不相同。本文采用3种较为常见的方法分别对简化后的三维树木模型枝干部分进行重构,分别为基于圆台、基于梯形平面和基于线性插值3种方法。

为了能够充分对比这3种重构方法的重构枝干的效果,本文以金莲花模型为例,采用以上3种重构方法进行重构,效果对比如图5所示。

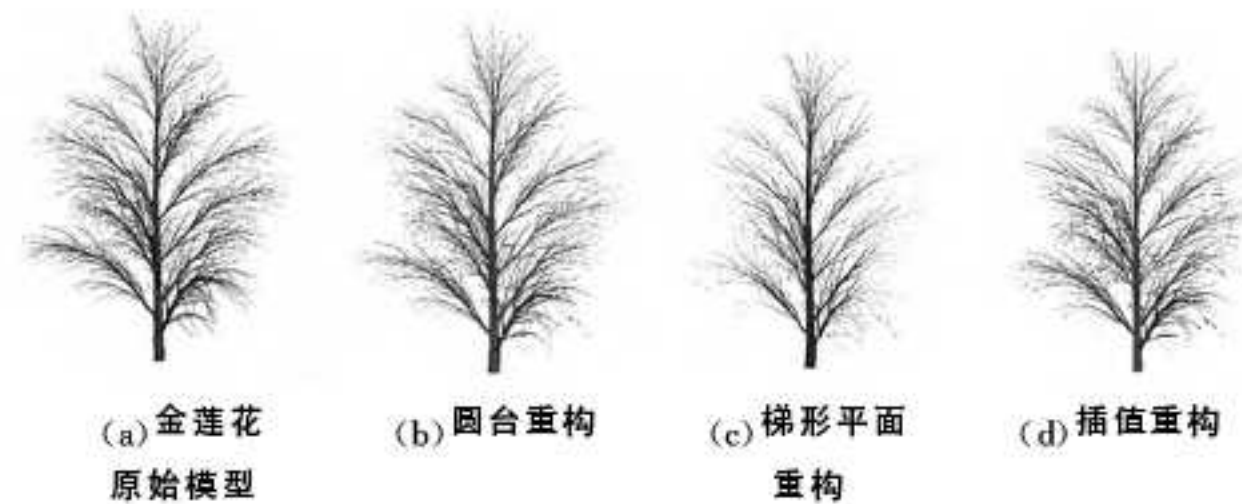


图5 3种模型重构对比图

其中图5(a)为金莲花原始模型,图5中(b)~(d)分别为利用圆台、梯形平面和插值重构的枝干模型。对比这些图片能够发现,采用基于圆台和插值重构方法得到的简化枝干模型与原始模型视觉上具有较高的相似度,采用基于梯形平面重构方法得到的简化枝干模型视觉效果缺乏立体感,因此视觉效果一般。

为了说明上述的简化和重构方法在保持树枝拓扑结构和外观形态的前提下,能有效减少模型的几何数据量,提高模型

的绘制效率,本文在对 4 种不同模型利用以上方法进行简化和重构时,对简化前后模型的存储空间、绘制时间和视觉质量 3 个方面进行对比,并采用 VC++ 和 OpenGL 实现了简化的三维树木模型枝干部分重构,对树木模型简化前后的绘制时间进行计算。计算机的配置如下:Microsoft Windows 7 操作系统,CPU 为 Intel(R) Core(TM) i3-2120 CPU 550@3.30 GHz,2048 MB (DDR2 SDRAM) 内存,AMD Radeon HD

6465 Graphics 显卡。
表 1 是树木模型简化前后存储容量和重构时间的统计信息,对比表 1 中 4 个原始树干模型文件与经过骨架提取后模型文件可以发现,三维树木枝干模型的骨架大小大约为原始枝干模型的 1/6。然而在骨架提取的基础上进行插值计算后的三维树木枝干模型的骨架大小大约为原始枝干模型的 1/15。

表 1 简化及重构信息对比

树种	文件大小			绘制时间			
	原始树干模型(MB)	骨架模型(kB)	插值(kB)	原始树干模型(s)	圆柱体(s)	梯形平面(s)	插值(s)
欧洲高山白蜡树	2.85	396	187	0.065	0.115	0.025	3.1891
冬青树	6.03	967	393	0.124	0.277	0.034	7.057
阿尔卑斯金莲花	6.10	958	346	0.119	0.284	0.033	6.55
庇里牛斯山栎	1.14	198	77.4	0.038	0.075	0.024	1.317

如表 1 所列,基于圆柱体重构的原始树干模型的绘制时间大于基于梯形平面的原始树干模型的绘制时间。这是因为在利用圆柱体重构枝干模型的过程中需经过大量的平移、旋转计算。同时,可以看出经过插值计算后基于圆柱体重构枝干模型的绘制时间最长。但是,考虑到在网络传输过程中文件传输的时延较大,因此在一定情况下可以忽略基于圆柱体和插值重构树干模型所带来的时间增量。本文通过计算枝干模型的原始图像与简化图像的显著性图^[15],进行简化模型视觉质量的比较与分析。图 6 中(a)~(d)分别给出了这 4 种树木模型经过本文的简化、重构方法后与原始枝干模型进行视觉质量分析的对比结果。

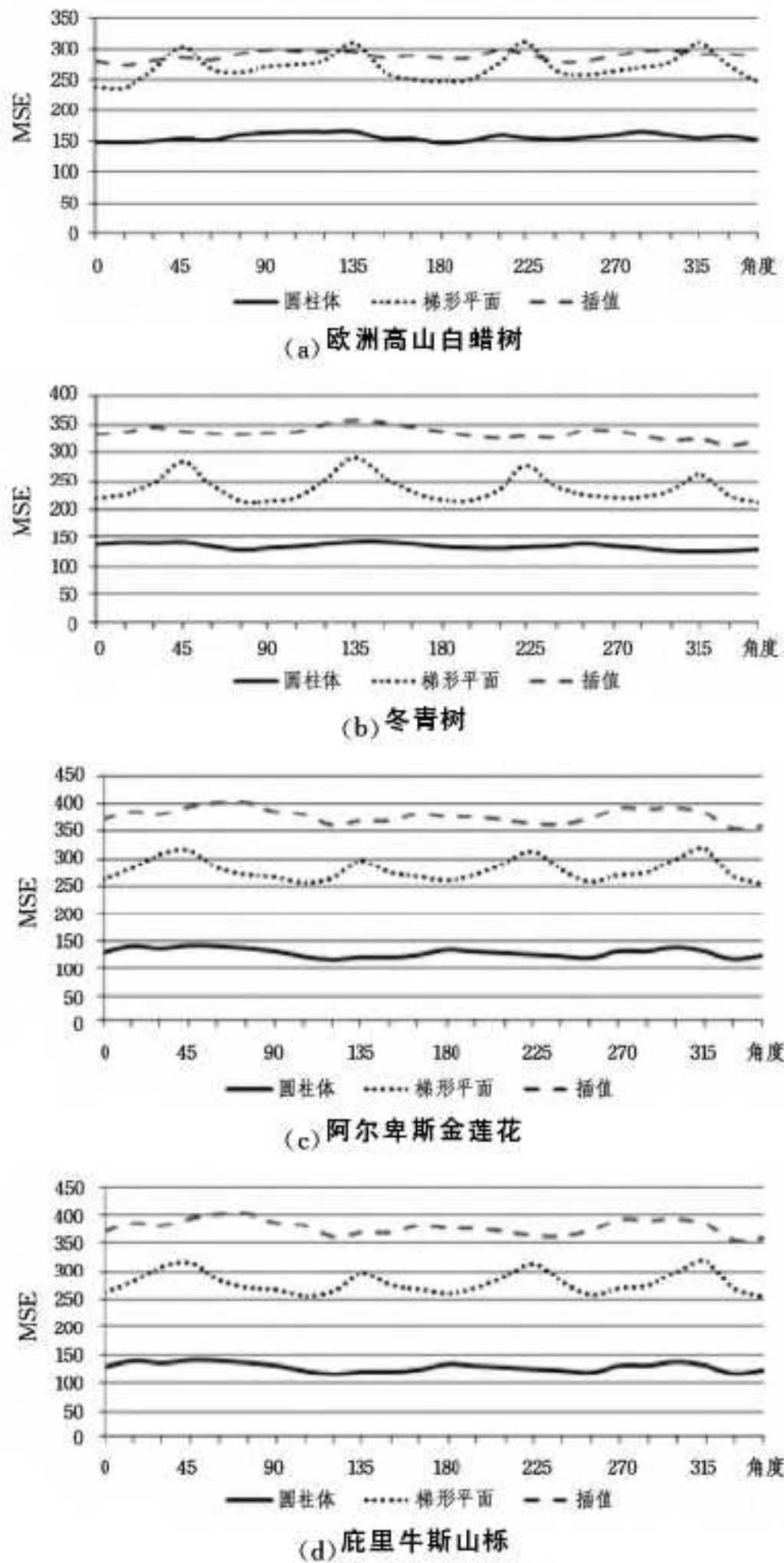


图 6 视觉质量分析

图中,X 轴为环绕枝干一周的视点位置,Y 轴为简化模型的 MSE 值,MSE 值越大则表明简化模型与原始模型的相似度越低。

4 基于纹理树冠的三维树木模型简化方法

基于几何表示的三维树木模型多分辨率简化技术能有效减少模型几何表面细节。但是,基于几何的多分辨率树木模型仍包含数量庞大的面片信息,无法满足基于网络传输绘制的实时渲染和交互要求。然而,基于图像的绘制方法却能很好地解决这一问题。基于图像绘制方法的本质是以极少的纹理图片构造和渲染高度复杂的自然场景,而无需考虑三维模型的几何复杂度。因此,本文采用基于凸包纹理树冠的方法,对三维树木模型的树冠部分进行简化和重构。

4.1 纹理树冠思想

虽然,现实生活中树木品种繁多,各品种树木之间的生长形态各异,但通常三维树木模型枝干部分能经过形状分解而得到不同的层次结构。无论什么品种的树木,其枝干部分通常拥有一个主干,在主干上生长着决定树木拓扑结构的主枝,而主枝之上又繁衍出细小的分支和枝条等。三维树木模型树冠部分由树叶的集合组成,树叶则通过叶脉与树枝相连,共同组成完整拓扑结构的树木模型。三维树木模型枝干部分的层次结构如图 7 所示。

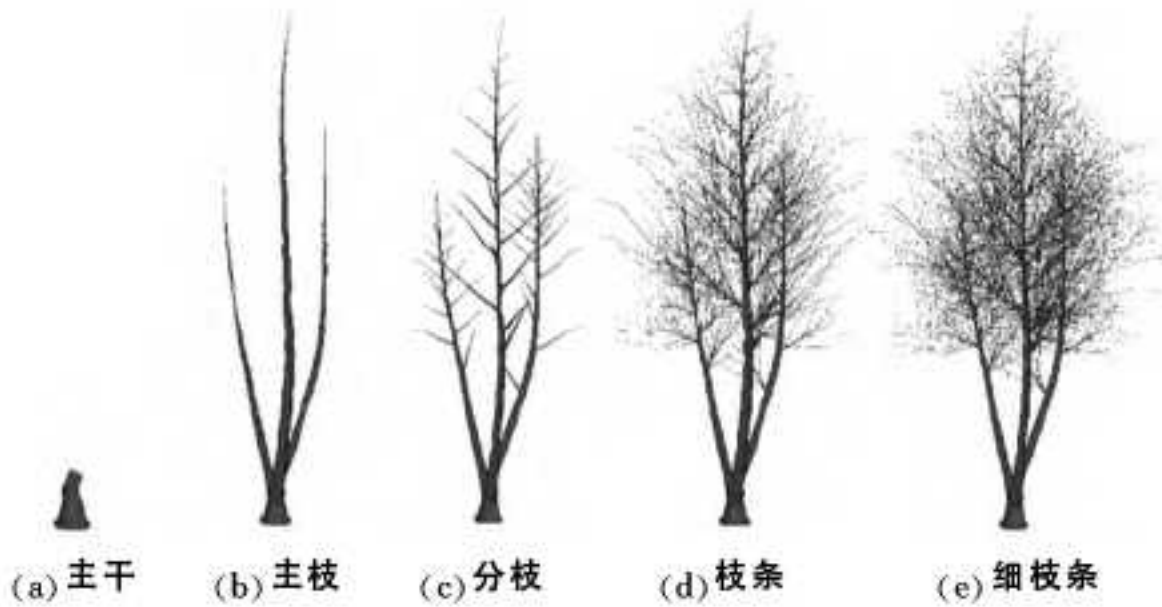


图 7 甜栗模型树干层次结构

图 8(a) 为甜栗模型整体形态,从图中可以看出大部分的枝条和细枝条都被浓密的树叶所遮挡,因此在视觉上也通常被忽略。为进一步简化树木模型,提高树木模型的网络传输和渲染效率,本文在三维树木模型枝干部分重构时,剔除模型中对模型外部拓扑结构影响较低的分支、枝条与细枝条部分,仅采用模型的主干和主枝(见图 7(a)、图 7(b))来表示模型枝

干部分的整体轮廓。然后,采用纹理贴图的方法来表示模型枝干部分的分支、枝条和细枝条以及模型树冠,以完成对三维树木模型树冠部分的简化。模型根据树木的层次拓扑结构划分为不同的聚簇,其基本过程如下:由于树木的每片树叶都通过叶脉与树枝相连,因此本文首先确定每片树叶与树枝的连接点,如图 8(b) 的甜栗模型树叶,图中的左侧圆点就是甜栗树叶与树枝的连接点。

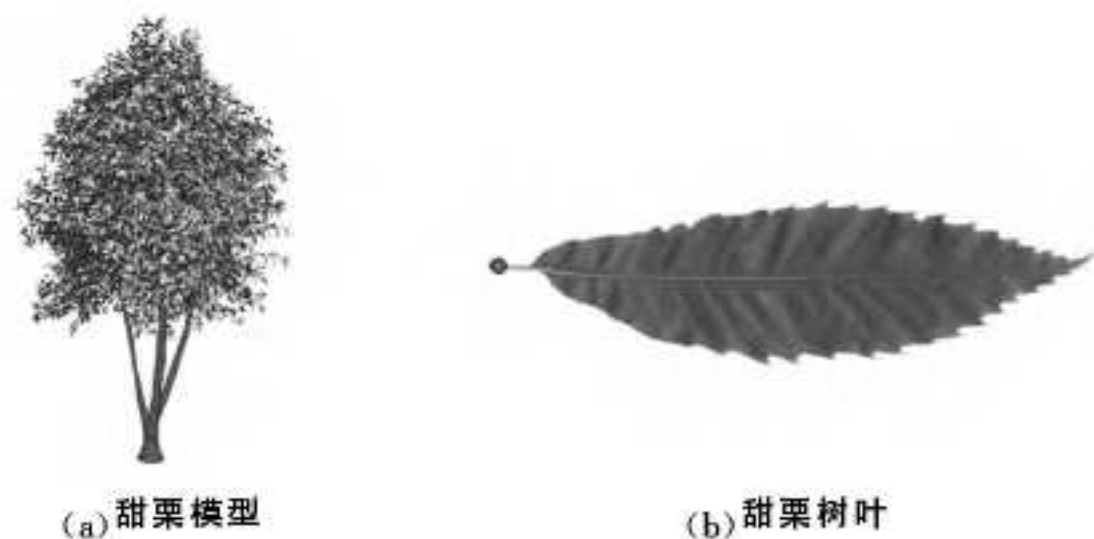


图 8 甜栗模型和甜栗树叶

分别计算每片树叶的连接点与所有主枝的生长点之间的距离,如式(1)所示:

$$dis(l, b) = \sqrt{(l_1 - b_1)^2 + (l_2 - b_2)^2 + (l_3 - b_3)^2} \quad (1)$$

其中, l 为树叶的连接点, b 为主枝的生长点。树枝的生长点就是在树枝上的叶片与树枝相连的点,即树叶生长在树枝上的点。

在划分聚簇时,首先选取距离每片树叶连接点最近的生长点,即 $dis(l, b)$ 最小的主枝,则这片树叶属于该主枝。随后,遍历树冠中所有的树叶,以确定每片树叶所生长的主枝。接下来,树冠中的每片树叶根据所生长的主枝,生成不同的树叶集。如图 9(a) 的甜栗模型,其树叶采用上述方法以模型的 3 根主枝生成了 3 个树叶聚簇,如图 9(b) 所示;树干的分枝、枝条等也被划分为聚簇。

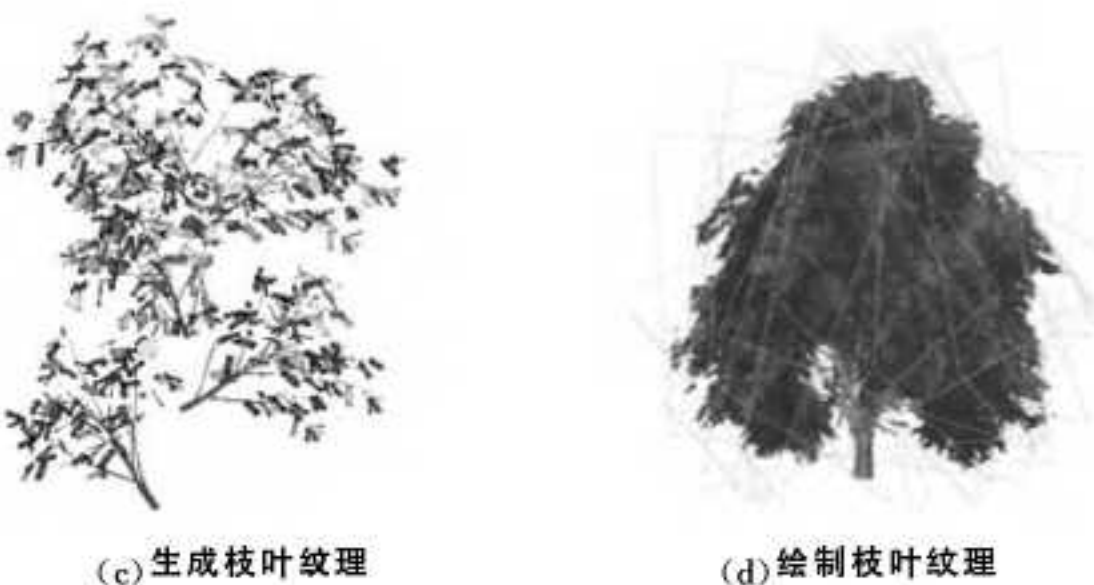
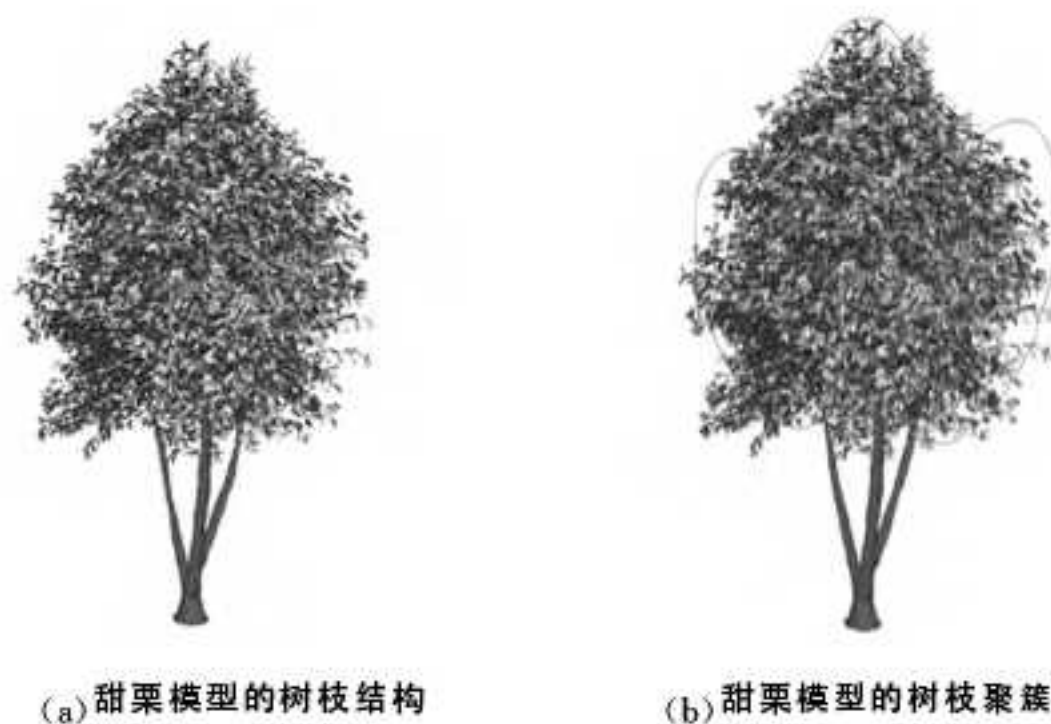


图 9 甜栗模型的枝叶聚簇

最后,根据每一个枝叶聚簇生成当前视点下的纹理图像如图 9(c) 所示,在树干重构时用纹理图像代替树叶、分枝聚

簇的几何表示,并绘制在对应的主枝上,如图 9(d) 所示。利用纹理贴图思想,分别依次对欧洲高山白蜡树、冬青树和庇里牛斯山栎树进行纹理贴图实验,如图 10(a)、图 10(b) 所示,图 10(a) 中由上至下依次为 3 种树的树木模型原图,图 10(b) 中依次为 3 种树的纹理贴图效果。实验中我们发现,图 10(b) 的贴图效果中,3 种树分别包含 266、481 和 184 个树冠纹理,这些纹理的总容量与原模型文件的容量相近,所以仅利用普通贴图思想并不能达到十分理想的三维简化效果,同样,在网络化应用当中的绘制和传输效率上也不能有明显的提高。

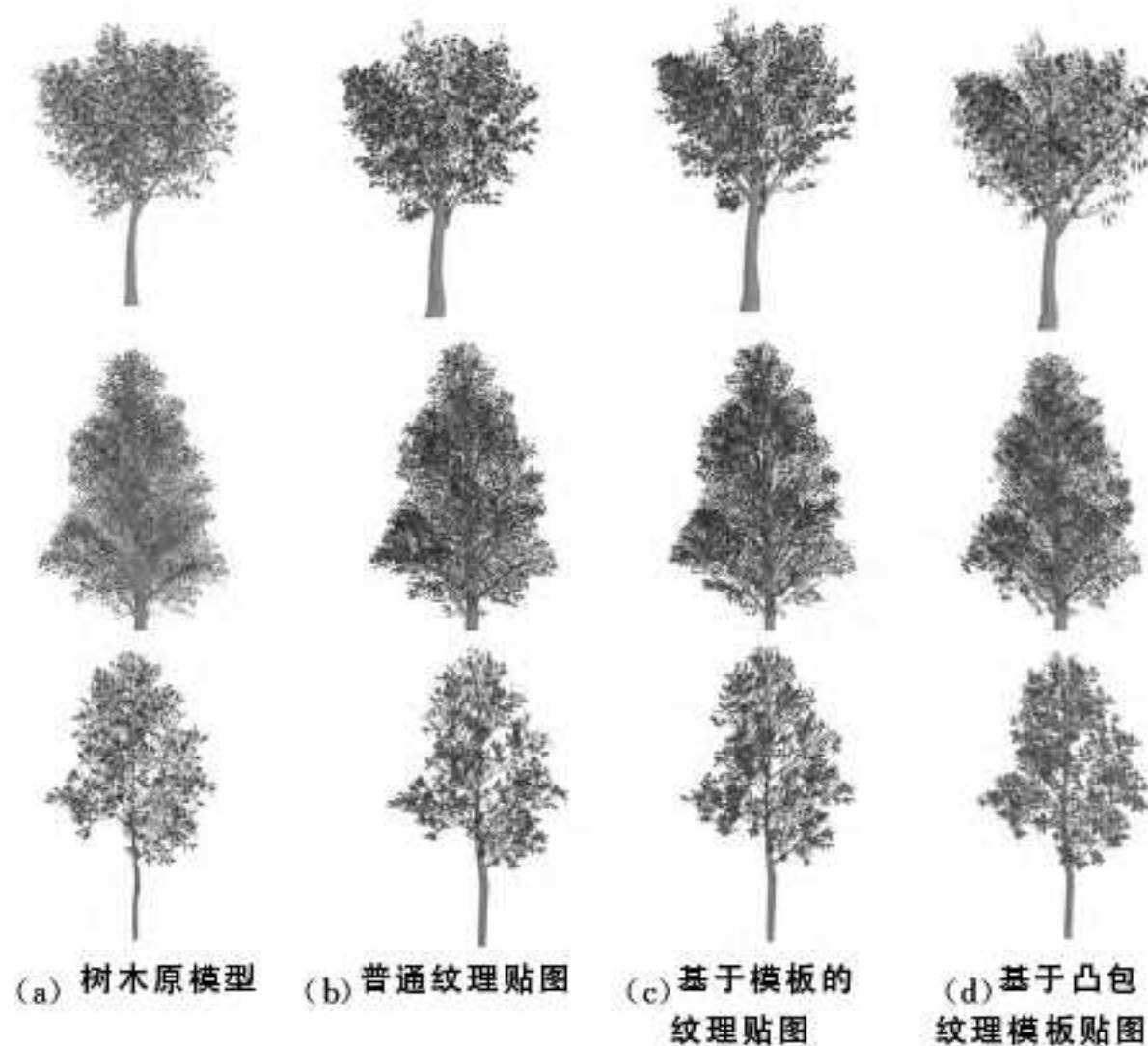


图 10 欧洲高山白蜡树、冬青树和庇里牛斯山栎树的贴图简化效果展示

为了改善上述问题,我们考虑用一些具有代表性的纹理模板去代替所有纹理进行贴图。首先,根据树枝包含的子树枝个数将所有纹理进行分组,拥有相同数量子树枝的树枝纹理划分到同一组;其次,根据树枝的生长方向将每组内的所有树枝分类,例如,将生长角方向按照每 45° 划分为一类;然后,在划分好的每一类纹理中选取一个纹理作为模板纹理;最后,在树冠重构时按照相同的规则选取模板纹理进行贴图。图 10(c) 中从上至下依次为欧洲高山白蜡树、冬青树和庇里牛斯山栎树的纹理模板贴图效果,且分别使用 32、36 和 20 个纹理模板进行贴图。分别对比图 10(a)、(b) 和 (c) 能够发现,图 10(c) 中各树既能够保持原模型的基本拓扑结构,又能减少重构树冠所需要的纹理数量,还能在带宽受限的网络应用中大大提升绘制和传输效率。然而,上述的普通纹理模板贴图重构树冠的方法只是粗略选取相应的树冠纹理模板进行贴图,而忽略了每一个枝叶聚簇所具有的特定形态结构,以至于不能够很好地保持树冠的整体轮廓。

4.2 凸包纹理模板贴图重构树冠

基于普通纹理模板的种种缺陷,本文采用一种基于凸包纹理模板贴图的树冠重构技术。该方法使用枝干模型中的主干、主枝和分支(图 7(a) — (c)) 来表示模型的树干部分,将模型枝干的表示进一步细化到了模型分支结构,使得重构的模型枝干与原始模型相似度进一步提高。同时,本文通过计算枝干模型分支所形成的外部轮廓,以凸包的形式表现出来,然后根据已有的纹理去填充相应的枝叶凸包,以完成对树木模型整体树冠的重构。具体流程如图 11 所示。



图 11 基于凸包纹理模板贴图的树冠重构技术

基于凸包的纹理模板重构方法需要根据树枝的凸包形状选取相应的纹理模板去填充凸包,相对于普通纹理贴图重构方法需要更小的纹理模板,因此能够进一步压缩重构树木模型所需要传输的文件容量。对比图 11 中的原始树枝和重构树枝图片可以发现,利用这种方法重构出来的局部枝叶能够很好地保持原始轮廓结构,因而相比普通纹理模板贴图,这种方法更具可行性。

图 11(d)中从上至下依次展示了欧洲高山白蜡树、冬青树和庇里牛斯山栎树利用基于凸包纹理模板方法贴图的效果,每个模型所使用的纹理模板均为 8 个。相比于普通纹理模板贴图方法而言,基于凸包纹理模板方法不仅减少了使用的纹理模板数量,而且更好地保持了树木整体的外表轮廓,从而使重构树木所需要的时间更短、重构出来的树木与原始模型具有更高的相似度。

此外,基于凸包的纹理模板贴图的树冠重构方法可以利用同一品种树木的纹理模板对相同品种的三维树木模型的树冠进行重构,这样对于同一品种树木简化时,纹理模板只需要传输一次即可,相同品种的三维树木模型在进行树冠重构时可以采用一组相同的纹理模板,这样既可以进一步压缩模型文件容量,又可以减少网络传输时间。如图 13 为庇里牛斯山

栎模型其他三维模型使用相同的纹理模板进行简化的对比结果,图 12 中(a)和(c)是原始模型,图 12 中(b)和(d)是简化后模型。

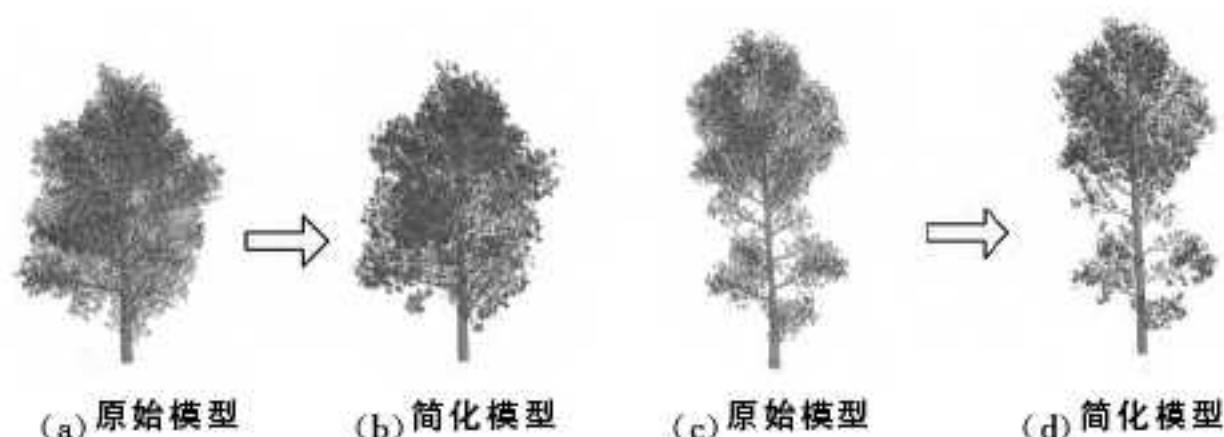


图 12 庇里牛斯山栎其他模型简化示例

5 实验结果和分析

为了验证本文提出树木模型简化方法的有效性,本文采用 VC++ 和 OpenGL 实现了三维树木模型简化重构系统,并对树木模型简化前后的绘制时间、网络传输时间、视觉质量和移动终端效果 4 个方面进行对比和分析。计算机的配置如下:Microsoft Windows 7 操作系统,CPU 为 Intel(R) Core (TM) i3-2120 CPU 550 @ 3.30GHz, 2048 MB (DDR2 SDRAM) 内存,AMD Radeon HD 6465 Graphics 显卡。

5.1 网络传输和绘制时间对比

由于树木模型简化主要面向网络应用,因此网络传输时间也是衡量该方法可行性的一个重要标准。本文还对模型简化前后在网络环境下传输速率和时间进行测试。测试环境为 3M 带宽,理论上在不考虑网络时延和数据包的起始位、停止位、同步字符等非数据字符各方面因素的情况下,理想传输速度为 384kB/s,然而实际速度远远达不到理想速度。综合测试结果如表 2 和表 3 所列。

表 2 原始模型文件信息

树种	文件大小			平均传输速度 (kb/s)	时间		
	原始模型(MB)	原始纹理(MB)	总体(MB)		传输(s)	绘制(s)	总体(s)
欧洲高山白蜡树	21.6	1.8	23.4	302	79.3	0.41	79.71
冬青树	21.2	2.8	25.0	264	96.9	0.29	97.19
庇里牛斯山栎	4.7	3.0	7.7	293	26.9	0.15	27.05

表 3 基于普通纹理模板贴图方法简化后模型信息

树种	文件大小			平均传输速度 (kb/s)	时间		
	树干模型(kB)	树冠模板纹理(MB)	总体(MB)		传输(s)	绘制(s)	总体(s)
欧洲高山白蜡树	562	1.5	2.0	297	6.6	0.15	6.75
冬青树	1200	1.7	2.9	311	9.5	0.17	9.67
庇里牛斯山栎	262	2.5	2.7	265	10.6	0.18	10.78

表 2 所列为原始模型基本信息统计,原始模型的容量包括树干和树冠等全部信息。本文对同一树木模型进行 10 次网络传输,并对总体时间进行统计,最后经过计算求出平均传输速度和传输时间,总体时间为传输和绘制的时间和。从表中的传输和绘制时间可以看出相对于传输时间而言绘制时间可以忽略不计。因此,减少传输时间是增加绘制效率的关键。

表 3 所列为基于普通纹理模板贴图方法简化后的模型信

息统计,简化模型由骨架树干信息和树冠纹理组成。其中,树干模型包括树干骨架数据和树冠纹理贴图所使用的相关数据,纹理树冠为相应的纹理模板。以同样的方法计算出平均传输速度、传输时间和总体时间。

表 4 所列为基于凸包纹理模板贴图方法简化后的模型信息统计,简化模型由骨架树干信息和树冠纹理组成。对比表 3 和表 4 可以发现该方法简化后的文件总容量更低,总体重构时间更短。

表 4 基于凸包纹理模板贴图方法简化后模型信息

树种	文件大小			平均传输速度 (kb/s)	时间		
	树干模型(kB)	树冠模板纹理(kB)	总体(MB)		传输(s)	绘制(s)	总体(s)
欧洲高山白蜡树	501	605	1.1	284	3.9	0.17	4.07
冬青树	1100	525	1.6	305	5.4	0.14	5.54
庇里牛斯山栎	236	807	1.0	287	3.6	0.11	3.71

对比表 2—表 4 中的数据,以冬青树为例,原始模型总容量为 25.0M,基于普通纹理模板贴图方法简化后的模型总容量为 2.9M,基于凸包纹理模板贴图方法简化后的模型总容量为 1.6M。在 3M 网络带宽环境下的传输时间分别为 108.3s、18.5s 和 6.3s,总体时间分别为 96.9s、9.5s 和 5.4s。可见,利用基于凸包纹理模板贴图方法对应用于网络的树木模型进行简化可提高绘制效率。

为了进一步验证利用本文提出的算法简化的三维树木模型在网络化应用中的效率,本文选取同一树种的 20 棵树木进

行网络传输,并在终端进行渲染。对于 20 个原始树木模型需要将其全部传输到网络终端进行绘制,总体时间约为表 2 中的 20 倍。对于同一树种的简化模型,本文应用相同的树冠模板纹理进行贴图,因此传输过程中树冠模板纹理只需要传输 1 次,对于树木骨架和枝叶凸包需要传输 20 次。具体统计信息如表 5 所列。由此可见利用本文的基于普通纹理贴图方法对网络化应用的三维场景中的树木进行简化,可以有效提高绘制效率,并且针对具有大量相同树种的三维场景,此方法更是有效。

表 5 20 个简化模型信息

树种	文件大小			平均传输速度		时间	
	树干骨架模型(MB)	树冠模板纹理(MB)	总体(MB)	(kb/s)	传输(s)	绘制(s)	总体(s)
欧洲高山白蜡树	11.3	1.4	12.7	277	46.9	3.52	50.43
冬青树	24.6	1.7	26.3	298	90.4	4.16	94.56
庇里牛斯山栎	6.6	2.5	9.1	303	30.7	5.84	36.54

5.2 视觉质量对比

为了验证本文提出的三维树木模型简化方法相比其他算法的有效性,本文在 PC 端对同种单株树木分别采用树叶合并算法、树叶随机裁剪算法、基于普通纹理模板贴图方法和基于凸包纹理模板贴图方法进行实验,同时对简化后的模型进行不同视角的对比。将 4 种算法分别应用于欧洲高山白蜡树和庇里牛斯山栎的模型简化后的结果,如图 13 和图 14 所示。

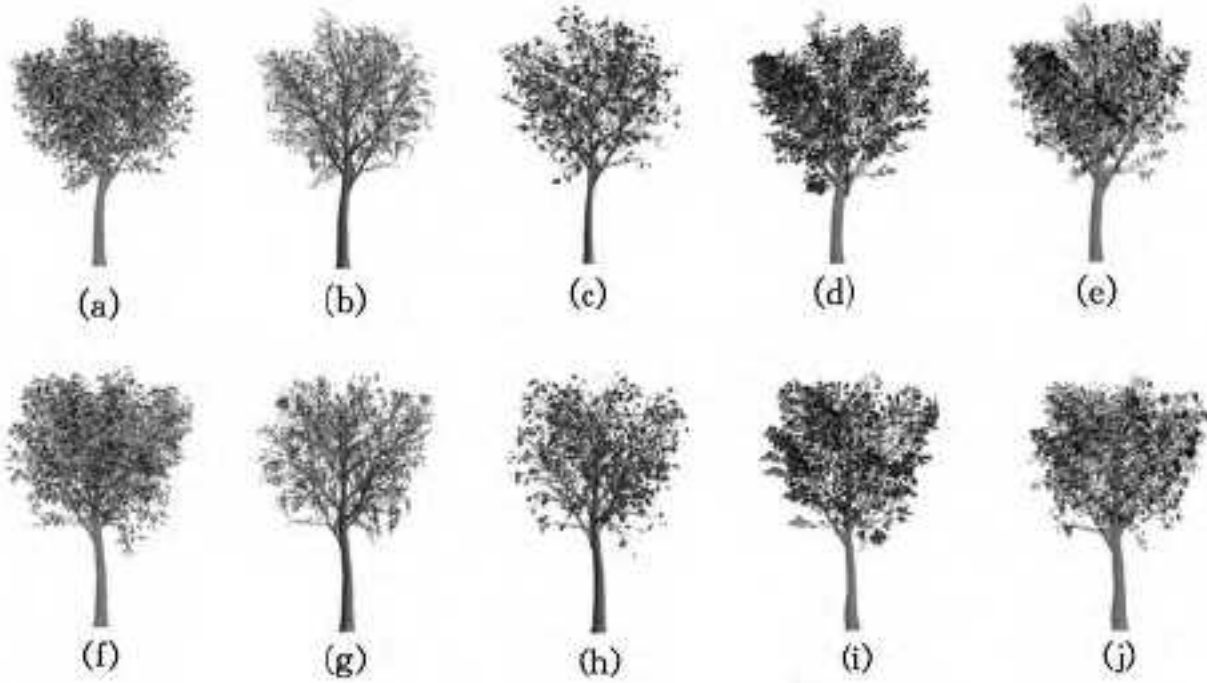


图 13 欧洲高山白蜡树模型的简化结果对比

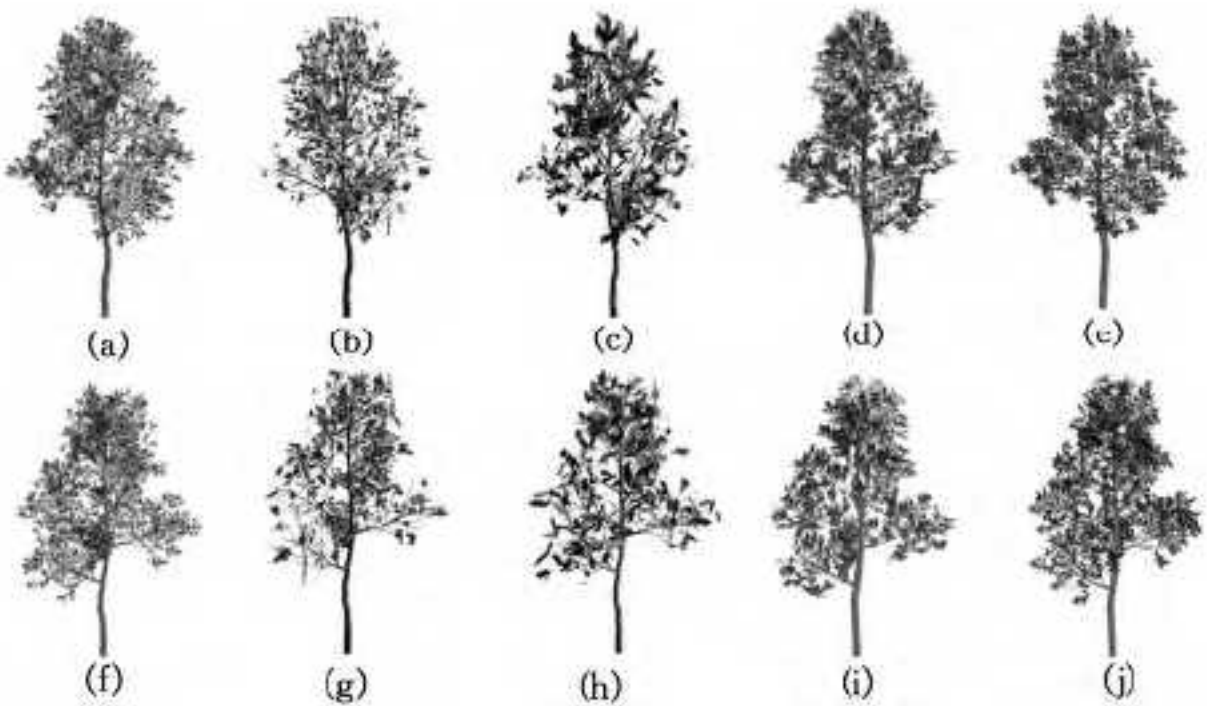


图 14 庇里牛斯山栎模型的简化结果对比

图 13 和图 14 中的(a)~(j)分别代表的是:(a)0°原始模型;(b)0°树叶合并结果;(c)0°随机裁剪结果;(d)0°基于普通纹理模板贴图简化结果;(e)0°基于凸包纹理模板贴图简化结果;(f)180°原始模型;(g)180°树叶合并结果;(h)180°随机裁剪结果;(i)180°基于普通纹理模板贴图简化结果;(j)180°基于凸包纹理模板贴图简化结果。

本文将树木模型原始图像与简化图像的均方差(MSE)作为评定简化后视觉效果的主观依据。两幅图像在视觉效果

上相似,则对应两幅图像的均方差较小。为了能够充分对比利用 3 种算法树木模型简化前后的视觉差异,选取环绕树木模型一周的多个视觉感知方向。利用上述 4 种算法对欧洲高山白蜡树和庇里牛斯山栎模型进行简化,并进行视觉质量分析。在利用本文提出的算法进行简化时,准备了 0°、90°、180°和 270°4 个视觉感知方向的纹理模板,选取距离当前具体感知方向最近的纹理模板进行贴图。视觉质量分析结果如图 15 所示。

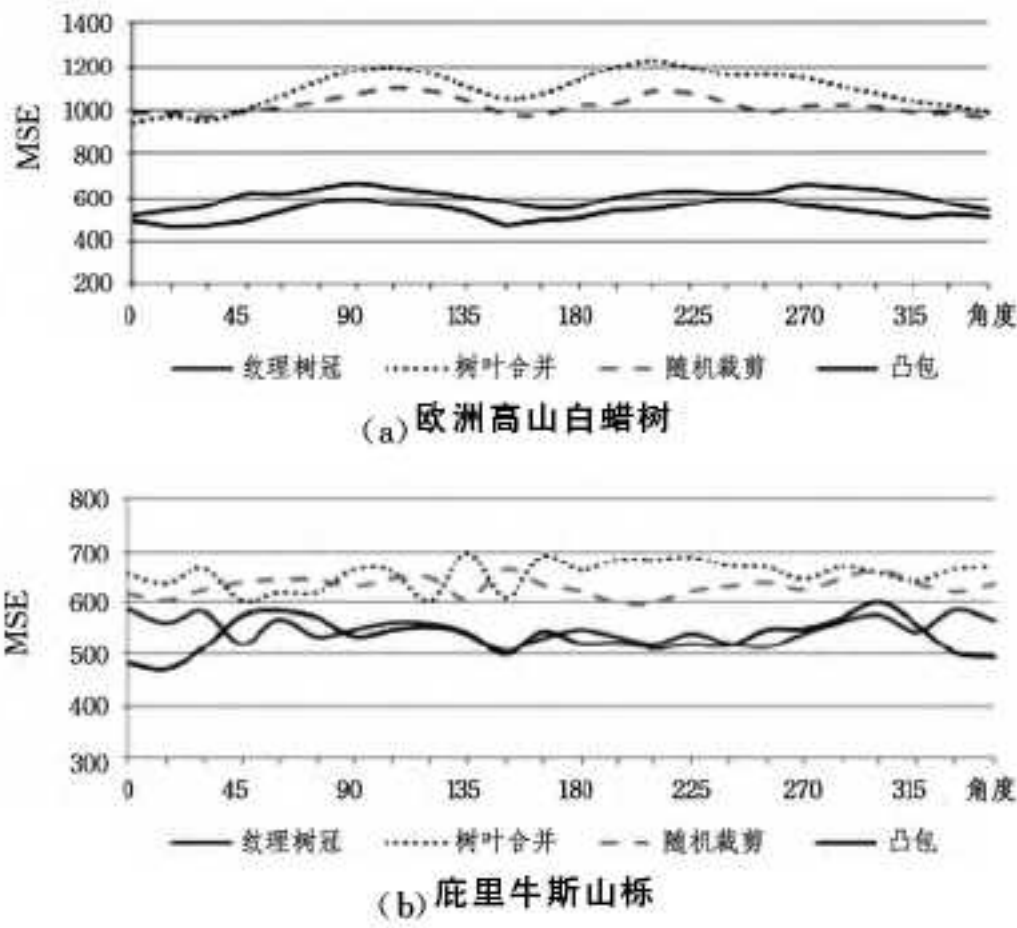


图 15 视觉质量分析

图中,X 轴为环绕树木一周的视点位置,Y 轴为 4 种简化算法生成的简化模型的 MSE 值(相比于原始模型)。从图中的曲线可以看出,利用本文提出基于普通纹理模板贴图简化方法和基于凸包纹理模板贴图简化方法所得模型,在视觉质量的整体效果上相比其他两种方法更好,视觉质量的整体曲线趋于平滑,而且每个视点间的 MSE 值差异较小,在实际应用或漫游过程中不会出现视觉效果突变的情况。基于凸包纹理模板贴图简化方法要比基于普通纹理模板贴图简化方法的视觉质量稍好,同时结合上述绘制时间、存储容量等因素对比,基于凸包纹理模板贴图简化方法具有更高的应用价值。

结束语 本文所提出的面向网络应用的植物模型简化技术,通过基于骨架的枝干简化和基于纹理的树冠简化两部分来完成对植物模型简化。并且,为了更好地保持植物模型的视觉感知特征,在模型简化过程中结合了基于植物模型视觉

(下转第 167 页)



图2 graf(800*640),lena(512*512)特征检测效果图

表1 测试结果数据表

图像大小	特征点数目 (个)	OpenCV 算法 耗时(ms)	本文算法耗时 (ms)	加速比
324*223	604	677	384	1.76
512*512	1109	1571	766	2.05
800*640	2907	3437	1391	2.47
1000*1348	4025	6616	2827	2.34
1600*1200	8413	10803	4676	2.31

由表1数据可看出本文算法实现了一定的加速,大尺寸图像比小尺寸图像取得了更好的效果,说明大尺寸图像更好地发挥了移动GPU的计算能力,掩盖了CPU与GPU之间数据传输的时间,但是图像尺寸继续增大后,加速效果不会进一步提升,这时数据传输开销过大,需要很多耗时。同时在移动平台上的实现较PC平台实现的接近10倍加速也有很大的差距^[10],这主要受限于移动终端的硬件性能。随着未来移动终端性能更加强大,可以取得更强大的加速能力。

结束语 本文实现了SIFT算法在移动平台的GPU加速,并针对安卓平台上的实现做了相应的优化,使得算法运行效率得到了提升。未来可以尝试更多的优化策略,进一步探索算法加速的可能性。同时可以尝试实现更多的图像处理加速算法,使移动GPU加速图像处理应用更加广泛。

(上接第148页)

重要性的非均匀简化策略。本文提出3种简化后的枝干重构方法,并对3种方法的优缺点进行分析对比。另外,还使用纹理模板进行树冠重构,并针对简化后植物模型的存储空间、绘制效率和视觉质量3个方面进行对比。实验数据表明,本文提出的面向网络应用的植物模型简化技术不但具有较高的模型简化率,且简化后的模型具有较高的视觉质量。因此,面向网络应用的树木模型简化方法能够大大提高网络应用中的三维树木模型的传输和绘制效率。

参考文献

- [1] Lee C H, Varshney A, Jacobs D W. Mesh saliency[J]. ACM Transactions on Graphics, 2005, 24(3): 659-666
- [2] Marco T, Nico P, Paolo C, et al. Practical quad mesh simplification[J]. Computer Graphics Forum, Blackwell Publishing Ltd, 2010, 29(2): 407-418
- [3] 金勇, 吴庆标, 刘利刚. 基于变分网格的曲面简化高效算法[J]. 软件学报, 2011, 22(5): 1097-1105
- [4] 偶春生, 张佑生. 基于特征保持的曲面网格简化[J]. 计算机应用研究, 2013, 30(10): 3162-3164
- [5] Morigi S, Rucci M. Multilevel mesh simplification [J]. Visual Computer, 2014, 30(5): 479-492
- [6] Remolar I, Chover M, Ribelles J, et al. Geometric Simplification of Foliage[C]//Proc of Eurographics. 2002
- [7] Bao Guan-bo, Li Hong-jun, Zhang Xiao-peng, et al. Realistic real-time rendering for large-scale forest scenes[C]//2011 IEEE

参考文献

- [1] Lowe D G. Distinctive Image Features from Scale-Invariant Keypoints[J]. International Journal of Computer Vision, 2004, 60(2): 91-110
- [2] Sinha S N, Frahm J M, Pollefeys M, et al. Feature tracking and matching in video using programmable graphics hardware[J]. Machine Vision & Applications, 2011, 22(1): 207-217
- [3] Kayombya G R. SIFT feature extraction on a Smartphone GPU using OpenGL ES2. 0[D]. Massachusetts: Massachusetts Institute of Technology, 2011
- [4] Rister B, Wang G, Wu M, et al. A fast and efficient sift detector using the mobile GPU[C]//2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2013: 2674-2678
- [5] Wang Wei-yan, Zhang Yun-quan, Long Guo-ping, et al. CLSIFT: An Optimization Study of the Scale Invariant Feature Transform on GPUs[C]//IEEE 15th International Conference on High Performance Computing and Communication. 2013
- [6] Wang G, Rister B, Cavallaro J R. Workload Analysis and Efficient OpenCL-based Implementation of SIFT Algorithm on a Smartphone[C]//Global Conference on Signal and Information Processing (GlobalSIP). IEEE, 2013: 759-762
- [7] Wang G. An easy-to use standalone SIFT library easy-to-use standalone SIFT library written in C/C++[CP/OL]. <http://sourceforge.net/projects/ezsift>
- [8] 张樱, 张云泉, 龙国平. 基于 OpenCL 的图像模糊化算法优化研究[J]. 计算机科学, 2012, 39(3): 260-264
- [9] 贾海鹏, 张云泉, 徐建良. 基于 OpenCL 的图像积分图算法优化研究[J]. 计算机科学, 2013, 40(2): 1-7
- [10] 王瑞, 梁华, 蔡宣平. 基于 GPU 的 SIFT 特征提取算法研究[J]. 现代电子技术, 2010, 33(15): 41-43
- [11] International Symposium on VR Innovation (ISVRI). IEEE, 2011: 217-223
- [12] Bao Guan-bo, Li Hong-jun, Zhang Xiao-peng, et al. Large-scale forest rendering: real-time, realistic, and progressive[J]. Computers & Graphics, 2012, 36(3): 140-151
- [13] Lee J, Kuo C C J. Tree model simplification with hybrid polygon billboard approach and human-centered quality evaluation[C]//2010 IEEE International Conference on Multimedia and Expo (ICME). IEEE, 2010: 932-937
- [14] Gumbau J, Chover M, Remolar I, et al. View-dependent pruning for real-time rendering of trees[J]. Computers & Graphics, 2011, 35(2): 364-374
- [15] Qin Xue-ying, Eihachiro N, Katsumi T, et al. Fast Photo Realistic Rendering of Trees in Daylight[J]. Computer Graphics Forum, Blackwell Publishing, 2003, 22(3): 243-252
- [16] Zeng Ji-guo, Zhang Yan, Zhan Shou-yi. 3D tree models reconstruction from a single image[C]//Sixth International Conference on Intelligent Systems Design and Applications (ISDA). IEEE, 2006: 445-450
- [17] Marcelo D G M, Walter M. A hybrid geometry and billboard-based model for trees[C]//2011 Brazilian Symposium on Games and Digital Entertainment (SBGAMES). IEEE, 2011: 17-25
- [18] Livny Y, Pirk S, Cheng Zhang-lin, et al. Texture-lobes for tree modelling[M]//ACM Transactions on Graphics (TOG). ACM, 2011: 76-79
- [19] Itti L, Koch C. A saliency-based search mechanism for overt and covert shifts of visual attention[J]. Vision Research, 2000, 40(10-12): 1489-1506