



中国科学院大学
University of Chinese Academy of Sciences

硕士学位论文

形状引导的过程式植物建模

作者姓名： 刘志浩

学位类别： 工学硕士

学科专业： 计算机应用技术

2021 年 6 月

Shape Driven Procedural Plant Modeling

**A thesis submitted to the
University of Chinese Academy of Sciences
in partial fulfillment of the requirement**

**for the degree of
Master of Science in Engineering
in Computer Applied Technology**

By

Liu Zhihao

June, 2021

中国科学院大学 学位论文原创性声明

本人郑重声明：所呈交的学位论文是本人在导师的指导下独立进行研究工作所取得的成果。尽我所知，除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的研究成果。对论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明或致谢。本人完全意识到本声明的法律结果由本人承担。

作者签名：

日期：

中国科学院大学 学位论文授权使用声明

本人完全了解并同意遵守中国科学院大学有关保存和使用学位论文的规定，即中国科学院大学有权保留送交学位论文的副本，允许该论文被查阅，可以按照学术研究公开原则和保护知识产权的原则公布该论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存、汇编本学位论文。

涉密及延迟公开的学位论文在解密或延期后适用本声明。

作者签名：

导师签名：

日期：

日期：

摘要

植物是现实环境和数字化场景中的重要组成元素。随着计算机图形学技术的日益进步，植物模型被大量应用于电子游戏、影视作品、城镇和园艺设计等场景中。但由于植物通常具有复杂的几何结构，因此如何进行高效逼真的植物建模一直是图形学领域的重要课题。基于规则和数据的植物建模方法往往面临规则参数设置复杂、植物外形不易控制和建模技能门槛高等问题。而现有的交互式植物建模方法虽然支持普通用户设计植物造型，但是大多基于二维草图实现，导致无法推断可靠的三维深度信息。

就公开文献来看，本文在国内外首次将虚拟现实与深度学习技术应用于植物三维建模，并根据不同类型的形状约束提出了三种新的植物三维建模方法。这些形状驱动的植物建模方法的共性是用户只需要关注植物的整体外形特征，而逼真的内部植物细节模型则通过过程式建模算法基于外形约束自动生成。本文的主要贡献如下：

1. 提出了一种二维形状驱动的植物建模方法。该方法结合了深度学习与过程式植物建模技术，通过将从二维推断三维信息的问题转化为图像间的翻译问题，实现了可靠的三维深度估计。首先，该方法使用神经网络从二维的植物轮廓草图和主枝干线条中预测两种特殊的深度图像，并据此恢复植物的三维外部轮廓和主骨架。接着以主骨架为基础，完整的植物模型则通过过程式建模算法在三维轮廓内自动生成。此外，本文还将该方法扩展以解决基于单幅图像的三维植物重建问题。

2. 提出了一种三维骨架驱动的植物建模技术。该方法探索了虚拟现实技术在植物建模中的应用。通过使用沉浸式VR设备，结合三维手势实现了对植物骨架的直接三维控制。该系统允许用户使用手柄直接绘制三维的骨架结构，并且支持对植物的任意部分进行灵活的修改和编辑。这种沉浸式的直接三维交互克服了传统建模软件在建模时的繁琐操作。此外，该方法还提供了画刷等高层语义工具来指定植物骨架的生长趋势，用来批量生成植物枝干。

3. 提出了一种三维轮廓驱动的植物建模技术。该方法结合三维手势，使用更少的交互实现了通过设计植物的三维轮廓来进行交互式植物建模的方法。用

户首先将目标植物的树冠区域拆解为若干简单的轮廓包体并用手势逐个绘制，而最终的植物模型则通过过程式建模方法自动生成。该方法的优势在于可以充分发挥人类的观察力和创造力，即用户可以最大限度地将复杂的植物结构抽象为简单的三维几何形状。此外，该方法支持用户用手势来操作和编辑已经创建好的轮廓形状。

实验证明本文提出的几种形状引导的植物建模方法能够有效帮助用户进行高效的植物造型创作，并可以充分激发用户的想象力和创造力。

关键词：植物建模，深度学习，虚拟现实，形状驱动

Abstract

Plants play an important role in both real and digital scenes. With the development of computer graphics technology, 3D plant models have been widely applied in areas such as video games, films, city planning and landscape design. However, due to the complexity of geometry, it's still challenging to efficiently model high quality plants. Rule-based and data-driven plant modeling methods usually suffer from the problems like complex parameter adjustment, high expertise requirement and little user intervention. Although existing 2D sketch-based approaches support the user to design plant models interactively, they cannot guarantee to generate plausible depth values.

This paper successfully bring the deep learning and virtual reality technology into the plant modeling area. We present three kinds of shape driven plant modeling methods, by which the users only focus on the overall shape of the target plant while the full detailed plant structure is automatically synthesized using procedural modeling algorithm. Our main contributions are as follows:

1. We present a 2D shape driven plant modeling method, which combines the strengths of both deep learning and procedural modeling technology. The key to address the problem of inferring plausible 3D depth is to recover 3D shape information using a deep neural network learned from a set of synthetic plant models. we first employ a cGAN network that takes 2D sketches of the silhouette and main branching structure of the tree, and predicts two depth images one for tree silhouette and one for the trunk skeleton. These depth images are then used to derive the 3D branching structure and the overall shape of the tree for a procedural model that generates the branches, twigs, and leaves within the obtained growth space (3D silhouette). Moreover, we extent the method for reconstructing 3D plant models from a single image.

2. We present a 3D skeleton driven plant modeling method which explores the application of virtual reality technology in plant modeling. It's possible to directly control the plant skeleton by using 3D gestures and immersive VR devices. Our system allows the user to freely sketch the 3D plant skeleton and manipulate any part of the

plant. The immersive 3D interaction overcomes the tedious operation during modeling plants with traditional software. What's more, advanced tools such as brush are provided to specify the growth trend of skeleton for creating a mass of branches.

3. We present a 3D silhouette driven plant modeling method which integrates mid-air finger 3D sketching in VR and automatic branch populating, and offers an ease-of-use and efficient immersive mid-air sketching interface for plant modeling and editing. The user just needs to sketch a few 3D strokes in mid-air to define the envelope of foliage and main trunks. The full 3D plant models are then generated automatically based on these shapes. The main advantage of the method is to give full play to the user's creativity and imagination. Additionally, the user can also modify the shape of plant model by freely deforming the lobes and strokes.

The experimental results demonstrate that our shape driven plant modeling approaches enable the user to design artistic plant models effectively and efficiently.

Keywords: Plant modeling, Deep learning, Virtual reality, Shape driven

目 录

第1章 绪论	2
1.1 研究背景	2
1.2 国内外研究现状	3
1.2.1 植物建模技术	3
1.2.2 深度学习技术	5
1.2.3 虚拟现实建模技术	6
1.3 主要研究内容	6
第2章 过程式植物建模方法	8
2.1 引言	8
2.2 空间资源竞争算法	9
2.3 顶端优势模拟	11
2.3.1 扩展的BH模型	11
2.3.2 优先级模型	12
2.4 植物几何模型构建	13
2.5 本章小结	15
第3章 二维形状驱动的植物建模	17
3.1 引言	17
3.2 基于对抗网络的植物形状预测	17
3.2.1 训练数据的表示和采集	18
3.2.2 条件对抗神经网络cGAN	20
3.2.3 三维形状的恢复和去噪	22
3.3 主枝干和轮廓约束下的植物建模	23
3.4 实验结果与分析	24
3.5 本章小结	29
第4章 三维骨架驱动的植物建模	31
4.1 引言	31
4.2 基于VR控制器的植物骨架控制	32
4.2.1 骨架的三维绘制与编辑	32
4.2.2 骨架画刷工具	34
4.3 实验结果与分析	36
4.4 本章小结	37

第5章 三维轮廓驱动的植物建模	38
5.1 引言	38
5.2 基于手势的植物三维轮廓设计	39
5.2.1 三维轮廓构建方法	39
5.2.2 轮廓形状编辑	42
5.3 实验结果与分析	44
5.3.1 植物建模结果	44
5.3.2 方法对比	47
5.3.3 用户研究	47
5.4 本章小结	49
第6章 总结与展望	50
6.0.1 本文工作总结	50
6.0.2 研究展望	50
参考文献	53
作者简历及攻读学位期间发表的学术论文与研究成果	59
致谢	60

图形列表

2.1 自组织的过程式植物建模的算法流程。	8
2.2 芽在生长时的空间资源竞争。	10
2.3 基于扩展BH模型实现的顶端优势的效果。	11
2.4 扩展BH模型和优先级模型的算法示意图。	12
2.5 采用优先级模型对植物形态的影响。	13
2.6 植物枝干的三维骨架及其最终网格模型。	14
2.7 在不同生长阶段中采用不同参数构建的一株植物模型。	15
2.8 采用过程式植物建模算法创建的不同种类植物模型。	15
3.1 基于二维形状（轮廓和主枝干草图）的过程式三维植物建模的算法流程。	17
3.2 训练图像样本对的示例。(a) 枝干样本对, (b) 轮廓样本对。	18
3.3 将三维轮廓转换为深度图像的过程。	19
3.4 用于恢复深度信息的条件对抗网络结构。	20
3.5 点云去噪以及主枝干构建的过程。	22
3.6 引入生长牵引方向的植物建模效果。(a)输入图像。(b)未使用牵引方向的枝干。(c)使用牵引方向的枝干。(d)重建的完整植物模型。	23
3.7 计算方向场的流程图。(a) 输入图像。(b)初始聚类。(c)二次划分后的聚类。(d)牵引骨架图。(e)最终的方向场。	24
3.8 三维主枝干和轮廓的恢复效果。	25
3.9 从真实照片中重建的三维植物模型。	26
3.10 轮廓边缘图中不同的内部细节对最终植物模型的影响。	27
3.11 用户手绘的线条所生成的植物模型。	28
3.12 同一种基于单幅图像的建模方法(Tan 等, 2008)的结果对比。	29
3.13 同一种基于多视角图像序列的建模方法(Guo 等, 2018)的结果对比。	29
4.1 基于手势交互的三维植物建模系统。	31
4.2 基于手势交互的几种三维骨架控制方法。	32
4.3 骨架三维编辑操作。(a) 直接手势编辑示例。(b) 基于触摸板的精确旋转控制。	32
4.4 直接利用三维手势绘制树木骨架。	34
4.5 使用画刷工具进行骨架绘制。	34
4.6 基于手柄触摸板的建模参数控制。	35

4.7 画刷牵引方向的原理和效果。	36
4.8 组合不同交互方式进行灵活的植物创作。	36
4.9 本章方法与一种基于二维草图的建模方法(Chen 等, 2008)的比较。 ...	37
5.1 基于三维轮廓的植物建模。	38
5.2 本章基于手势交互的建模系统的硬件环境及其界面。	39
5.3 从手绘线条逐步构建封闭的轮廓包体的过程。	40
5.4 从两根线条形成三角网格的策略。顶部为两根闭合线条，底部为一根 开放线条和闭合线条。	41
5.5 用于寻找最佳起始匹配点对的两种策略的比较。	42
5.6 对线条和轮廓包体的编辑操作。	42
5.7 对照盆景植物照片创建的植物模型。	44
5.8 不同的主枝干对最终植物骨架结构的影响。	45
5.9 通过自由摆放轮廓包体进行更灵活的艺术设计。	45
5.10 对特定形状的植物造型的建模结果。	46
5.11 同一种基于二维骨架草图的建模方法(Chen 等, 2008)的对比。	47
5.12 用户填写的调查问卷的统计结果。	48
5.13 在用户研究实验中参与者根据同一个照片重建的植物模型示例。 ...	48

表格列表

3.1 部分植物的重建时间统计（秒）。	27
5.1 部分植物的形状信息及分步骤建模时间统计。	46

表格列表

第1章 绪论

1.1 研究背景

近年来随着虚拟现实（VR）、游戏和影视等产业的发展，用户对高质量的数字化场景提出了更高需求。植物作为自然环境的不可缺少的组成部分，对其进行逼真的建模可以显著提高用户在虚拟环境中的沉浸感和体验感。此外植物建模在城市及园林景观设计、农林业管理、生态环境仿真等领域也应用广泛。

现实世界的植物通常具有错综复杂的枝干结构，并存在大范围的叶片和阴影遮挡。同时植物界也包含着丰富多样的种类，导致树种间纹理图案、细节特征等的表现千差万别，且同种植物在不同的生长阶段也会表现出多样的形态特征。而传统基于CAD软件的手工建模方法由于需要专业的建模知识，重建植物这种高细节对象的效率往往是极低的。因此，探索逼真且高效的三维植物建模仍然是图形学领域的难点问题。为了提高三维建模的效率和质量，研究人员提出了基于规则(Prusinkiewicz 等, 1990a; Honda, 1971)、图像（序列）(Tan 等, 2008; Neubert 等, 2007)、激光点云(Xu 等, 2007; Yan 等, 2009)等方式的植物建模方法。但是这些方法大多面临着参数控制复杂、专业知识要求高、用户干预不够灵活等缺点(杨垠晖 等, 2018)。

关于虚拟现实系统的研究可追溯到20世纪60年代，Sutherland (1965)最先对虚拟现实系统做出了定义，指出VR系统应为图形显示、用户交互、触觉反馈及声音反馈的混合应用，能够模拟现实世界中的人类感知。近年来大量硬件厂商开出了多种消费级头戴显示器（HMD），如HTC Vive、Pico和Oculus等，这些头戴显示器大多配备专门的手柄控制器用以实现手势交互。此外，一些厂商也开发了专用的手势捕捉传感器或者传感手套，如LeapMotion和Cyberglove等。其中LeapMotion基于双目视觉实现了对双手的完全捕捉，可以较为准确的获取手指的细节动作。这些VR系统的出现，使得我们可以基于三维手势开发出更加灵活的交互式系统。

深度学习技术的出现在诸多领域掀起了方法革新的热潮，如计算机视觉、自然语言处理和生物信息学等。一些工作(Girdhar 等, 2016; Sinha 等, 2017)也将深度学习带入了三维重建领域。但是由于三维建模问题自身的复杂性和庞大的

数据计算量，现有方法的重建结果大多为低细节的体素表示，难以满足植物等物体的高质量建模需求。因此深度神经网络在三维重建领域的应用仍面临诸多困难。

本文基于深度学习和虚拟现实技术提出了三种形状驱动的植物建模方法。首先，通过将植物的三维建模问题转化为图像间的翻译问题，本文基于深度神经网络提出了一种二维形状驱动的植物建模算法，并成功将其扩展以解决基于单幅图像的三维重建问题。此外，基于虚拟现实技术，我们还分别开发了由三维骨架和轮廓驱动的交互式植物建模方法，使得用户可以只关注和描绘植物的大致形态特征，而完整的植物细节模型则通过过程化建模算法基于用户指定的形状约束自动生成。

1.2 国内外研究现状

1.2.1 植物建模技术

过程式植物建模： 基于语法或规则的过程式植物建模技术提供了一种有效的方式来创建复杂的植物模型(Deussen 等, 2006)。例如，最初由Lindenmayer (1968)提出的L系统可以通过定义一系列文法规则来描述不同种类植物的形态和几何特征，后来该方法又被加以拓展采用更多新方式来构建更复杂的植物模型(Prusinkiewicz 等, 1990b, 1994, 1996)。另一种被广泛使用的过程式方法通过将植物视作递归的分枝结构来生成植物模型，这些分枝结构可以用少量的简单参数即可描述，如使用分枝夹角和连续枝段的长度比例(Honda, 1971)、隐函数(Bloomenthal, 1985)、分形模式(Oppeheimer, 1986)等。de Reffye 等 (1988)也使用了一个规则的几何来模拟分枝结构，但是这些规则是受到植物生长模型的启发而制定的。Stava 等 (2014)使用了一种目标多边形模型来作为约束，提出了一种逆向的过程式建模方法，而非直接制定生成规则。他们使用马尔可夫链(MCMC) 来估计过程式模型的最优参数以获得与输入形态相近的植物模型。

考虑到环境因素和枝干彼此的影响，空间殖民算法(Runions 等, 2007; Beneš 等, 2009)被提出以生成更加真实的植物模型。Pirk 等 (2012)提出来了一种允许复杂植物模型同环境进行交互的动态植物建模和表示技术。Palubicki 等 (2009)使用自组织建模的概念并通过模拟芽和枝干对空间和光照等资源的竞争，来获得适于环境的植物模型。然后，Yi 等 (2018)通过集成多种植物的生长等式扩展了

这种自组织建模方法。但是这一类方法大多需要专业的植物学知识，并且使用这种不直观的规则对于用户手工调整植物结构来说非常费力。

真实树木重建：为了减少建模负担，研究人员提出了大量的根据真实世界的数据来重建植物模型的方法。激光扫描是一种有效的获取现实世界中植物的几何数据的方式。根据扫描得到的三维点云，[Xu 等 \(2007\)](#)和[Livny 等 \(2010\)](#)首先利用最小生成树来重建树木的三维骨架，然后随机添加小的细枝和树叶来形成树冠结构。[Livny 等 \(2011\)](#)后续又提出了一种基于轮廓包体（Lobe）的树木表示方法来近似输入点云的几何形状，然后通过使用一些预先定义的树枝片段来实例化每个轮廓包体，从而合成完整的植物模型。[Yan 等 \(2009\)](#)、[Raumonen 等 \(2013\)](#)和[Pfeifer 等 \(2004\)](#)提出了一种柱形匹配算法，通过局部地拟合和查找柱状体来实现植物建模。在一些其他的工作中，[Friedman 等 \(2013\)](#)从输入点云数据的小波变换中推断植物的形状语法规则，然后基于这些规则来生成植物模型。[Li 等 \(2013\)](#)则通过捕捉植物的点云序列来分析一个生长中的植物的演化部件。

基于图像的技术通常使用多视角图像序列提取植物的各种几何信息来指导建模过程，例如可视化包体([Shlyakhter 等, 2001](#))、体积表示([Reche-Martinez 等, 2004; Neubert 等, 2007](#))、从SFM算法获得的点云数据([Quan 等, 2006; Tan 等, 2007; Bradley 等, 2013](#))等。[Diener 等 \(2006\)](#)和[Li 等 \(2011\)](#)也试图从视频中重建植物模型。此外，近年来研究人员也相继提出了一些基于单幅图像的植物重建方法。[Tan 等 \(2008\)](#)等要求用户在单幅图像上绘制至少两条笔划来标记树冠和可见主枝干的范围，然后一种生长引擎将在这些二维信息的指导下合成三维的植物模型。[Guénard 等 \(2013\)](#)提出了一种分析合成方法，通过结合单幅图像的信息和一些树种的先验知识来重建三维植物模型。[Argudo 等 \(2016\)](#)提出了一种在使用极少用户交互的前提下用来重建植物模型的完整流程。其核心是通过求解一个能量最小化问题来粗糙地估计树冠区域的形状，然后通过一个简化的SFS方法（Shape from shading）来添加内部的枝干细节。

基于深度学习的植物建模方法也受到了图形学领域的广泛关注。[Huang 等 \(2017\)](#)提出了一种根据用户绘制的草图合成形状的方法，他们使用一个卷积神经网络（CNN）从草图中预测过程式建模的参数。然而这种方法只适用于合成二维的植物。最近，[Isokane 等 \(2018\)](#)使用深度学习技术从多视角图像序列中重建三维植物模型。他们结合了一种使用贝叶斯拓展的图像到图像的翻译技术以及

三维融合技术来获得一个三维枝干的深度概率图。

交互式植物建模：为了方便用户控制植物的形状，一些交互式植物建模方法也陆续被提出。[Okabe 等 \(2005\)](#)基于枝干之间的夹角应尽可能大的假设，设计了一种从二维骨架草图创建三维植物模型的系统。[Chen 等 \(2008\)](#)提出了一种更鲁棒的草图植物建模方法，通过使用一个植物的数据集作为先验知识，利用马尔可夫链来从手绘二维骨架线条中推断植物的三维信息。[Ijiri 等 \(2006\)](#)对L系统进行了扩展，使得植物的生长可以被单个用户绘制的线条所控制。[Longay 等 \(2012\)](#)提出了TreeSketch系统用以在平板电脑上通过多手指交互来创建更具艺术感的植物模型。然而这些方法都是基于二维交互实现的，因此用户难以灵活精确地进行三维植物的艺术创作。

1.2.2 深度学习技术

随着AlexNet([Krizhevsky 等, 2012](#))的提出，深度学习技术在计算机视觉学界引起了广泛的关注，并在图像分类([Simonyan 等, 2015; Szegedy 等, 2015; He 等, 2016](#))，目标检测([Girshick 等, 2014; Girshick, 2015; Ren 等, 2015; Redmon 等, 2016](#))，图像分割([Badrinarayanan 等, 2017; He 等, 2017](#))等领域取得了巨大成功。

图像到图像的翻译技术：所谓图像到图像的翻译是指将属于某种风格和类型域的图像转换到另一种风格或类型域。[Goodfellow 等 \(2014\)](#)最先提出了对抗网络（GAN）的概念并应用到图像生成领域。此后大量研究人员也尝试基于GAN及其变种来解决图像到图像的翻译问题。例如，[Isola 等 \(2017\)](#)基于条件对抗网络([Mirza 等, 2014](#))提出了pix2pix框架，用来从大量成对图像的数据集中学习图像翻译的内在规则。[CycleGAN\(Zhu 等, 2017\)](#)和[DiscoGAN\(Kim 等, 2017\)](#)利用循环一致性损失解决了无需使用成对数据集训练的图像到图像翻译问题。[Choi 等 \(2018\)](#)提出了starGAN实现了使用单个生成器来完成多域映射的图像翻译。

三维物体的深度学习：由于深度学习在图像领域取得了巨大成功，大量研究人员也尝试将深度学习应用到三维物体生成或重建中去。[Wu 等 \(2016\)](#)基于GAN网络来生成尺寸为 $64 \times 64 \times 64$ 的体素结构的三维物体。[Girdhar 等 \(2016\)](#)提出了一种TL-embedding网络，该网络包含一系列卷积层和全连接层，用于从二维图像中预测尺寸为 $20 \times 20 \times 20$ 的三维体素。[Rezende 等 \(2016\)](#)等改进了cGAN网络用来从二维图像中重建更准确的三维体素结构。然而，体素是一种精度很低

的三维物体的表示形式。[Lun 等 \(2017\)](#)等通过多视图的卷积网络从多视角的二维草图中重建三维点云。[SurfNet](#)则通过深度残差网络来直接重建三维的网格表面([Sinha 等, 2017](#))。然而这些方法仅仅适用于重建一些人造的规则物体, 如汽车、桌子、床和椅子等。相比之下, 植物模型的结构则更为复杂, 没有连续光滑的表面, 并且植物通常由大量杂乱的树叶和枝干组成。

1.2.3 虚拟现实建模技术

随着消费级虚拟现实设备的发展和普及, 使得用户可以直接使用三维交互来绘制或者建模。目前, 大量基于沉浸式虚拟现实设备的三维建模工业软件被开发出来。例如, [ShapeLab \(2018\)](#)和[OculusMedium \(2016\)](#)提供了多种雕刻操作来精确地编辑网格形状。另外一些系统([Schkolne 等, 2001](#); [GravitySketch, 2018](#); [Keefe 等, 2001](#))则允许用户通过手柄控制器直接刷出连续的三维表面。但是上述系统由于需要精确地三维控制, 均要求用户有专业的建模技巧, 并且绘制复杂物体的所有细节也是非常耗时的。[TiltBrush \(2018\)](#)软件以及[Rosales 等 \(2019\)](#)提出的SurfaceBrush系统支持用户绘制大量的三维线条来覆盖物体的表面, 流形的网格表面将基于这些三维线条自动构建。然而用户难以使用线条来准确描述物体表面的形状特征, 因此该类方法难以构建精细的三维模型, 同时大量的绘制操作也会导致用户的疲劳。同时, ([Arora 等, 2017](#))指出由于人类在深度感知上的不精确以及空间定位能力的缺乏, 使用手柄控制器来进行精确的三维绘制是较为困难的。为了简化交互操作并提高用户的使用体验, [Arora 等 \(2018\)](#)混合了相对精确的二维草图绘制和相对粗糙的三维手势交互, 设计了SymbiosisSketch软件以实现相对精准的几何建模。然而, 所有这些基于VR的建模方法均无法满足植物这种形态复杂的物体的建模需要。

1.3 主要研究内容

通过将深度学习和虚拟现实技术同过程式植物建模算法融合起来, 本文提出了三种形状驱动的植物建模方法:

- **二维形状驱动的植物建模:** 我们利用深度学习技术提出了一种基于二维形状的植物三维重建方法。其核心是通过一个深度神经网络从二维轮廓图和枝干线条中恢复植物的大致三维形状, 包括三维轮廓和主骨架。然后完整的植物模型在这些形状的约束下由过程式植物建模算法自动生成。此外, 我们还成功

地将该方法进行扩展来实现从单幅图像中重建植物模型。

- **三维骨架驱动的植物建模：**该方法作为虚拟现实技术在植物建模领域的初次探索，通过借助VR头戴显示器和手柄控制器，从而基于手势操作实现了植物骨架的直接三维交互。该系统允许用户在三维空间中直接绘制骨架线条，同时支持用户灵活地编辑植物的任意子树结构。此外，我们也设计了画刷等高层语义的交互工具来约束植物骨架的生长趋势，从而自动生成大量枝干。

- **三维轮廓驱动的植物建模：**我们提出了一种在虚拟现实环境中使用手指交互来描述植物的三维轮廓和主枝干，然后结合过程式建模获得高质量植物模型的方法。该方法充分发挥了人类观察和抽象事物的能力，用户在交互过程中只需要关注植物的大致轮廓形状并将树冠结构描述为几个简单的轮廓包体。与骨架驱动的建模相比，该方法需要的交互量更少，每一个轮廓包体只需要用几笔简单的线条即可描述。此外，该系统也提供了更多的操作方式以支持用户在灵活地摆放和编辑三维轮廓。

三种形状驱动的建模方法以过程式植物建模为基础，

本文包含六个部分，具体的章节安排如下：

第1章为绪论部分。主要讨论了植物三维建模领域的研究背景与研究现状，以及深度学习和虚拟现实等技术的基本情况和实际应用，同时简要介绍了本文的主要工作内容。

第2章介绍了过程式植物建模算法，并详细探讨了如何模拟植物生长过程中的资源竞争和顶端优势等现象。为后续章节介绍形状驱动的植物建模做铺垫。

第3-5章依次介绍了本文提出的三个形状驱动的植物建模方法，即二维形状驱动、三维骨架驱动、三维轮廓驱动的植物建模。

第6章对全文进行了总结，归纳了本文工作的主要贡献和创新性意义，并展望了未来的研究工作。

第2章 过程式植物建模方法

2.1 引言

过程式植物建模是一种自组织的建模算法，用户通过给定一组植物的形态参数，算法便会自动根据这些参数生成相应树种的三维植物模型。为了便于在后续章节中介绍形状驱动的建模，本章先对过程式植物建模这一共性问题进行讨论。常见的过程式植物建模方法主要采用L系统(Prusinkiewicz 等, 1990a, 1996)和空间殖民算法(Runions 等, 2007; Palubicki 等, 2009)等方式实现。L系统通过定义一系列文法规则来表达不同的树种特征，但是构建这样的文法规则非常费时并且需要专业的植物学知识。因此本文采用了一种基于空间资源竞争的植物建模算法(Palubicki 等, 2009)，该方法无需定义大量的规则，只需要通过少量的参数控制便可以生成多种不同树种的植物模型。

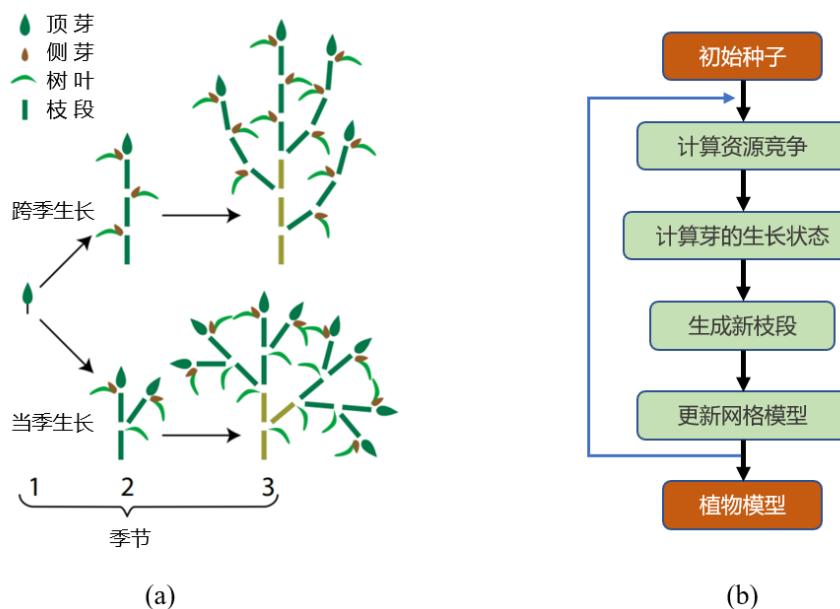


图 2.1 自组织的过程式植物建模的算法流程。

Figure 2.1 The process of self organizing procedural plant modeling.

首先介绍将要出现的几个术语。我们将植物的结构描述为一种多层次的树状组织结构。结点（Node）是指组成枝干的顶点，可以附着多枚叶片。一个枝段（Internode）表示由两个结点所组成的一小段枝干。芽分为顶芽和侧芽，分别位于一条枝干的顶端或两侧。顶芽在生长过程中由于顶端优势通常具有优先

权。

芽可能有四种不同的命运：产生新的枝段、产生花朵、保持现状或者丢弃。新枝段可以是连续不断的生成也可以是有节奏的生成。受到不同季节变换的影响，温带植物通常是有节奏的生成。例如在春季新枝段的序列通常是以单个分支冒出的形式生长。图 2.1(a)描述了芽在生成新的枝段序列时的两种可选形式。连年生长的枝段可以线性生长，单轴分支结构，这种方式多出现侧枝跨季生长的情况。另外，在当季生长的情况下，枝干的生长也可以是多分枝形式的，这种情形下的植物大多没有明显的主干。

图 2.1(b)介绍了自组织的过程式植物建模的基本流程。从初始的种子点开始，计算当前所有芽的空间等资源的分配情况。再根据顶端优势和资源配置，决定所有芽的命运。接着，对于处于活动状态的芽产生新的枝段。最后根据当前的树木结构构建新的植物几何结构。整个过程可以不断迭代从而获得不同年龄的植物模型。

2.2 空间资源竞争算法

空间殖民算法模拟了在植物的生长过程中，芽为了生长出新的枝段而互相争夺空间等资源的现象。空间资源采用一系列均匀分布的空间点（Markers）来表示，一个芽的生长状态将受到其所能争夺到的空间资源总量的影响。空间殖民算法中的芽有如下两个重要的参数：

- **独占区**（Occupancy zone）：是以芽为中心的半径为 ρ 的球形空间，该范围内的空间点被该芽所独占，记为OZ。在图 2.2(a)中用灰色球形表示。
- **探测区**（Perception volume）：是从芽发出的一个锥形的探测体，用于争夺空间点来作为芽生长的空间资源，其半径为 r 夹角为 θ ，并记为PV。在图 2.2(a)中用红色扇形表示。

图 2.2(a)描述了空间殖民算法的最基本形式。在植物的每轮生长过程中，首先将位于芽A的独占区内的所有空间点都删除以表示这部分空间被芽所征用。然后芽将争夺位于其锥形探索区内的空间点，若探索区内有可使用的空间点，则芽A下一步的生长方向可以计算为：

$$V_{opt} = \frac{1}{N(A)} \sum_{i \in N(A)} \text{normalize}(P_i - P_A) \quad (2.1)$$

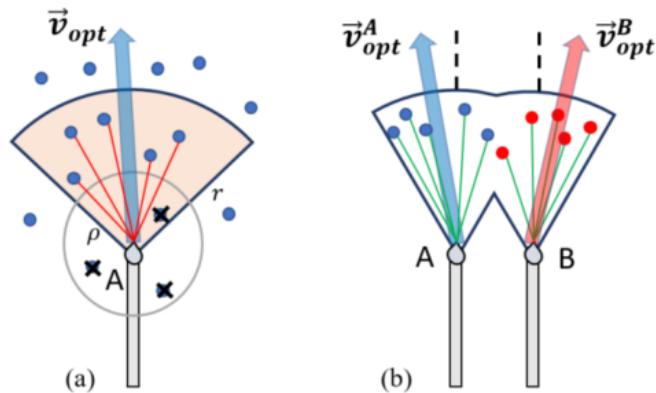


图 2.2 芽在生长时的空间资源竞争。

Figure 2.2 Space colonization for buds.

即生长方向等于每个位于探测区内的空间点与芽坐标所形成的归一化向量的平均方向，式中 P_A 为芽A的三维坐标， $N(A)$ 表示位于芽A探测区范围内的空间点集合。对于多个芽的探测区重叠的情况，如图 2.2(b)所示，我们将位于重叠区域的空间点分配给距离最近的芽。这种策略可以有效避免新生长的枝段靠的太近而造成的枝段交叉。

阴影传播：阴影传播是上述空间殖民算法的一种替代方案(Bornhofen 等, 2009)，该方法采用光源作为竞争的资源对象，并用一种粗糙的估计方式来计算每个芽的曝光量。空间首先被划分为若干体素组成的网格。绑定在每个体素 (i, j, k) 上的阴影值 s 初始化为0。对于一个芽A所在的体素 (I, J, K) 将向下传播一个金字塔状的半影区域。这些被芽A的阴影区域所影响的体素索引为 $(i, j, k) = (I \pm p, J - q, K \pm p)$ ，其中 $q = 0, 1, \dots, q_{max}$ 且 $p = 0, 1, \dots, q$ 。由于芽A的出现，每个被影响的体素的阴影值 s 将会增加 $\Delta s = ab^{-q}$ ，式中 $a > 0$ 和 $b > 1$ 为用户定义的参数。此时，另一个位于体素 (i, j, k) 内的芽B的曝光量Q将计算为

$$Q = \max(C - s + a, 0) \quad (2.2)$$

式中 C 是一个表示完全曝光的常量，增加的项 a 相当于使得芽不会对自身产生阴影遮挡。此时，芽的生长方向 V_{opt} 可以用阴影值的负梯度方向表示，或者在所有位于芽探测区内且临近体素 (i, j, k) 的周围体素集合中选择具有最低阴影值的体素作为其生长方向。

生长方向 v_{opt} 代表了根据资源竞争算法所计算出来的最优生长方向，枝干沿着这个方向生长时可以获得最多的资源配置。为了获得更加丰富的植物形

态，枝干的最终生长方向将由芽原始方向 $v_{default}$ 、最优生长方向 v_{opt} 和重力方向 $v_{gravity}$ 共同决定，即：

$$v_{final} = \alpha v_{default} + \beta v_{opt} + \gamma v_{gravity} \quad (2.3)$$

式中 α 、 β 和 γ 为三个方向的权重参数，重力方向 $v_{gravity} = (0, 1, 0)^T$ 控制植物在纵向上的生长趋势，当 $\gamma < 0$ 时，新生成的枝干会呈现出向下垂钓的形态。

2.3 顶端优势模拟

在植物学中顶端优势现象通常指：在植物的生长过程中对顶芽和侧芽的资源通常是不均匀分配的，由于资源总量通常是一定的，因此顶芽通常得到比侧芽更多的资源因而生长的更加旺盛，而侧芽生长则受到一定抑制。本节讲介绍空间和光照等环境资源对芽的生长状态的影响。通过调整顶端优势模拟的参数，植物可以呈现出如图 2.3 所示的不同植物形态。通常当侧芽分配资源较多时，植物会更多表现为灌木形式；而当顶芽获得资源更多时，则呈现出有明显主枝干的乔木形式。



图 2.3 基于扩展BH模型实现的顶端优势的效果。

Figure 2.3 The effect of apical control with the extended BH model.

2.3.1 扩展的BH模型

BH模型(Borchert 等, 1984)最初是一种通过控制生长资源在芽中的调度分布来控制枝干生长程度的内源性机制。对于自组织的植物生长模拟，我们可以扩展并引入BH模型来使用芽接收的光照资源来指导资源的分配。算法包括两个逆向的过程：

• **自顶向下资源累积:** 首先光照资源照射到每个芽上并被吸收, 然后自顶向下地沿枝干骨架的路径积累到根部。如图 2.4(a)所示, 每一个枝段所存储的光照资源量 Q 来自于其附属的的主枝干积累的光量 Q_m 和侧枝积累的光量 Q_l , 即 $Q = Q_m + Q_l$ 。所有的光照资源最终积累到植物根部。

• **自底向上资源分配:** 根部资源的累积值 Q_{base} 决定了在第二步自底向上资源分配时的资源总量 v_{base} 。经过生化作用, 植物可分配的资源总量变化为 $v_{base} = \alpha Q_{base}$, 式中 α 是一个比例系数, 默认为2。假设到达一个结点的资源量为 v , 后续分配给该结点所延续的主分枝的资源量为 v_m , 分配给其侧枝的资源量为 v_l , 则该处的资源配置情况可以用下面的等式描述:

$$v_m = v \frac{\lambda Q_m}{\lambda Q_m + (1 - \lambda) Q_l}, \quad v_l = v \frac{(1 - \lambda) Q_l}{\lambda Q_m + (1 - \lambda) Q_l} \quad (2.4)$$

参数 $\lambda \in [0, 1]$ 控制了资源配置的偏向程度。 $\lambda > 0.5$ 时表示资源更倾向于分配给主枝干, $\lambda = 0.5$ 表示没有偏向, $\lambda < 0.5$ 表示资源配置偏向于侧枝。最终根部的资源将分配给所有芽。若一个芽所接收的光照资源量为 v , 则 $\lfloor v \rfloor$ 表示在本轮迭代中该芽可以生长出的枝段数量, 每根枝段的长度为 $v/\lfloor v \rfloor$ 。

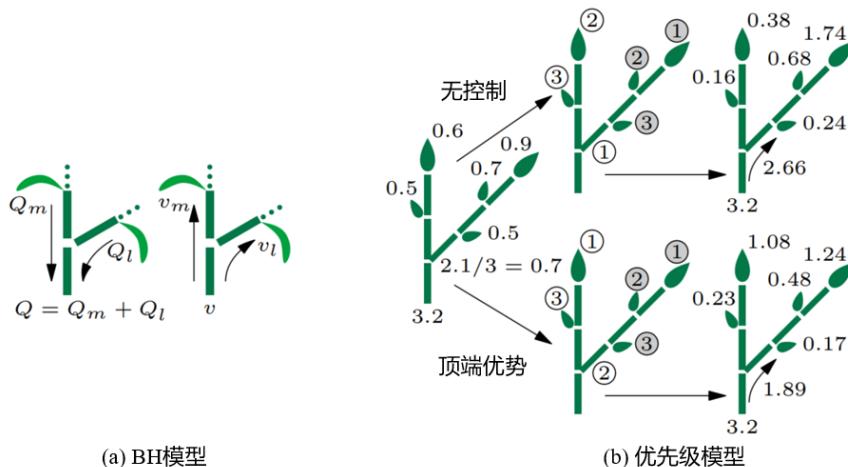


图 2.4 扩展BH模型和优先级模型的算法示意图。

Figure 2.4 The descriptions of the extended BH model and priority model.

2.3.2 优先级模型

BH模型通过依次考虑枝干结点来向上逐步分配资源。而接下来介绍的优先级模型 (Priority model) 则是一次性考虑整个轴线上的所有同级结点。首先, 我们自顶向下的将芽所接收的光照资源积累到根结点处, 如图 2.4(b)的左侧所

示。接着，计算每个结点处的支撑枝干从其附属的芽上所接收的光源的平均值，即用接收的光量总值除以该枝干上的芽数，如图 2.4(b)的左侧中出现的 $2.1/3 = 0.7$ 。据此，我们便对于每个枝干上的所有同级结点进行优先级排序，排序结果如图 2.4(b)的中间部分所示。如果需要引入顶点优势效果，只需要将顶芽放在排序序列的首位即可。最后在向上分配资源时，芽和支撑枝干所分配到的资源数量取决于它在队列中的优先级位置，即：

$$v_i = v \frac{Q_i \omega_i}{\sum_{j=1}^N Q_j \omega_j}, \quad i = 1, 2, \dots, N \quad (2.5)$$

式中 v 是结点待分配的光照资源量， N 为该结点后续的整条枝干上芽和支撑枝干的数目，并按照前面计算的优先级序列进行排序。 v_i 是第*i*个芽或支撑枝干所分配到的资源量。 ω_i 为资源配置的权重并采用如下所示的分段函数来定义：

$$\omega_i = \begin{cases} \omega_{max} - \frac{\omega_{max} - \omega_{min}}{\kappa N} i & i \leq \kappa N \\ \omega_{min} & i > \kappa N \end{cases} \quad (2.6)$$

式中 ω_{max} 和 ω_{min} 的默认值分别为1和0.006， $\kappa \in [0, 1]$ 的取值可以控制植物合轴形态。图 2.5展示了当 κ 分别取0.5和0.35时的植物模型，当 κ 的值越小植物越倾向于主枝干明显的形态。此外，与BH模型相同，光照资源在从顶向下积累到根部时会通过生化作用而发生变化，即 $v_{base} = \alpha Q_{base}$ 。并且最终分配给每个芽的光照资源量决定了将要生长的枝段的数目和长度。

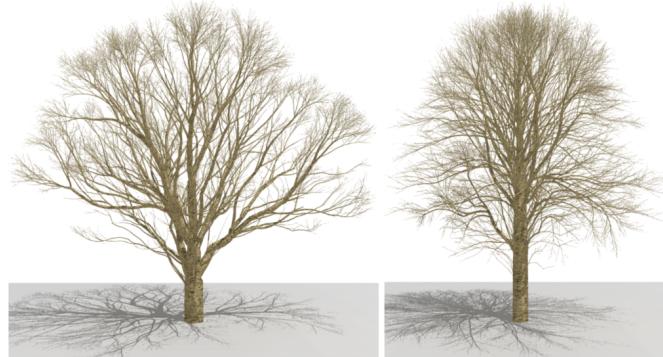


图 2.5 采用优先级模型对植物形态的影响。

Figure 2.5 The effect of priority model on plant structure.

2.4 植物几何模型构建

在得到植物骨架的多叉树数据结构后便可以构建出相应的网格模型。图 2.6展示了一组植物的多叉树骨架和相应的三维模型的案例。本文采用广义圆

柱体来构建枝干的网格结构。所谓广义圆柱体是指顶面和底面不一定不平行的柱形结构。对于一个中间结点及其所衔接的两个枝段，我们让结点一侧的圆柱体顶面和另一侧的柱体地面贴合在一起即可消除掉结点处的缝隙。图 2.6(c)展示了一段枝干的网格构建结果。

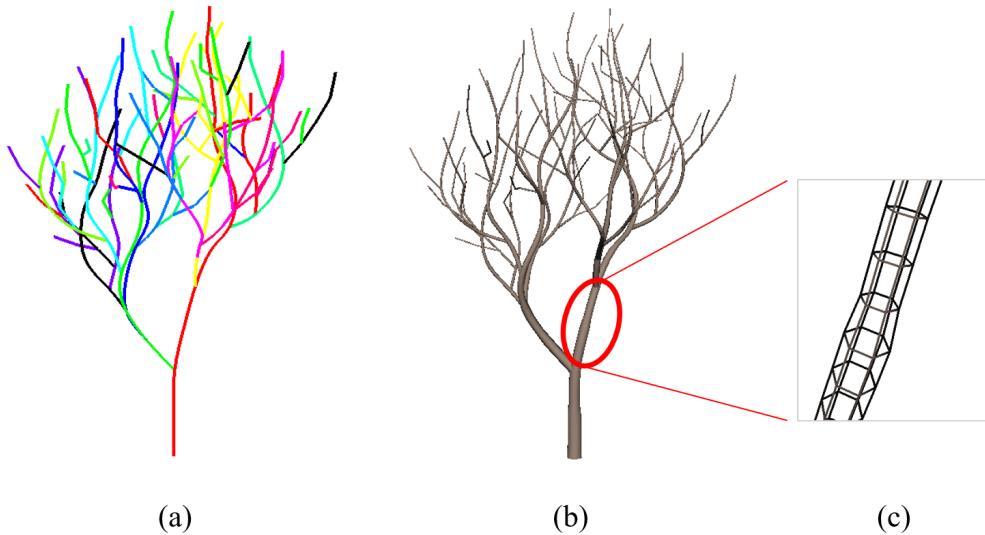


图 2.6 植物枝干的三维骨架及其最终网格模型。

Figure 2.6 The 3D skeleton and final geometric mesh of the branching structure.

为了在保证建模效果的同时尽可能减少模型的面片数，我们发现使用六面体来代替圆柱也能达到不错的视觉效果。为了让植物枝干看起来更自然，我们还对每一条超过3个结点的枝干使用三次Hermite插值进行光滑处理。对于植物的叶片，我们采用由两个三角形组成的方形纹理来表示。

此外，植物枝干的直径也是影响植物外观的重要因素。本文采用了一种自顶向下的直径计算策略，即根据其所有子枝干的直径大小来估算父枝干的直径，下式给出了枝干半径的植物学约束：

$$d^n = \sum_{i \in Child} d_i^n \quad (2.7)$$

式中， d_i 为第*i*个子枝干的直径， $n \in (2, 3)$ 为枝干直径的传递系数。植物最末端的枝干直径均被初始化为0.01。

图 2.7展示了一个植物在不同年龄时的建模结果。通过控制顶端优势的参数，我们可以对植物在不同年龄时采用不同样式的枝干风格。在这个例子中，起初由于顶芽获得了更多的资源从而得到有明显主枝干的幼苗，但在后续的迭

代中，我们减弱顶端优势的影响从而构建出丰满的树冠结构。图 2.8则展示了更多使用不同参数构建的植物模型。例如最左侧植物，我们让优先级模型中的参数 $\kappa=0.25$ 以及重力方向的权重 $\gamma = 0.8$ 从而得到一株笔直生长的乔木模型。对于图 2.8中间的植物，我们则通过在不同的年龄阶段修改重力方向的权重 γ 来获得形如柳树的三维模型。

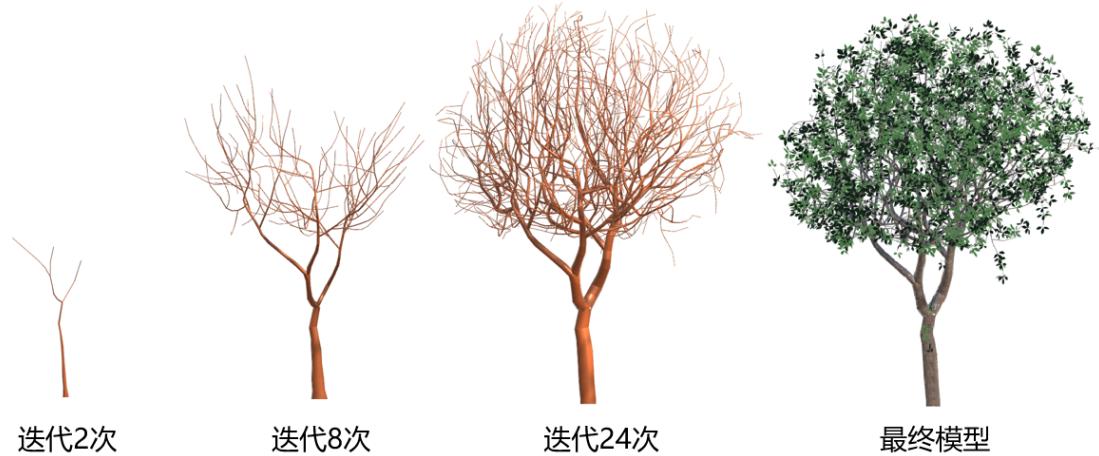


图 2.7 在不同生长阶段中采用不同参数构建的一株植物模型。

Figure 2.7 A plant model generated with various parameters at different iteration steps.



图 2.8 采用过程式植物建模算法创建的不同种类植物模型。

Figure 2.8 The resulting plant models with different species.

2.5 本章小结

基于迭代生成的植物结构非常依赖于用户定义的枝干分布规则，这导致该类方法通常需要大量的参数调控。用户需要丰富的植物学知识才能精确调控其

中的每处参数和规则。本章介绍了一种仅由少量参数控制的过程式建模植物建模方法，该方法有效模拟了植物在生长过程中对空间和光照等资源的竞争以及顶端优势现象，从而实现了植物枝干的自组织生成。本章的实验结果证明了该方法对生成不同种类植物的有效性。同时，该方法也为下文的研究工作做了铺垫，并将作为后续章节中介绍的几种形状驱动的植物建模的通用技术基础。

第3章 二维形状驱动的植物建模

3.1 引言

本章成功地将深度学习和过程式建模结合起来，并提出了一种二维形状驱动的三维植物建模方法。其核心是采用条件对抗神经网络（cGAN, conditional generative adversarial network）从二维轮廓草图和手绘枝干线条中，预测植物的三维形状信息（三维轮廓和主枝干），最终完整的植物模型将根据预测得到的三维形状采用过程化建模自动生成。该方法通过在一个合成的三维植物模型库中训练神经网络，从而实现了三维深度信息的可靠估计。此外，通过从图像中提取的边缘图来代替二维轮廓草图输入，该方法也可以用于解决单幅图像的植物三维重建问题。具体的算法流程如图 3.1 所示。首先，用户可以直接绘制植物的二维轮廓草图，或者对一张植物图像使用 Canny 算子(Canny, 1986) 提取边缘信息作为轮廓图（如图 3.1(a)），同时用彩色线条标记主枝干（如图 3.1(b)）。接着，cGAN 网络将基于这两个二维形状预测两个深度图，分别表示树木整体三维轮廓和主枝干的三维骨架。最后，过程式建模算法将以三维主枝干为基础在三维轮廓内部生成完整的植物三维模型。

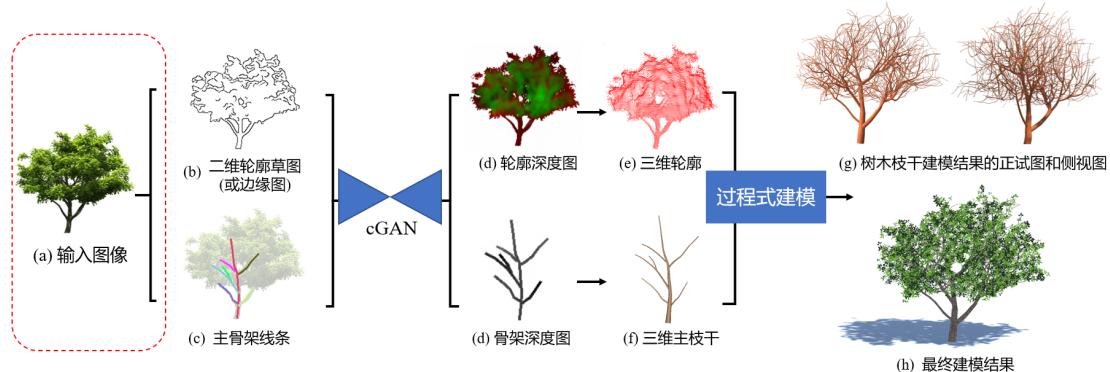


图 3.1 基于二维形状（轮廓和主枝干草图）的过程式三维植物建模的算法流程。

Figure 3.1 Procedural plant modeling based on 2D shapes (silhouette and main trunk sketches).

3.2 基于对抗网络的植物形状预测

大多数基于深度学习的物体重建方法通常以体素或点云的形式来表示和预测三维形状，并且仅限于重建相对简单和光滑的形状，比如汽车或椅子。然而，

由于树木自身具有更为复杂、细致的枝干结构和丰富的枝叶，直接预测三维树木模型是相当困难的。本章中，我们通过将深度学习和过程式植物建模相结合，从一幅图像中生成高质量的树模型。首先，利用深度神经网络对树木的基本三维形状进行预测，然后根据该预测的形状进一步利用过程式建模生成完整细致的植物模型。这里使用两种表示方法来描述树的基本三维形状：

- 可见主枝干：用于精确表示主要可见树干。神经网络需要从表示主枝干的二维线条中预测三维的主枝干骨架。
- 整体轮廓：用于包围整个树的粗糙轮廓点云。神经网络需要从树木的二维边缘图像中恢复出轮廓信息。

本文采用两种特殊的二维深度图像来表示这样的骨架和轮廓。因此该问题可以被简化成图像对图像的任务（Image to image task），而不是直接从图像估计三维形状结构。因此，我们使用cGAN网络从输入图像中推断出这些深度图像。

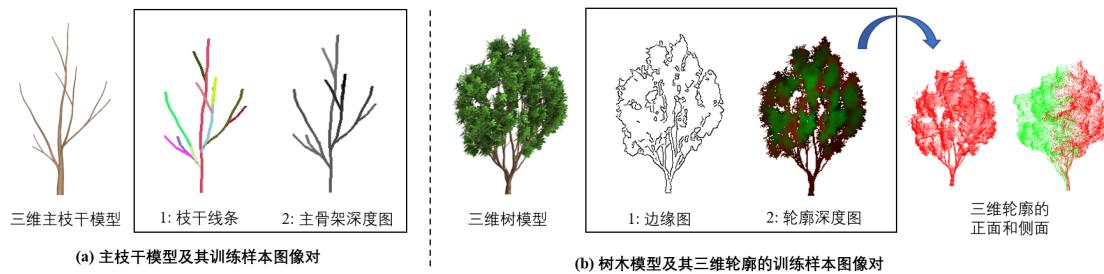


图 3.2 训练图像样本对的示例。(a) 枝干样本对, (b) 轮廓样本对。

Figure 3.2 Examples of the training image pair. (a) 3D skeleton training pair, (b) 3D silhouette training pair.

3.2.1 训练数据的表示和采集

由于目前没有公开的大量植物数据集，本文首先利用过程式建模自动生成大量的三维植物模型用于获取样本对。通过使用不同的建模参数，可以自动大量生成不同种类的植物。接着，我们利用这些三维树木模型用投影和深度缓冲的方式获得植物的图像数据集。本文实验所用的图像的大小都设置为 256×256 ，并使用正交投影将三维模型映射为二维图像。下面将分别介绍可见主枝干和轮廓的图像样本对的生成方法。

主枝干样本对：为了预测主枝干的三维信息，神经网络以枝干骨架的二维线条作为输入，以相应的深度图像作为输出（如图 3.2(a)）。在给定一个三维树

模型的情况下，通过对每段枝干使用不同颜色进行渲染着色，可以直接获得二维线条，而深度图像则可以从深度缓冲区中提取。在枝干重叠的交叉处，不同的颜色标注可以清楚地表示枝干前后的相对位置。这种方式有助于神经网络的收敛。本文中，对于所有树木进行标记所采用的颜色是固定的。

此外，所有枝干都设置为4像素宽度，因为较细的线条会导致收敛困难，而较宽的线条则可能导致线段之间的严重遮挡。最后，我们将渲染窗口的尺寸设置为 256×256 ，并使用正交投影来渲染每组图像样本对。在这种情况下，深度值将限制在[0, 1]的区间内，深度取0时表示最近的平面，取1表示最远的平面。

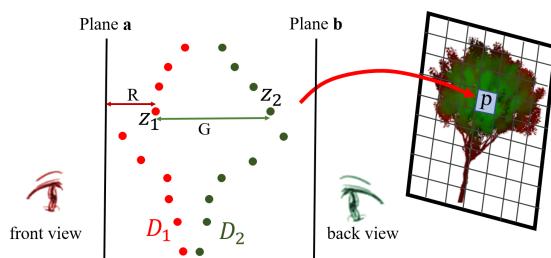


图 3.3 将三维轮廓转换为深度图像的过程。

Figure 3.3 Illustration of casting 3D silhouette to a depth image.

轮廓样本对：为了估计树木的三维轮廓，网络将树木图像的边缘信息像作为输入，输出用于描绘植物的三维轮廓的深度图（如图 3.2(b)）。

对于合成的三维植物的图像，我们采用高斯滤波结合Canny检测器得到植物的边缘图，以减少纹理和光照带来的影响。接着，我们需要将植物的三维轮廓信息存储到一个RGB深度图像中。我们观察到，使用前后视图的深度信息足够覆盖整个植物的外形。因此通过将相机在放置在植物的前后两个位置，我们可以使用正交投影将两个视角对应的深度图像分别保存下来，分别记为 D_1 和 D_2 。接着，我们需要把这两个深度图像放在相同的坐标系中表示。为了匹配这两个深度图像之间的像素，首先在水平方向上将深度图 D_2 镜像翻转。我们使用 D_s 来表示三维轮廓的最终合成深度图像，并且分别使用 $z_{1,x,y}$ 和 $z_{2,x,y}$ 来表示图像 D_1 和 D_2 中像素 (x,y) 处的深度值。如图 3.3所示，当位于前方观察时的相机截锥体将平面a作为其近平面，b作为远平面，而后视图时则相反。对于位于截锥体中的任何三维点 v ，正交投影将其深度计算为 $z_v = dist(v, near) / dist(near, far)$ ，其 $dist()$ 表示计算欧氏距离的函数。两个视图的深度值都介于[0, 1]之间。因此，为了将深度图 D_2 转换到与 D_1 相同的坐标中，我们将 D_2 中像素 (x,y) 的新深度值设

置为 $(1.0 - z_{2,x,y})$ 。最后对于表示3D轮廓的最终组合深度图像 D_s , 我们使用红色通道(R)存储前视图的深度, 使用绿色通道(G)存储前视图和后视图之间的间隔。

3.2.2 条件对抗神经网络cGAN

本文以 Isola 等 (2017)提出的条件对抗生成网络(cGAN)为基础并进行扩展, 用来解决图像到图像的转换问题, 从而输出蕴含三维信息的深度图像。我们的cGAN网络输入从单个图像中提取的枝干图或边缘图, 输出相应的主枝干和轮廓的深度图像。图 3.4展示了本文所采用的神经网络的结构和参数配置。本章实验中所有图像的大小均设置为 256×256 。用于预测3D轮廓时的输入是1通道的边缘图像, 输出则是2通道的轮廓深度图, 2个通道分别表示图像在正面和背面观察时的深度信息。对于主枝干预测任务, 网络输入3通道的枝干线条图像, 并输出1通道深度图像。

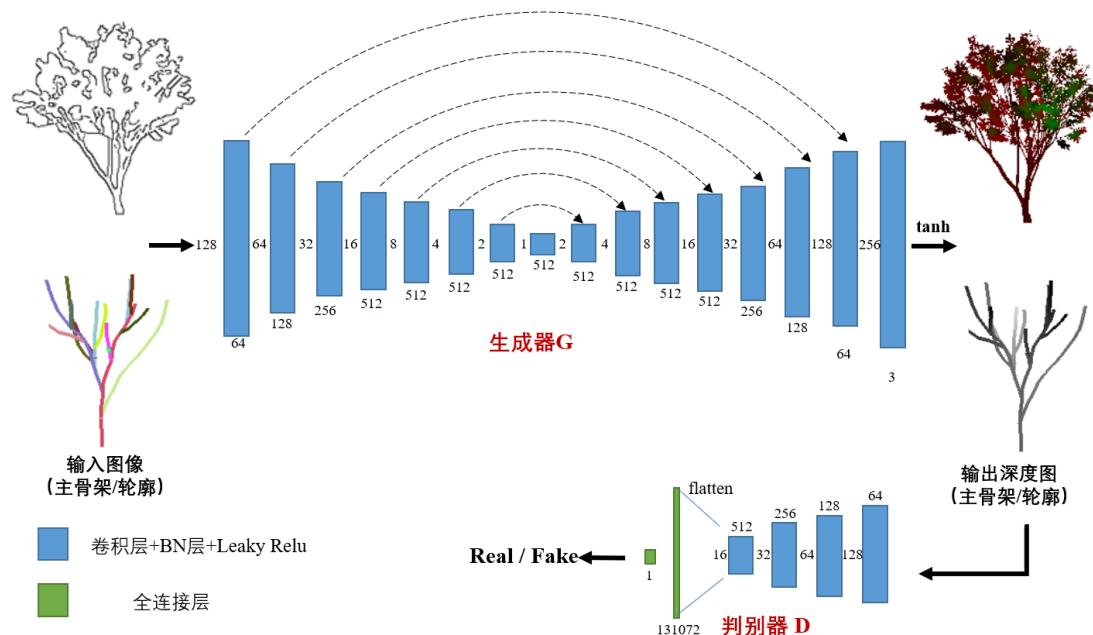


图 3.4 用于恢复深度信息的条件对抗网络结构。

Figure 3.4 Adversarial network architecture for recovering depth information.

生成器G: cGAN网络的核心是一个生成器G, 它将一个域的图像作为输入, 并将其转换为另一个域。我们采用U-Net (Ronneberger 等, 2015)作为本章生成器的基本结构, 该结构在许多基于二维图像的任务中是一种非常流行的选择。U-Net的本质是一种编码器-解码器结构 (Encoder -Decoder), 该结构将编码器和解

码器中除了常规的串联式连接外，还将处于镜像位置的两个卷积层之间采用跳跃连接的方式，如图3.4所示。编码器和解码器的结构是对称的，都由一系列的卷积层组成。所有卷积操作都使用 4×4 的卷积核，卷积步长为2。所有中间层均采用Batch normalization操作，并采用Leaky ReLU作为激活函数。解码器中的最后一个卷积层采用tanh激活函数，并用于输出预测的深度图像。

判别器D：引入和训练判别器D的目的是为了尽可能地区分真实深度图像和生成器G输出的虚假的深度图像。一个良好的判别器能够提高生成器G预测输出的深度图像的质量。判别器的结构也由一系列卷积层组成。其中间卷积也使用Batch normalization操作和Leaky ReLU激活函数。所有卷积操作都使用尺寸为4和的卷积核，步长设置为2。在最后一个卷积层后，我们采用全连接层（Fully connected layer）和Sigmoid激活函数以映射到一维的输出。判别器D的一维输出表示生成器D所预测的深度图像与真实深度图像相似的概率。

损失函数：cGAN网络的损失函数描述了生成器G和判别器D之间的博弈关系，即生成器G希望预测的深度图像尽可能不被判别器D检测出，而判别器D则希望尽可能辨别出生成器G所预测的虚假的深度图像。该种博弈关系可以定义为：

$$L_{cGAN}(G, D) = E_{x,y}[\log D(y|x)] - E_x[\log(D(G(x)|x))] \quad (3.1)$$

其中， x 是输入图像， y 是数据集中真实的深度图像。在训练过程中，生成器G试图最小化该函数，而判别器D试图最大化该函数。

为了提高网络的收敛质量，我们引入了一个带掩码的深度损失函数 L_{depth} 。该损失函数使用L1距离来测量预测深度和真实深度之间的逐像素差异。此外，与大多数图像翻译任务不同，我们希望网络只关心树木主体的深度而不必学习背景部分的深度，因此我们还利用二元掩码来屏蔽背景部分的像素。 L_{depth} 的计算如下式所示：

$$L_{depth}(G) = \sum_p m_p \|d_p - \tilde{d}_p\|_1 \quad (3.2)$$

其中， m_p 是像素 p 处的二进制掩码值，若该像素为前景时取1，为背景时则取0。 d_p 和 \tilde{d}_p 分别表示像素 p 处真实的像素深度和预测的像素深度。综上所述，

我们的最终损失函数 L 可以表示为：

$$L = L_{cGAN}(G, D) + \lambda L_{depth}(G) \quad (3.3)$$

为了有效地训练网络，我们交替训练判别器和生成器，并采用Adam优化器(Kingma 等, 2014)对网络进行优化。

3.2.3 三维形状的恢复和去噪

利用调整后的cGAN网络获得深度图像后，便可以分别恢复出轮廓和主枝干的三维点云数据。然而采用直接预测得到的点云往往含有大量的噪声点，如图3.5所示。因此我们首先需要对推断出的深度图像进行去噪。

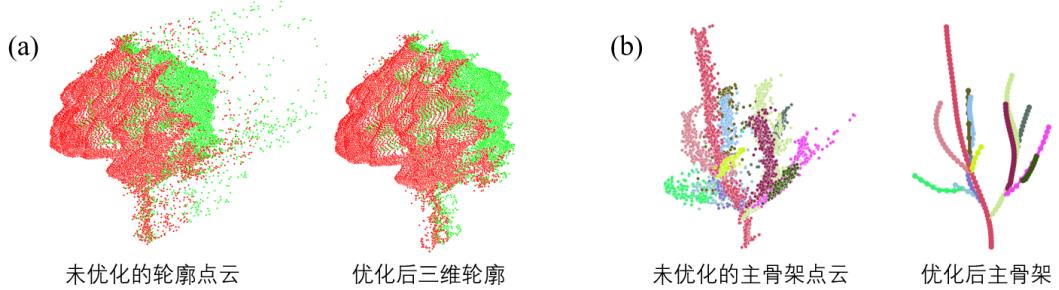


图 3.5 点云去噪以及主枝干构建的过程。

Figure 3.5 Data processing for point cloud denoising and visible branch tracing.

通过观察放大后的深度图像和预测的点云，我们首先发现大部分噪声都位于前景和背景的边界处。因此我们将位于边界上的所有像素都标记为噪声点，因为它们的深度值并不准确。接下来，我们对深度图像的每个通道进行如下处理。对于前景中的每个像素 p^k ($k = R, G, B$)，我们计算 p^k 与其所有相邻像素点 p_i^k 之间的平均深度差 d^k ：

$$d^k = \frac{1}{N(p^k)} \sum_{i \in N(p^k)} |p^k - p_i^k|, \quad (3.4)$$

其中， $i \in N(p^k)$ ， $N(p^k)$ 表示 p^k 周围距离2像素的所有相邻像素点。对于每个相邻像素，我们还使用相同的方法计算其平均距离 d_i^k 。如果像素 p^k 的所有邻域像素的平均深度差 d_i^k 的方差大于给定阈值，则将该像素 p^k 视为噪声点，即：

$$\delta^2 = \frac{\sum_{i \in N(p^k)} (d_i^k - d_{avg}^k)^2}{N(p^k)} > \epsilon \quad (3.5)$$

其中, $d_{avg}^k = \frac{1}{N(p^k)} \sum_{i \in N(p^k)} (d_i^k - d_{avg}^k)$ 表示平均深度差 d_i^k 的平均值, 阈值 ϵ 的默认值为 0.019。最后对于所有噪声点用淋雨像素的平均深度值来作为噪声点处的深度。图 3.5(a)展示了对轮廓的三维点云的去噪效果。

主枝干的追踪构建。为了重建可见的枝干, 我们需要将骨架点云组织成一个连通的图结构。本文采用广度优先搜索的方法, 以点云的最低点作为根部, 并从根部开始遍历主枝干深度图像的像素。我们使用一个队列 q 存储待访问的像素, 使用列表 l 记录已找到的枝干。已经找到的枝干颜色采用该处的像素颜色标识, 每个枝干都维护一个结点列表。在开始时, 将根结点放入队列 q 中, 并在 l 中添加一个起始枝干。然后, 以队列首部的像素为中心, 沿着图像中的八个方向搜索邻域像素, 搜索步长为 4 (即默认的笔划宽度)。对于队列 q 中的每个像素, 可以通过计算其半径为 4 的相邻像素的中心位置来形成一个新的树枝结点, 并将该结点加入到队列 q 和 l 中相应的枝干中。如果遇到新的像素颜色, 我们会在 l 中新建一个枝干。当队列 q 变为空时, 算法结束。最后, 根据枝干列表 l 中的邻接关系和端点处的距离关系形成连通的树状结构。图 3.5(b)显示了主枝干骨架的构建效果。

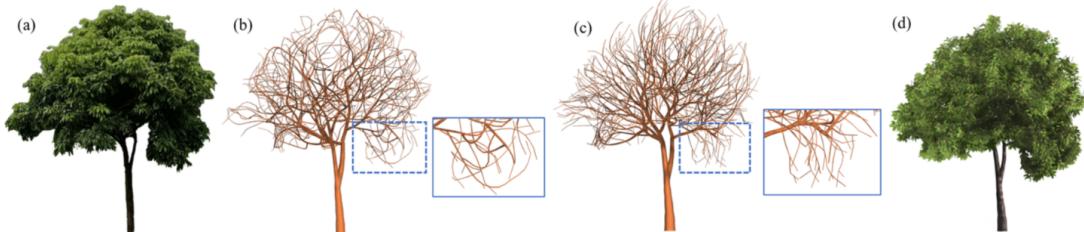


图 3.6 引入生长牵引方向的植物建模效果。(a)输入图像。(b)未使用牵引方向的枝干。(c)使用牵引方向的枝干。(d)重建的完整植物模型。

Figure 3.6 The effect of using attraction direction. (a) the input image. (b) branches without attraction. (c) branches with attraction. (d) final tree model.

3.3 主枝干和轮廓约束下的植物建模

在主枝干和三维轮廓的约束下, 我们可以采用第二章所描述的基于空间资源竞争的过程式植物建模方法来自动生成完整的植物三维模型。然而单纯借助过程式建模可能并不能让植物模型的枝干分布完全与原始图像保持一致。如 3.6(b)所示, 虚线框内重建处的枝干过于杂乱的分布而非顺着图像中的趋势向

下生长。为了解决这一问题，我们基于一个合成的生长方向图，为空间中的每一个空间点（marker）额外引入了一个枝干生长的牵引方向 v_{trac} 。

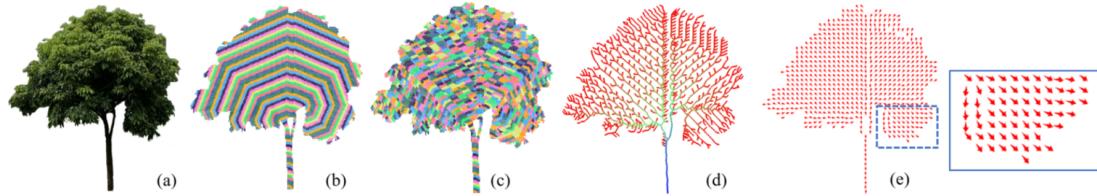


图 3.7 计算方向场的流程图。**(a)** 输入图像。**(b)** 初始聚类。**(c)** 二次划分后的聚类。**(d)** 牵引骨架图。**(e)** 最终的方向场。

Figure 3.7 The processing of computing the direction map: **(a)** input image. **(b)** preliminary bins. **(c)** bins after the second time division. **(d)** attraction map. **(e)** final direction map.

如图 3.7 所示，我们首先将照片中所有的前景像素看待为二维顶点，因此整个植物图像可以转换为稀疏的二维点云。我们首先应用 Dijkstra 算法计算从根结点到其他所有顶点的最短路径。根据每个顶点到根结点的路径长度，我们可以将点云划分为若干个聚类，如图 3.7(b) 所示，每个聚类采用不同的颜色标注。接着，我们根据一个给定的内部最小距离将每个聚类做进一步的划分以得到更小的聚类，如图 3.7(c) 所示。最后，从根部所属的聚类开始，一个二维骨架可以通过逐个连接邻接聚类的中心位置而构建得到（图 3.7(d)）。最后，前景像素的生长牵引方向 v_{trac} 便可以通过在二维骨架图中计算其所有相邻枝干的生长方向的平均值得到。最终的生长方向图如图 3.7(e) 所示。

为了将从图像获得的二维方向图应用到三维的空间点上，我们简单的讲二维方向场绕根结点旋转一周，以为每一个空间点分配一个生长牵引方向。最后，在同时考虑基于空间资源竞争计算得到的方向 v_{opt} 和生长牵引方向 v_{trac} 后，一个芽的最终生长方向可以用下式得出：

$$v_{final} = v_{opt} + \alpha \frac{1}{N} \sum_i^N \vec{v}_{trac} \quad (3.6)$$

图 3.6c 展示了引入生长牵引方向后的植物效果。虚线框内的枝干能够较好的按照图像的趋势向下生长。

3.4 实验结果与分析

在本节中，我们进行了大量实验以证明我们方法的有效性。本节首先讨论

并评价了使用cGAN网络预测三维形状的效果，然后探讨了对真实照片中的植物的三维重建效果。我们也与最新的相关植物建模方法进行了比较和分析，并进一步挖掘了本章方法的可能性和未来发展。所有的模型都是在一台配备有Intel i7-7700HQ处理器的台式计算机上获得的，其频率为2.8GHz，RAM为8GB。

训练设置：神经网络的训练在一台配备有NVIDIA GeForce GTX 1080 Ti GPU（11GB显存）的工作站上进行，深度学习算法采用Python和PyTorch进行编写。为了训练cGAN网络，我们分别为主枝干和轮廓构建了20K组图像对用于训练。通过使用不同的参数，数据集可以包含20种不同形态的树种，每一种植物包含1000个三维模型。训练过程耗时4天，但是在测试阶段仅仅需要约0.07秒就能从给定的输入图像中预测出深度图像。

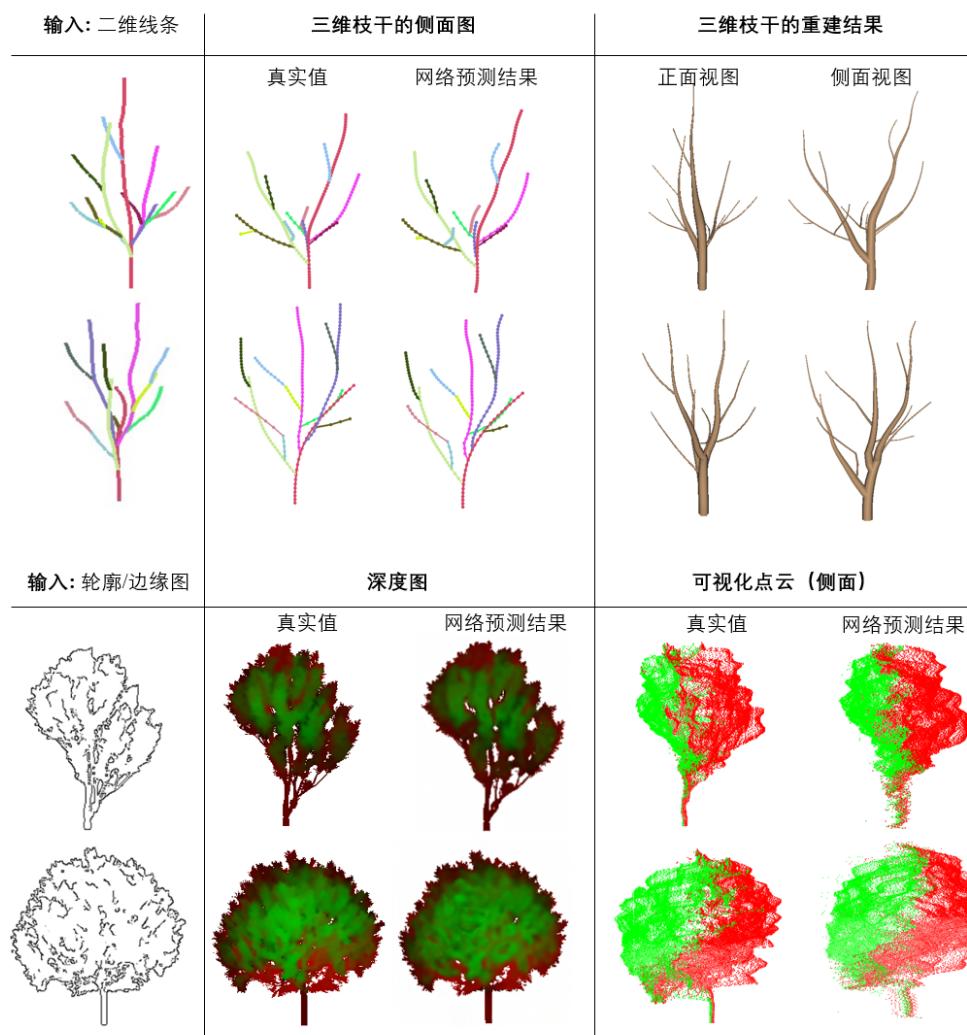


图 3.8 三维主枝干和轮廓的恢复效果。

Figure 3.8 Reconstruction results of 3D main trunks and silhouettes.

网络效果评估：为了评估神经网络预测深度图的准确性，我们构建了2K组图像对作为测试数据集。测试集同样包含20种不同的植物，并将用于定性和定量的分析评价。图 3.8 的上半部分展示了部分主枝干的重建效果案例。通过观察第二列的重建后枝干的侧视图，我们发现预测的枝干很好的保持了原始的深度位置，尽管有些枝干由于深度图的噪声和误差而稍显弯曲。图 3.8 的下半部分展示了部分三维轮廓的重建效果案例。其第二列分别为轮廓图的真实深度图像和预测深度图像。我们发现轮廓图的外形得到了很好的保持和恢复，尤其是轮廓图中的孔洞部分也能够较好的恢复出来，且树叶和可见枝干部分也能够清晰地辨认区分。第三列则为相应三维点云的侧视图，其中红色和绿色的点分别为前视图和后视图所观察到的点云。通过观察深度图像和侧视图的轮廓，表明我们的算法能够有效大致保持主枝干的轮廓的形状特征，并给出合理的近似预测深度。



图 3.9 从真实照片中重建的三维植物模型。

Figure 3.9 Modeling results of our method from real plant images.

接下来，我们使用两种方式来定量地评估网络的预测能力：深度误差和Hau-

sdorff距离。深度误差表示网络预测的深度图像和真实深度图之间的平均深度差。经过在测试集上的计算，主枝干和轮廓的平均深度误差分别为0.032和0.041（深度值的范围为[0, 1]）。此外，针对主枝干的重构，我们还采用Hausdorff距离来度量重构后的骨架与真实主枝干之间的相似程度。在测试集上，我们用重构后的骨架和真实骨架的树结点所构成的两个点集进行计算，最终得到平均Hausdorff距离为0.098。深度误差和Hausdorff距离的取值表明，该方法能有效地保持和恢复数据集中植物的三维信息。

表 3.1 部分植物的重建时间统计（秒）。

Table 3.1 Time (second) for modeling several trees.

No.	主枝干线条绘制	主枝干恢复	三维轮廓恢复	树木生长
Fig. 3.9a	10.80	0.069	0.061	1.53
Fig. 3.9b	17.13	0.083	0.058	1.22
Fig. 3.9c	11.80	0.076	0.072	1.07

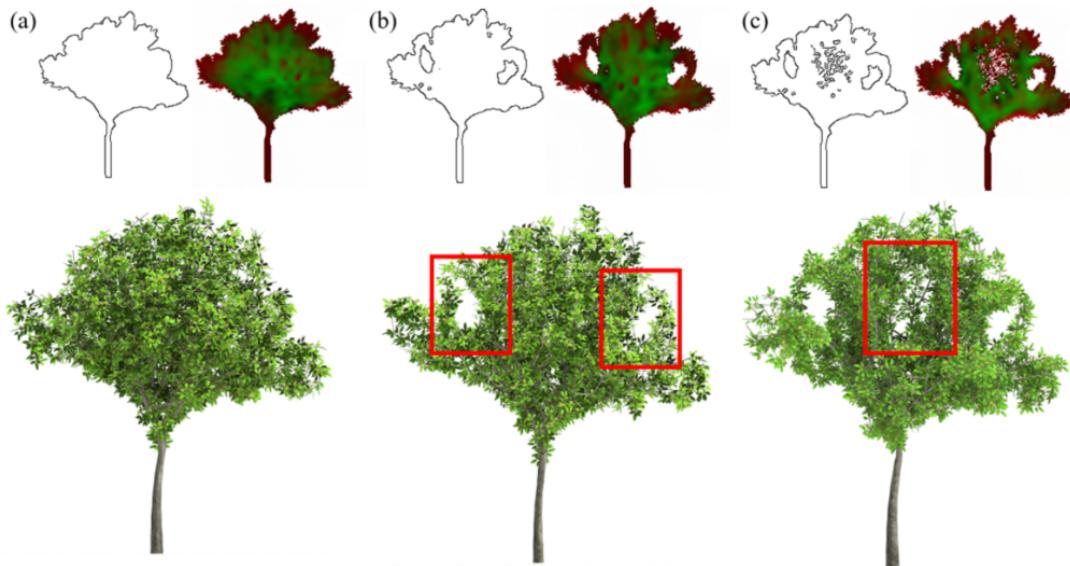


图 3.10 轮廓边缘图中不同的内部细节对最终植物模型的影响。

Figure 3.10 Different inner details can make effect on the tree structure.

植物建模效果评估：为了证明本文方法的植物重建质量，我们使用训练好的cGAN网络对植物的真实图像进行测试。图 3.9展示了几组从不同形态特征的植物照片中重建的三维植物模型的效果图，证明我们的方法可以有效地应用于不同类型的树的建模。此外，表 3.1给出了几组植物案重建过程中的分步时间统

计。除了交互绘制所需要的时间外，整个重建过程非常高效的。通过cGAN预测三维骨架和轮廓是实时的，且通过过程式建模从主枝干和轮廓生成一棵完整的树也需要大约1-2秒的时间。

我们还设置了一组交互实验以证明边缘图内部的细节对于最终植物建模效果的影响。如图 3.10所示，我们首先完全擦除一个植物边缘图的内部细节，再用不同程度的细节再次填充并生成植物模型。图 3.10的三个例子依次为：没有细节、有两个大洞、再添加杂乱的小点。结果表明我们的方法可以很好的对边缘图中的不同细节做出反应。即缺失的大洞代表空洞，而零碎的小点代表稀疏的树叶。除了真实世界的植物照片，我们的方法还可以用来模拟更复杂的例子，如图 3.11，我们的输出树忠实地反映了用户设计的手绘形状，证明了本章方法的良好泛化能力。

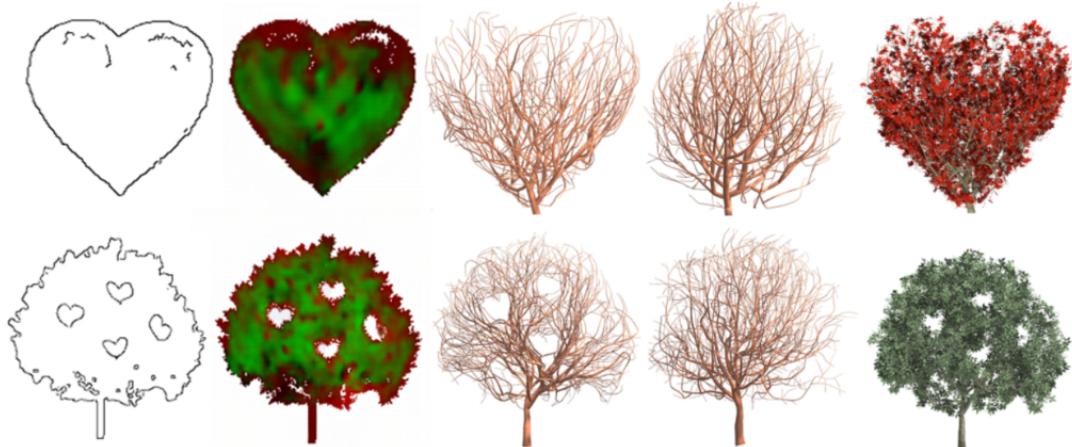


图 3.11 用户手绘的线条所生成的植物模型。

Figure 3.11 Ornamental tree models created from the shapes designed by a user.

比较：我们将本章算法同最新的基于单幅图像的植物建模方法 ([Tan 等, 2008](#))进行比较，该方法在2.4GHz CPU的PC上重建植物模型大约需要20分钟，而我们的方法可以在2.8GHz CPU上只需要几秒钟便可以完成。并且由于他们的方法只考虑了树的外部边界，故对于树叶稀疏或有内部空洞的植物进行重建的效果不佳，而我们的方法则能更好地应对这种复杂情况。图 3.12展示了两个方法的建模对比效果。此外，本章方法通过神经网络来预测三维轮廓和骨架，使得侧视图总是多变的。但是 [Tan 等 \(2008\)](#)生成的树在前视图和侧视图之间看起来总是相似的。此外 [Tan 等 \(2008\)](#)的方法只是基于不同枝干之间的距离尽可能大的假设，简单地旋转枝干，这对于描述真实的骨架来说太过简单。而通过神经

网络，本章方法可以有效地从二维线条中推断出更精确的三维主枝干骨架。

我们还对比了一种基于多幅图像（图像序列）的植物重建算法(Guo 等, 2018)。该方法使用SFM算法从图像序列中恢复植物的三维点云，然后通过过程式建模得到完整的枝干结构。但由于获得的点云精度质量不高，导致无法很好地恢复植物的细节，例如图 3.13(b)的红框区域。而本文方法通过用不同颜色标注枝干，可以准确地恢复该区域的枝干交叉情况。并且，本文方法只需要采集单幅图像即可得到高质量的植物模型。



图 3.12 同一种基于单幅图像的建模方法(Tan 等, 2008)的结果对比。

Figure 3.12 Comparison with a previous single image based reconstruction method.

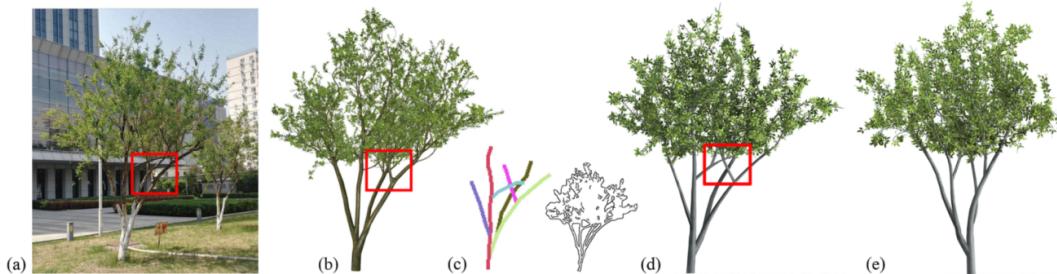


图 3.13 同一种基于多视角图像序列的建模方法(Guo 等, 2018)的结果对比。

Figure 3.13 Comparison with a method that reconstructs plants from multiple view images.

3.5 本章小结

本章通过结合深度学习和过程式植物建模技术，提出了一种二维形状驱动的植物建模方法。该方法基于一个cGAN神经网络从二维轮廓草图和手绘线条中预测得到植物的大致三维形状（三维轮廓和主枝干骨架），接着完整的植物模型则通过一种改进的过程式植物建模算法自动生成。通过在一个合成得到的包含不同树种的植物模型数据集中训练一个神经网络，该方法有效地解决了预测可

靠三维深度信息的难题。同时，通过从图像中提取边缘图作为网络的二维轮廓输入，我们也将该方法拓展为基于单幅图像的植物建模方法，本章最后与几种现有的基于图像的植物建模方法进行对比，证明该方法在恢复三维结构和得到高质量植物模型方面的有效性。

第4章 三维骨架驱动的植物建模

4.1 引言

随着虚拟现实（VR）技术的发展，用户可以完全沉浸在一个虚拟环境中，并能够借助手柄控制器等设备进行灵活和连续的交互。在此基础上，本文基于虚拟现实平台提出了一种基于三维骨架驱动的交互式植物建模方法。该方法结合了虚拟现实设备和过程式建模的优点，基于三维手势操作设计了一套交互方案包括：直接枝干绘制、直接仿射变换（旋转和平移）、骨架画刷工具等。用户通过三维手势交互可以创作植物的基本三维骨架，而更加丰富的细节则由过程式建模完成。如图 4.1所示，该系统包含一个用于显示的VR头盔和用于交互的6自由度手柄控制器。通过使用VR头盔和手势交互，用户可以同时执行旋转和平移操作并且能够直接通过头部旋转来改变观察视角，而无需像传统二维交互式设备一样直行大量控制和调整操作。借助手柄控制器上的触摸板，用户还可以在交互的过程中任意修改建模参数。

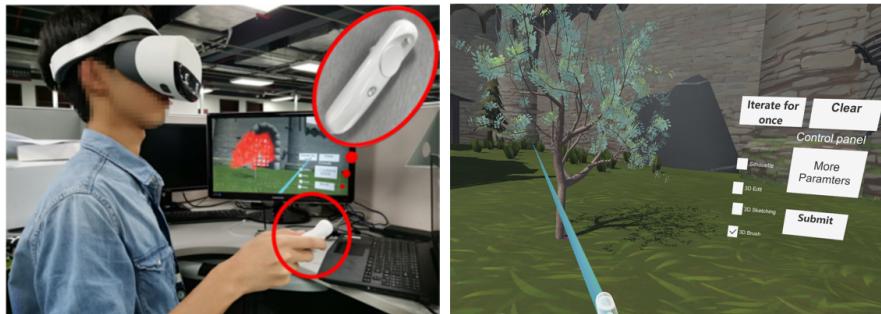


图 4.1 基于手势交互的三维植物建模系统。

Figure 4.1 Our 3D gesture based interactive plant modeling system.

图 4.2列举了本章方法所提供了几种三维骨架控制方法。图 4.2(a)展示了使用三维手势直接绘制枝干的例子。图 4.2(b)则展示了用户对植物的一个部分进行整体的移动编辑操作。同时，本章算法可以避免编辑的部分与其余的植物枝干之间出现交叉和碰撞。图 4.2(c)则展示了通过画刷工具指定骨架的生长趋势和范围从而自动生成相应形状的植物模型。

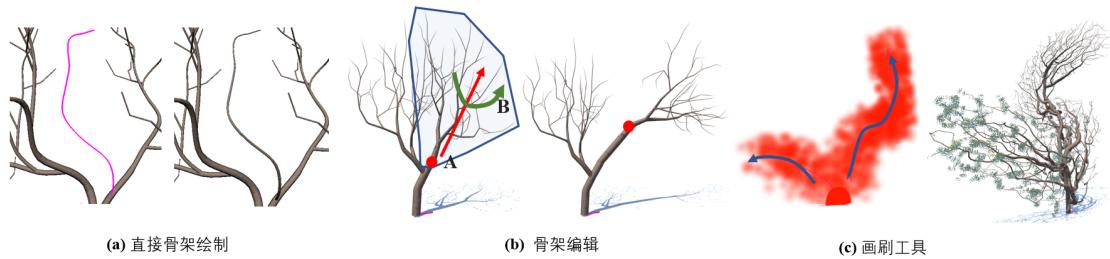


图 4.2 基于手势交互的几种三维骨架控制方法。

Figure 4.2 3D Skeleton control based on gestures.

4.2 基于VR控制器的植物骨架控制

本文采用Pico Neo虚拟现实设备进行开发，包括一个VR头戴显示器以及一个6自由度的手柄。本节将介绍如何基于三维手势实现植物骨架的一套交互操作。

4.2.1 骨架的三维绘制与编辑

对于传统的二维输入设备，如鼠标或手写笔等，用户在设计三维模型时的编辑操作往往非常不方便且不连续。例如，由于所有操作被局限于二维空间，用户必须分别执行旋转和平移操作，并且在操作三维模型时必须不断调整相机位置。相比之下，6自由度的手柄控制器则允许用户用更直接、更连续三维手势进行操作。

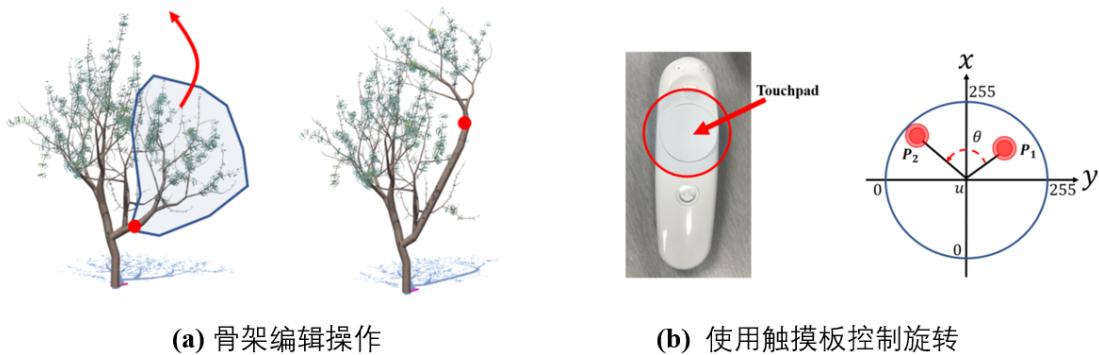


图 4.3 骨架三维编辑操作。(a) 直接手势编辑示例。(b) 基于触摸板的精确旋转控制。

Figure 4.3 3D skeleton editing. (a) Direct gesture editing. (b) Precise rotation using touchpad.

本节实现的直接三维编辑，允许用户通过选中植物枝干的某个结点，然后通过手势自由地移动该结点所对应的整棵子树的枝干。当然受益于三维手势交互，用户在移动枝干时可以同时进行平移和旋转的操作，而无需分别进行。图

4.3(a)展示了骨架编辑操作的效果，用户选择红色点所对应的结点，在对该部分子树进行转动的同时并向上拔高子树的位置。

为了保证移动后的枝干相互不碰撞，我们在移动的过程中对选中的子树进行碰撞交叉检测。具体来说，在选中树干结点的同时，先为所选择子树部分和剩余未选择子树部分分别构建一个凸包，然后便对这两个凸包进行碰撞检测。首先是凸包的构建，本节通过Gift wrapping算法实现。该算法将一个子树的所有结点看作一个点集，首先从点集中选择一个保证位于凸包表面的初始三角面f，然后找到面f中的一条另一侧的面还没有被找到的边e，接着从剩余的顶点中找到一个与边e和面f夹角最小的点以构建新的三角面，算法不断重复直到凸包闭合起来。在获得闭合的凸包后，对两个凸包的碰撞检测则通过Gilbert-Johnson-Keerthi算法实现，该算法首先计算两个凸包的闵可夫斯基差，如果坐标原点位于闵可夫斯基差所对应的多面体内则表示两个凸包发生了碰撞交叉。

一般情况下，直接使用三维手势的移动可以大致满足旋转精度的要求。然而，对于一些高精度要求的情况是不够的，因为手腕的旋转程度很难被准确控制。因此，我们提供了另一种借助手柄触摸板实现的精确旋转控制方案，这种方法同样适用于帮助市面上另一种常见的3自由度手柄实现旋转操作。如图4.3(b)所示，用户通过在触摸板上滑动手指来旋转的旋转的角度。子树的旋转角度 θ 与手指按压位置移动的角度相关，即：

$$\theta = \tau \cdot \arccos \left\{ \frac{(\mathbf{P}_1 - \mathbf{u}) \cdot (\mathbf{P}_2 - \mathbf{u})}{|\mathbf{P}_1 - \mathbf{u}| |\mathbf{P}_2 - \mathbf{u}|} \right\} \quad (4.1)$$

式中 $\mathbf{P}'_1 = (\mathbf{P}_1.x, \mathbf{P}_1.y, 0)^T$ 和 $\mathbf{P}'_2 = (\mathbf{P}_2.x, \mathbf{P}_2.y, 0)^T$ 分别是手指滑动前后的坐标位置， $\mathbf{u} = (127.5, 127.5)^T$ 表示触摸板的坐标原点。 $\tau = \text{sign}((\mathbf{P}_1 - \mathbf{u}) \times (\mathbf{P}_2 - \mathbf{u})) . z$ 表示旋转的方向， \times 表示叉积。

除了编辑操作外，借助6自由度的手柄控制器可以直接在空间中自由绘制三维线条。基于此我们提供了一种基于三维手势的三维枝干绘制方案（如图4.4所示）。用户从现有树木骨架的一处枝干开始，在三维空间中绘制连续的线条。由于手势绘制会导致一定程度的抖动，因此线条首先采用稀疏的点采样来减少抖动带来的影响。但是直接将这些稀疏采样点顺序连接会导致生成的枝干呈折线状，尤其是弯折出会非常尖锐不自然。因此，我们采用三次Hermite插值使得枝干线条呈现光滑自然的形态。

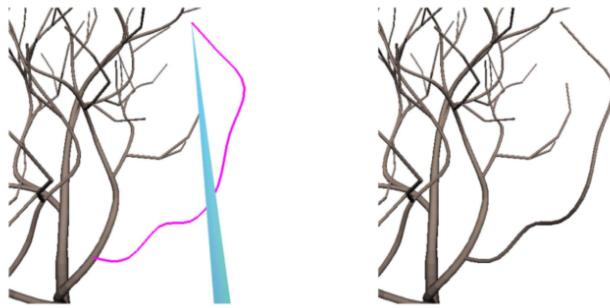


图 4.4 直接利用三维手势绘制树木骨架。

Figure 4.4 Sketching the skeleton directly with 3D gesture.

4.2.2 骨架画刷工具

TreeSketch (Longay 等, 2012)曾提出一种在二维平板电脑上的交互建模方式, 即采用一种画刷工具使用多个手指绘制一个区域, 然后在这个区域内用过程化建模的方式将枝干填充生长。但是TreeSketch的二维操作局限于二维, 因此他们采用一种简单的膨胀算法将二维形状扩展为三维形状, 这导致该算法生成的植物在三维中观察较为不自然。本节则基于6自由度手柄提供了一种基于三维手势的画刷工具。如图 4.5所示, 用户可以仅仅通过画刷直接绘制一个三维区域, 蓝色箭头表示鼠标绘制的路径, 红色的区域为播撒的空间点。然后根据这些空间点, 植物的枝干可以通过空间殖民算法生长出来。

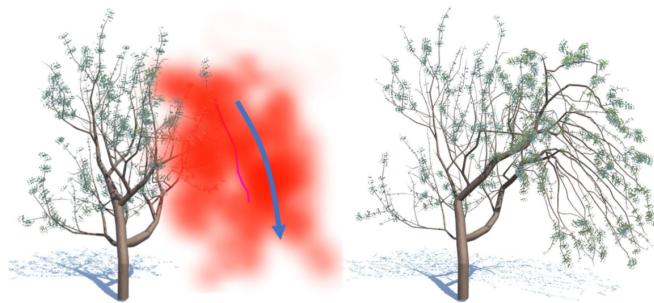


图 4.5 使用画刷工具进行骨架绘制。

Figure 4.5 Skeleton sketching using the brush tool.

具体来说, 手柄触摸板的按键按下时画刷绘制开始。随着手柄的移动, 算法首先会以手柄的当前位置为中心在一个球形空间内均匀生成空间点。在绘制过程中, 球的半径和空间点的密度可以用手柄的触摸板随意连续修改, 如图 4.6所示, 手指在触摸板上横向滑动时代表修改球的半径, 而纵向滑动时表示

修改空间点的密度。在手柄按键抬起后，植物当前枝干上的芽会检测是否位于画刷所绘制的区域范围内，如果芽处于该范围内，则视为活动状态并争夺空间点资源，直到没有足够的空间点可以争夺为止。

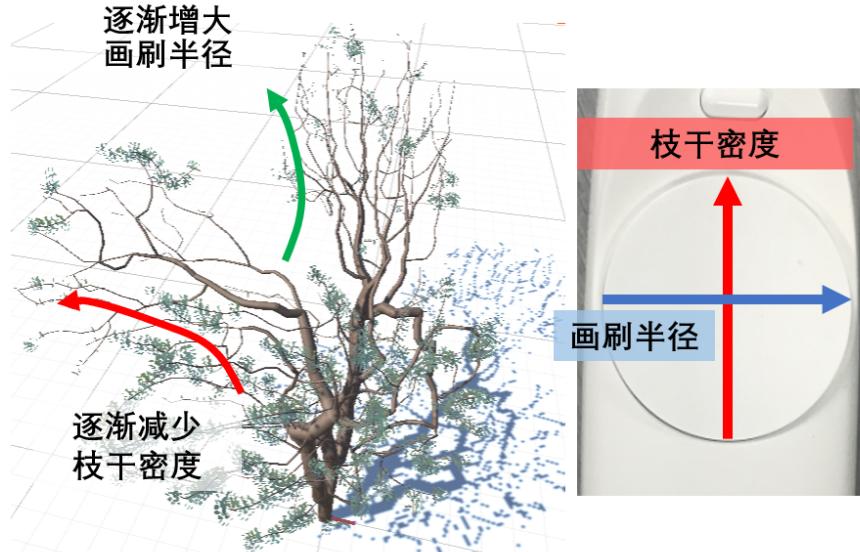


图 4.6 基于手柄触摸板的建模参数控制。

Figure 4.6 Parameter adjustment using the touchpad.

单纯使用基于空间殖民的过程式建模容易导致如图 4.7(a)所示的问题。即，由于画刷半径较小，芽在生长时极易触碰到区域边界，导致产生这种杂乱的螺旋状的不自然形态。为了使枝干的生长更加可控，我们引入画刷引导方向来解决这一现象。如图 4.7(c)所示，对于每个空间点*i*，我们首先记录该处的笔划方向 $V_{stroke}(i)$ ，并计算该处笔划方向对于空间点*i*在空间竞争时的权重 ω_i 。 ω_i 的计算可以通过将空间点*i*到笔划的垂直距离 d_i 带入到正态分布中计算得到，即：

$$\omega_i = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(d_i/\sigma)^2} \quad (4.2)$$

在这种情况下，芽的最终生长方向 \vec{V}_{final} 将由空间殖民算法计算出的最优生长方向和画刷的路径方向共同描述：

$$\vec{V}_{final} = \text{norm} \left(\vec{V}_{opt} + \sum_i^N \omega_i \vec{V}_{stroke}(i) \right) \quad (4.3)$$

式中， \vec{V}_{opt} 为空间殖民算法所计算得到的原始生长方向。这样枝干将会沿着笔划的绘制方向更加自然地生长，其效果如图 4.7(b)所示。

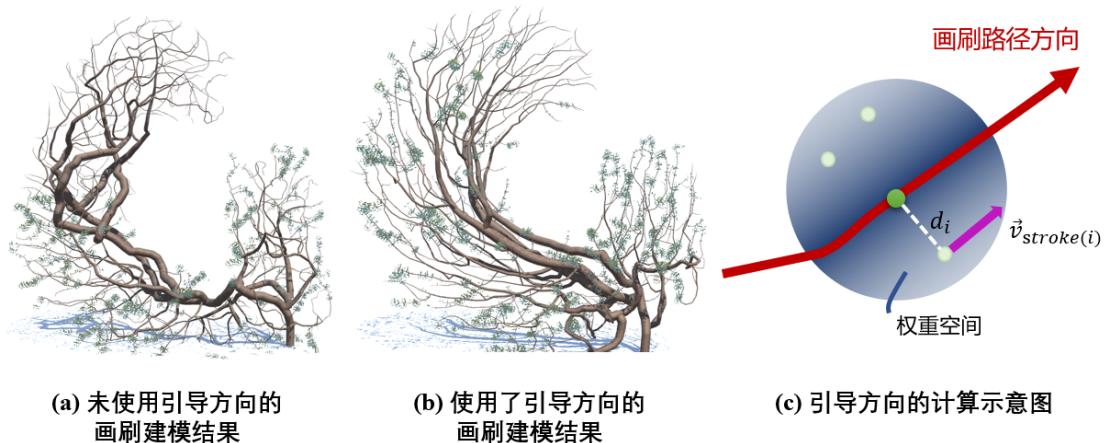


图 4.7 画刷牵引方向的原理和效果。

Figure 4.7 The description and effect of brush guiding direction.

4.3 实验结果与分析

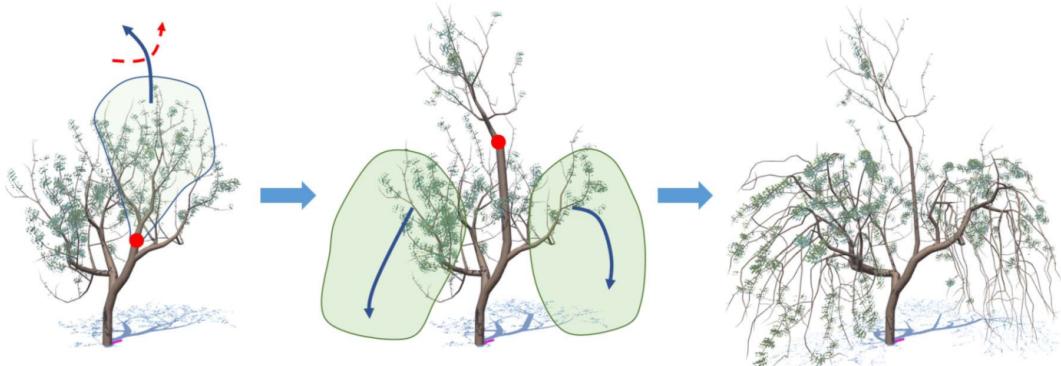


图 4.8 组合不同交互方式进行灵活的植物创作。

Figure 4.8 Plant design with the combination of different interactions.

本文基于Pico-Neo一体机开发了一种基于骨架的交互式植物建模系统，该系统提供了直接三维绘制与编辑、骨架画刷等方式来实现骨架形状的控制。用户可以灵活组合本文提供的几种交互方式以充分发挥创造性，获得更具艺术感的植物模型。图 4.8展示了组合多种交互方式的例子。用户首先选择植物骨架一个结点，然后移动（同时平移和旋转）该结点所属的整棵子树，接着用户在树木的左右两侧使用画刷工具绘制出垂直向下生长的枝干。

我们将本章系统同一种基于二维草图的植物建模方法(Chen 等, 2008)进行对比。图 4.9(a)为Chen 等 (2008)的方法采用二维草图恢复的三维枝干模型结果，该方法采用马尔可夫链从输入的二维骨架草图中推测三维信息，这一步骤需要耗

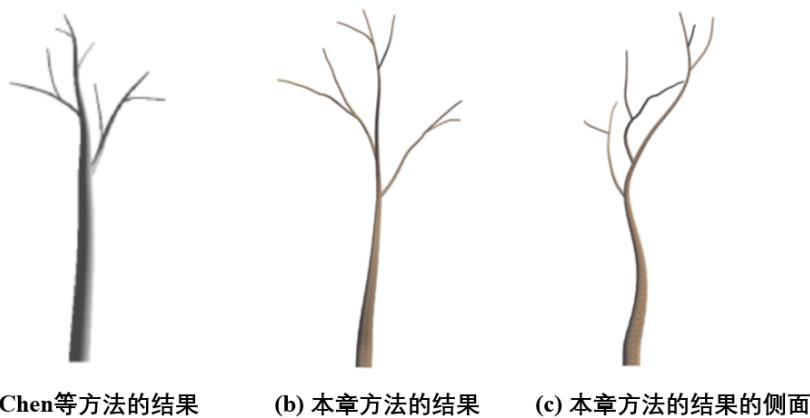


图 4.9 本章方法与一种基于二维草图的建模方法(Chen 等, 2008)的比较。

Figure 4.9 Comparison with a 2D sketch based method.

费大量的时间，并且受限于二维的交互方式，用户无法自由表达三维空间中的形状，只能被动接收系统推测的结果。而本文由于采用直接三维手势交互，用户可以随意在三维空间中绘制。图 4.9(b)为采用本文系统创建的一个相似的植物枝干模型。但是受益于三维交互，用户可以使枝干在侧面的角度上自由弯曲，图 4.9(c)展示了本文系统创建的枝干模型的侧面视图。

4.4 本章小结

本章将虚拟现实技术引用到植物建模领域，从而提出了一种三维骨架驱动的植物建模方法。用户可以直接通过三维手势来进行骨架的绘制和编辑操作。此外，我们也结合程式建模设计了画刷工具，用户可以使用画刷来绘制植物骨架的生长趋势和范围，从而快速进行植物的艺术造型设计。

第5章 三维轮廓驱动的植物建模

5.1 引言

基于二维形状的建模往往受限于推断真实的三维信息，并且用户难以对植物的三维结构做出有效的交互式控制。基于三维骨架的直接手势交互虽然能解决这一问题，但是由于用户需要绘制大量精确的枝干骨架，这也导致了交互工作量较大的缺点。因此，为了进一步扩展虚拟现实设备和三维手势在交互式建模中的优势，本章提出了一种基于三维轮廓的植物建模方法。

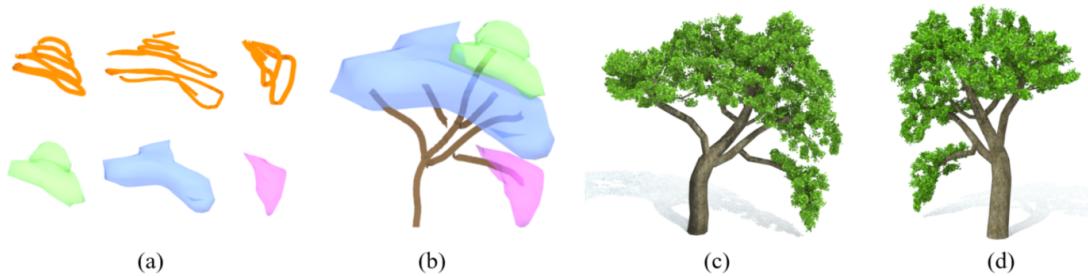


图 5.1 基于三维轮廓的植物建模。

Figure 5.1 Tree modeling based on 3D silhouette.

本章主要探讨了在虚拟现实环境中，如何利用手指绘制设计更高层的语义交互来进行植物建模的方法。该方法同样混合了手势交互和过程式植物建模。但不同于三维骨架的交互方式，这里用户只需要关注并绘制植物的大致形状即可获得期望的植物模型。图 5.1展示了本文算法的基本流程，用户首先在空中用手指绘制一些简单三维笔划以定义树木的三维轮廓包体和主枝干，然后我们的算法会根据这些形状约束自动生成完整的三维植物模型。此外，用户还可以通过手势在空间中自由地移动、变形这些三维轮廓包体。本章将通过大量的实验分析和用户研究来探讨该方法在树木建模和整体形状控制上的易用性、效率和灵活性。

图 5.2分别展示了实际的硬件环境和用户界面。我们通过HTC Vive头戴显示器来获得虚拟环境的沉浸式体验，并利用安装在头戴显示器前方的LeapMotion来追踪用户的手势动作。对于我们的可视化操作界面，用户可以在一个虚拟环境中用手势进行自由的绘制和编辑操作。在虚拟环境中用户也可以看到位于前方的控制面板。控制面板中提供了一系列按钮用来帮助用户切换手势操作的类型，

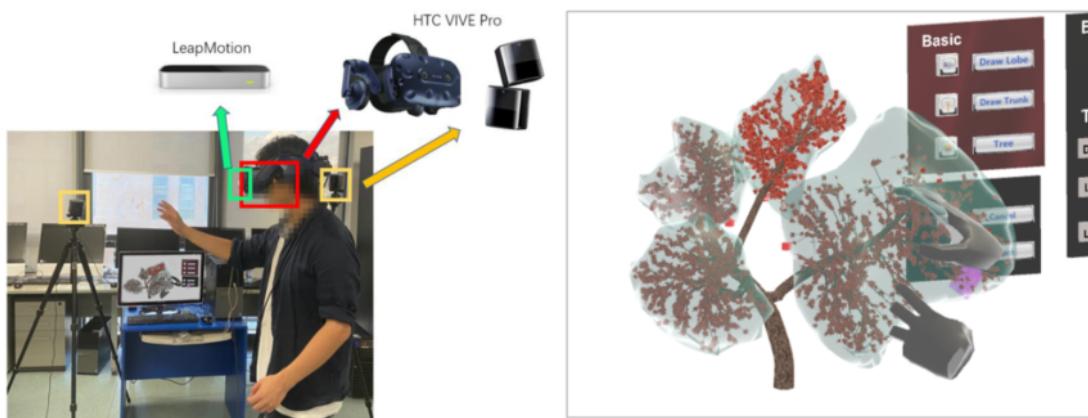


图 5.2 本章基于手势交互的建模系统的硬件环境及其界面。

Figure 5.2 The hardware setup and interface of our gesture based modeling system.

比如绘制三维轮廓、绘制主枝干、网格编辑等。此外，用户也可以通过控制面板提供的参数控制来调控植物的生长参数以获得不同种类的植物模型。

5.2 基于手势的植物三维轮廓设计

由于植物模型大多具有复杂的枝干和树叶结构，直接设计植物的所有细节是极其费时费力的，同时这也导致植物模型通常需要大量的面片和内存空间来存储。[Livny 等 \(2011\)](#)曾提出一种基于Texture lobes的树木表示方式用于压缩植物模型的细节。该方法的策略源自于对人类观察植物时的特点的分析，即：人类在观察植物时并不会仔细地观察植物地每一处细节，而是只留意植物大致的外部轮廓以及可见的主枝干。而植物的外部轮廓则可以拆解成若干个稍小的包体，他们称这些轮廓包体为Lobe。本文基于这一思想设计了一种三维轮廓驱动的交互式植物建模方法，用户仅仅通过绘制植物树冠部分的三维轮廓包体以及可见的主枝干，接着基于这些形状约束，完整的植物模型则通过过程式建模方法来获得。例如图 5.1b，我们用三个轮廓包体（Lobe）来描述整个植物树冠区域的外部轮廓。

5.2.1 三维轮廓构建方法

用户可以通过绘制若干三维线条的方式来生成轮廓包体（Lobe）。如图 5.3所示，在用户依次绘制的同时，线条将进行两两间的点对点匹配以形成流形的网格结构。网格的更新过程是实时同步的。在网格封闭后，算法将通过网格重构（Remesh）操作来获得均匀分布的三角面片。绘制过程中的网格满足流形约束

将有利于进行网格重构和网格编辑操作。

接下来我们将讨论如何在两个相邻的线条之间生成规则的网格面片。图 5.4展示了在两个线条之间生成网格的详细过程。用户绘制的线条有两种类型:闭合线条和开放线条。不同类型的线条间相互组合可以表达更加复杂的形状。当一个线条的起点和终点的距离小于一个给定阈值时,该线条将被视为闭合线条;反之,当起点终点间隔较大时则视其为开放线条。我们引入开放线条的主要目的是用于将闭合线条所产生的开口封闭,以获得密闭的网格形状。下面我们将按照不同的组合情况来介绍线条的点对点匹配过程。

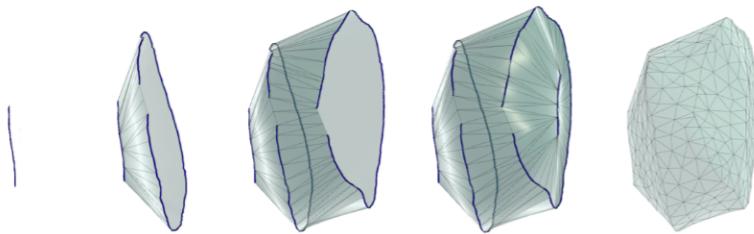


图 5.3 从手绘线条逐步构建封闭的轮廓包体的过程。

Figure 5.3 Lobe generation from strokes.

情况1 - 两个闭合线条: 图 5.4的上半部分展示了两个闭合线条的网格构建过程。红色点为线条上的采样点。为了方便后续的点对点匹配,我们首先让两个线条的顶点数保持一致。具体来说,对于采样点较少的线条,我们采用三次Hermite插值在当前间距最大的两个采样点之间插入一个新的顶点,这个过程不断重复直到两个线条的采样点数目相同。图中的绿色点即新插入的顶点。

接下来,我们执行点对点匹配,按照顺序依次将线条的顶点连接形成三角面片。但是在此之前,我们必须决定好一个好的起始点对使得网格面片能够正确的构建。一个直观的想法是采用欧式距离最近的两个顶点 (Funkhouser 等, 2004),但是这种方法对于许多极端境况并不稳定。如图 5.5所示的情况,此时右边笔划的底部远离左边笔划的底部,但却靠近左边笔划的顶部。如果简单的采纳欧式距离最近的两个点作为起始点对则会产生如图 5.5a所示的错误匹配。然而我们所期待的正确匹配的起始点对则应该是图 5.5b所示的结果,即两个线条的底部能够实现匹配。为了解决这一问题,我们经过实验发现重心方向是一种很好的特性来寻找正确匹配的起始点对。

点的重心方向描述了该点与整个线条重心的相对位置关系,因此可以很好

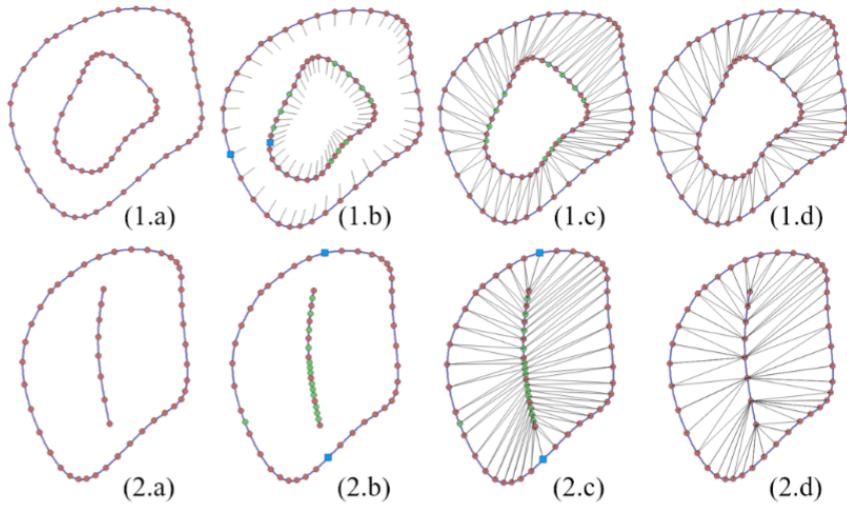


图 5.4 从两根线条形成三角网格的策略。顶部为两根闭合线条，底部为一根开放线条和闭合线条。

Figure 5.4 Mesh strip generation from two strokes. Top row: two closed strokes. Bottom row: an open and a closed stroke.

的反应出该点处整个线条形状的趋势特征。点 p_i 处的重心方向可以近似地通过该点到线条上其他点 p_k 的方向向量的平均值计算得到，即：

$$dir_{barycentric}(p_i) = \text{normalize}(\sum_{k=0}^N p_k - p_i), k \neq i \quad (5.1)$$

图 5.4(1.b)中附着在每个顶点上的灰色线段即该点处的重心方向。两个具有最接近的重心方向的顶点将被视作一个好的起始点对，图中我们用蓝色方块来表示起始点对。然后，我们便可以通过顺序连接两个线条上的顶点来形成网格面片，如图 5.4(1.c)所示。由于这两根线条还可能会与其他线条连接形成新的网格，为了保证所构成的网格满足仍然流形约束，我们最后还需要将新插值的顶点（绿色点）移除。这里，我们直接将每个新插值的点及其所属的三角面连接关系合并到最近的原始采样点上。最终形成的网格面片如图 5.4(1.d)所示。

情况2 - 闭合线条和开放线条： 图 5.4 的下半部分展示了在一个闭合线条和一个开放线条间形成网格面片的详细流程。开放线条的作用是连接闭合线条形成的空洞从而构建一个封闭的轮廓包体。在本情况下，我们需要将闭合线条根据开放线条的方位分成两个部分，以分别与开放线条的对应一侧进行连接。为此需要在闭合线条上寻找两个连接端点。首先，我们采用三次Hermite插值使得闭合线条的顶点数等于开放线条的顶点数的2倍+2。接着，对闭合线条的每个顶点计算其重心方向，闭合线条上的两个连接端点（蓝色方块）将被设置为重

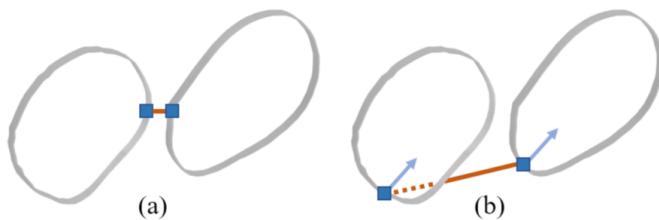


图 5.5 用于寻找最佳起始匹配点对的两种策略的比较。

Figure 5.5 Comparison between two methods for finding the best starting point pair.

心方向与开放线条两端延长线方向最为接近的两个顶点。然后我们便可以为开放线条的每一侧分别执行点对点匹配，从而得到如图 5.4(2.c)所示的初步网格面片。最后，我们采用与情况1相同的策略将新插值的顶点及其所属的三角面连接关系合并到最相邻的原始采样点中，最终形成的三角网格如图 5.4(2.d)所示。

通过上述两种策略，我们可以对任意两个用户绘制的线条构建三角网格。对于多个笔划的情形，系统可以随着用户的每次绘制而同步更新网格。本文系统允许用户不按照笔划的排布顺序绘制，当用户在现有的两个线条之间绘制新的线条时，系统会根据当前的邻接关系更新局部的网格面片。最后在得到封闭的轮廓包体后，我们还采用增量网格重构算法(Botsch 等, 2004)来获得更加均匀的三角网格。

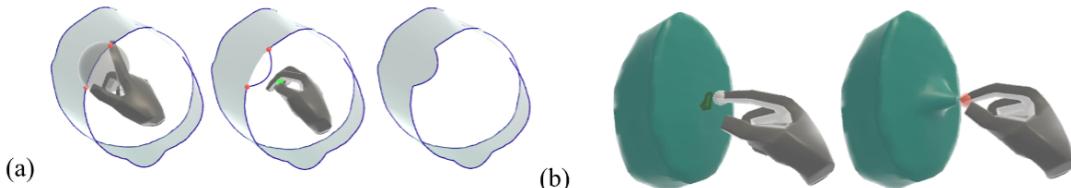


图 5.6 对线条和轮廓包体的编辑操作。

Figure 5.6 Editing tools for modifying the strokes and lobes.

5.2.2 轮廓形状编辑

由于用户常常难以一次性绘制好预期的线条或者希望进一步调整轮廓包体的形状，因此本文系统为用户提供了两种编辑方式，分别用于编辑手绘线条的形状和编辑轮廓胞体的网格（如图 5.6）。

线条编辑： 本文基于二次贝塞尔曲线算法实现了一种线条编辑工具。如图 5.6(a)所示，用户首先用拇指和食指在曲线上选择两个端点 P_1 和 P_2 用来指定待

编辑的线条区域。接着用户通过拖拽一个绿色的控制点 P_c 来控制该区域的线条形状。根据二次贝塞尔公式，移动控制点 P_c 后所形成的局部曲线可以用下式表示。

$$P(t) = (1-t)^2 P_1 + 2t(1-t)P_c + t^2 P_2 \quad (5.2)$$

式中的 $t \in [0, 1]$ 在取不同的值时可以表示曲线上的不同点，当 $t = 0$ 和 $t = 1$ 时分别代表该局部曲线的两个端点 P_1 和 P_2 。

网格编辑：为了允许用户编辑轮廓包体的网格，本文基于拉普拉斯网格变形算法(Sorkine 等, 2004)，设计了一种编辑网格的交互方式。图 5.6(b)展示了网格编辑的基本操作，用户用手指触及网格表面的一个点 P_c 作为拉普拉斯变形的控制点，然后通过拖拽该点实现网格形状控制。

拉普拉斯网格变形的基本原理是让变形前后的拉普拉斯坐标尽可能保持不变。一个顶点的拉普拉斯坐标可以反应该顶点所处位置的局部几何特征。在将三角网格近似为一个光滑曲面后，顶点 P_i 的拉普拉斯坐标 δ_i 可以通过下式计算得到：

$$\delta_i = \frac{1}{d_i} \sum_{j \in N(i)} (P_i - P_j) \quad (5.3)$$

这里 $N(i)$ 表示顶点 P_i 的邻域顶点。为了方便计算，我们引入矩阵的描述形式以一次性求取所有顶点的拉普拉斯坐标，即：

$$\delta = L \cdot V \quad (5.4)$$

式中， $\delta = (\delta_1, \delta_2, \dots, \delta_n)$ 表示所有顶点的拉普拉斯坐标的集合， $V = (V_1, V_2, \dots, V_n)$ 表示所有顶点世界坐标的集合。 $L = I - D^{-1}A$ 是尺寸为 $n \times n$ 的拉普拉斯矩阵， D 为 $n \times n$ 的对角矩阵，其对角线上的元素分别为第*i*个顶点的度数。 A 为记录网格顶点的连接关系的稀疏矩阵，当顶点 P_i 和 P_j 之间存在边连接时 $A_{i,j} = 1$ ，反之 $A_{i,j} = 0$ 。

这时当控制点 P_c 移动到新位置 P'_c 后，新的网格顶点集合 V' 可以通过求解一个带控制点位置约束的优化问题来获得，即：

$$\arg \min_{V'} \left(\|LV' - \delta\|^2 + \omega \|P'_c - P_c\|^2 \right) \quad (5.5)$$

式中 ω 表示控制点位置保持的权重参数。由于上述优化问题可以转换为求解一个超定线性方程组，因此可以通过最小二乘法直接求得变形后的网格坐标 V' 。

5.3 实验结果与分析

5.3.1 植物建模结果

为了验证本文方法的建模效果，本节设计了几组实验并展示了一系列不同种类和形态的植物模型。图 5.7展示了三组根据植物盆景的实物照片所重建的植物模型。由于盆景植物通常具有较高的艺术性，其枝干错综弯曲且树冠可以拆解为多个形状各异的轮廓包体，因此传统方法难以重建这类植物造型。而本文方法受益于直接的手势交互，用户可以将复杂的盆栽造型拆解为若干较小的独立轮廓包体并逐个绘制，因此可以重建出与原始照片造型相似的高质量艺术植物造型。



图 5.7 对照盆景植物照片创建的植物模型。

Figure 5.7 Resulting models with reference to the real Bonsai plants.

本文系统允许用户通过手势灵活的移动绘制好的轮廓包体和枝干，这有利于用户充分发挥自己的创造力。用户可以通过随意排布这些绘制好的组件来获得不同的植物造型。图 5.9(a)和(b)展示了一组采用相同的轮廓包体但是采用不同的放置排布方式的建模效果。图 5.9(c)则展示了一组更加形象的例子，用户可以通过摆放代表翅膀的轮廓包体，来让天鹅造型的植物产生翅膀振动的效果。

为了进一步探讨主枝干对于植物模型的影响，我们设计了如图 5.8所示的实

验。在使用相同轮廓包体的前提下，用户绘制不同的主枝干并观察最终的植物模型的异同。我们发现虽然最终的整体造型几乎一致，但是在主枝干所在的区域对新生长的枝干分布产生了明确的影响。图 5.10则展示了四种由用户创建的艺术造型的植物，分别为兔子、苹果、战车和酒杯，证明了本文算法在构建复杂轮廓形状时的鲁棒性。

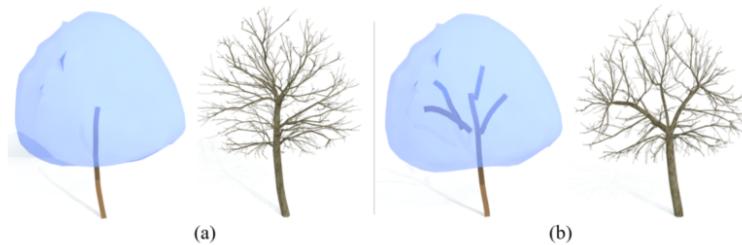


图 5.8 不同的主枝干对最终植物骨架结构的影响。

Figure 5.8 The effect of using different main trunks.

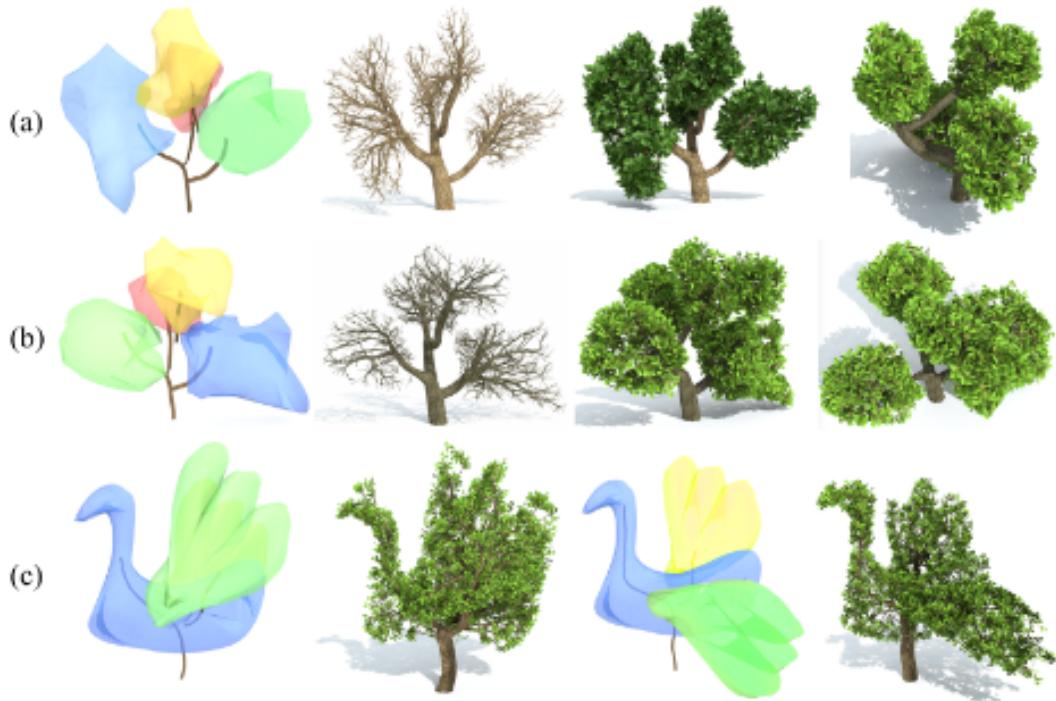


图 5.9 通过自由摆放轮廓包体进行更灵活的艺术设计。

Figure 5.9 Artistic design with various placement of lobes.

本节中出现的所有植物模型都可以在1-5分钟左右创建完成，建模时间的差异取决于植物形状的复杂程度。由于植物的建模过程包含轮廓包体绘制、主枝

于绘制和树木生长三维步骤，我们在表 5.1 中记录了在创建几株典型的植物模型时的基本信息和每个步骤的耗时。图 5.8(a)所示的植物由于只有一个包体，因此绘制时间非常短。而对于图 5.9(c)所示的植物则使用了大量的时间来绘制三维轮廓，这是由于为了获得接近天鹅造型的植物模型，用户需要非常细致的绘制和调整轮廓的细节。但是，一旦用户绘制完主枝干和轮廓包体，植物的枝干生长过程僵尸完全实时的。此外，我们也注意到，用户在描述非常复杂的形状时，往往不是使用单一的轮廓包体，而是倾向于将复杂形状拆解为若干独立的小型包体，这样做可以降低形状的绘制难度，也表明我们的系统可以充分发挥用户的创造力和想象力。

表 5.1 部分植物的形状信息及分步骤建模时间统计。

Table 5.1 Statistics of the shape information and modeling time for several plants.

植物	轮廓包体数	轮廓线条数	轮廓绘制	枝干绘制	树木生长
5.8 (a)	1	5	0.3min	2s	0.096s
5.9 (a-b)	4	22	1.2min	15s	0.310s
5.9 (c)	3	16	4.6min	10s	0.172s

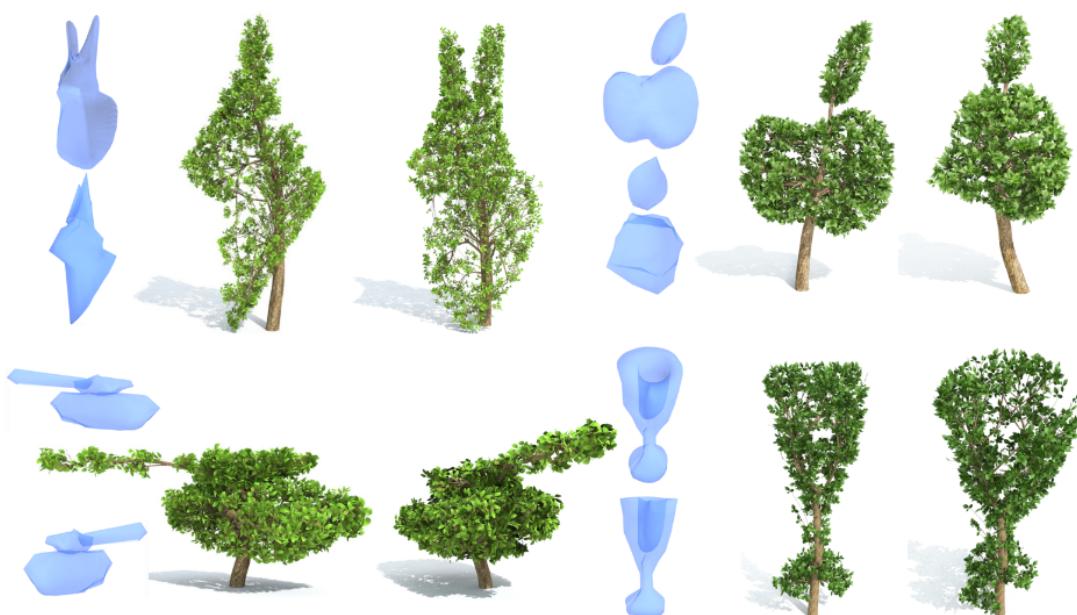


图 5.10 对特定形状的植物造型的建模结果。

Figure 5.10 Modeling results of the plants with specific shapes.

5.3.2 方法对比

我们首先将本文方法同一种最新的基于二维草图的植物建模算法(Chen 等, 2008)进行了对比。该方法允许用户在二维平面上绘制植物的二维骨架草图, 枝干的三维信息可以通过马尔可夫随机场推断得出。图 5.11(a)是使用Chen 等(2008)的方法所创建的一个植物模型, 可以发现用户需要精确地绘制大量的二维线条来作为植物骨架, 这显然是非常麻烦的。并且受限于二维输入, 该方法还需要推断可靠的深度信息, 而这往往也会耗费大量的计算时间。图 5.11(b)则是采用本文方法创建的一颗形态相近的植物模型, 用户仅仅需要绘制一个简单的轮廓包体和几根主枝干便可以非常高效地获得同样质量的植物模型。



图 5.11 同一种基于二维骨架草图的建模方法(Chen 等, 2008)的对比。

Figure 5.11 Comparison with a 2D sketch-based modeling approach.

5.3.3 用户研究

本节设计了几组用户实验以获得用户反馈并评估我们的系统。我们邀请了10名没有任何建模经验的新手用户来体验三种交互式植物建模系统, 分别为一种二维草图建模系统(Okabe 等, 2005)、TreeSketch(Longay 等, 2012)以及本文系统。在任务开始前, 我们为每位用户提供了关于三个系统所有功能的教程以学习其使用方法。我们首先邀请这些用户分别使用这些系统来随意创造植物模型, 并在体验结束后填写一份调查问卷。

调查问卷的统计结果如图 5.12所示。我们提出了三个问题以收集用户的主观评价, 分别涉及: 易用性、建模满意度和创造性。用户打分的范围为[1,10], 其中1分表示非常不同意, 10分表示非常同意。我们很高兴的看到, 用户对于本文系统的评分相较于其他两种系统来说得分更高。所有用户都反映说本文系统更容易上手使用, 直观的手势交互让他们觉得比二维绘制更加容易发挥自己的创造性, 并且将复杂的植物模型抽象为简单的轮廓包体对于初学者来说更容易学习和使用。我们收集了用户对于另外两种系统以相对较低评价的原因。而对

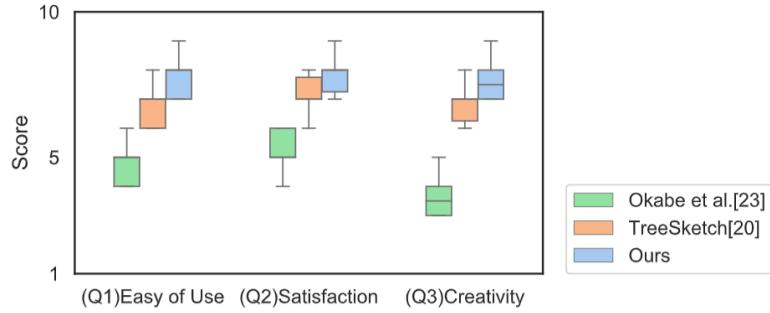


图 5.12 用户填写的调查问卷的统计结果。

Figure 5.12 The statistical results collected by questionnaire from participants.

于Okabe等人的系统，用户普遍反映说不喜欢这种需要细心绘制大量枝干线条的方式，绘制完整的骨架草图对于使用者来说非常繁琐。对于TreeSketch系统，用户则表示二维画刷难以充分表达更加丰富的三维形状，并且他们的实现方式较为简单导致生成的植物模型的外观也略显不自然。

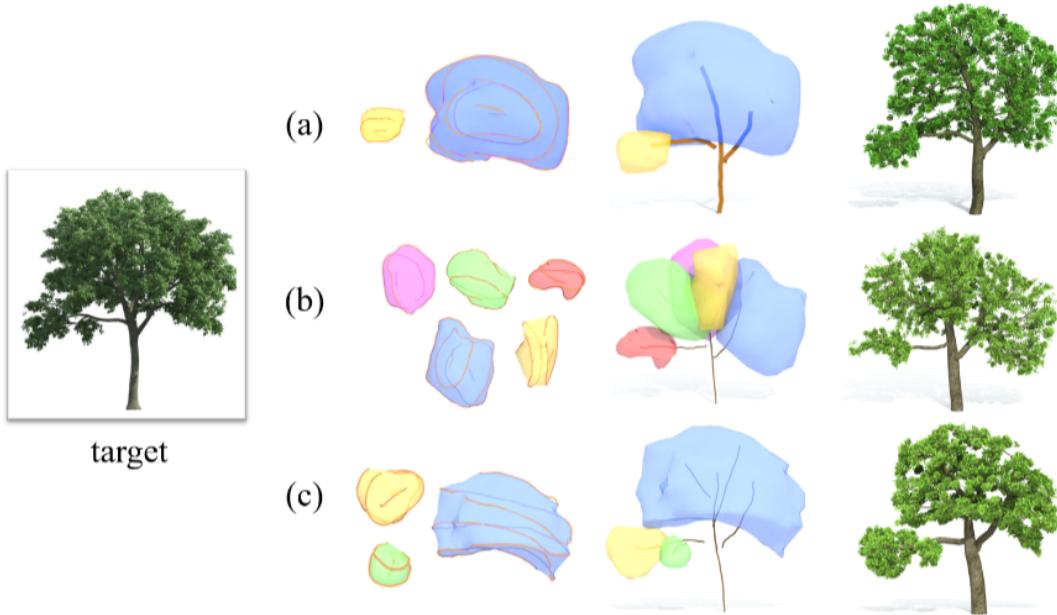


图 5.13 在用户研究实验中参与者根据同一个照片重建的植物模型示例。

Figure 5.13 Plant examples created by participants targeted to a same photo for user study.

此外，用户也指出了本文系统的存在的弱点，并提供了一些改进意见。有一位用户反映说，基于LeapMotion的手势绘制需要用户长时间抬起手臂，保持这种姿势可能会导致轻微的手臂疲劳。也有两人指出长时间佩戴虚拟现实眼镜会导致轻微的眩晕感。另一位用户则给我们提出了一条有用的建议，他希望在

未来能够提供更多的植物形态的参数控制以创建更加丰富种类的植物模型。

为了观察用户是如何表达植物的大致形状，我们也要求参与者们根据一副给定的照片来重建相应的植物模型，部分重建的结果如图 5.13 所示。我们惊喜地发现所有模型都与照片有较高的相似性。此外用户在表达相同的树木形状时采用了不同数目和形状的轮廓包体，这表明个体观察习惯的差异会影响使用者采用不同的设计方式来表达对应的形状。这也充分证明了本文系统能够充分地发挥用户的想象力。

5.4 本章小结

本章基于三维手势交互提出了一种三维轮廓驱动的植物建模方法。首先，用户通过将植物抽象为若干简单的轮廓包体（Lobes）和少量的主枝干并用手势逐个绘制。接着，我们的算法将自动根据这些三维轮廓形状生成预期的植物模型。该方法的主要优点在于交互控制少，并且可以充分发挥用户的观察力与创造性。我们设计了一组用户研究实验并邀请一些志愿者来体验我们的系统。通过用户的反馈和统计结果，证明了本文方法提供了良好的交互体验以及高效的植物造型设计能力。

第6章 总结与展望

6.0.1 本文工作总结

逼真且高效的植物建模是计算机图形学领域的一个重要问题。本文成功地将深度学习和虚拟现实技术引入到植物建模领域，并基于不同类型的形状约束提出了三种新的植物建模方法，具体包括二维形状驱动、三维骨架驱动和三维轮廓驱动的建模算法。我们进行了大量的实验分析和用户研究，证明本文方法均能够高效且高质量地进行植物建模。本文的主要贡献具体包含以下几点：

(1) 三种方法的共性是用户只需关注植物模型的大致形状特征，然后基于给出的形状约束运用过程式建模算法自动生成高质量的植物三维模型。

(2) 将深度学习技术与过程式植物建模成功结合了起来。在二维形状驱动建模方法中，我们将二维到三维形状的推断问题转化为了图像间的翻译问题。首先使用cGAN网络从二维轮廓图和枝干线条中分别预测出植物的外部三维轮廓和三维主骨架。接着基于预测得到的主骨架，完整的植物模型将通过过程式建模算法在三维轮廓内的空间中生成。

(3) 我们也将二维形状的植物建模扩展为基于单幅图像的植物建模方法。首先对输入的植物照片提取边缘图作为轮廓草图，然后用户交互式地标注照片中的可见枝干，即可作为cGAN网络的输入并预测出相应的三维轮廓和主骨架。

(4) 通过虚拟现实技术，结合三维手势提出了新的交互式植物建模方法，且实验证明使用手势的交互方法可以最大限度上发挥用户的创造力。通过使用三维骨架驱动的方法，用户可以直接在空间中使用手势或者画刷工具绘制立体的骨架结构，同时可以对植物的任意部分进行灵活的骨架编辑。而基于三维轮廓驱动的方法则允许使用更少的交互来建模，用户可以使用手势直接绘制和编辑植物的三维轮廓，而植物的细节则通过过程式建模进行完善和补充。

6.0.2 研究展望

本文提出了三种基于形状驱动的植物建模方法，并取得了一定成果。实验结果表明本文方法能够有效地帮助设计具有复杂艺术造型的植物模型。但是，本文方法仍然存在一些改进空间，并且在相关研究领域中尚有不少问题和挑战等待解决。

首先，在二维形状驱动的建模方法中，目前在得到预测的三维信息后需要人工指定程式建模的参数。因此在下一步的工作中我们希望改进神经网络，使其也能从输入图像或草图中预测出最符合的树种类型，或者直接预测出程式建模的参数。

此外，当前的虚拟现实技术仍然不够完善。部分用户反映使用沉浸式虚拟环境的交互建模系统时会产生轻微的晕眩感。并且长时间的手势交互操作也会导致一定程度的手臂疲劳。因此，如何尽可能提高用户在使用VR设备时的沉浸感体验也是下一步的研究内容。

将来我们也会继续探索更多基于三维手势的植物建模的交互操作，例如实现手势控制下的植物动画等。同时，思考如何将三维手势应用到更多类型的物体建模上也是非常有意义的研究工作。

参考文献

- 杨垠晖, 王锐. 树木的真实感建模与绘制综述 [J]. 计算机辅助设计与图形学学报, 2018, 030(002): 191-216.
- Argudo O, Chica A, Andujar C. Single-picture reconstruction and rendering of trees for plausible vegetation synthesis [J]. Computers & Graphics, 2016, 57: 55-67.
- Arora R, Kazi R H, Anderson F, et al. Experimental evaluation of sketching on surfaces in vr [C]//CHI: volume 17. 2017: 5643-5654.
- Arora R, Habib Kazi R, Grossman T, et al. Symbiosissketch: Combining 2d & 3d sketching for designing detailed 3d objects in situ [C]//Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems. 2018: 1-15.
- Badrinarayanan V, Kendall A, Cipolla R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation [J]. IEEE transactions on pattern analysis and machine intelligence, 2017, 39(12): 2481-2495.
- Beneš B, Andryšco N, Št’ava O. Interactive modeling of virtual ecosystems [C]//Proceedings of the Fifth Eurographics conference on Natural Phenomena. 2009: 9-16.
- Bloomenthal J. Modeling the mighty maple [J]. SIGGRAPH Comput. Graph., 1985, 19(3): 305-311.
- Borchert R, Honda H. Control of development in the bifurcating branch system of tabebuia rosea: a computer simulation [J]. Botanical Gazette, 1984, 145(2): 184-195.
- Bornhofen S, Lattaud C. Competition and evolution in virtual plant communities: a new modeling approach [J]. Natural Computing, 2009, 8(2): 349-385.
- Botsch M, Kobbett L. A remeshing approach to multiresolution modeling [C]//Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing. 2004: 185-192.
- Bradley D, Nowrouzezahrai D, Beardsley P. Image-based reconstruction and synthesis of dense foliage [J]. ACM Trans. on Graphics (Proc. SIGGRAPH), 2013, 32(4): 74.
- Canny J. A computational approach to edge detection [J]. IEEE Transactions on pattern analysis and machine intelligence, 1986(6): 679-698.
- Chen X, Neubert B, Xu Y Q, et al. Sketch-based tree modeling using markov random field [C]//ACM SIGGRAPH Asia 2008 papers. 2008: 1-9.
- Choi Y, Choi M, Kim M, et al. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 8789-8797.
- de Reffye P, Edelin C, Françon J, et al. Plant models faithful to botanical structure and development [J]. ACM SIGGRAPH, 1988, 22: 151-158.

- Deussen O, Lintermann B. Digital design of nature: computer generated plants and organics [M]. Springer Science & Business Media, 2006.
- Diener J, Reveret L, Fiume E. Hierarchical retargetting of 2d motion fields to the animation of 3d plant models [C]//Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation. Eurographics Association, 2006: 187-195.
- Friedman S, Stamos I. Automatic procedural modeling of tree structures in point clouds using wavelets [C]//International Conference on 3D Vision. IEEE, 2013: 215-222.
- Funkhouser T, Kazhdan M, Shilane P, et al. Modeling by example [C]//ACM SIGGRAPH 2004. 2004: 652–663.
- Girdhar R, Fouhey D F, Rodriguez M, et al. Learning a predictable and generative vector representation for objects [C]//European Conference on Computer Vision. Springer, 2016: 484-499.
- Girshick R. Fast r-cnn [C]//Proceedings of the IEEE international conference on computer vision. 2015: 1440-1448.
- Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation [C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580-587.
- Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets [C]//Advances in neural information processing systems. 2014: 2672-2680.
- GravitySketch. Gravitysketch [EB/OL]. 2018. <https://www.gravitysketch.com/>.
- Guénard J, Morin G, Boudon F, et al. Reconstructing Plants in 3D from a Single Image Using Analysis-by-Synthesis [M]//Lecture Notes in Computer Science: volume 8033 Advances in Visual Computing. 2013: 322-332.
- Guo J, Xu S, Yan D M, et al. Realistic procedural plant modeling from multiple view images [J]. IEEE Transactions on Visualization and Computer Graphics, 2018, PP(99): 1-1.
- He K, Zhang X, Ren S, et al. Deep residual learning for image recognition [C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.
- He K, Gkioxari G, Dollár P, et al. Mask r-cnn [C]//Proceedings of the IEEE international conference on computer vision. 2017: 2961-2969.
- Honda H. Description of the form of trees by the parameters of the tree-like body: Effects of the branching angle and the branch length on the shape of the tree-like body [J]. Journal of Theoretical Biology, 1971, 31(2): 331 - 338.
- Huang H, Kalogerakis E, Yumer E, et al. Shape synthesis from sketches via procedural models and convolutional networks [J]. IEEE Trans. on Vis. and Comp. Graphics, 2017, 23(8): 2003-2013.
- Ijiri T, Owada S, Igarashi T. The sketch l-system: Global control of tree modeling using free-form strokes [C]//International Symposium on Smart Graphics. Springer, 2006: 138-146.

- Isokane T, Okura F, Ide A, et al. Probabilistic plant modeling via multi-view image-to-image translation [C]//The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018.
- Isola P, Zhu J Y, Zhou T, et al. Image-to-image translation with conditional adversarial networks [C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 1125-1134.
- Keefe D F, Feliz D A, Moscovich T, et al. Cavepainting: a fully immersive 3d artistic medium and interactive experience [C]//Proceedings of the 2001 symposium on Interactive 3D graphics. 2001: 85-93.
- Kim T, Cha M, Kim H, et al. Learning to discover cross-domain relations with generative adversarial networks [C]//Proceedings of the 34th International Conference on Machine Learning-Volume 70. JMLR. org, 2017: 1857-1865.
- Kingma D P, Ba J. Adam: A method for stochastic optimization [J]. arXiv preprint arXiv:1412.6980, 2014.
- Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks [J]. Advances in neural information processing systems, 2012, 25: 1097-1105.
- Li C, Deussen O, Song Y Z, et al. Modeling and generating moving trees from video [J]. ACM Trans. on Graphics (Proc. SIGGRAPH Asia), 2011, 30(6): 127:1-127:12.
- Li Y, Fan X, Mitra N J, et al. Analyzing growing plants from 4d point cloud data [J]. ACM Trans. on Graphics (Proc. SIGGRAPH Asia), 2013, 32(6): 157:1-157:10.
- Lindenmayer A. Mathematical models for cellular interactions in development. ii. simple and branching filaments with two-sided inputs. [J]. Journal of Theoretical Biology, 1968, 18: 300-315.
- Livny Y, Yan F, Olson M, et al. Automatic reconstruction of tree skeletal structures from point clouds [J]. ACM Trans. on Graphics (Proc. SIGGRAPH), 2010, 29(6): 151.
- Livny Y, Pirk S, Cheng Z, et al. Texture-lobes for tree modeling [C]//ACM Trans. on Graphics (Proc. SIGGRAPH). 2011: 1.
- Longay S, Runions A, Boudon F, et al. Treesketch: Interactive procedural modeling of trees on a tablet [C]//SBIM '12: Proceedings of the International Symposium on Sketch-Based Interfaces and Modeling. Goslar Germany, Germany: Eurographics Association, 2012: 107-120.
- Lun Z, Gadelha M, Kalogerakis E, et al. 3d shape reconstruction from sketches via multi-view convolutional networks [C]//2017 International Conference on 3D Vision (3DV). IEEE, 2017: 67-77.
- Mirza M, Osindero S. Conditional generative adversarial nets [J]. Computer Science, 2014: 2672-2680.

- Neubert B, Franken T, Deussen O. Approximate image-based tree-modeling using particle flows [J]. ACM Trans. on Graphics (Proc. SIGGRAPH), 2007, 26(3).
- OculusMedium. Oculusmedium [EB/OL]. 2016. <https://oculus.com/medium/>.
- Okabe M, Owada S, Igarash T. Interactive design of botanical trees using freehand sketches and example-based editing [J]. Computer Graphics Forum (Proc. EUROGRAPHICS), 2005, 24(3): 487-496.
- Oppenheimer P E. Real time design and animation of fractal plants and trees [J]. SIGGRAPH Comput. Graph., 1986, 20(4): 55-64.
- Palubicki W, Horel K, Longay S, et al. Self-organizing tree models for image synthesis [C]//volume 28. 2009.
- Pfeifer N, Gorte B, Winterhalder D, et al. Automatic reconstruction of single trees from terrestrial laser scanner data [C]//Proceedings of 20th ISPRS Congress. 2004: 114-119.
- Pirk S, Stava O, Kratt J, et al. Plastic trees: Interactive self-adapting botanical tree models [J]. ACM Trans. on Graphics (Proc. SIGGRAPH), 2012, 31(4): 50:1-50:10.
- Prusinkiewicz P, Lindenmayer A, Hanan J. The algorithmic beauty of plants [J]. The virtual laboratory (USA), 1990.
- Prusinkiewicz P, Lindenmayer A. The algorithmic beauty of plants [M]. New York, USA: Springer-Verlag New York, Inc., 1990b.
- Prusinkiewicz P, James M, Měch R. Synthetic topiary [C]//SIGGRAPH '94: Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques. 1994: 351-358.
- Prusinkiewicz P, Hammel M, Hanan J, et al. L-systems: from the theory to visual models of plants [C]//Proceedings of the 2nd CSIRO Symposium on Computational Challenges in Life Sciences: volume 3. 1996: 1-32.
- Quan L, Tan P, Zeng G, et al. Image-based plant modeling [C]//ACM Trans. on Graphics (Proc. SIGGRAPH): volume 25. 2006: 599-604.
- Raumonen P, Kaasalainen M, ?kerblom M, et al. Fast automatic precision tree models from terrestrial laser scanner data [J]. Remote Sensing, 2013, 5(2): 491.
- Reche-Martinez A, Martin I, Drettakis G. Volumetric reconstruction and interactive rendering of trees from photographs [J]. ACM Trans. on Graphics (Proc. SIGGRAPH), 2004, 23(3): 720-727.
- Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection [C]// Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 779-788.
- Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks [J]. arXiv preprint arXiv:1506.01497, 2015.
- Rezende D J, Eslami S A, Mohamed S, et al. Unsupervised learning of 3d structure from images [C]//Advances in Neural Information Processing Systems. 2016: 4996-5004.

- Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation [C]//International Conference on Medical image computing and computer-assisted intervention. Springer, 2015: 234-241.
- Rosales E, Rodriguez J, Sheffer A. Surfacebrush: from virtual reality drawings to manifold surfaces [J]. arXiv preprint arXiv:1904.12297, 2019.
- Runions A, Lane B, Prusinkiewicz P. Modeling trees with a space colonization algorithm [C]// Proceedings of the Third Eurographics Conference on Natural Phenomena. 2007: 63-70.
- Schkolne S, Pruett M, Schröder P. Surface drawing: creating organic 3d shapes with the hand and tangible tools [C]//Proceedings of the SIGCHI conference on Human factors in computing systems. 2001: 261-268.
- ShapeLab. Shapelab [EB/OL]. 2018. <https://store.steampowered.com/app/571890/ShapeLab/>.
- Shlyakhter I, Rozenoer M, Dorsey J, et al. Reconstructing 3d tree models from instrumented photographs [J]. IEEE Comput. Graph. Appl., 2001, 21(3): 53-61.
- Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition [C]//3rd International Conference on Learning Representations, ICLR. 2015.
- Sinha A, Unmesh A, Huang Q, et al. Surfnet: Generating 3d shape surfaces using deep residual networks [C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 6040-6049.
- Sorkine O, Cohen-Or D, Lipman Y, et al. Laplacian surface editing [C]//Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing. 2004: 175-184.
- Stava O, Pirk S, Kratt J, et al. Inverse procedural modelling of trees [J]. Computer Graphics Forum, 2014, 33(6): 118-131.
- Sutherland I. The ultimate display [J]. 1965.
- Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions [C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 1-9.
- Tan P, Zeng G, Wang J, et al. Image-based tree modeling [C]//ACM Trans. on Graphics (Proc. SIGGRAPH): volume 26. 2007: 87.
- Tan P, Fang T, Xiao J, et al. Single image tree modeling [J]. ACM Trans. on Graphics (Proc. SIGGRAPH Asia), 2008, 27(5): 108:1-108:7.
- TiltBrush. Google tiltbrush [EB/OL]. 2018. <https://tiltbrush.com/>.
- Wu J, Zhang C, Xue T, et al. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling [C]//Advances in neural information processing systems. 2016: 82-90.
- Xu H, Gossett N, Chen B. Knowledge and heuristic-based modeling of laser-scanned trees [J]. ACM Trans. on Graphics, 2007, 26(4): 19.

- Yan D M, Wintz J, Mourrain B, et al. Efficient and robust reconstruction of botanical branching structure from laser scanned points [C]//Proc. 11th IEEE Int. Conf. CAD/Graph. Comput. 2009: 572-575.
- Yi L, Li H, Guo J, et al. Tree growth modelling constrained by growth equations [C]//Computer Graphics Forum: volume 37. Wiley Online Library, 2018: 239-253.
- Zhu J Y, Park T, Isola P, et al. Unpaired image-to-image translation using cycle-consistent adversarial networks [C]//Proceedings of the IEEE international conference on computer vision. 2017: 2223-2232.

作者简历及攻读学位期间发表的学术论文与研究成果

已发表(或正式接受)的学术论文:

硕士期间共发表4篇国际期刊和会议，均为第一作者/共同第一作者。

1. **Zhihao Liu**, Kai Wu, Jianwei Guo, Yunhai Wang, Oliver Deussen, Zhanglin Cheng. Single Image Tree Reconstruction via Adversarial Network. Graphical Models. 2021. **CCF-B类期刊**
2. Fanxing Zhang*, **Zhihao Liu***(共同第一作者), Zhanglin Cheng, Oliver Deussen, Baoquan Chen, Yunhai Wang. Mid-air Finger Sketching for Tree Modeling. IEEE VR. 2021. **CCF-A类会议**
3. **Zhihao Liu**, Fanxing Zhang, Zhanglin Cheng. BuildingSketch: Freehand Mid-Air Sketching for Building Modeling. IEEE ISMAR. 2021. **CCF-B类会议**
4. **Zhihao Liu**, Ce Shen, Zhi Li, Tingyu Weng, Oliver Deussen, Zhanglin Cheng, Dangxiao Wang. Interactive Modeling of Trees using VR Devices. The 9th International Conference on Virtual Reality and Visualization. 2019.

申请或已获得的专利:

三维交互式植物建模方法、装置、终端及存储介质：

CN201911395795.X[P].2019-12-30.

基于手势交互的植物建模方法及其植物建模装置、设备：

CN202110203699.1[P].2021-02-23.

参加的研究项目及获奖情况:

国家自然科学基金面上项目，基于众包和主动采集数据的城市大场景三维重建研究，61972388，2020.1-2023.12

深圳市基础研究学科布局项目，面向灾害应急的无人机遥感对地观测关键技术研发，JCYJ20180507182222355，2019.4-2022.3

致 谢

感谢灰原哀、逢坂大河和长门有希，对咱的不离不弃。

感谢派蒙祖宗发力，让我的可莉抽到了四命。

感谢陌生的队友们耐心的把我的LOL抬到白金段。

感谢实验室的伙伴们在每个夜晚陪我"加班"开黑送温暖。

在希望世界和平之前，也希望楼下卖烤冷面的大叔以后能不要再把自己的汗甩到冷面上，也希望他能不要一边抠鼻子一边拿香菜.....